08\_IYCF

#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

icy\_df <- read\_csv(here("data", "raw", "iycf\_national\_zaf.csv"))

## Rows: 23 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(icy\_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 795971 Children… 87.4 1 ZA South Afri… 1998   
## 3 ZAF 795973 Children… 38.9 1 ZA South Afri… 1998   
## 4 ZAF 621666 Children… 6.9 1 ZA South Afri… 1998   
## 5 ZAF 621667 Children… 6.3 1 ZA South Afri… 1998   
## 6 ZAF 621670 Children… 40.9 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)

icy\_df <- icy\_df[-1, ]

#dimensions

dim(icy\_df)

## [1] 22 29

# Inspect Duplicated rows

dup\_check <- icy\_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>, …

icy\_df <- icy\_df %>%  
 distinct(Indicator, SurveyYear, CharacteristicId, Value, .keep\_all = TRUE)

# Missing Values

# 1. Remove completely empty columns  
icy\_df <- icy\_df %>% select(where(~!all(is.na(.))))  
  
# 2. Impute numeric columns with median  
num\_cols <- icy\_df %>% select(where(is.numeric)) %>% names()  
icy\_df <- icy\_df %>%  
 mutate(across(all\_of(num\_cols), ~ ifelse(is.na(.), median(., na.rm = TRUE), .)))  
  
# 3. Impute character/categorical columns with mode  
cat\_cols <- icy\_df %>% select(where(is.character)) %>% names()  
get\_mode <- function(x) {  
 ux <- na.omit(x)  
 if(length(ux) == 0) return(NA\_character\_)  
 names(sort(table(ux), decreasing = TRUE))[1]  
}  
icy\_df <- icy\_df %>%  
 mutate(across(all\_of(cat\_cols), ~ ifelse(is.na(.), get\_mode(.), .)))  
  
# 4. Summary after handling missing values  
missing\_summary <- data.frame(  
 Column = names(icy\_df),  
 Missing\_Percentage = paste0(round(colMeans(is.na(icy\_df)) \* 100, 2), "%"),  
 Missing\_Count = colSums(is.na(icy\_df))  
)  
  
cat("Total remaining NAs:", sum(is.na(icy\_df)), "\n")

## Total remaining NAs: 0

cat("Missing value summary per column:\n")

## Missing value summary per column:

print(missing\_summary)

## Column Missing\_Percentage Missing\_Count  
## ISO3 ISO3 0% 0  
## DataId DataId 0% 0  
## Indicator Indicator 0% 0  
## Value Value 0% 0  
## Precision Precision 0% 0  
## DHS\_CountryCode DHS\_CountryCode 0% 0  
## CountryName CountryName 0% 0  
## SurveyYear SurveyYear 0% 0  
## SurveyId SurveyId 0% 0  
## IndicatorId IndicatorId 0% 0  
## IndicatorOrder IndicatorOrder 0% 0  
## IndicatorType IndicatorType 0% 0  
## CharacteristicId CharacteristicId 0% 0  
## CharacteristicOrder CharacteristicOrder 0% 0  
## CharacteristicCategory CharacteristicCategory 0% 0  
## CharacteristicLabel CharacteristicLabel 0% 0  
## ByVariableId ByVariableId 0% 0  
## IsTotal IsTotal 0% 0  
## IsPreferred IsPreferred 0% 0  
## SDRID SDRID 0% 0  
## SurveyYearLabel SurveyYearLabel 0% 0  
## SurveyType SurveyType 0% 0  
## DenominatorWeighted DenominatorWeighted 0% 0  
## DenominatorUnweighted DenominatorUnweighted 0% 0

data.frame(  
 Column = names(icy\_df),  
 Missing\_Data = paste0(colSums(is.na(icy\_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 IsTotal 0  
## 19 IsPreferred 0  
## 20 SDRID 0  
## 21 SurveyYearLabel 0  
## 22 SurveyType 0  
## 23 DenominatorWeighted 0  
## 24 DenominatorUnweighted 0

#check data types

data.frame(  
 Column = names(icy\_df),  
 paste0(sapply(icy\_df, typeof))  
)

## Column paste0.sapply.icy\_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 IsTotal double  
## 19 IsPreferred double  
## 20 SDRID character  
## 21 SurveyYearLabel double  
## 22 SurveyType character  
## 23 DenominatorWeighted double  
## 24 DenominatorUnweighted double

#Check The structure of the dataset

str(icy\_df)

## tibble [22 × 24] (S3: tbl\_df/tbl/data.frame)  
## $ ISO3 : chr [1:22] "ZAF" "ZAF" "ZAF" "ZAF" ...  
## $ DataId : chr [1:22] "795971" "795973" "621666" "621667" ...  
## $ Indicator : chr [1:22] "Children ever breastfed" "Children who started breastfeeding within 1 hour of birth" "Children exclusively breastfed" "Children breastfeeding and consuming plain water only" ...  
## $ Value : chr [1:22] "87.4" "38.9" "6.9" "6.3" ...  
## $ Precision : chr [1:22] "1" "1" "1" "1" ...  
## $ DHS\_CountryCode : chr [1:22] "ZA" "ZA" "ZA" "ZA" ...  
## $ CountryName : chr [1:22] "South Africa" "South Africa" "South Africa" "South Africa" ...  
## $ SurveyYear : chr [1:22] "1998" "1998" "1998" "1998" ...  
## $ SurveyId : chr [1:22] "ZA1998DHS" "ZA1998DHS" "ZA1998DHS" "ZA1998DHS" ...  
## $ IndicatorId : chr [1:22] "CN\_BRFI\_C\_EVR" "CN\_BRFI\_C\_1HR" "CN\_BRFS\_C\_EXB" "CN\_BRFS\_C\_WAT" ...  
## $ IndicatorOrder : num [1:22] 1.04e+08 1.04e+08 1.04e+08 1.04e+08 1.04e+08 ...  
## $ IndicatorType : chr [1:22] "I" "I" "I" "I" ...  
## $ CharacteristicId : num [1:22] 1000 1000 295001 295001 295001 ...  
## $ CharacteristicOrder : num [1:22] 0 0 21001 21001 21001 ...  
## $ CharacteristicCategory: chr [1:22] "Total" "Total" "Age in months (other groupings)" "Age in months (other groupings)" ...  
## $ CharacteristicLabel : chr [1:22] "Total" "Total" "0-5" "0-5" ...  
## $ ByVariableId : chr [1:22] "0" "0" "0" "0" ...  
## $ IsTotal : num [1:22] 1 1 1 1 1 1 1 1 1 1 ...  
## $ IsPreferred : num [1:22] 1 1 1 1 1 1 1 1 1 1 ...  
## $ SDRID : chr [1:22] "CNBRFICEVR" "CNBRFIC1HR" "CNBRFSCEXB" "CNBRFSCWAT" ...  
## $ SurveyYearLabel : num [1:22] 1998 1998 1998 1998 1998 ...  
## $ SurveyType : chr [1:22] "DHS" "DHS" "DHS" "DHS" ...  
## $ DenominatorWeighted : num [1:22] 2010 2010 499 499 499 ...  
## $ DenominatorUnweighted : num [1:22] 2041 2041 505 505 505 ...

#Convert Data Types

icy\_df <- icy\_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 <- icy\_df %>%  
 map\_df(~ tibble(  
 n\_unique = n\_distinct(.),  
 sample\_values = paste(head(unique(.), n\_sample), collapse = ", ")  
 ), .id = "column")  
  
  
summary\_tbl

## # A tibble: 24 × 3  
## column n\_unique sample\_values   
## <chr> <int> <chr>   
## 1 ISO3 1 ZAF   
## 2 DataId 22 795971, 795973, 621666   
## 3 Indicator 14 Children ever breastfed, Children who started breas…  
## 4 Value 22 87.4, 38.9, 6.9   
## 5 Precision 1 1   
## 6 DHS\_CountryCode 1 ZA   
## 7 CountryName 1 South Africa   
## 8 SurveyYear 2 1998, 2016   
## 9 SurveyId 2 ZA1998DHS, ZA2016DHS   
## 10 IndicatorId 14 CN\_BRFI\_C\_EVR, CN\_BRFI\_C\_1HR, CN\_BRFS\_C\_EXB   
## # ℹ 14 more rows

# Drop the Redundant Columns

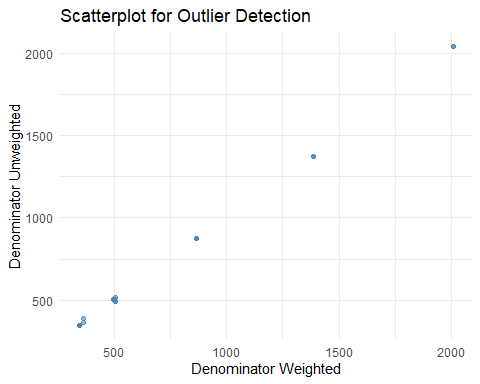
icy\_df <- icy\_df %>%  
 select(  
 -ISO3,   
 -DHS\_CountryCode,   
 -CountryName,   
 -SurveyId,  
 -ByVariableId,   
   
 -IsTotal,  
   
 -SurveyYearLabel,   
 -SurveyType,  
 -CharacteristicOrder  
   
 )

* Columns removed because they were constant, redundant, or not analytically useful:
* ISO3, DHS\_CountryCode, CountryName, SurveyId, ByVariableId, IsTotal, SurveyYearLabel, SurveyType, CharacteristicOrder
* These columns either contained a single value, duplicated information, or survey metadata that does not impact analysis. # Assumed pattern, the missing values can be filled with the previous non missing value in the opposite attribute

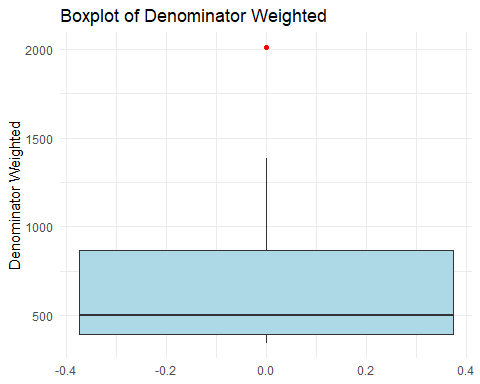
library(dplyr)  
library(tidyr)  
  
icy\_df <- icy\_df %>%  
 fill(DenominatorWeighted, DenominatorUnweighted, .direction = "up")  
  
icy\_df[  
 c("DataId","DenominatorWeighted", "DenominatorUnweighted")]

## # A tibble: 22 × 3  
## DataId DenominatorWeighted DenominatorUnweighted  
## <chr> <dbl> <dbl>  
## 1 795971 2010 2041  
## 2 795973 2010 2041  
## 3 621666 499 505  
## 4 621667 499 505  
## 5 621670 499 505  
## 6 621664 499 505  
## 7 621143 505 514  
## 8 796663 502. 505  
## 9 719834 1386 1376  
## 10 719833 1386 1376  
## # ℹ 12 more rows

ggplot(icy\_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()



ggplot(icy\_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()



dim(icy\_df)

## [1] 22 15

#Outlier Handling

# Calculate IQR boundaries  
Q1\_w <- quantile(icy\_df$DenominatorWeighted, 0.25, na.rm = TRUE)  
Q3\_w <- quantile(icy\_df$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(icy\_df$DenominatorUnweighted, 0.25, na.rm = TRUE)  
Q3\_uw <- quantile(icy\_df$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  
icy\_df <- icy\_df %>%  
 mutate(  
 DenominatorWeighted = pmin(pmax(DenominatorWeighted, lower\_w), upper\_w),  
 DenominatorUnweighted = pmin(pmax(DenominatorUnweighted, lower\_uw), upper\_uw)  
 )

Problem: DenominatorWeighted and DenominatorUnweighted contained extreme values that could skew analyses.

Solution: IQR-based capping:

Calculate bounds:

* Lower bound = Q1 – 1.5 × IQR
* Upper bound = Q3 + 1.5 × IQR
* Cap extreme values:
* Values below lower bound → set to lower bound
* Values above upper bound → set to upper bound
* Visualize: Scatterplots and boxplots were used to confirm the effect of outlier capping.
* Outcome: Extreme values were mitigated while retaining all rows, improving robustness for analysis. #save cleaned data

write\_csv(icy\_df, here("data","processed", "iycf\_cleaned.csv"))