12_Toilet_Facilities_National

#Loading Libraries

```
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(tidyr)
library(stringr)
library(readr)
library(here)

## here() starts at C:/Users/morul/School/3rd
Year/BIN381/BIN381_PROJECT/BIN381_PROJECT

library(ggplot2)
```

#Load Dataset

```
t_df <- read_csv(here("data","raw","toilet-facilities_national_zaf.csv"))

## Rows: 47 Columns: 29
## ── Column specification
────────────────────────────────────────────────────────

## Delimiter: ","
## chr (17): ISO3, DataId, Indicator, Value, Precision, DHS_CountryCode,
Countr...
## dbl  (8): IndicatorOrder, CharacteristicId, CharacteristicOrder, IsTotal,
Is...
## lgl  (4): RegionId, CILow, CIHigh, LevelRank
##
## ℹ Use `spec()` to retrieve the full column specification for this data.
## ℹ Specify the column types or set `show_col_types = FALSE` to quiet this
message.
```

#Display Dataset content

```
head(t_df)

## # A tibble: 6 × 29
##   ISO3   DataId Indicator Value Precision DHS_CountryCode CountryName
```

```
SurveyYear
##   <chr>  <chr>  <chr>     <chr> <chr>     <chr>           <chr>
<chr>
## 1 #coun… #meta… #indicat… #ind… #indicat… <NA>            #country+n…
#date+year
## 2 ZAF    795762 Househol… 50.1  1         ZA              South Afri… 1998
## 3 ZAF    795768 Househol… 38.3  1         ZA              South Afri… 1998
## 4 ZAF    795760 Househol… 31.2  1         ZA              South Afri… 1998
## 5 ZAF    795764 Househol… 6     1         ZA              South Afri… 1998
## 6 ZAF    795765 Househol… 11.6  1         ZA              South Afri… 1998
## # ℹ 21 more variables: SurveyId <chr>, IndicatorId <chr>, IndicatorOrder
<dbl>,
## #   IndicatorType <chr>, CharacteristicId <dbl>, CharacteristicOrder
<dbl>,
## #   CharacteristicCategory <chr>, CharacteristicLabel <chr>,
## #   ByVariableId <chr>, ByVariableLabel <chr>, IsTotal <dbl>,
## #   IsPreferred <dbl>, SDRID <chr>, RegionId <lgl>, SurveyYearLabel <dbl>,
## #   SurveyType <chr>, DenominatorWeighted <dbl>, DenominatorUnweighted
<dbl>,
## #   CILow <lgl>, CIHigh <lgl>, LevelRank <lgl>
```

#Remove the first row(meta data)

```
t_df <- t_df[-1, ]
```

#dimensions

```
dim(t_df)

## [1] 46 29
```

#Inspect Duplicated rows

```
dup_check <- t_df %>%
  group_by(Indicator, SurveyYear, CharacteristicId, Value) %>%
  filter(n() > 1)

dup_check

## # A tibble: 0 × 29
## # Groups:   Indicator, SurveyYear, CharacteristicId, Value [0]
## # ℹ 29 variables: ISO3 <chr>, DataId <chr>, Indicator <chr>, Value <chr>,
## #   Precision <chr>, DHS_CountryCode <chr>, CountryName <chr>,
## #   SurveyYear <chr>, SurveyId <chr>, IndicatorId <chr>, IndicatorOrder
<dbl>,
## #   IndicatorType <chr>, CharacteristicId <dbl>, CharacteristicOrder
<dbl>,
## #   CharacteristicCategory <chr>, CharacteristicLabel <chr>,
## #   ByVariableId <chr>, ByVariableLabel <chr>, IsTotal <dbl>,
## #   IsPreferred <dbl>, SDRID <chr>, RegionId <lgl>, SurveyYearLabel <dbl>,
…
```

#Percentage Missing Values

```
data.frame(
  Column = names(t_df),
  Missing_Percentage = paste0(round(colMeans(is.na(t_df)) * 100, 2), "%")
  )
```

```
##                    Column Missing_Percentage
## 1                    ISO3                 0%
## 2                  DataId                 0%
## 3               Indicator                 0%
## 4                   Value                 0%
## 5               Precision                 0%
## 6         DHS_CountryCode                 0%
## 7             CountryName                 0%
## 8              SurveyYear                 0%
## 9                SurveyId                 0%
## 10            IndicatorId                 0%
## 11         IndicatorOrder                 0%
## 12          IndicatorType                 0%
## 13       CharacteristicId                 0%
## 14    CharacteristicOrder                 0%
## 15 CharacteristicCategory                 0%
## 16    CharacteristicLabel                 0%
## 17           ByVariableId                 0%
## 18        ByVariableLabel               100%
## 19                IsTotal                 0%
## 20            IsPreferred                 0%
## 21                  SDRID                 0%
## 22               RegionId               100%
## 23        SurveyYearLabel                 0%
## 24             SurveyType                 0%
## 25    DenominatorWeighted               8.7%
## 26  DenominatorUnweighted               8.7%
## 27                  CILow               100%
## 28                 CIHigh               100%
## 29              LevelRank               100%
```

```
data.frame(
  Column = names(t_df),
  Missing_Data = paste0(colSums(is.na(t_df)))
  )
```

```
##                    Column Missing_Data
## 1                    ISO3            0
## 2                  DataId            0
## 3               Indicator            0
## 4                   Value            0
## 5               Precision            0
## 6         DHS_CountryCode            0
## 7             CountryName            0
## 8               SurveyYear            0
## 9                 SurveyId            0
## 10             IndicatorId            0
## 11          IndicatorOrder            0
## 12           IndicatorType            0
## 13        CharacteristicId            0
## 14     CharacteristicOrder            0
## 15  CharacteristicCategory            0
## 16     CharacteristicLabel            0
## 17            ByVariableId            0
## 18         ByVariableLabel           46
## 19                 IsTotal            0
## 20             IsPreferred            0
## 21                   SDRID            0
## 22                RegionId           46
## 23         SurveyYearLabel            0
## 24              SurveyType            0
## 25     DenominatorWeighted            4
## 26   DenominatorUnweighted            4
## 27                   CILow           46
## 28                  CIHigh           46
## 29               LevelRank           46
```

#check data types

```
data.frame(
  Column = names(t_df),
  paste0(sapply(t_df, typeof))
)
```

```
##                      Column paste0.sapply.t_df..typeof..
## 1                      ISO3                    character
## 2                    DataId                    character
## 3                 Indicator                    character
## 4                     Value                    character
## 5                 Precision                    character
## 6           DHS_CountryCode                    character
## 7               CountryName                    character
## 8                SurveyYear                    character
## 9                  SurveyId                    character
## 10              IndicatorId                    character
## 11           IndicatorOrder                       double
## 12            IndicatorType                    character
## 13         CharacteristicId                       double
## 14      CharacteristicOrder                       double
## 15   CharacteristicCategory                    character
## 16      CharacteristicLabel                    character
## 17             ByVariableId                    character
## 18          ByVariableLabel                    character
## 19                  IsTotal                       double
## 20            IsPreferred                              double
## 21                  SDRID                           character
## 22               RegionId                             logical
## 23        SurveyYearLabel                              double
## 24             SurveyType                           character
## 25    DenominatorWeighted                              double
## 26  DenominatorUnweighted                              double
## 27                  CILow                             logical
## 28                 CIHigh                             logical
## 29              LevelRank                             logical
```

#Check The structure of the dataset

```
str(t_df)
```

```
## tibble [46 × 29] (S3: tbl_df/tbl/data.frame)
##  $ ISO3                    : chr [1:46] "ZAF" "ZAF" "ZAF" "ZAF" ...
##  $ DataId                  : chr [1:46] "795762" "795768" "795760" "795764"
...
##  $ Indicator               : chr [1:46] "Households with an improved
sanitation facility" "Households with an unimproved sanitation facility"
"Households with a pit latrine without a slab or an open pit" "Households
with a bucket toilet" ...
##  $ Value                   : chr [1:46] "50.1" "38.3" "31.2" "6" ...
##  $ Precision               : chr [1:46] "1" "1" "1" "1" ...
##  $ DHS_CountryCode         : chr [1:46] "ZA" "ZA" "ZA" "ZA" ...
##  $ CountryName             : chr [1:46] "South Africa" "South Africa" "South
Africa" "South Africa" ...
##  $ SurveyYear              : chr [1:46] "1998" "1998" "1998" "1998" ...
##  $ SurveyId                : chr [1:46] "ZA1998DHS" "ZA1998DHS" "ZA1998DHS"
"ZA1998DHS" ...
##  $ IndicatorId             : chr [1:46] "WS_TLET_H_IMP" "WS_TLET_H_NIM"
"WS_TLET_H_NPT" "WS_TLET_H_NBK" ...
##  $ IndicatorOrder          : num [1:46] 2.5e+08 2.5e+08 2.5e+08 2.5e+08
2.5e+08 ...
##  $ IndicatorType           : chr [1:46] "I" "I" "I" "I" ...
##  $ CharacteristicId        : num [1:46] 1000 1000 1000 1000 1000 1000 1000
1000 1000 1000 ...
##  $ CharacteristicOrder     : num [1:46] 0 0 0 0 0 0 0 0 0 0 ...
##  $ CharacteristicCategory: chr [1:46] "Total" "Total" "Total" "Total" ...
##  $ CharacteristicLabel     : chr [1:46] "Total" "Total" "Total" "Total" ...
##  $ ByVariableId            : chr [1:46] "0" "0" "0" "0" ...
##  $ ByVariableLabel         : chr [1:46] NA NA NA NA ...
##  $ IsTotal                 : num [1:46] 1 1 1 1 1 1 1 1 1 1 ...
##  $ IsPreferred             : num [1:46] 1 1 1 1 1 1 1 1 1 1 ...
##  $ SDRID                   : chr [1:46] "WSTLETHIMP" "WSTLETHNIM"
"WSTLETHNPT" "WSTLETHNBK" ...
##  $ RegionId                : logi [1:46] NA NA NA NA NA NA ...
##  $ SurveyYearLabel         : num [1:46] 1998 1998 1998 1998 1998 ...
##  $ SurveyType              : chr [1:46] "DHS" "DHS" "DHS" "DHS" ...
##  $ DenominatorWeighted   : num [1:46] 12247 12247 12247 12247 12247 ...
##  $ DenominatorUnweighted : num [1:46] 12247 12247 12247 12247 12247 ...
##  $ CILow                 : logi [1:46] NA NA NA NA NA NA ...
##  $ CIHigh                : logi [1:46] NA NA NA NA NA NA ...
##  $ LevelRank             : logi [1:46] NA NA NA NA NA NA ...
```

#Convert Data Types

```
t_df <- t_df %>%
  mutate(
        Value = as.numeric(Value),
    Precision = as.numeric(Precision),
    SurveyYear = as.integer(SurveyYear),
    IndicatorOrder = as.integer(IndicatorOrder),
    CharacteristicId = as.integer(CharacteristicId),
    CharacteristicOrder = as.integer(CharacteristicOrder),
    IsTotal = as.logical(as.integer(IsTotal)),
    IsPreferred = as.logical(as.integer(IsPreferred)),
    SurveyYearLabel = as.integer(SurveyYearLabel),
    DenominatorWeighted = as.numeric(DenominatorWeighted),
    DenominatorUnweighted = as.numeric(DenominatorUnweighted),
  )
```

#check for unique values

```
library(dplyr)
library(purrr)

# Summary table: column name, number of unique values, sample of unique
values
n_sample <- 3

summary_tbl <- t_df %>%
  map_df(~ tibble(
    n_unique = n_distinct(.),
    sample_values = paste(head(unique(.), n_sample), collapse = ", ")
  ), .id = "column")


summary_tbl

## # A tibble: 29 × 3
##    column          n_unique sample_values
##    <chr>              <int> <chr>
##  1 ISO3                   1 ZAF
##  2 DataId                46 795762, 795768, 795760
##  3 Indicator             32 Households with an improved sanitation
facility, Ho…
##  4 Value                 37 50.1, 38.3, 31.2
##  5 Precision              2 1, 0
##  6 DHS_CountryCode        1 ZA
##  7 CountryName            1 South Africa
##  8 SurveyYear             2 1998, 2016
##  9 SurveyId               2 ZA1998DHS, ZA2016DHS
## 10 IndicatorId           32 WS_TLET_H_IMP, WS_TLET_H_NIM, WS_TLET_H_NPT
## # ℹ 19 more rows
```

#Drop the countries only onw unqiue value: reason, there is no useful information - county is also always za

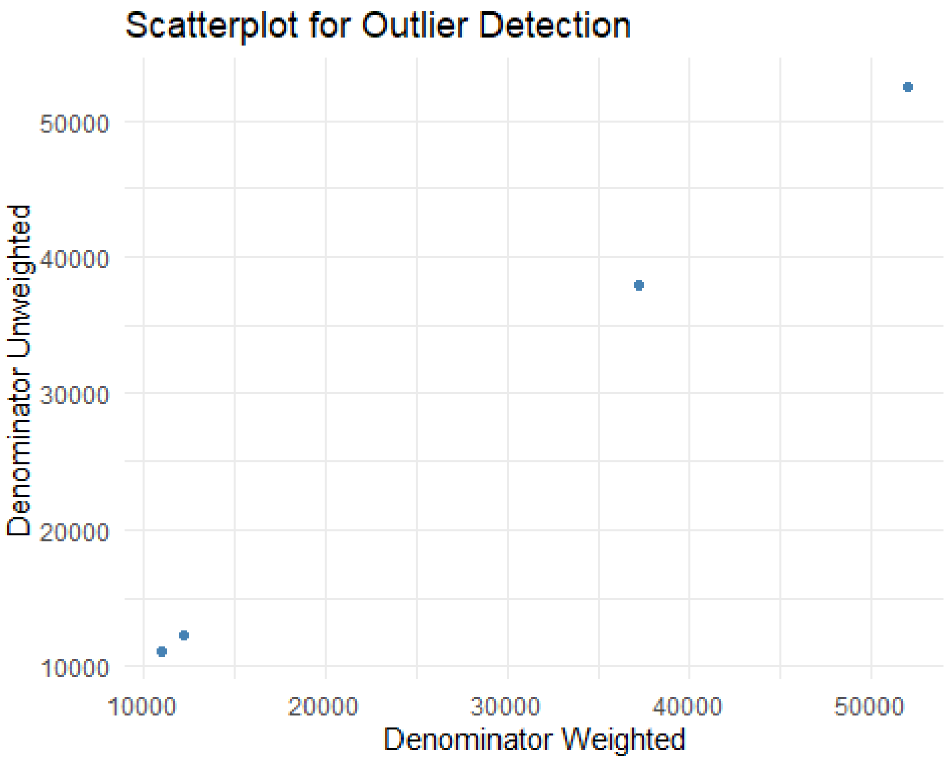
```
t_df <- t_df %>%

 select(
     -ISO3,
    -DHS_CountryCode,
    -CountryName,
    -SurveyId,
    -ByVariableId,
    -ByVariableLabel,
    -IsTotal,
    -RegionId,
    -SurveyYearLabel,
    -SurveyType,
    -CharacteristicOrder
  )
```

#Assumed pattern, the missing values can be filled with the previous non missing value in the opposite attribute

```
library(dplyr)
library(tidyr)

imm_df <- t_df %>%
  fill(DenominatorWeighted, DenominatorUnweighted, .direction = "down")

t_df[
       c("DenominatorWeighted", "DenominatorUnweighted")]

## # A tibble: 46 × 2
##    DenominatorWeighted DenominatorUnweighted
##                  <dbl>                 <dbl>
##  1               12247                 12247
##  2               12247                 12247
##  3               12247                 12247
##  4               12247                 12247
##  5               12247                 12247
##  6               12247                 12247
##  7               12247                 12247
##  8                  NA                 12247
##  9               12247                    NA
## 10               52007                  52465
## # ℹ 36 more rows
```

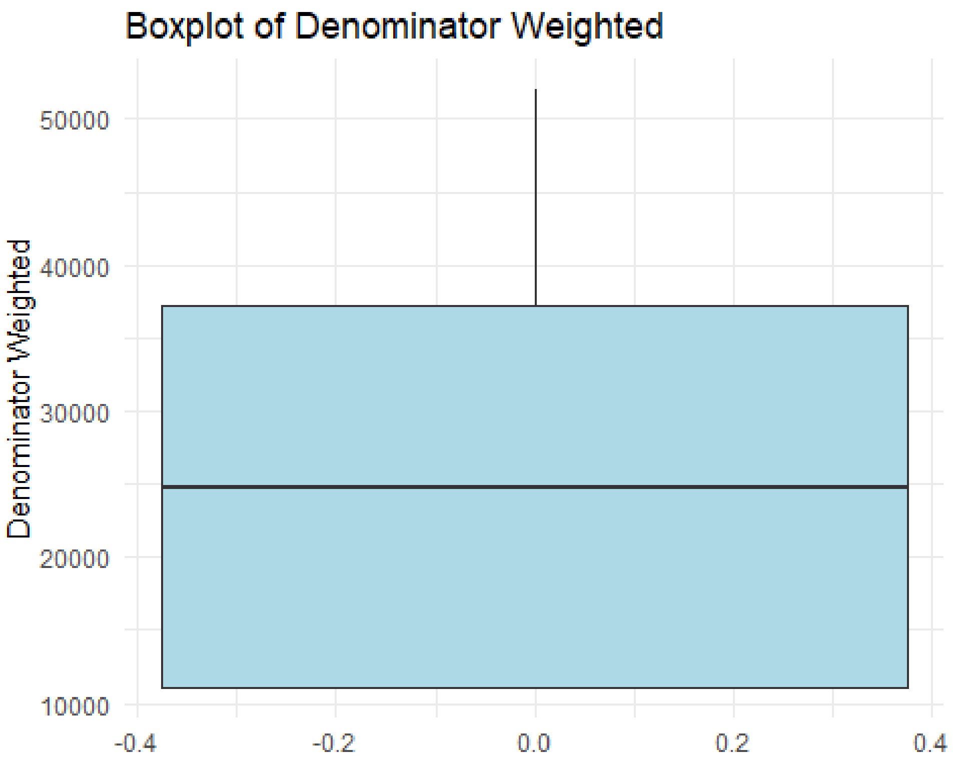
```
ggplot(t_df, aes(x = DenominatorWeighted, y = DenominatorUnweighted)) +
  geom_point(alpha = 0.6, color = "steelblue") +
  labs(title = "Scatterplot for Outlier Detection",
       x = "Denominator Weighted",
       y = "Denominator Unweighted") +
  theme_minimal()
```

```
## Warning: Removed 8 rows containing missing values or values outside the
scale range
## (`geom_point()`).
```



```
ggplot(t_df, aes(y = DenominatorWeighted)) +
  geom_boxplot(fill = "lightblue", outlier.color = "red", outlier.shape = 16)
+
  labs(title = "Boxplot of Denominator Weighted",
       y = "Denominator Weighted") +
  theme_minimal()
```

```
## Warning: Removed 4 rows containing non-finite outside the scale range
## (`stat_boxplot()`).
```



```
dim(t_df)

## [1] 46 18
```

#Outlier Handling

```
# Calculate IQR boundaries
Q1_w <- quantile(t_df$DenominatorWeighted, 0.25, na.rm = TRUE)
Q3_w <- quantile(t_df$DenominatorWeighted, 0.75, na.rm = TRUE)
IQR_w <- Q3_w - Q1_w
lower_w <- Q1_w - 1.5 * IQR_w
upper_w <- Q3_w + 1.5 * IQR_w

Q1_uw <- quantile(t_df$DenominatorUnweighted, 0.25, na.rm = TRUE)
Q3_uw <- quantile(t_df$DenominatorUnweighted, 0.75, na.rm = TRUE)
IQR_uw <- Q3_uw - Q1_uw
lower_uw <- Q1_uw - 1.5 * IQR_uw
upper_uw <- Q3_uw + 1.5 * IQR_uw

# Cap values to the IQR limits
t_df <- t_df %>%
  mutate(
    DenominatorWeighted = pmin(pmax(DenominatorWeighted, lower_w), upper_w),
    DenominatorUnweighted = pmin(pmax(DenominatorUnweighted, lower_uw),
upper_uw)
  )
```

#save cleaned data

```
write_csv(t_df, here("data","processed", "toilet_cleaned.csv"))
```