we c— title: “Assignment 2.1” output: html_notebook

Description of Demographics

In this section, describe all the demographic variables that you intend to use in your analysis. In addition to your write-up, it should include relevant numerical measures (including tables) and graphs.

Name: Sammarieo Brown ID: 620142596

Package Management -> Importing the necessary packages that will be used in this project

```
library(tibble)
library(tidyr)
library(haven)
library(readr)
library(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
```

```
library(ggplot2)
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.2
## --
```

```
## v purrr   1.0.1     v forcats 0.5.2
## v stringr 1.5.0
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(ggrepel)
```

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("C:/Users/samma/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))
```

```
# coverting the SLC_2007.sav to csv

converted_df <- write.table(x=dataframe,file="C:/Users/samma/DataSpell/Statistical Computing Project/da
```

Set the seed and take 90% of the random sample

```
set.seed(710)

# import dataset
SLC_2007.Data <- read.csv("C:/Users/samma/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


print(colnames(SLC_2007.Sample))
```

```
##  [1] "hh_num"                "final_weight"          "water_bill"
##  [4] "hh_size_all"           "hh_size_mem"           "per_cap_con_all"
##  [7] "per_cap_con_mem"       "per_cap_pop_quint"     "type_dwelling"
## [10] "material_walls"        "num_rooms"             "type_toilet"
## [13] "toilet_shared"         "kitchen_shared"        "own_dwelling"
## [16] "other_dwelling"        "renter"                "rent_amt"
## [19] "rent_period"           "rent_helper"           "pay_mortgage"
## [22] "mortgage_payment"      "mortgage_num"          "mortgage_period"
## [25] "pay_taxes"             "tax_period"            "water_source"
## [28] "water_lock"            "water_meter"           "water_bill_latest"
## [31] "water_bill_months"     "water_source_shared"   "water_source_dist"
## [34] "water_source_dist_code" "light_source"         "electric_bill"
## [37] "light_bill_months"     "own_land_tele"         "own_cell_tele"
## [40] "land_tele_bill"        "cell_tele_bill"        "land_tele_use"
## [43] "cell_tele_use"         "garbage_dispose"       "area_code"
```

```
# 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,area_code,hh_size_all,hh_size_mem,
                          per_cap_con_all,water_bill,water_source, water_bill_latest

)
print(colnames(SLC_2007.Subset))
```

```
##  [1] "area_code"         "type_dwelling"     "type_toilet"
##  [4] "toilet_shared"     "kitchen_shared"    "own_dwelling"
##  [7] "water_source"      "hh_size_all"       "hh_size_mem"
## [10] "per_cap_con_all"   "water_bill"        "water_bill_latest"
```

```
print(head(SLC_2007.Subset))
```

```
##      area_code type_dwelling type_toilet toilet_shared kitchen_shared
```

```
## 2584         2             1           3             2              3
## 4431         3             1           2             2              2
## 3392         3             1           2             1              1
## 1288         3             1           2             1              1
## 6879         2             1           3             2              2
## 6611         4             2           2             1              1
##      own_dwelling water_source hh_size_all hh_size_mem per_cap_con_all
## 2584            5            3           1           1        33030.90
## 4431            1            3          11           9        61097.74
## 3392            1            1           1           1       402235.81
## 1288            1            7           2           2        41369.73
## 6879            1            1           1           1       100486.94
## 6611            1            1           5           5       126883.12
##      water_bill water_bill_latest
## 2584          0                NA
## 4431          0                NA
## 3392       4200               350
## 1288          0                NA
## 6879       4920               410
## 6611       9600               800
```

Data Wrangling

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

```
# 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,

  ) %>%
  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
    ),

  )

print(head(SLC_2007.Subset))
```

```
##       area_code           type_dwelling     type_toilet toilet_shared
## 2584 Other Town SEPARATE HOUSE DETACHED              PIT        SHARED
## 4431      Rural SEPARATE HOUSE DETACHED W.C. NOT LINKED        SHARED
## 3392      Rural SEPARATE HOUSE DETACHED W.C. NOT LINKED EXCLUSIVE USE
## 1288      Rural SEPARATE HOUSE DETACHED W.C. NOT LINKED EXCLUSIVE USE
## 6879 Other Town SEPARATE HOUSE DETACHED              PIT        SHARED
## 6611          4     SEMI-DETACHED HOUSE W.C. NOT LINKED EXCLUSIVE USE
##      kitchen_shared own_dwelling        water_source hh_size_all hh_size_mem
## 2584           NONE            5    Public standpipe           1           1
```

```
## 4431         SHARED          YES    Public standpipe          11           9
## 3392  EXCLUSIVE USE          YES     Indoor tap/pipe           1           1
## 1288  EXCLUSIVE USE          YES Trucked water (NWC)           2           2
## 6879         SHARED          YES     Indoor tap/pipe           1           1
## 6611  EXCLUSIVE USE          YES     Indoor tap/pipe           5           5
##      per_cap_con_all water_bill water_bill_latest
## 2584        33030.90          0                NA
## 4431        61097.74          0                NA
## 3392       402235.81       4200               350
## 1288        41369.73          0                NA
## 6879       100486.94       4920               410
## 6611       126883.12       9600               800
```

Data Visualization

Description of Demographic Variables

```
library(tidyverse)
library(gtsummary)
library(gapminder)

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()

table1
```

```
## Table printed with `knitr::kable()`, not {gt}. Learn why at
## https://www.danieldsjoberg.com/gtsummary/articles/rmarkdown.html
## To suppress this message, include `message = FALSE` in code chunk header.
```

Characteristic	**N**	**N = 6,278**
Area Code	6,278	
4		182 (2.9%)
5		233 (3.7%)
KMA		919 (15%)
Other Town		1,208 (19%)
Rural		3,736 (60%)

```
SLC_2007.Subset2 <- SLC_2007.Subset %>%
        rename("Type of Toilet" = type_toilet)
table2 <- SLC_2007.Subset2 %>%
        select("Type of Toilet") %>%
```

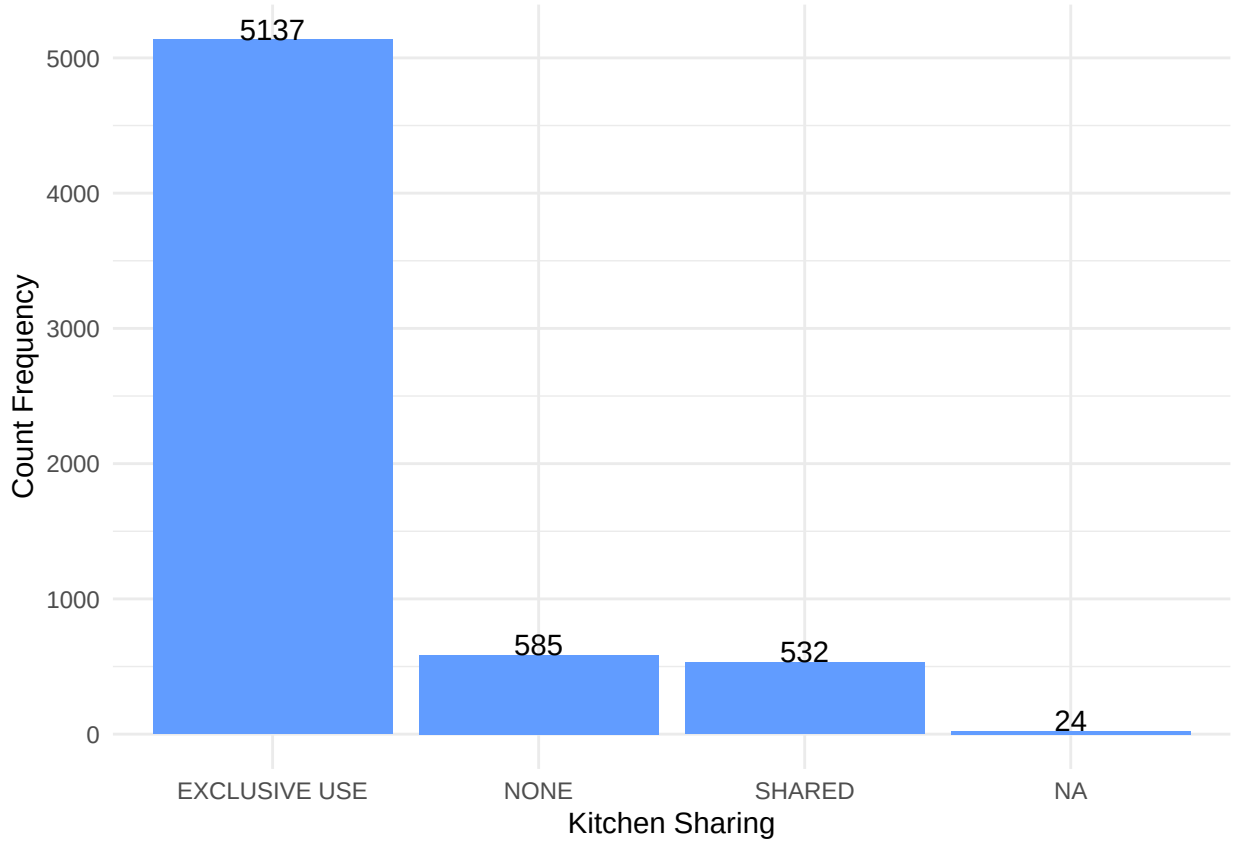
```
        tbl_summary(
                missing = "no"
        ) %>%
        add_n() %>% # add column with total number of non-missing observations
        modify_header(label = "**Characteristic**") %>% # update the column header
        bold_labels()



table2
```

```
## Table printed with `knitr::kable()`, not {gt}. Learn why at
## https://www.danieldsjoberg.com/gtsummary/articles/rmarkdown.html
## To suppress this message, include `message = FALSE` in code chunk header.
```

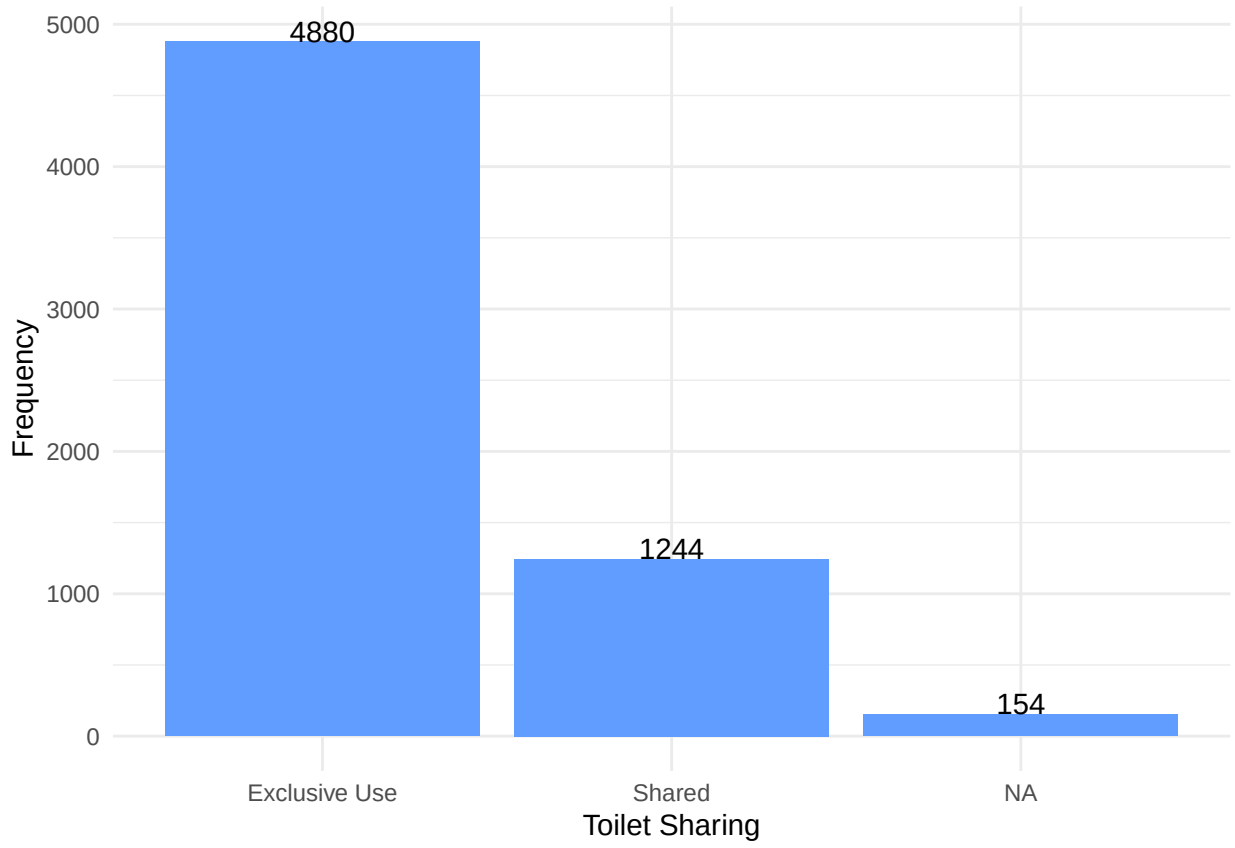
Characteristic	**N**	**N = 6,278**
Type of Toilet	6,255	
NONE		110 (1.8%)
OTHER		7 (0.1%)
PIT		2,931 (47%)
W.C. LINKED TO SEWER		957 (15%)
W.C. NOT LINKED		2,250 (36%)

```
ggplot(SLC_2007.Subset, aes(x = kitchen_shared)) +
        geom_bar(aes(y = ..count..), fill = "#619CFF") +
        labs(
             x = "Kitchen Sharing",
             y = "Count Frequency") +
        geom_text(aes(label = ..count..), stat = "count", vjust = 0.01, colour = "black")+
        theme_minimal()
```

```
## Warning: The dot-dot notation (`..count..`) was deprecated in ggplot2 3.4.0.
## i Please use `after_stat(count)` instead.
```




```
library(ggplot2)

ggplot(SLC_2007.Subset, aes(x = toilet_shared)) +
        geom_bar(fill = "#619CFF") +
        labs(
             x = "Toilet Sharing",
             y = "Frequency") +
        scale_x_discrete(labels = c("Exclusive Use", "Shared")) +
        geom_text(aes(label = ..count..), stat = "count", vjust =0.01, colour = "black")+

        theme_minimal()
```



Description of Key Variables

```
# Calculate correlation coefficient and format as table
cor_table <- data.frame(Correlation = round(cor(SLC_2007.Subset$per_cap_con_all, SLC_2007.Subset$hh_size
rownames(cor_table) <- "per_cap_con_all vs. hh_size_all"
cor_table
```

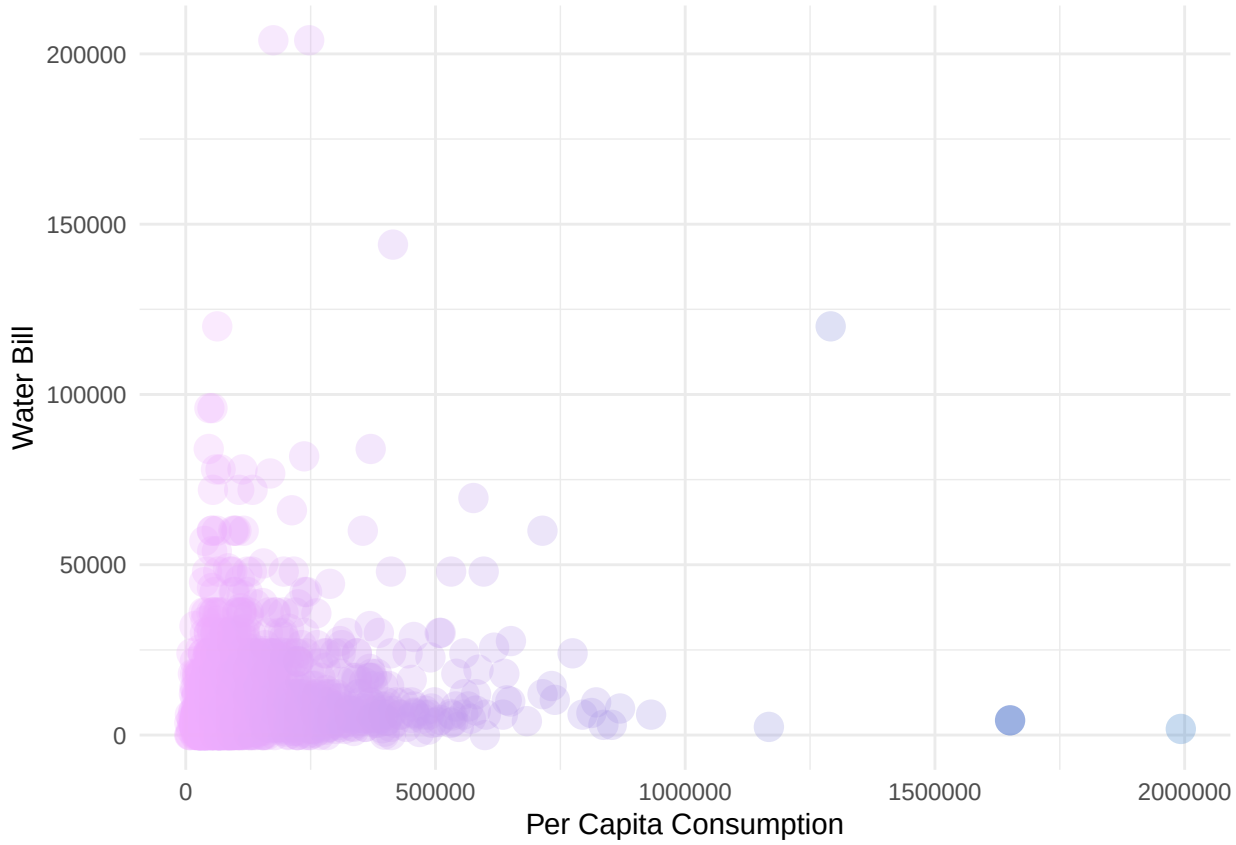
```
##                                Correlation
## per_cap_con_all vs. hh_size_all     -0.2949
```

```
# Generate summary table of water_bill variable
summary(SLC_2007.Subset$water_bill)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##       0       0       0    4171    6000  204000      21
```

```
# Remove rows with missing values
SLC_2007.Subset <- na.omit(SLC_2007.Subset)

# Add first principal component
SLC_2007.Subset$pc <- predict(prcomp(~per_cap_con_all + water_bill, SLC_2007.Subset))[,1]

# Add density for each point
SLC_2007.Subset$density <- fields::interp.surface(
        MASS::kde2d(SLC_2007.Subset$per_cap_con_all, SLC_2007.Subset$water_bill), SLC_2007.Subset[,c("p
```

```r
# Plot with title
ggplot(SLC_2007.Subset, aes(per_cap_con_all, water_bill, color = pc, alpha = 1/density)) +
        geom_point(shape = 16, size = 5, show.legend = FALSE) +

        theme_minimal() +
        scale_color_gradient(low = "#2473c5", high = "#f2aeff") +
        scale_alpha(range = c(.25, .6)) +
        labs(#title = "Relationship between Per Capita Consumption and Water Bill with PC and density",
             x = "Per Capita Consumption",
             y = "Water Bill")
```

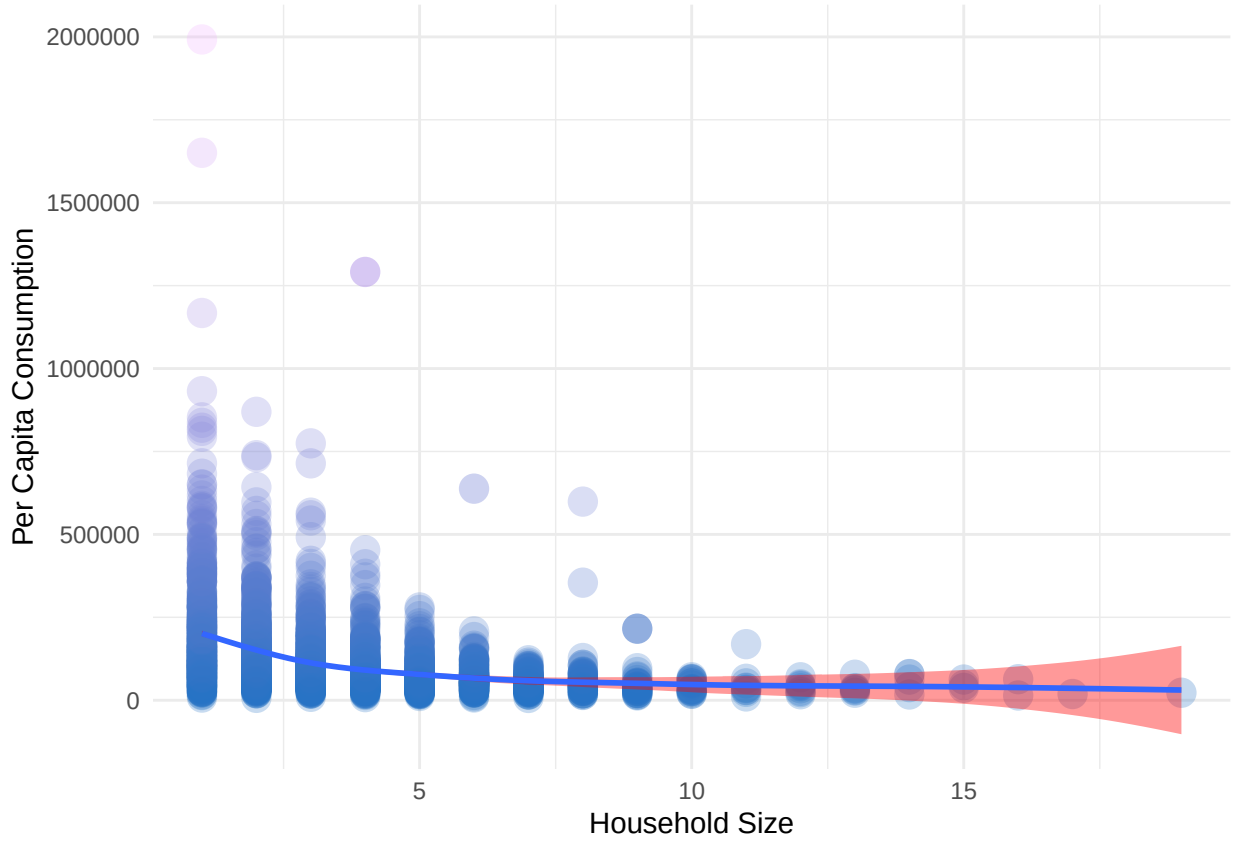



```r
# Remove rows with missing values
SLC_2007.Subset <- na.omit(SLC_2007.Subset)

# Add first principal component
SLC_2007.Subset$pc <- predict(prcomp(~hh_size_all + per_cap_con_all, SLC_2007.Subset))[,1]

# Add density for each point
SLC_2007.Subset$density <- fields::interp.surface(
        MASS::kde2d(SLC_2007.Subset$hh_size_all, SLC_2007.Subset$per_cap_con_all), SLC_2007.Subset[,c("

# Plot with title and updated y-axis label and variable
ggplot(SLC_2007.Subset, aes(hh_size_all, per_cap_con_all, color = pc, alpha = 1/density)) +
        geom_point(shape = 16, size = 5, show.legend = FALSE) +
        geom_smooth(fill='red') +
        theme_minimal() +
```

```
        scale_color_gradient(low = "#2473c5", high = "#f2aeff", guide = "none") +
        scale_alpha(range = c(.25, .6), guide = "none") +
        labs(#title = "Relationship between Household Size and Per Capita Consumption with PC and densi
               x = "Household Size",
               y = "Per Capita Consumption")
```

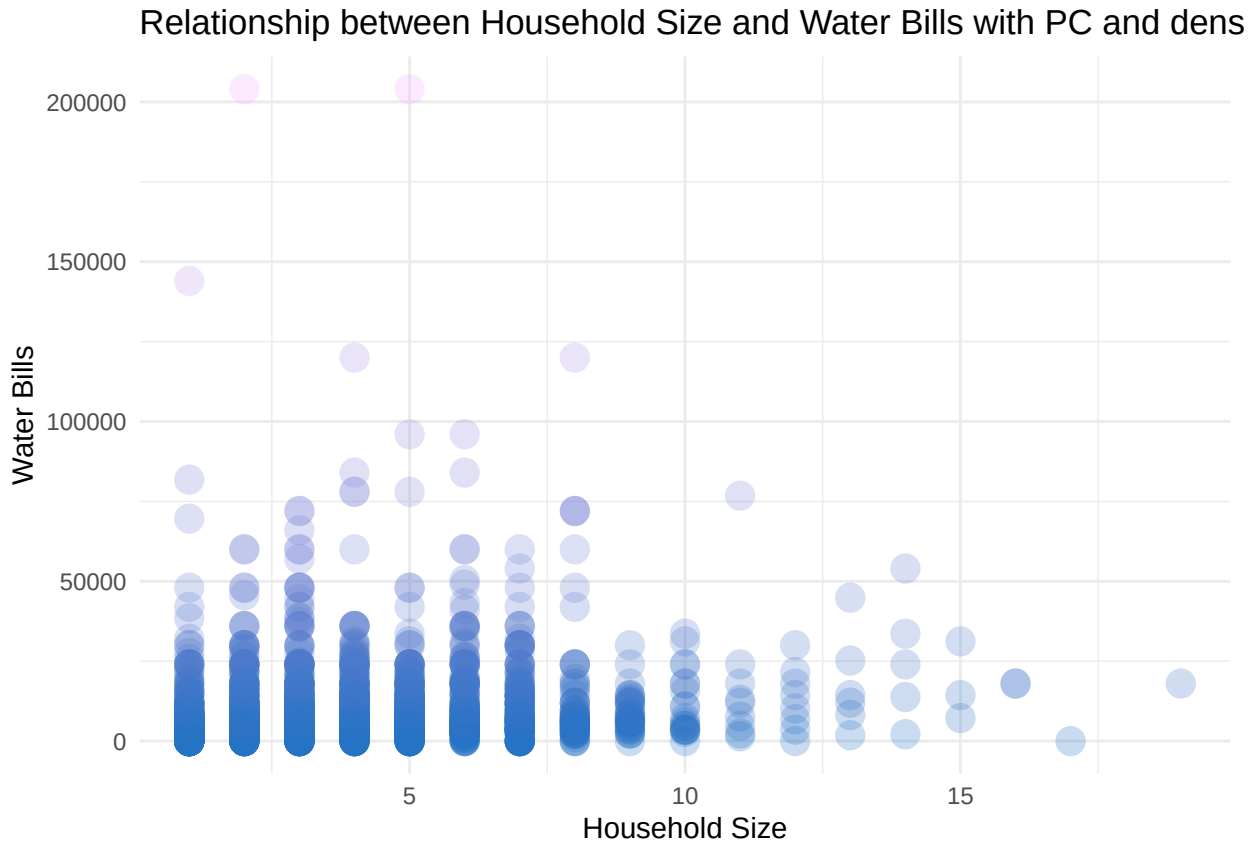
```
## `geom_smooth()` using method = 'gam' and formula = 'y ~ s(x, bs = "cs")'
```

```
## Warning: The following aesthetics were dropped during statistical transformation:
## colour, alpha
## i This can happen when ggplot fails to infer the correct grouping structure in
##   the data.
## i Did you forget to specify a `group` aesthetic or to convert a numerical
##   variable into a factor?
```

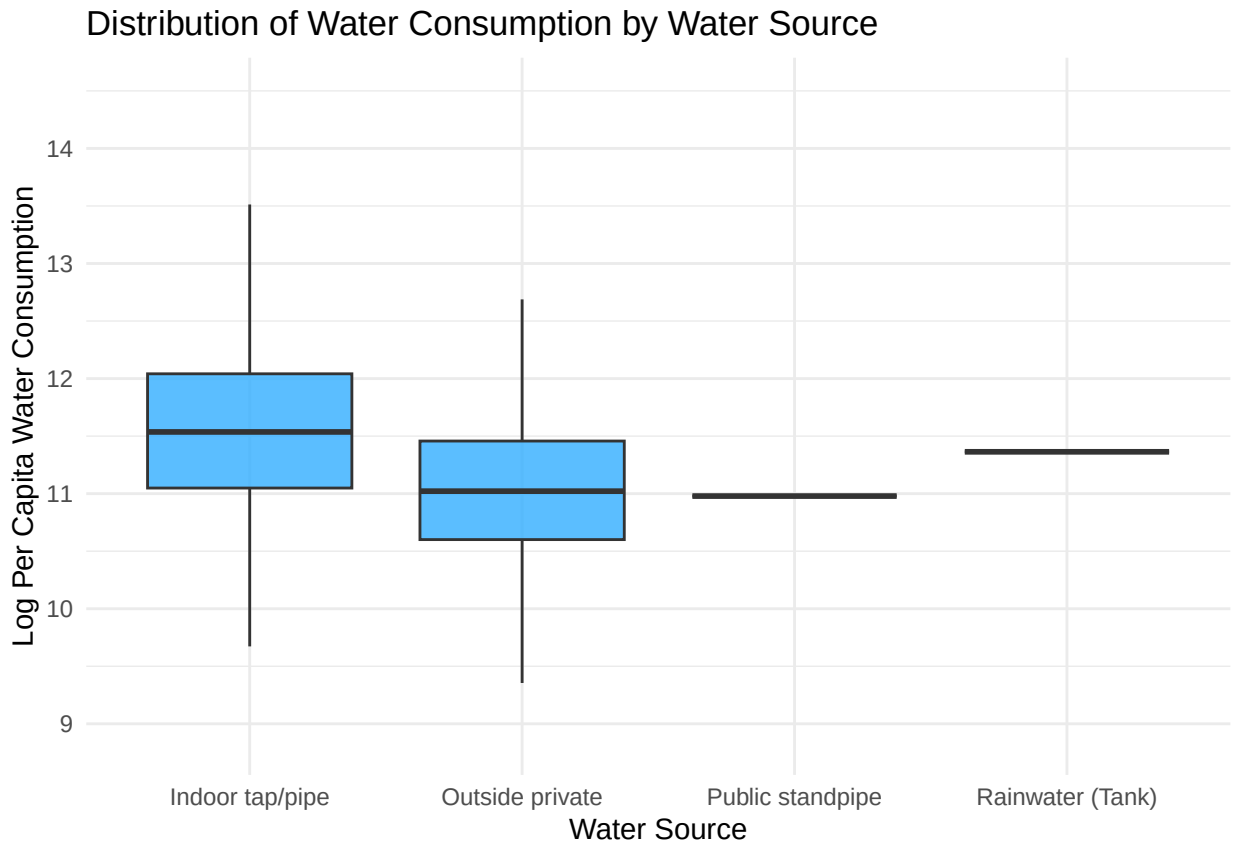



```
# Remove rows with missing values
SLC_2007.Subset <- na.omit(SLC_2007.Subset)

# Add first principal component
SLC_2007.Subset$pc <- predict(prcomp(~hh_size_all + water_bill, SLC_2007.Subset))[,1]

# Add density for each point
SLC_2007.Subset$density <- fields::interp.surface(
        MASS::kde2d(SLC_2007.Subset$hh_size_all, SLC_2007.Subset$water_bill), SLC_2007.Subset[,c("hh_si

# Plot with title and updated y-axis label and variable
ggplot(SLC_2007.Subset, aes(hh_size_all, water_bill, color = pc, alpha = 1/density)) +
```

```
        geom_point(shape = 16, size = 5, show.legend = FALSE) +
        theme_minimal() +
        scale_color_gradient(low = "#2473c5", high = "#f2aeff") +
        scale_alpha(range = c(.25, .6)) +
        labs(title = "Relationship between Household Size and Water Bills with PC and density",
             x = "Household Size",
             y = "Water Bills")
```

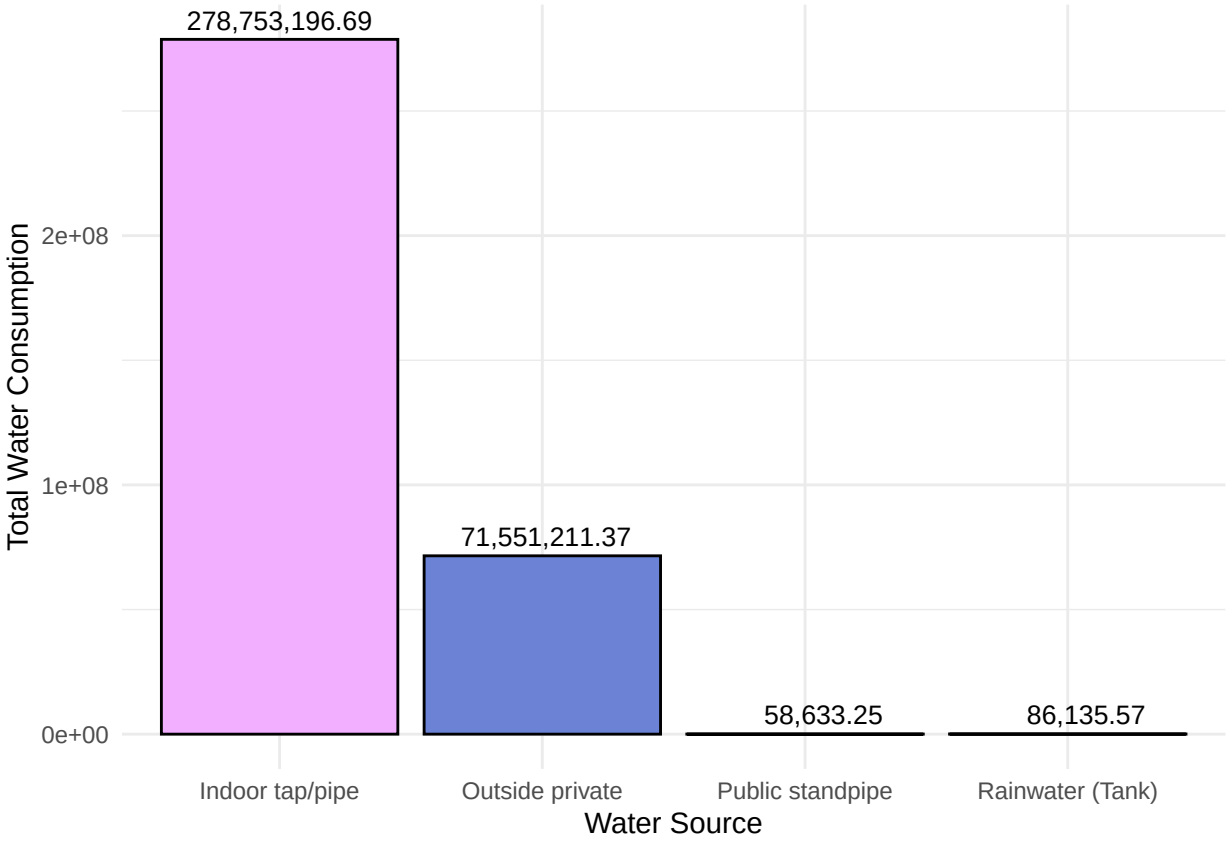



```
# Remove rows with missing values
SLC_2007.Subset <- na.omit(SLC_2007.Subset)

# Group water source categories
SLC_2007.Subset$water_source_group <- ifelse(SLC_2007.Subset$water_source %in% c("Public tap/standpipe"

# Transform water consumption variable by  logarithmic  transformation to compress the range of values.
SLC_2007.Subset$log_per_cap_con_all <- log(SLC_2007.Subset$per_cap_con_all)

# Create boxplot of water consumption by water source without outliers
ggplot(SLC_2007.Subset, aes(water_source_group, log_per_cap_con_all)) +
        geom_boxplot(fill = "#32aeff", alpha = 0.8, outlier.color = NA, outlier.shape = NA) +
        theme_minimal() +
        labs(title = "Distribution of Water Consumption by Water Source",
             x = "Water Source",
             y = "Log Per Capita Water Consumption")
```




```
# Summarize water consumption by water source
water_summary <- aggregate(per_cap_con_all ~ water_source, SLC_2007.Subset, FUN = sum)

# Create bar graph of water consumption by water source
ggplot(water_summary, aes(x = water_source, y = per_cap_con_all, fill = per_cap_con_all)) +
        geom_bar(stat = "identity", color = "black", show.legend = FALSE) +
        scale_fill_gradient(low = "#2473c5", high = "#f2aeff", guide = FALSE) +
        geom_text(aes(label = format(round(per_cap_con_all, 2), big.mark = ","), y = per_cap_con_all),
        theme_minimal() +
        labs(#title = "Water Consumption by Water Source",
             x = "Water Source",
             y = "Total Water Consumption")
```

```
## Warning: The `guide` argument in `scale_*()` cannot be `FALSE`. This was deprecated in
## ggplot2 3.3.4.
## i Please use "none" instead.
```


278,753,196.69
71,551,211.37
58,633.25
86,135.57
2e+08
1e+08
0e+00
Total Water Consumption
Indoor tap/pipe
Outside private
Public standpipe
Rainwater (Tank)
Water Source