05_DHS_Quickstats

DHS QuickStats (National, South Africa)

Load Libraries

```
# Data manipulation
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(readr)
library(here)

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

library(purrr)

# Visualization and summaries
library(ggplot2)
library(skimr)
library(visdat)
```

Load the DHS QuickStats dataset

```
dhs_df <- read_csv(here("data", "raw", "dhs-quickstats_national_zaf.csv"))

## Rows: 53 Columns: 29
## ── Column specification
────────────────────────────────────────────────────────

## Delimiter: ","
## chr (17): ISO3, DataId, Indicator, Value, Precision, DHS_CountryCode,
Countr...
## dbl (10): IndicatorOrder, CharacteristicId, CharacteristicOrder, IsTotal,
Is...
## lgl  (2): RegionId, 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.
```

```r
# Remove first row if it contains metadata
dhs_df <- dhs_df[-1, ]

# Reset row names
rownames(dhs_df) <- NULL

cat("DHS QuickStats dataset loaded successfully.\n")
```

```
## DHS QuickStats dataset loaded successfully.
```

```r
cat("Dimensions:", dim(dhs_df), "\n")
```

```
## Dimensions: 52 29
```

Initial Data Assessment

```r
# Quick glimpse
glimpse(dhs_df)
```

```
## Rows: 52
## Columns: 29
## $ ISO3                   <chr> "ZAF", "ZAF", "ZAF", "ZAF", "ZAF", "ZAF",
"ZAF"…
## $ DataId                 <chr> "796527", "795692", "795693", "795515",
"795357…
## $ Indicator              <chr> "Total fertility rate 15-49", "Married
women cu…
## $ Value                  <chr> "2.9", "56.3", "55.1", "16.5", "75.7",
"24.2", …
## $ Precision              <chr> "1", "1", "1", "1", "1", "1", "1", "0",
"0", "0…
## $ DHS_CountryCode        <chr> "ZA", "ZA", "ZA", "ZA", "ZA", "ZA", "ZA",
"ZA",…
## $ CountryName            <chr> "South Africa", "South Africa", "South
Africa",…
## $ SurveyYear             <chr> "1998", "1998", "1998", "1998", "1998",
"1998",…
## $ SurveyId               <chr> "ZA1998DHS", "ZA1998DHS", "ZA1998DHS",
"ZA1998D…
## $ IndicatorId            <chr> "FE_FRTR_W_TFR", "FP_CUSM_W_ANY",
"FP_CUSM_W_MO…
## $ IndicatorOrder         <dbl> 11763080, 32633010, 32633020, 32933030,
3293315…
## $ IndicatorType          <chr> "I", "I", "I", "I", "I", "I", "I", "I",
"I", "I…
## $ CharacteristicId       <dbl> 1000, 1000, 1000, 1000, 1000, 1000, 1000,
1000,…
## $ CharacteristicOrder    <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
```

```
0, 0,…
## $ CharacteristicCategory <chr> "Total", "Total", "Total", "Total",
"Total", "T…
## $ CharacteristicLabel    <chr> "Total", "Total", "Total", "Total",
"Total", "T…
## $ ByVariableId           <chr> "0", "0", "0", "0", "0", "0", "0", "14001",
"14…
## $ ByVariableLabel        <chr> NA, NA, NA, NA, NA, NA, NA, "Five years
precedi…
## $ IsTotal                <dbl> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
1, 1,…
## $ IsPreferred            <dbl> 1, 1, 1, 1, 1, 1, 1, 1, 0, 1, 0, 1, 0, 0,
1, 1,…
## $ SDRID                  <chr> "FEFRTRWTFR", "FPCUSMWANY", "FPCUSMWMOD",
"FPNA…
## $ RegionId               <lgl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,
NA,…
## $ SurveyYearLabel        <dbl> 1998, 1998, 1998, 1998, 1998, 1998, 1998,
1998,…
## $ SurveyType             <chr> "DHS", "DHS", "DHS", "DHS", "DHS", "DHS",
"DHS"…
## $ DenominatorWeighted    <dbl> NA, 5077, 5077, 5077, 3695, NA, NA, NA, NA,
NA,…
## $ DenominatorUnweighted  <dbl> NA, 4948, 4948, 4948, 3590, NA, NA, NA, NA,
NA,…
## $ CILow                  <dbl> NA, NA, NA, NA, NA, NA, NA, 38, 37, 50, 50,
77,…
## $ CIHigh                 <dbl> NA, NA, NA, NA, NA, NA, NA, 53, 48, 68, 63,
223…
## $ LevelRank              <lgl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,
NA,…
```

```r
# Summary of missingness
skim(dhs_df)
```

Data summary

Name	dhs_df
Number of rows	52
Number of columns	29

Column type frequency:	
character	17
logical	2
numeric	10

Group variables None

Variable type: character

skim_variable	n_missing	complete_rate	min	max	empty	n_unique	whitespace
ISO3	0	1.00	3	3	0	1	0
DataId	0	1.00	3	6	0	52	0
Indicator	0	1.00	15	76	0	27	0
Value	0	1.00	2	4	0	51	0
Precision	0	1.00	1	1	0	2	0
DHS_CountryCode	0	1.00	2	2	0	1	0
CountryName	0	1.00	12	12	0	1	0
SurveyYear	0	1.00	4	4	0	2	0
SurveyId	0	1.00	9	9	0	2	0
IndicatorId	0	1.00	13	13	0	27	0
IndicatorType	0	1.00	1	1	0	1	0
CharacteristicCategory	0	1.00	5	11	0	2	0
CharacteristicLabel	0	1.00	5	11	0	2	0
ByVariableId	0	1.00	1	6	0	6	0
ByVariableLabel	33	0.37	12	32	0	5	0
SDRID	0	1.00	10	10	0	27	0
SurveyType	0	1.00	3	3	0	1	0

Variable type: logical

skim_variable	n_missing	complete_rate	mean	count
RegionId	52	0	NaN	:
LevelRank	52	0	NaN	:

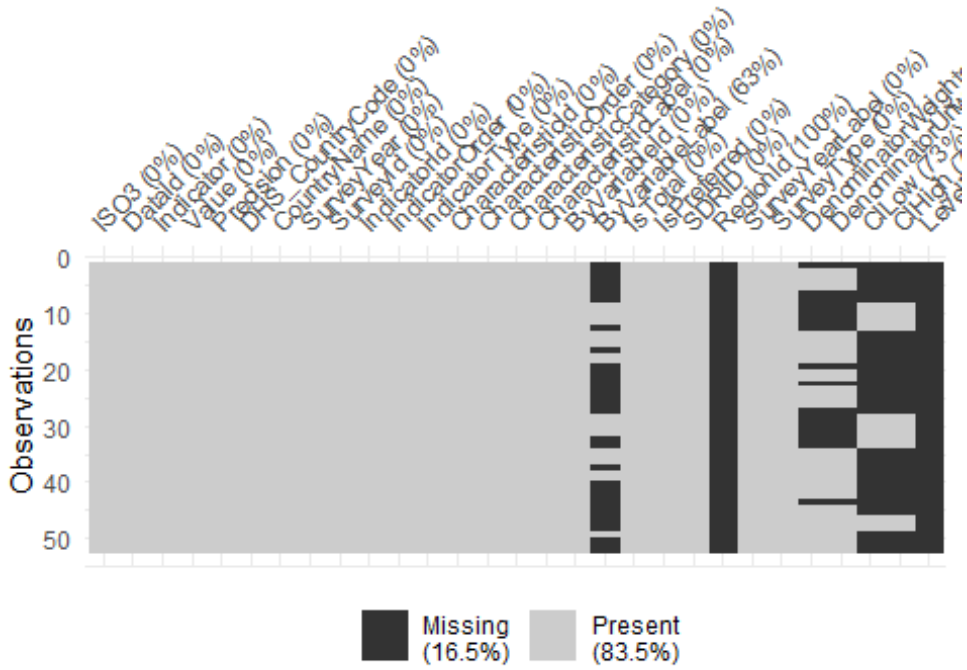
Variable type: numeric

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100	hist
IndicatorOrder	0	1.00	93178551.54	61143758.08	11763080.0	60330295.00	83566070.0	104261072.50	260321010	

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100	hist
CharacteristicId	0	1.00	2557.69	3438.04	1000.0	1000.00	1000.0	1000.00	10000	
CharacteristicOrder	0	1.00	1730.77	3820.05	0.0	0.00	0.0	0.00	10000	
IsTotal	0	1.00	1.00	0.00	1.0	1.00	1.0	1.00	1	
IsPreferred	0	1.00	0.79	0.41	0.0	1.00	1.0	1.00	1	
SurveyYearLabel	0	1.00	2008.73	8.92	1998.0	1998.00	2016.0	2016.00	2016	
DenominatorWeighted	18	0.65	3832.21	3328.81	246.0	1414.50	3050.0	5055.75	12247	
DenominatorUnweighted	18	0.65	3999.03	3682.57	256.0	1470.75	2841.0	4948.00	12247	
CILow	38	0.27	67.67	79.98	11.5	27.75	37.5	50.00	270	

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100	hist
CIHigh	38	0.27	160.54	258.61	17.3	45.00	52.0	66.75	802	▅▇▁▁▁▅

```r
# Visualize missing values
vis_miss(dhs_df)
```




```r
# Standardize column names
dhs_df <- dhs_df %>% janitor::clean_names()
colnames(dhs_df)

##  [1] "iso3"                    "data_id"
##  [3] "indicator"               "value"
##  [5] "precision"               "dhs_country_code"
##  [7] "country_name"            "survey_year"
##  [9] "survey_id"               "indicator_id"
```

```
## [11] "indicator_order"         "indicator_type"
## [13] "characteristic_id"       "characteristic_order"
## [15] "characteristic_category" "characteristic_label"
## [17] "by_variable_id"          "by_variable_label"
## [19] "is_total"                "is_preferred"
## [21] "sdrid"                   "region_id"
## [23] "survey_year_label"       "survey_type"
## [25] "denominator_weighted"    "denominator_unweighted"
## [27] "ci_low"                  "ci_high"
## [29] "level_rank"
```

Glimpse of the Dataset

A quick look at the first few rows and column types revealed:

- 17 character columns (e.g., ISO3, Indicator, CountryName)
- 10 numeric columns (e.g., Value, Precision)
- 2 logical columns (RegionId, LevelRank)

Summary of Missing Values

The skimr package summarized missingness:

- Columns such as ByVariableLabel, DenominatorWeighted, CILow, and CIHigh contained missing values.
- These columns would require imputation or handling in later steps.

Visualization

vis_miss() was used to create a visual map of missing data, which helped identify columns with high missingness at a glance.

Why this step matters: Understanding missing data is crucial for selecting appropriate imputation methods or deciding if columns should be dropped.

Rename Columns Meaningfully

```r
# Replace generic col_1, col_2, ... with actual names
colnames(dhs_df) <- c(
  "iso3", "data_id", "indicator", "value", "precision",
  "dhs_country_code", "country_name", "survey_year", "survey_id",
  "indicator_id", "indicator_order", "indicator_type", "characteristic_id",
  "characteristic_order", "characteristic_category", "characteristic_label",
  "by_variable_id", "by_variable_label", "is_total", "is_preferred",
  "sdrid", "region_id", "survey_year_label", "survey_type",
  "denominator_weighted", "denominator_unweighted", "ci_low", "ci_high",
  "level_rank"
```

```r
)

cat("Columns renamed to meaningful names.\n")

## Columns renamed to meaningful names.

colnames(dhs_df)

##  [1] "iso3"                    "data_id"
##  [3] "indicator"               "value"
##  [5] "precision"               "dhs_country_code"
##  [7] "country_name"            "survey_year"
##  [9] "survey_id"               "indicator_id"
## [11] "indicator_order"         "indicator_type"
## [13] "characteristic_id"       "characteristic_order"
## [15] "characteristic_category" "characteristic_label"
## [17] "by_variable_id"          "by_variable_label"
## [19] "is_total"                "is_preferred"
## [21] "sdrid"                   "region_id"
## [23] "survey_year_label"       "survey_type"
## [25] "denominator_weighted"    "denominator_unweighted"
## [27] "ci_low"                  "ci_high"
## [29] "level_rank"
```

Column names were standardized to snake_case using janitor::clean_names().

Additionally, descriptive names were assigned to generic column names (e.g., col_1, col_2) to improve readability.

Example:

- ISO3 → iso3
- DataId → data_id
- Value → value
- Precision → precision
- Benefit: This ensures consistency across analysis scripts and improves interpretability for readers.

Remove Duplicates

```r
# Check for exact duplicates
exact_dups <- sum(duplicated(dhs_df))
cat("Exact duplicate rows:", exact_dups, "\n")

## Exact duplicate rows: 0

# Remove duplicates, keeping first occurrence
dhs_df <- dhs_df %>%
```

```
  distinct(indicator, survey_year, characteristic_id, value, .keep_all =
TRUE)

cat("Dimensions after duplicate removal:", dim(dhs_df), "\n")

## Dimensions after duplicate removal: 52 29
```

Remove Redundant & Empty Columns

```
all_na_cols <- dhs_df %>%
  summarise(across(everything(), ~all(is.na(.)))) %>%
  pivot_longer(everything(), names_to = "column", values_to = "all_na") %>%
  filter(all_na) %>%
  pull(column)

if(length(all_na_cols) > 0) {
  dhs_df <- dhs_df %>% select(-all_of(all_na_cols))
  cat("Removed 100% missing columns:\n")
  print(all_na_cols)
} else {
  cat("No columns were 100% missing.\n")
}

## Removed 100% missing columns:
## [1] "region_id"  "level_rank"
```

Removing Duplicates and Empty Columns

Exact duplicate rows were checked; none were found.

The dataset was then filtered to retain unique combinations of Indicator, Survey Year, Characteristic ID, and Value.

Fully empty columns (region_id and level_rank) were removed.

Convert Data Types Safely

```
# Numeric columns
numeric_cols <- intersect(c("value", "precision", "ci_low", "ci_high"),
colnames(dhs_df))

# Integer columns
integer_cols <- intersect(c("survey_year", "indicator_order",
"characteristic_id",
                            "characteristic_order", "survey_year_label",
"by_variable_id"), colnames(dhs_df))

# Logical columns
logical_cols <- intersect(c("is_total", "is_preferred"), colnames(dhs_df))

# Apply type conversion
```

```r
dhs_df <- dhs_df %>%
  mutate(
    across(all_of(numeric_cols), as.numeric),
    across(all_of(integer_cols), as.integer),
    across(all_of(logical_cols), ~as.logical(as.integer(.)))
  )

str(dhs_df)
```

```
## tibble [52 × 27] (S3: tbl_df/tbl/data.frame)
##  $ iso3                   : chr [1:52] "ZAF" "ZAF" "ZAF" "ZAF" ...
##  $ data_id                : chr [1:52] "796527" "795692" "795693" "795515"
...
##  $ indicator              : chr [1:52] "Total fertility rate 15-49"
"Married women currently using any method of contraception" "Married women
currently using any modern method of contraception" "Unmet need for family
planning" ...
##  $ value                  : num [1:52] 2.9 56.3 55.1 16.5 75.7 24.2 18.4
45 42 59 ...
##  $ precision              : num [1:52] 1 1 1 1 1 1 1 0 0 0 ...
##  $ dhs_country_code       : chr [1:52] "ZA" "ZA" "ZA" "ZA" ...
##  $ country_name           : chr [1:52] "South Africa" "South Africa"
"South Africa" "South Africa" ...
##  $ survey_year            : int [1:52] 1998 1998 1998 1998 1998 1998 1998
1998 1998 1998 ...
##  $ survey_id              : chr [1:52] "ZA1998DHS" "ZA1998DHS" "ZA1998DHS"
"ZA1998DHS" ...
##  $ indicator_id           : chr [1:52] "FE_FRTR_W_TFR" "FP_CUSM_W_ANY"
"FP_CUSM_W_MOD" "FP_NADM_W_UNT" ...
##  $ indicator_order        : int [1:52] 11763080 32633010 32633020 32933030
32933150 41633090 51703090 63206030 63206030 63206050 ...
##  $ indicator_type         : chr [1:52] "I" "I" "I" "I" ...
##  $ characteristic_id      : int [1:52] 1000 1000 1000 1000 1000 1000 1000
1000 1000 1000 ...
##  $ characteristic_order   : int [1:52] 0 0 0 0 0 0 0 0 0 0 ...
##  $ characteristic_category: chr [1:52] "Total" "Total" "Total" "Total" ...
##  $ characteristic_label   : chr [1:52] "Total" "Total" "Total" "Total" ...
##  $ by_variable_id         : int [1:52] 0 0 0 0 0 0 0 14001 14003 14001 ...
##  $ by_variable_label      : chr [1:52] NA NA NA NA ...
##  $ is_total               : logi [1:52] TRUE TRUE TRUE TRUE TRUE TRUE ...
##  $ is_preferred           : logi [1:52] TRUE TRUE TRUE TRUE TRUE TRUE ...
##  $ sdrid                  : chr [1:52] "FEFRTRWTFR" "FPCUSMWANY"
"FPCUSMWMOD" "FPNADMWUNT" ...
##  $ survey_year_label      : int [1:52] 1998 1998 1998 1998 1998 1998 1998
1998 1998 1998 ...
##  $ survey_type            : chr [1:52] "DHS" "DHS" "DHS" "DHS" ...
##  $ denominator_weighted   : num [1:52] NA 5077 5077 5077 3695 ...
##  $ denominator_unweighted : num [1:52] NA 4948 4948 4948 3590 ...
##  $ ci_low                 : num [1:52] NA NA NA NA NA NA NA 38 37 50 ...
##  $ ci_high                : num [1:52] NA NA NA NA NA NA NA 53 48 68 ...
```

Columns were converted to appropriate types:

- Numeric columns: value, precision, ci_low, ci_high
- Integer columns: survey_year, indicator_order, characteristic_id, etc.
- Logical columns: is_total, is_preferred
- Purpose: Correct data types ensure proper calculations, comparisons, and visualizations

Handle Missing Values

```
# Define mode function for categorical imputation
impute_mode <- function(x) {
  ux <- na.omit(x)
  if(length(ux) == 0) return(x)
  rep(names(sort(table(ux), decreasing = TRUE))[1], length(x))
}

# Impute missing values
dhs_df <- dhs_df %>%
  mutate(
    # Numeric → median
    across(where(is.numeric), ~ifelse(is.na(.), median(., na.rm = TRUE), .)),

    # Character → mode
    across(where(is.character), ~ifelse(is.na(.), impute_mode(.), .)),

    # Logical → FALSE
    across(where(is.logical), ~ifelse(is.na(.), FALSE, .))
  )

# Ensure survey_year_label filled
dhs_df <- dhs_df %>%
  mutate(survey_year_label = ifelse(is.na(survey_year_label), survey_year,
survey_year_label))

# Recalculate missing values
missing_summary <- data.frame(
  Column = colnames(dhs_df),
  n_missing = colSums(is.na(dhs_df)),
  total_rows = nrow(dhs_df),
  missing_percent = round(colSums(is.na(dhs_df))/nrow(dhs_df)*100, 2)
)

missing_summary %>% arrange(desc(missing_percent))

##                                          Column n_missing total_rows
## iso3                                       iso3         0         52
```

```
## data_id                                  data_id         0          52
## indicator                              indicator         0          52
## value                                      value         0          52
## precision                              precision         0          52
## dhs_country_code               dhs_country_code         0          52
## country_name                       country_name         0          52
## survey_year                         survey_year         0          52
## survey_id                             survey_id         0          52
## indicator_id                       indicator_id         0          52
## indicator_order                 indicator_order         0          52
## indicator_type                   indicator_type         0          52
## characteristic_id             characteristic_id         0          52
## characteristic_order       characteristic_order         0          52
## characteristic_category characteristic_category         0          52
## characteristic_label       characteristic_label         0          52
## by_variable_id                   by_variable_id         0          52
## by_variable_label             by_variable_label         0          52
## is_total                               is_total         0          52
## is_preferred                       is_preferred         0          52
## sdrid                                     sdrid         0          52
## survey_year_label             survey_year_label         0          52
## survey_type                         survey_type         0          52
## denominator_weighted       denominator_weighted         0          52
## denominator_unweighted   denominator_unweighted         0          52
## ci_low                                   ci_low         0          52
## ci_high                                 ci_high         0          52
##                         missing_percent
## iso3                                  0
## data_id                               0
## indicator                             0
## value                                 0
## precision                             0
## dhs_country_code                      0
## country_name                          0
## survey_year                           0
## survey_id                             0
## indicator_id                          0
## indicator_order                       0
## indicator_type                        0
## characteristic_id                     0
## characteristic_order                  0
## characteristic_category               0
## characteristic_label                  0
## by_variable_id                        0
## by_variable_label                     0
## is_total                              0
## is_preferred                          0
## sdrid                                 0
## survey_year_label                     0
## survey_type                           0
```

```
## denominator_weighted                    0
## denominator_unweighted                  0
## ci_low                                  0
## ci_high                                 0
```

Handling Missing Values

Strategy:

1. Numeric columns: Imputed using the median
2. Character columns: Imputed using the mode (most frequent value)
3. Logical columns: Missing values set to FALSE

Special handling: survey_year_label was filled with survey_year where missing.

Outlier Detection

```r
# Identify potential outliers using IQR
numeric_cols <- intersect(c("value", "precision"), colnames(dhs_df))

for(col in numeric_cols) {
  Q1 <- quantile(dhs_df[[col]], 0.25, na.rm = TRUE)
  Q3 <- quantile(dhs_df[[col]], 0.75, na.rm = TRUE)
  IQR_val <- Q3 - Q1
  lower <- Q1 - 1.5*IQR_val
  upper <- Q3 + 1.5*IQR_val
  dhs_df[[paste0(col, "_outlier_flag")]] <- dhs_df[[col]] < lower |
dhs_df[[col]] > upper
}

# Winsorize 'value' at 1st and 99th percentile
lower_cap <- quantile(dhs_df$value, 0.01, na.rm = TRUE)
upper_cap <- quantile(dhs_df$value, 0.99, na.rm = TRUE)

dhs_df <- dhs_df %>%
  mutate(value = pmax(pmin(value, upper_cap), lower_cap))

summary(dhs_df$value)

##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   0.604  23.450  55.700  69.002  81.525 504.890
```

- Method: Interquartile Range (IQR) to identify extreme values
- Treatment: Values outside 1.5×IQR were flagged, then Winsorized at the 1st and 99th percentiles.

Save Cleaned Data

```r
write_csv(dhs_df, here("data", "processed", "dhs_quickstats_cleaned.csv"))
cat("Cleaned DHS QuickStats dataset saved.\n")

## Cleaned DHS QuickStats dataset saved.
```