Unicef Water

Belicia Rodriguez

5/10/2019

Contents

Introduction

UNICEF collects data on the accessibility and conditions of drinking water, sanitation, and hygiene (WASH) around the world. The organization divides the evaluation for each criteria into different hierarchial evaluations.

Drinking Water

1. *Safely managed drinking water service*: Water from an improved source that is located on premises, available when needed and free from contamination.
2. *Basic drinking water service*: An improved water source within 30 minutes roundtrip
3. *Limited drinking water service*: An improved water source that takes over 30 minutes roundtrip to access
4. *Unimproved water sources*: Water source without improvement
5. *Surface water*: Water taken directly from lakes, rivers, etc

Sanitation

1. *Safely managed sanitation service*: Improved sanitation facilities not shared with other households with excreta safely disposed in situ or removed and treated offsite
2. *Basic sanitation service*: Improved sanitation facilites not shared with other households
3. *Limited sanitation service*: Improved facilities shared with other households
4. *Unimproved sanitation service*: Sanitation without improvement
5. *Open defecation*: Go out in fields, bushes, forests, open bodies of water, etc (Proportion of population using improved sanitation facilities (excluding shared))

Hygiene

1. *Basic*: Handwashing facilities with soap or water
2. *Limited*: Handwashing facilities without water or soap
3. *No facility*: No facilities for handwashing

This investigation specifically outlines the WASH accessibility and conditions of 232 countries in 2000 and 2015. In this analysis, I will be investigating which countries have low drinking water, sanitation, and/or hygiene and showing whether there has been improvement between 2000 to 2015.

I will be using the excel sheet UNICEF uploaded on their WASH overview post in July 2017 to do an analysis on the progession of countries towards improving WASH conditions.

I analyze each evaluation through three country levels: national, urban, and rural.

1. *National*: encompasses the urban, suburban, and rural levels
2. *Urban*: refers to cities in the region and suburban areas
3. *Rural*: everything outside of urban areas

Code Explanation

Although the investigation portion of this report will not include code, this section will breakdown the steps taken to import, clean, and tidy the data and show glimpses of code. This section will not analyze any of the data, but I decided to include this section in order to explain how the data was prepared for analysis. I've allowed for the code to be accessible through a "code" button on the side, but I will explain what each section will discuss.

1. Downloading Data from Website

Explain why download.file() was used to download data from UNICEF's website, and I also breakdown what components of read_excel() were used to set up the data.

2. Tidying Data

Mainly used select(), separate(), gather(), drop_na(), mutate_at() with as.factor(), and mutate() with str_remove() to tidy the four dataframes created from the UNICEF's excel files. I also use the drinking water and sanitation dataframes to create two dataframes containing rates on improved sanitation facilities dataframe and improved water supply.

3. Functions

I created six functions for this report: add_UN_region(), water_supply_graph(), ldc_graph(), unicef_summary_table(), unicef_two_tables(), and retrieve_data(). In general, these functions either added more classifications for the UNICEF dataframes or generated similar-looking graphs and tables to decrease the amount of copying and pasting in the code. retrieve_data() was primarily used to reduce the amount of code written in the r embedded code that I use in the text parts of the reports.

Downloading Data from Website

- download.file()

Reproductibility is an important component of creating an R Markdown report; if I shared my R Markdown document with another person who is using a different computer than mine, I want that person to be able to reproduce my document without difficulty. That is why I used download.file() to import the UNICEF excel file; on UNICEF's website, the excel file has its own website link. I used the link as the main argument in download.files, which then saved the file to my computer in the same folder where my R markdown document is saved; another argument of download.files() allowed me to save the file as "unicef_water.xlsx".

- read_excel()

The UNICEF's excel file contained four sheets; therefore, I used read_excel four times, each time assigning a different sheet to a different R object. I used the col_names argument to rewrite the column names. I suspect there's a way to use regex to fix the columns instead of rewriting the more than 15 column names, but I haven't mastered regex enough to comfortably use them here. I'm also not sure if I can use regex in

read_excel, but I will investigate more on regex at a later time. I also used the na argument to replace "-", which signified an empty cell, to NA.

```
# Downloun unicef web file to computer
download.file("https://data.unicef.org/wp-content/uploads/2015/12/Drinking-Water-Sanitation-Hygiene-Data

# Create drinking_water dataframe
drinking_water <- read_excel("unicef_water.xlsx", sheet = 1, col_names = c("iso_code", "countries", "yea

# Delete empty column in drinking_water
drinking_water <- subset(drinking_water, select = -DELETE)

# Create sanitation dataframe
sanitation <- read_excel("~/Downloads/Drinking-Water-Sanitation-Hygiene-Database-July-2017.xlsx", sheet

# Delete empty column in sanitation
sanitation <- subset(sanitation, select = -DELETE)

# Create hygiene dataframe
hygiene <- read_excel("~/Downloads/Drinking-Water-Sanitation-Hygiene-Database-July-2017.xlsx", sheet = 3
```

After using download.file() and read_excel(), the dataframes were formatted this way.

```
## # A tibble: 464 x 36
##    iso_code countries  year `national/at_le~ `national/limit~
##    <chr>    <chr>     <dbl>            <dbl>            <dbl>
##  1 AFG      Afghanis~  2000             27.1             2.36
##  2 AFG      Afghanis~  2015             63.0             5.55
##  3 ALB      Albania    2000             87.6             9.27
##  4 ALB      Albania    2015             91.4             4.74
##  5 DZA      Algeria    2000             89.8             5.50
##  6 DZA      Algeria    2015             93.5             5.21
##  7 ASM      American~  2000             98.5            NA
##  8 ASM      American~  2015             99.2            NA
##  9 AND      Andorra    2000            100              NA
## 10 AND      Andorra    2015            100              NA
## # ... with 454 more rows, and 31 more variables:
## #   `national/unimproved` <dbl>, `national/surface_water` <dbl>,
## #   `national/annual_basic_change_rate` <dbl>,
## #   `rural/at_least_basic` <dbl>, `rural/limited` <dbl>,
## #   `rural/unimproved` <dbl>, `rural/surface_water` <dbl>,
## #   `rural/annual_basic_change_rate` <dbl>, `urban/at_least_basic` <dbl>,
## #   `urban/limited` <dbl>, `urban/unimproved` <dbl>,
## #   `urban/surface_water` <dbl>, `urban/annual_basic_change_rate` <dbl>,
## #   `national/iws_safely_managed` <dbl>,
## #   `national/iws_accessible_premise` <dbl>,
## #   `national/iws_available_needed` <dbl>,
## #   `national/iws_free_from_contamination` <dbl>,
## #   `national/iws_piped` <dbl>, `national/iws_non_piped` <dbl>,
## #   `rural/iws_safely_managed` <dbl>,
## #   `rural/iws_accessible_premise` <dbl>,
## #   `rural/iws_available_needed` <dbl>,
## #   `rural/iws_free_from_contamination` <dbl>, `rural/iws_piped` <dbl>,
## #   `rural/iws_non_piped` <dbl>, `urban/iws_safely_managed` <dbl>,
## #   `urban/iws_accessible_premise` <dbl>,
```

```
## #   `urban/iws_available_needed` <dbl>,
## #   `urban/iws_free_from_contamination` <dbl>, `urban/iws_piped` <dbl>,
## #   `urban/iws_non_piped` <dbl>
```

Tidying Data

Tidying these dataframes created the biggest chunk of code in my report; however, the process was usually a variation of a repetition of several core steps. I will explain the core steps here.

Drinking Water, Sanitation, Hygiene, Improved

- Separating dataframes

Both the drinking water and sanitation dataframes have data on the improvement of both facility categories, and I decided it was best to create new dataframes that focused solely on improvement. For drinking water, I created an improved_water_supply object and assigned variables that only contained "iws" and the other important variables (percentage, countries, iso_code, year). I then deleted the "iws" variables from the drinking_water dataframe. The same process was done for the sanitation/improved_sanitation_facilities dataframes.

- Gather & separate columns

An important component of tidying data is that every row has to represent an observation and columns have to be variable names (not values). Therefore, I decided to gather "national/at_least_basic" to "urban/annual_basic_change_rate" columns and create two new columns: evaluation, containing the old column names, and percentage, containing all the percentages that were underneath the columns.

After gathering, I separated the evaluation columns, which has the "national/at_least_basic" to "urban/annual_basic_change_rate" names in them, by the "/" in each of the names. The separation created two columns: country_level (national, rural, urban) and water_supply (at_least_basic, limited, etc).

- Remove rows with NAs

Now that each row is an observation, I can remove the rows that have no observation, or simply NA in their percentage column. This is done by using the drop_na() function.

- Creating categorical variables

There are three categorical variables in each dataframe: country_level (national, urban, rural), level/improvement (at least basic, limited, etc), and year (2000, 2015). I made them categorical variables in order to make it easier to investigate the data in later graphs and tables.

- Round percentage variable

The percentage columns had more than 8-10 digits in each column, so I decided to set the cap at the number of digits to 6 in order to read the numbers with more ease.

- Snip

Specifically for the improved_water_supply and improved_sanitation_facilities, the improvement values all had either "iws" or "isf" in the name. These letters were no longer necessary; therefore, I snipped them from the values using mutate() and str_remove().

```
## Separate Dataframes
# Drinking Water -> Drinking Water and Improved Water Supply
improved_water_supply <- drinking_water %>% select(iso_code, countries, year, contains("iws"))
drinking_water <- drinking_water %>% select(-contains("iws"))

# Sanitation -> Sanitation and Improved Sanitation Facilities
improved_sanitation_facilities <- sanitation %>% select(iso_code, countries, year, contains("isf"))
sanitation <- sanitation %>% select(-contains("isf") )
```

```r
## Drinking Water Dataframe
# Gather columns
drinking_water <- gather(drinking_water, evaluation, percentage, "national/at_least_basic":"urban/annual

# Separate evaluation column
drinking_water <- separate(drinking_water, evaluation, c("country_level", "water_supply"), sep = "/")

# Remove rows with NAs
drinking_water <- drop_na(drinking_water)

# change country_level, water_supply, and year into categorical variables
drinking_water <- drinking_water %>% mutate_at(c("country_level","water_supply","year"), as.factor)

# Round percentage variable to six digits
drinking_water <- drinking_water %>% mutate(percentage = round(drinking_water$percentage, digits = 6))

## Improved Water Supply Dataframe
# Gather columns
improved_water_supply <- gather(improved_water_supply, evaluation, percentage, "national/iws_safely_mana

# Separate evaluation column
improved_water_supply <- separate(improved_water_supply, evaluation, c("country_level", "water_improveme

# Remove rows with NAs
improved_water_supply <- drop_na(improved_water_supply)

# snip iws from water_improvement variable
improved_water_supply <- improved_water_supply %>% mutate(water_improvement = str_remove(water_improveme

# change country_level, water_improvement, and years into categorical variables
improved_water_supply <- improved_water_supply %>% mutate_at(c("country_level", "water_improvement", "y

# Round percentage variable to six digits
improved_water_supply <- improved_water_supply %>% mutate(percentage = round(improved_water_supply$perc

## Hygiene Dataframe
# Gather columns
hygiene <- gather(hygiene, evaluation, percentage, "national/basic":"urban/no_facility")

# Separate evaluation column
hygiene <- separate(hygiene, evaluation, c("country_level", "hygiene_level"), sep = "/")

# Remove rows with NAs
hygiene <- drop_na(hygiene)

# change country_level, hygiene_level, and year to categorical variables
hygiene <- hygiene %>% mutate_at(c("country_level", "hygiene_level", "year"), as.factor)

# Round percentage variable to six digits
hygiene <- hygiene %>% mutate(percentage = round(hygiene$percentage, digits = 6))

## Sanitation Dataframe
# Gather columns
```

```
sanitation <- gather(sanitation, evaluation, percentage, "national/at_least_basic":"urban/annual_rate_c

# Separate evaluation column
sanitation <- separate(sanitation, evaluation, c("country_level", "sanitation_level"), sep = "/")

# Remove rows with NAs
sanitation <- drop_na(sanitation)

# Change country_level, sanitation_level, and year to categorical variables
sanitation <- sanitation %>% mutate_at(c("country_level", "sanitation_level", "year"), as.factor)

# Round percentage variable to six digits
sanitation <- sanitation %>% mutate(percentage = round(sanitation$percentage, digits = 6))

## Improved Sanitation Facilities
# Gather columns
improved_sanitation_facilities <- gather(improved_sanitation_facilities, evaluation, percentage, "natio

# Separate evaluation columns
improved_sanitation_facilities <- separate(improved_sanitation_facilities, evaluation, c("country_level

# Remove rows with NAs
improved_sanitation_facilities <- drop_na(improved_sanitation_facilities)

# snip isf from sanitation_improvement column
improved_sanitation_facilities <- improved_sanitation_facilities %>% mutate(sanitation_improvement = st

# Change country_level, sanitation_improvement, and year to categorical variables
improved_sanitation_facilities <- improved_sanitation_facilities %>% mutate_at(c("country_level", "sani
```

After tidying, the drinking water dataframe was formatted this way.

```
## # A tibble: 5,217 x 6
##    iso_code countries      year  country_level water_supply   percentage
##    <chr>    <chr>          <fct> <fct>         <fct>                <dbl>
##  1 AFG      Afghanistan    2000  national      at_least_basic        27.1
##  2 AFG      Afghanistan    2015  national      at_least_basic        63.0
##  3 ALB      Albania        2000  national      at_least_basic        87.6
##  4 ALB      Albania        2015  national      at_least_basic        91.4
##  5 DZA      Algeria        2000  national      at_least_basic        89.8
##  6 DZA      Algeria        2015  national      at_least_basic        93.5
##  7 ASM      American Samoa 2000  national      at_least_basic        98.5
##  8 ASM      American Samoa 2015  national      at_least_basic        99.2
##  9 AND      Andorra        2000  national      at_least_basic       100
## 10 AND      Andorra        2015  national      at_least_basic       100
## # ... with 5,207 more rows
```

Rates

- Two separate dataframes

1. Drinking Water

By using filter() and select(), I obtained only the observations that included "annual_basic_change_rate." I also changed the variable name "water supply" to "change rates" and dropped the levels that I removed when only selecting rate values (i.e at least basic, limited, etc). I also changed the name "annual_basic_change_rate"

to "drinking_water_basic" by using fct_recode().

2. Sanitation

Since the sanitation dataframe contained I used rbind() to combine the sanitation dataframe filtered by "annual_rate_change_open_defecation" AND the sanitation dataframe filtered by "annual_rate_change_basic". Afterwards, I used mutate() to droplevels() and change the variable name "sanitation_level" to "change rates."

- Bind two dataframes & remove from global environment

After creating rates1 (from drinking water) and rates2 (from sanitation), I used rbind() to combine the two dataframes into one, titled "annual_rate_change." I then removed rates1 and rates2 using rm() since I don't like my global environment cluttered with unused dataframes.

- Remove duplicate rows

Since the percentages were duplicated for both year 2000 and 2015, I decided to remove all the rows where year = 2000 in order to delete duplicate rows.

- Remove rows in drinking water & sanitation

Now that I created an "annual_rate_change" dataframe, the same data that is in drinking water and sanitation dataframes can be removed. I used subset to remove the observations from the two dataframes and used droplevels() to removed unused levels.

```
## Rates Dataframe: drinking water and sanitation annual rate changes for certain water/sanitation cond
# Create rates1: contains drinking_water rate values
rates1 <- drinking_water %>% filter(water_supply == "annual_basic_change_rate") %>% mutate(change_rates

# Recode factor name for drinking_water's rate values
rates1$change_rates <- fct_recode(rates1$change_rates, "drinking_water_basic" = "annual_basic_change_ra

# Create rates2: contains sanitation rate values
rates2 <-rbind(sanitation %>% filter(sanitation_level == "annual_rate_change_open_defecation"), sanitat

# Recode factor names for sanitation's rate values
rates2$change_rates <- fct_recode(rates2$change_rates, "sanitation_basic" = "annual_rate_change_basic",

# Bind rates1 and rates2 into a single dataframe
annual_rate_change <- rbind(rates1, rates2)

# Remove rates1 and rates2 from global environment
rm(rates1, rates2)

# Remove duplicate rows
annual_rate_change <- annual_rate_change %>% subset(year == 2000) %>% select(-year)

# Remove rates from drinking_water & sanitation
drinking_water <- drinking_water %>% subset(water_supply != "annual_basic_change_rate") %>% mutate(wate

sanitation <- sanitation %>% subset(sanitation_level != "annual_rate_change_basic" & sanitation_level !=
```

After tidying, the annual rates change dataframe was formatted this way.

```
## # A tibble: 1,606 x 5
##    iso_code countries          country_level percentage change_rates
##    <chr>    <chr>              <fct>              <dbl> <fct>
##  1 AFG      Afghanistan        national            2.39   drinking_water_bas~
```

```
##  2 ALB      Albania            national          0.254  drinking_water_bas~
##  3 DZA      Algeria            national          0.242  drinking_water_bas~
##  4 ASM      American Samoa     national          0.0488 drinking_water_bas~
##  5 AND      Andorra            national          0      drinking_water_bas~
##  6 AGO      Angola             national          0.216  drinking_water_bas~
##  7 AIA      Anguilla           national          0.368  drinking_water_bas~
##  8 ATG      Antigua and Barbu~ national         -0.101  drinking_water_bas~
##  9 ARG      Argentina          national          0.0410 drinking_water_bas~
## 10 ARM      Armenia            national          0.203  drinking_water_bas~
## # ... with 1,596 more rows
```

Functions

Most of the functions I created were to reduce the amount of times I'd copy and paste codes that made graphs or tables. However, the add_UN_regions() function added two categorical variables to the datasets in order to paint a bigger picture about the information in the datasets. The retrieve_data() reduced the amount of code that would be embedded into the text under each section of this report and make it easier for me to read when typing.

add_UN_regions(df)

This functino adds two categorical variables to the dataframes.

1. Regions 2. Least Developed Country:

I used the region designations in the Sustainable Development Goals (SDG) that were used in the 2017 Report and Statistical annex. Several graphs on the UNICEF's WASH website had used these groupings.

- **Created a vector of countries for each region**: I copied and reformatted all of the countries in each region and assigned them to their own R object.
- **Created an empty regions and LDC (least developed country) column**: I used mutate() to create a regions column filled with NAs and a LDC column filled with FALSE.
- **Created a list with regions**: I created a regions object that was a list where each component was a region vector with its countries, which was created previously.
- **For loop**: There are three for loops nested within themselves; one for loop goes through each region, another for loop goes through each country in each region

    1. Regions for loop:

    The first for loop goes through each region in the regions list created earlier.

    2. Countries for loop:

    The second loop goes through each country in whichever region is chosen in the previous loop. This loop also matches which rows in the dataframe has the country the loop is on. This is done by having the country (c) variable equal the dataframe country column, then using as.numeric and saving this result to the row_matches album. The row_matches object is then used in the next for loop to match the region to the correct row.

    3. Rows for loop:

    The third for loop goes through every row in the dataframe and adds the correct region designation to the country that is in that region. The result fills up the regions column or LDC column that was added to the dataframe earlier.

    There's an if else statement in the loop that decides whether the region would be one of the seven (sub_saharan_africa, northern_africa_western_asia, central_southern_asia, eastern_southeastern_asia, latin_america_caribbean, oceania, europe_northern_america) or LDC (least_developed_countries). This is determined by the region_id_number, which is a variable defined

at the beginning of all the for loops. Plus one is added to region_id_number at the end of the first for loop in order to count which region the loop is on.

If the region_id_number equals 8, meaning the regions loop is on the least_developed_countries object, then the function begins deciding whether a row in the dataframe has a country that is on the least developed countries list. If a country is on the list, then TRUE replaces the FALSE that is already in the column.

- **Make region and LDC columns categorical variables**: This is done by using mutate_at() and as.factor on the two columns.
- **End of function**: After the function is created, I use the function on each individual dataframe and reassign the dataframe returned in the function to the dataframe in the global environment in order for it to be updated.

```
# sustainable development goals (sdg) regional groupings
add_UN_regions <- function(df) {
  # vector of countries and their regions (total: 240)
    sub_saharan_africa <- c("Angola", "Benin", "Botswana", "Burkina Faso", "Burundi", "Cabo Verde", "Ca

    northern_africa_western_asia <- c("Algeria", "Egypt", "Libya", "Morocco", "Sudan", "Tunisia", "West

    central_southern_asia <- c("Kazakhstan", "Kyrgyzstan", "Tajikistan", "Turkmenistan", "Uzbekistan",

    eastern_southeastern_asia <- c("China", "China, Hong Kong Special Administrative Region", "China, Ma

    latin_america_caribbean <- c("Anguilla", "Antigua and Barbuda", "Aruba", "Bahamas", "Barbados", "Bo

    oceania <- c("Australia","Christmas Island", "Cocos (Keeling) Islands", "Heard Island & McDonald Isl

    europe_northern_america <- c("Bermuda", "Canada", "Greenland", "United States of America", "Bulgaria

    least_developed_countries <- c("Afghanistan", "Angola", "Bangladesh", "Benin", "Bhutan", "Burkina Fa

  # add region & LDC variables to dataframe
  df <- df %>% mutate(region = NA_character_, LDC = FALSE)

  # creating list with regions
  regions <- list(sub_saharan_africa = sub_saharan_africa, northern_africa_western_asia = northern_afric

  # loop for dataframe to have each country checked against the 6 regions
  region_id_number = 1
  for (r in regions) {
    # loop for individual region to be evaluated against every country in dataframe
    for (c in r) {
      # which rows match the specific region
      row_matches <- as.numeric(c == df[["countries"]])
      # go through each row and add region name if there's a match
      for (n in 1:nrow(df)) {
        if (region_id_number == 8) {
          if (row_matches[n] == 1) {
            df$LDC[n] <- TRUE
          }
        }
        else if (row_matches[n] == 1) {
          df$region[n] <- names(regions)[region_id_number]
```

```
        }
      }
    }
  # add to region id number for naming
    region_id_number = region_id_number + 1
  }

   # make region & LDC variable a categorical variable
    df <- df %>% mutate_at(c("region", "LDC"), as.factor)

  # return new df
  return(df)
}

# add regions from UN website
annual_rate_change <- add_UN_regions(annual_rate_change)
drinking_water <- add_UN_regions(drinking_water)
hygiene <- add_UN_regions(hygiene)
improved_sanitation_facilities <- add_UN_regions(improved_sanitation_facilities)
improved_water_supply <- add_UN_regions(improved_water_supply)
sanitation <- add_UN_regions(sanitation)
```

After using the add_UN_regions(), the drinking water dataframe was formatted this way.

```
## # A tibble: 4,133 x 8
##    iso_code countries year  country_level water_supply percentage region
##    <chr>    <chr>     <fct> <fct>         <fct>              <dbl> <fct>
##  1 AFG      Afghanis~ 2000  national      at_least_ba~        27.1 centr~
##  2 AFG      Afghanis~ 2015  national      at_least_ba~        63.0 centr~
##  3 ALB      Albania   2000  national      at_least_ba~        87.6 europ~
##  4 ALB      Albania   2015  national      at_least_ba~        91.4 europ~
##  5 DZA      Algeria   2000  national      at_least_ba~        89.8 north~
##  6 DZA      Algeria   2015  national      at_least_ba~        93.5 north~
##  7 ASM      American~ 2000  national      at_least_ba~        98.5 ocean~
##  8 ASM      American~ 2015  national      at_least_ba~        99.2 ocean~
##  9 AND      Andorra   2000  national      at_least_ba~       100   europ~
## 10 AND      Andorra   2015  national      at_least_ba~       100   europ~
## # ... with 4,123 more rows, and 1 more variable: LDC <fct>
```

The following functions are used only to reproduce the graphs, tables, and numbers that are seen in the "Investigating Data" portion of the

unicef_summary_table(df, ws, yr, country, title = NA, colnames = c(NA))

- This function creates a singular table on the dataframe with the specified water_supply, year, and country_level designated. The table includes summary statistics. The information is grouped by region and includes a column on least developed regions.
- The function's arguments correlate to three of the column names (water_supply, year, country_level) that are found in all the dataframes. The dataframe argument (df) selects which dataframe to create the table from, and the title and colnames arguments define the title and column names, respectively, of the table.
- The table is created using the summary_table() function in the qwraps2 package, but in order for the table to properly show up in the r markdown, the rlang package is also required. The rlang package has the ".data" pronoun, which is used in the list of table components (the second argument in the

summary_table function). It is also required to have *options(qwraps2_markup = "markdown")* set as well in order for the table to work in r markdown.

- The first part of the function creates two dataframes that only has the information inputted in the arguments i.e dataframe, water supply, country level, and year specifications. The difference in the dataframes is that one is being used for region information while the other is specifically for least developed countries information.
- The summarystats variable is a list with the different summary statistics on the country's population percentage. The table includes minimum, maximum, median, IQR, IQR range, mean, and standard deviation. The median (IQR) and mean (SD) use qwraps2 functions median_iqr() and mean_sd(), respectively. min() and max() come from the base package, and IQR() comes from the stat package. I use round() on min(), max(), and IQR() to round the statistics to the second decimal place in order to make the numbers more readable.
- The names(summarystats) assigns the table_name to the table, and the name depends on the year the dataframe is using. The yr variable is then turned as a character to become the name of the table.
- The second part of the function creates the table objects by using the summary_table() function twice: once on the region dataframe and the other on the least developed countries dataframe. Both use summarystats as their second argument. Afterwards, cbind() is used to combine both table objects into one.
- The if else statement determines what the function should return. If the title and colnames argument of the function are defined, then the print() function is used to print the table with its title (using the rtitle argument) and column names (using the cnames argument). The rtitle and cnames argument are unique to when using print() on qwraps2_summary_table objects. The qwraps2 CRAN pdf, which was linked above, explains that in section 3, titled "Building a Data Summary Table."

unicef_two_tables(dataframe, water, country_level, title_name)

- As much as I love using the qwraps2 table, the only downside was I couldn't use the filter() function from dplyr with the .data pronoun to make 2000 and 2015 sections in one table. Neither could I just use the argument variable x. Therefore, I created another function that automatically made two tables with the only difference being their year, title, and colname arguments. The second table automatically sets the title and colname arguments to repeatedly producing "-".

retrieve_data(df, ws, country, yr = NA, reg = NA, column = FALSE)

- Embedding lines of R code into my text prevents me from having to look at my table and copy and paste the summary statistics into my texts (and possibly copying down the wrong number). However, the r embedded code got increasingly complicated to write because of the amount of filtering required to get the dataframe I want. Therefore, I created the retrieve_data() function to help decrease the amount of code in my texts.
- Occassionally, I may not want to specify the region or year for my dataframe; I may want all of the regions' percentages when calculating statistics or I may want percentages form both 2000 and 2015. Therefore, I made the region and year arguments optional, and I added several if else statements that signify what to do if either, both, or neither were specified.
- I may want the function to return either a dataframe or the percentage column of the dataframe. Therefore, I created an if else statement that evaluated what to do if the column was either TRUE or FALSE. If column is TRUE, then the percentage column is returned. If FALSE, then the dataframe is returned.

water_supply_graph(df, ws, title_name)

- This function creates similar boxplot graphs using ggplot2; the only difference is the specified dataframe, water supply, and title name. I will explain the layers of the graph by function.

- scale_x_discrete() was used to name the x axis ("Region") and rename the labels of each region, which makes the region label names look better instead of having the names of the variables that contain underscores and lowercase letters.
- labs() gave the graph a title (which was set by the function's title_name argument) and a pre-set subtitle and y axis title. The x axis title was set in the scale_x_discrete() function.
- geom_boxplot() creates the boxplot graph.
- facet_grid() splits the graph's data into panels defined by two categorical variables. For these graphs specifically, horizontally, the graphs were divided by the country_level variable, whereas vertically the graphs were separated by the year variable. I used the labeller argument to capitalize the level names for the country_level. The ggplot2 R documentation says facet_grid() "forms a matrix of panels defined by row and column faceting variables" and is most useful when "all combinations of the [discete] variables exist in the data."
- theme_minimal() was one of the pre-set theme options I chose for my graphs. I used this website to browse for a theme that I wanted.
- coord_flip() flipped the x and y axis so that regions was now on the y axis and percentage was on the x axis. I wanted to do this in the aes() function, but the graphs did not turn out correctly. It was easier to flip the x and y axis at the end.

ldc_graph(df, ws, title_text)

- There are only a couple of differences between the water_supply_graph() function and this function. The main difference is that the x argument in aes() is LDC and not region. There is no scale_x_discrete() because the tick marks (which were TRUE and FALSE in ldc_graph) didn't need to be renamed. There is also no subtitle. Everything else about water_supply_graph() and ldc_graph() is identical.

outliers_region

-

```
# function to make tables
unicef_summary_table <- function(df, ws, yr, country, title = NA, colnames = c(NA)) {
  # dataframe with filter
  dtfr <- df %>% filter(water_supply == ws & year == yr & country_level == country) %>% select(region, p
  ldc <- df %>% filter(LDC == TRUE & water_supply == ws & country_level == country & year == yr) %>% sel
  non_ldc <- df %>% filter(LDC == FALSE & water_supply == ws & country_level == country & year == yr) %

  # summary stats to take from dataframe
  summarystats <- list(
    table_name =
    list("Minimum" = ~ round(min(.data$percentage), 2),
         "Maximum" = ~ round(max(.data$percentage), 2),
         "Median (QR1,QR3)" = ~ qwraps2::median_iqr(.data$percentage),
         "IQR" = ~ round(IQR(.data$percentage), 2),
         "Mean (SD)" = ~ qwraps2::mean_sd(.data$percentage)
    ))

  names(summarystats) <- as.character(yr)

  # create table
  world <- summary_table(dtfr, summarystats)
  obj <- summary_table(dtfr %>% dplyr::group_by(region), summarystats)
  obj_ldc <- summary_table(ldc, summarystats)
  obj_non_ldc <- summary_table(non_ldc, summarystats)
  combo <- cbind(cbind(world, cbind(obj, obj_ldc)), obj_non_ldc)
```

```
  # print column names & return
  if (!is.na(title) & !is.na(colnames)) {
    return (print(combo, rtitle = title, cnames = colnames))
  } else {
   # only return table
    return(combo)
  }
}

# function to make two tables under each graph
unicef_two_tables <- function(dataframe, water, country_level, title_name) {
  unicef_summary_table(df = dataframe, ws = water, yr = 2000, country = country_level, title = title_nar

  unicef_summary_table(df = dataframe, ws = water, yr = 2015, country = country_level, title = title_nar
}

# retrieve data with desired restrictions
retrieve_data <- function(df, ws, country, yr = NA, reg = NA, column = FALSE) {
  # retrieve a specific region's dataframe or general dataframe
  if (!is.na(reg) & !is.na(yr)) {
    dataframe <-
      df %>% filter(water_supply == ws &
                      country_level == country & year == yr & region == reg)

  } else if (!is.na(reg) & is.na(yr)) {
    dataframe <-
      df %>% filter(water_supply == ws &
                      country_level == country & region == reg)

  } else if (is.na(reg) & !is.na(yr)) {
    dataframe <-
      df %>% filter(water_supply == ws &
                      country_level == country & year == yr)

  } else if (is.na(reg) & is.na(yr)) {
    dataframe <-
      df %>% filter(water_supply == ws &
                      country_level == country)
  }

  # retrive entire dataframe or only percentage column
  if (column == FALSE) {
    return(dataframe)

  } else if (column == TRUE) {
    col <- pull(dataframe %>% select(percentage))
    return(col)
  }
}

# function to make desired ggplot boxplots
## water supply
```

```r
water_supply_graph <- function(df, ws, title_name) {
  ggplot((df %>% filter(water_supply == ws)), aes(x = region, y = percentage)) +
  scale_x_discrete(name = "Region", labels = c("sub_saharan_africa" = "Sub-Saharan Africa", "oceania" =
  labs(title = title_name, subtitle = "Comparing country levels and year of observed percentages", y =
  geom_boxplot() +
  facet_grid(year ~ country_level, labeller = labeller(country_level = c(national = "National", rural =
  theme_minimal() +
  coord_flip()
}

## least developed country & water supply
ldc_graph <- function(df, ws, title_text) {
  ggplot((df %>% filter(water_supply == ws )), aes(x = LDC, y = percentage)) +
  labs(title = title_text, y = "Percentage (%)") +
  geom_boxplot() +
    facet_grid(year ~ country_level, labeller = labeller(country_level = c(national = "National", rural
  theme_minimal() +
  coord_flip()
}

## outliers
outliers_region <- function(dataframe, water_supply, cl, year, region) {
  # create dataframe object & other statistics
  IQR <- IQR(retrieve_data(dataframe, water_supply, cl, year, region) %>% select(percentage) %>% pull()

  Q1 <- quantile(retrieve_data(dataframe, water_supply, cl, year, region) %>% select(percentage) %>% pu

  df_object <- retrieve_data(dataframe, water_supply, cl, year, region) %>% filter(percentage < (Q1 - (

  # number of countries
  num_of_countries <- df_object %>% count() %>% as.numeric()

  # no outliers
  if (num_of_countries == 0) {
    return("NONE")
  }

  # for loop
  string <- ""
  for (i in 1:num_of_countries) {
    country <- (df_object %>% select(countries) %>% pull())[i]

    percent <- round((df_object %>% select(percentage) %>% pull())[i], 2)

    string <- str_c(string, country, " (", percent, "), ")
  }

  string <- substr(string, 1, nchar(string)-2)

  return(string)
}
```

Investigating Data

All Countries

```
##                                        countries
## 1                                          Angola
## 2                                           Benin
## 3                                        Botswana
## 4                                    Burkina Faso
## 5                                         Burundi
## 6                                      Cabo Verde
## 7                                        Cameroon
## 8                          Central African Republic
## 9                                            Chad
## 10                                        Comoros
## 11                                          Congo
## 12                                 Côte d'Ivoire
## 13                Democratic Republic of the Congo
## 14                                      Djibouti
## 15                             Equatorial Guinea
## 16                                       Eritrea
## 17                                      Ethiopia
## 18                                         Gabon
## 19                                        Gambia
## 20                                         Ghana
## 21                                        Guinea
## 22                                 Guinea-Bissau
## 23                                         Kenya
## 24                                       Lesotho
## 25                                       Liberia
## 26                                    Madagascar
## 27                                        Malawi
## 28                                          Mali
## 29                                    Mauritania
## 30                                     Mauritius
## 31                                       Mayotte
## 32                                    Mozambique
## 33                                       Namibia
## 34                                         Niger
## 35                                       Nigeria
## 36                                       Réunion
## 37                                        Rwanda
## 38                          Sao Tome and Principe
## 39                                       Senegal
## 40                                    Seychelles
## 41                                  Sierra Leone
## 42                                       Somalia
## 43                                  South Africa
## 44                                   South Sudan
## 45                                     Swaziland
## 46                                          Togo
## 47                                        Uganda
## 48                     United Republic of Tanzania
## 49                                        Zambia
## 50                                      Zimbabwe
```

```
## 51                                        Algeria
## 52                                          Egypt
## 53                                          Libya
## 54                                        Morocco
## 55                                          Sudan
## 56                                        Tunisia
## 57                                 Western Sahara
## 58                                     Azerbaijan
## 59                                        Armenia
## 60                                        Bahrain
## 61                                         Cyprus
## 62                                        Georgia
## 63                                           Iraq
## 64                                         Israel
## 65                                         Jordan
## 66                                         Kuwait
## 67                                        Lebanon
## 68                              State of Palestine
## 69                                           Oman
## 70                                          Qatar
## 71                                   Saudi Arabia
## 72                             Syrian Arab Republic
## 73                                         Turkey
## 74                             United Arab Emirates
## 75                                          Yemen
## 76                                   Saint Helena
## 77                          West Bank and Gaza Strip
## 78                                     Kazakhstan
## 79                                     Kyrgyzstan
## 80                                     Tajikistan
## 81                                   Turkmenistan
## 82                                     Uzbekistan
## 83                                    Afghanistan
## 84                                     Bangladesh
## 85                                         Bhutan
## 86                                          India
## 87                          Iran (Islamic Republic of)
## 88                                       Maldives
## 89                                          Nepal
## 90                                       Pakistan
## 91                                      Sri Lanka
## 92                                          China
## 93  China, Hong Kong Special Administrative Region
## 94      China, Macao Special Administrative Region
## 95           Democratic People's Republic of Korea
## 96                                          Japan
## 97                                       Mongolia
## 98                                Republic of Korea
## 99                                Brunei Darussalam
## 100                                      Cambodia
## 101                                     Indonesia
## 102              Lao People's Democratic Republic
## 103                                      Malaysia
## 104                                       Myanmar
```

```
## 105                                  Philippines
## 106                                    Singapore
## 107                                     Thailand
## 108                                  Timor-Leste
## 109                                     Viet Nam
## 110                                     Anguilla
## 111                           Antigua and Barbuda
## 112                                        Aruba
## 113                                      Bahamas
## 114                                     Barbados
## 115               Bonaire, Sint Eustatius and Saba
## 116                        British Virgin Islands
## 117                               Cayman Islands
## 118                                         Cuba
## 119                                     Curaçao
## 120                                    Dominica
## 121                           Dominican Republic
## 122                                      Grenada
## 123                                   Guadeloupe
## 124                                        Haiti
## 125                                     Jamaica
## 126                                   Martinique
## 127                                   Montserrat
## 128                                 Puerto Rico
## 129                          Saint Kitts and Nevis
## 130                                  Saint Lucia
## 131               Saint Vincent and the Grenadines
## 132                      Sint Maarten (Dutch part)
## 133                                    Suriname
## 134                          Trinidad and Tobago
## 135                        Turks and Caicos Islands
## 136                    United States Virgin Islands
## 137                                    Honduras
## 138                                  Costa Rica
## 139                                 El Salvador
## 140                                   Guatemala
## 141                                      Mexico
## 142                                   Nicaragua
## 143                                      Panama
## 144                                   Argentina
## 145                                      Belize
## 146               Bolivia (Plurinational State of)
## 147                                      Brazil
## 148                                       Chile
## 149                                    Colombia
## 150                                     Ecuador
## 151                                French Guiana
## 152                     Falkland Islands (Malvinas)
## 153     South Georgia & the South Sandwich Islands
## 154                                      Guyana
## 155                                    Paraguay
## 156                                        Peru
## 157                                     Uruguay
## 158              Venezuela (Bolivarian Republic of)
```

```
## 159                                  Australia
## 160                            Christmas Island
## 161                    Cocos (Keeling) Islands
## 162                Heard Island & McDonald Islands
## 163                             Norfolk Island
## 164                                New Zealand
## 165                                       Fiji
## 166                               New Caledonia
## 167                             Papua New Guinea
## 168                              Solomon Islands
## 169                                     Vanuatu
## 170                                    Kiribati
## 171                             Marshall Islands
## 172               Micronesia (Federated States of)
## 173                                      Nauru
## 174                     Northern Mariana Islands
## 175                                      Palau
## 176                                       Guam
## 177                             French Polynesia
## 178                     Wallis and Futuna Island
## 179                                   Pitcairn
## 180                                Cook Islands
## 181                                       Niue
## 182                                    Tokelau
## 183                                      Tonga
## 184                                     Tuvalu
## 185                               American Samoa
## 186                                      Samoa
## 187                     Wallis and Futuna Islands
## 188                                    Bermuda
## 189                                     Canada
## 190                                  Greenland
## 191                     United States of America
## 192                                   Bulgaria
## 193                                    Belarus
## 194                               Czech Republic
## 195                                    Hungary
## 196                          Republic of Moldova
## 197                                     Poland
## 198                                    Romania
## 199                           Russian Federation
## 200                                   Slovakia
## 201                                    Ukraine
## 202                                Åland Islands
## 203                              Channel Islands
## 204                                    Denmark
## 205                                    Estonia
## 206                                Faroe Islands
## 207                                    Finland
## 208                                Isle of Man
## 209                              United Kingdom
## 210                                    Iceland
## 211                                    Ireland
## 212                                     Latvia
```

```
## 213                                      Lithuania
## 214                                         Norway
## 215                                         Sweden
## 216                                        Albania
## 217                                        Andorra
## 218                          Bosnia and Herzegovina
## 219                                        Croatia
## 220                                         Greece
## 221                                          Italy
## 222                                          Malta
## 223                                     Montenegro
## 224                                       Portugal
## 225                                     San Marino
## 226                                         Serbia
## 227                                       Slovenia
## 228                                          Spain
## 229                                      Gibraltar
## 230      The former Yugoslav Republic of Macedonia
## 231                                        Austria
## 232                                        Belgium
## 233                                    Switzerland
## 234                                        Germany
## 235                                         France
## 236                                  Liechtenstein
## 237                                     Luxembourg
## 238                                         Monaco
## 239                                    Netherlands
## 240                        Saint Pierre and Miquelon
##                           region   LDC
## 1             sub_saharan_africa  TRUE
## 2             sub_saharan_africa  TRUE
## 3             sub_saharan_africa FALSE
## 4             sub_saharan_africa  TRUE
## 5             sub_saharan_africa  TRUE
## 6             sub_saharan_africa FALSE
## 7             sub_saharan_africa FALSE
## 8             sub_saharan_africa  TRUE
## 9             sub_saharan_africa  TRUE
## 10            sub_saharan_africa  TRUE
## 11            sub_saharan_africa FALSE
## 12            sub_saharan_africa FALSE
## 13            sub_saharan_africa  TRUE
## 14            sub_saharan_africa  TRUE
## 15            sub_saharan_africa  TRUE
## 16            sub_saharan_africa  TRUE
## 17            sub_saharan_africa  TRUE
## 18            sub_saharan_africa FALSE
## 19            sub_saharan_africa  TRUE
## 20            sub_saharan_africa FALSE
## 21            sub_saharan_africa  TRUE
## 22            sub_saharan_africa  TRUE
## 23            sub_saharan_africa FALSE
## 24            sub_saharan_africa  TRUE
## 25            sub_saharan_africa  TRUE
```

```
## 26            sub_saharan_africa  TRUE
## 27            sub_saharan_africa  TRUE
## 28            sub_saharan_africa  TRUE
## 29            sub_saharan_africa  TRUE
## 30            sub_saharan_africa FALSE
## 31            sub_saharan_africa FALSE
## 32            sub_saharan_africa  TRUE
## 33            sub_saharan_africa FALSE
## 34            sub_saharan_africa  TRUE
## 35            sub_saharan_africa FALSE
## 36            sub_saharan_africa FALSE
## 37            sub_saharan_africa  TRUE
## 38            sub_saharan_africa  TRUE
## 39            sub_saharan_africa  TRUE
## 40            sub_saharan_africa FALSE
## 41            sub_saharan_africa  TRUE
## 42            sub_saharan_africa  TRUE
## 43            sub_saharan_africa FALSE
## 44            sub_saharan_africa  TRUE
## 45            sub_saharan_africa FALSE
## 46            sub_saharan_africa  TRUE
## 47            sub_saharan_africa  TRUE
## 48            sub_saharan_africa  TRUE
## 49            sub_saharan_africa  TRUE
## 50            sub_saharan_africa FALSE
## 51  northern_africa_western_asia FALSE
## 52  northern_africa_western_asia FALSE
## 53  northern_africa_western_asia FALSE
## 54  northern_africa_western_asia FALSE
## 55  northern_africa_western_asia  TRUE
## 56  northern_africa_western_asia FALSE
## 57  northern_africa_western_asia FALSE
## 58  northern_africa_western_asia FALSE
## 59  northern_africa_western_asia FALSE
## 60  northern_africa_western_asia FALSE
## 61  northern_africa_western_asia FALSE
## 62  northern_africa_western_asia FALSE
## 63  northern_africa_western_asia FALSE
## 64  northern_africa_western_asia FALSE
## 65  northern_africa_western_asia FALSE
## 66  northern_africa_western_asia FALSE
## 67  northern_africa_western_asia FALSE
## 68  northern_africa_western_asia FALSE
## 69  northern_africa_western_asia FALSE
## 70  northern_africa_western_asia FALSE
## 71  northern_africa_western_asia FALSE
## 72  northern_africa_western_asia FALSE
## 73  northern_africa_western_asia FALSE
## 74  northern_africa_western_asia FALSE
## 75  northern_africa_western_asia  TRUE
## 76  northern_africa_western_asia FALSE
## 77  northern_africa_western_asia FALSE
## 78         central_southern_asia FALSE
## 79         central_southern_asia FALSE
```

```
## 80         central_southern_asia FALSE
## 81         central_southern_asia FALSE
## 82         central_southern_asia FALSE
## 83         central_southern_asia  TRUE
## 84         central_southern_asia  TRUE
## 85         central_southern_asia  TRUE
## 86         central_southern_asia FALSE
## 87         central_southern_asia FALSE
## 88         central_southern_asia FALSE
## 89         central_southern_asia  TRUE
## 90         central_southern_asia FALSE
## 91         central_southern_asia FALSE
## 92     eastern_southeastern_asia FALSE
## 93     eastern_southeastern_asia FALSE
## 94     eastern_southeastern_asia FALSE
## 95     eastern_southeastern_asia FALSE
## 96     eastern_southeastern_asia FALSE
## 97     eastern_southeastern_asia FALSE
## 98     eastern_southeastern_asia FALSE
## 99     eastern_southeastern_asia FALSE
## 100    eastern_southeastern_asia  TRUE
## 101    eastern_southeastern_asia FALSE
## 102    eastern_southeastern_asia  TRUE
## 103    eastern_southeastern_asia FALSE
## 104    eastern_southeastern_asia  TRUE
## 105    eastern_southeastern_asia FALSE
## 106    eastern_southeastern_asia FALSE
## 107    eastern_southeastern_asia FALSE
## 108    eastern_southeastern_asia  TRUE
## 109    eastern_southeastern_asia FALSE
## 110      latin_america_caribbean FALSE
## 111      latin_america_caribbean FALSE
## 112      latin_america_caribbean FALSE
## 113      latin_america_caribbean FALSE
## 114      latin_america_caribbean FALSE
## 115      latin_america_caribbean FALSE
## 116      latin_america_caribbean FALSE
## 117      latin_america_caribbean FALSE
## 118      latin_america_caribbean FALSE
## 119      latin_america_caribbean FALSE
## 120      latin_america_caribbean FALSE
## 121      latin_america_caribbean FALSE
## 122      latin_america_caribbean FALSE
## 123      latin_america_caribbean FALSE
## 124      latin_america_caribbean  TRUE
## 125      latin_america_caribbean FALSE
## 126      latin_america_caribbean FALSE
## 127      latin_america_caribbean FALSE
## 128      latin_america_caribbean FALSE
## 129      latin_america_caribbean FALSE
## 130      latin_america_caribbean FALSE
## 131      latin_america_caribbean FALSE
## 132      latin_america_caribbean FALSE
## 133      latin_america_caribbean FALSE
```

```
## 134      latin_america_caribbean FALSE
## 135      latin_america_caribbean FALSE
## 136      latin_america_caribbean FALSE
## 137      latin_america_caribbean FALSE
## 138      latin_america_caribbean FALSE
## 139      latin_america_caribbean FALSE
## 140      latin_america_caribbean FALSE
## 141      latin_america_caribbean FALSE
## 142      latin_america_caribbean FALSE
## 143      latin_america_caribbean FALSE
## 144      latin_america_caribbean FALSE
## 145      latin_america_caribbean FALSE
## 146      latin_america_caribbean FALSE
## 147      latin_america_caribbean FALSE
## 148      latin_america_caribbean FALSE
## 149      latin_america_caribbean FALSE
## 150      latin_america_caribbean FALSE
## 151      latin_america_caribbean FALSE
## 152      latin_america_caribbean FALSE
## 153      latin_america_caribbean FALSE
## 154      latin_america_caribbean FALSE
## 155      latin_america_caribbean FALSE
## 156      latin_america_caribbean FALSE
## 157      latin_america_caribbean FALSE
## 158      latin_america_caribbean FALSE
## 159                      oceania FALSE
## 160                      oceania FALSE
## 161                      oceania FALSE
## 162                      oceania FALSE
## 163                      oceania FALSE
## 164                      oceania FALSE
## 165                      oceania FALSE
## 166                      oceania FALSE
## 167                      oceania FALSE
## 168                      oceania  TRUE
## 169                      oceania  TRUE
## 170                      oceania  TRUE
## 171                      oceania FALSE
## 172                      oceania FALSE
## 173                      oceania FALSE
## 174                      oceania FALSE
## 175                      oceania FALSE
## 176                      oceania FALSE
## 177                      oceania FALSE
## 178                      oceania FALSE
## 179                      oceania FALSE
## 180                      oceania FALSE
## 181                      oceania FALSE
## 182                      oceania FALSE
## 183                      oceania FALSE
## 184                      oceania  TRUE
## 185                      oceania FALSE
## 186                      oceania FALSE
## 187                      oceania FALSE
```

```
## 188      europe_northern_america FALSE
## 189      europe_northern_america FALSE
## 190      europe_northern_america FALSE
## 191      europe_northern_america FALSE
## 192      europe_northern_america FALSE
## 193      europe_northern_america FALSE
## 194      europe_northern_america FALSE
## 195      europe_northern_america FALSE
## 196      europe_northern_america FALSE
## 197      europe_northern_america FALSE
## 198      europe_northern_america FALSE
## 199      europe_northern_america FALSE
## 200      europe_northern_america FALSE
## 201      europe_northern_america FALSE
## 202      europe_northern_america FALSE
## 203      europe_northern_america FALSE
## 204      europe_northern_america FALSE
## 205      europe_northern_america FALSE
## 206      europe_northern_america FALSE
## 207      europe_northern_america FALSE
## 208      europe_northern_america FALSE
## 209      europe_northern_america FALSE
## 210      europe_northern_america FALSE
## 211      europe_northern_america FALSE
## 212      europe_northern_america FALSE
## 213      europe_northern_america FALSE
## 214      europe_northern_america FALSE
## 215      europe_northern_america FALSE
## 216      europe_northern_america FALSE
## 217      europe_northern_america FALSE
## 218      europe_northern_america FALSE
## 219      europe_northern_america FALSE
## 220      europe_northern_america FALSE
## 221      europe_northern_america FALSE
## 222      europe_northern_america FALSE
## 223      europe_northern_america FALSE
## 224      europe_northern_america FALSE
## 225      europe_northern_america FALSE
## 226      europe_northern_america FALSE
## 227      europe_northern_america FALSE
## 228      europe_northern_america FALSE
## 229      europe_northern_america FALSE
## 230      europe_northern_america FALSE
## 231      europe_northern_america FALSE
## 232      europe_northern_america FALSE
## 233      europe_northern_america FALSE
## 234      europe_northern_america FALSE
## 235      europe_northern_america FALSE
## 236      europe_northern_america FALSE
## 237      europe_northern_america FALSE
## 238      europe_northern_america FALSE
## 239      europe_northern_america FALSE
## 240      europe_northern_america FALSE
```

```
## # A tibble: 7 x 2
##   region                          n
##   <fct>                       <int>
## 1 europe_northern_america        53
## 2 sub_saharan_africa             50
## 3 latin_america_caribbean        49
## 4 oceania                        29
## 5 northern_africa_western_asia   27
## 6 eastern_southeastern_asia      18
## 7 central_southern_asia          14
```

Least Developed Countries only

```
##                            countries                       region
## 1                              Angola           sub_saharan_africa
## 2                               Benin           sub_saharan_africa
## 3                        Burkina Faso           sub_saharan_africa
## 4                             Burundi           sub_saharan_africa
## 5           Central African Republic           sub_saharan_africa
## 6                                Chad           sub_saharan_africa
## 7                             Comoros           sub_saharan_africa
## 8  Democratic Republic of the Congo           sub_saharan_africa
## 9                            Djibouti           sub_saharan_africa
## 10                 Equatorial Guinea           sub_saharan_africa
## 11                            Eritrea           sub_saharan_africa
## 12                           Ethiopia           sub_saharan_africa
## 13                             Gambia           sub_saharan_africa
## 14                             Guinea           sub_saharan_africa
## 15                      Guinea-Bissau           sub_saharan_africa
## 16                            Lesotho           sub_saharan_africa
## 17                            Liberia           sub_saharan_africa
## 18                         Madagascar           sub_saharan_africa
## 19                             Malawi           sub_saharan_africa
## 20                               Mali           sub_saharan_africa
## 21                         Mauritania           sub_saharan_africa
## 22                         Mozambique           sub_saharan_africa
## 23                              Niger           sub_saharan_africa
## 24                             Rwanda           sub_saharan_africa
## 25             Sao Tome and Principe           sub_saharan_africa
## 26                             Senegal           sub_saharan_africa
## 27                        Sierra Leone           sub_saharan_africa
## 28                             Somalia           sub_saharan_africa
## 29                         South Sudan           sub_saharan_africa
## 30                                Togo           sub_saharan_africa
## 31                              Uganda           sub_saharan_africa
## 32       United Republic of Tanzania           sub_saharan_africa
## 33                              Zambia           sub_saharan_africa
## 34                               Sudan northern_africa_western_asia
## 35                               Yemen northern_africa_western_asia
## 36                         Afghanistan        central_southern_asia
## 37                          Bangladesh        central_southern_asia
## 38                              Bhutan        central_southern_asia
## 39                               Nepal        central_southern_asia
## 40                            Cambodia    eastern_southeastern_asia
## 41 Lao People's Democratic Republic     eastern_southeastern_asia
```

```
## 42                          Myanmar    eastern_southeastern_asia
## 43                       Timor-Leste    eastern_southeastern_asia
## 44                            Haiti      latin_america_caribbean
## 45                  Solomon Islands                      oceania
## 46                          Vanuatu                      oceania
## 47                         Kiribati                      oceania
## 48                           Tuvalu                      oceania
```

```
## # A tibble: 6 x 2
##   region                          n
##   <fct>                       <int>
## 1 sub_saharan_africa             33
## 2 central_southern_asia           4
## 3 eastern_southeastern_asia       4
## 4 oceania                         4
## 5 northern_africa_western_asia    2
## 6 latin_america_caribbean         1
```

WORLD STATISTICS & GRAPHS?

WASH Divisions

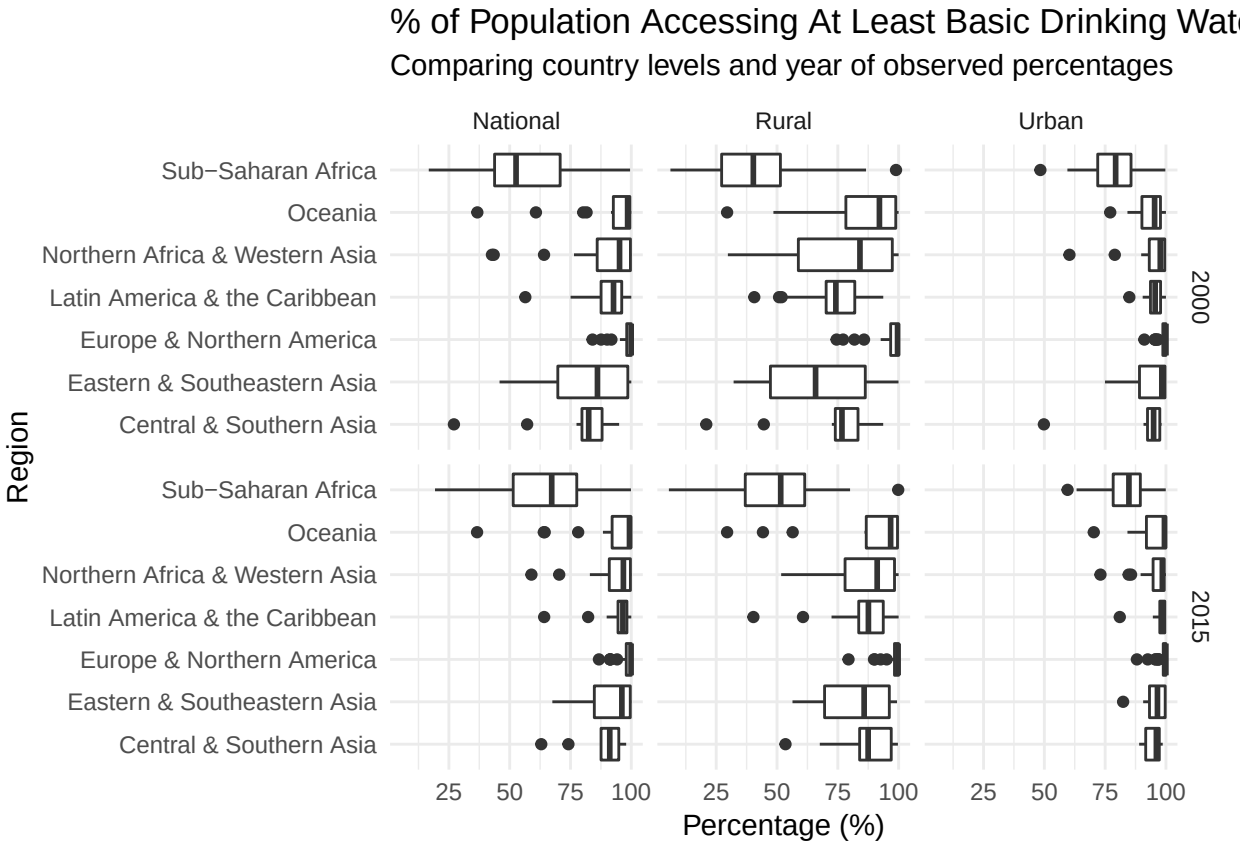
Each WASH division (drinking water, sanitation, hygiene) is divided into separate data frames; therefore, they will be evaluated separately in their own sections. Each summary statistic will be evaluted at the region level, and I will also talk about certain counties and where they lie on the region's percentage range.

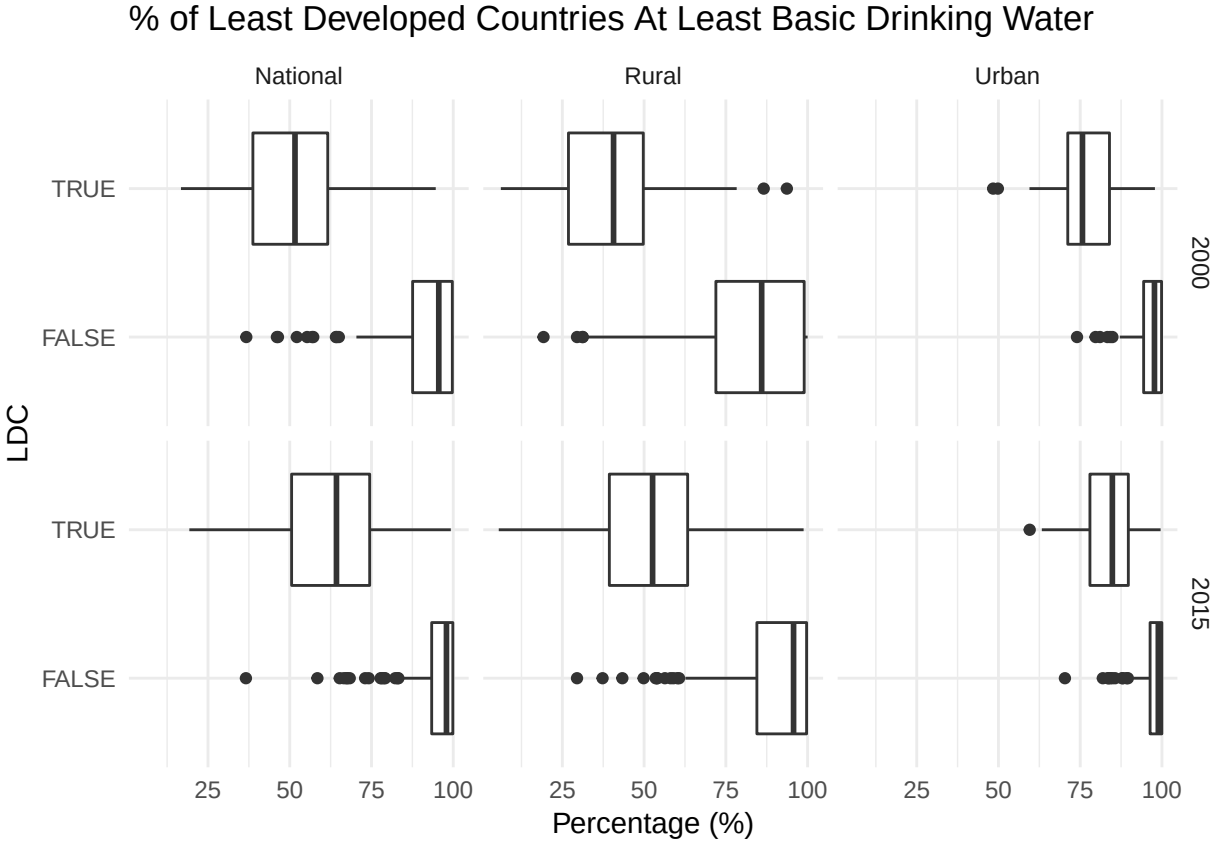
Drinking Water

Drinking water section evaluates what percentages of country's populations has access to clean drinking water, and whether the sources of drinking water have improved or not. The annual rate change dataframe, which is a separate dataframe, documents the percentage rate change per year of countries. It is able to show the rate at which countries improved their drinking water resources.

Some countries may not have observations in every category.

At Least Basic





National

National: At Least Basic	World	Central & Southern Asia	Eastern & Southeastern Asia	Europe & Northern America	Latin America & Caribbean	Northern Africa & Western Asia	Oceania	Sub-Saharan Africa	Least Developed Countries	Not LDC
2000										
Minimum	16.73	27.07	45.83	84.06	56.42	42.71	36.71	16.73	16.73	36.71
Maximum	100	94.99	100.00	100.00	100.00	100.00	100.00	99.50	94.65	100
Median (QR1,QR3)	91.89 (74.94, 99.18)	82.48 (79.71, 87.98)	86.09 (69.79, 98.57)	99.94 (98.11, 100.00)	92.69 (87.57, 96.06)	95.17 (85.96, 99.64)	98.49 (92.61, 99.40)	52.62 (43.79, 70.75)	51.60 (38.72, 61.62)	95.58 (87.58, 99.72)
IQR	24.25	8.27	28.79	1.89	8.49	13.67	6.79	26.96	22.9	12.14
Mean (SD)	82.40 ± 21.42	78.97 ± 17.52	81.57 ± 19.36	98.31 ± 3.45	90.30 ± 8.41	87.64 ± 17.08	91.10 ± 15.61	55.43 ± 20.33	51.50 ± 18.79	90.98 ± 12.24

National: At Least Basic	World	Central & Southern Asia	Eastern & Southeastern Asia	Europe & Northern America	Latin America & Caribbean	Northern Africa & Western Asia	Oceania	Sub-Saharan Africa	Least Developed Countries	Not LDC
2015										
Minimum	19.29	62.98	67.54	86.69	64.17	58.93	36.60	19.29	19.29	36.6
Maximum	100	97.88	100.00	100.00	100.00	100.00	100.00	99.87	99.26	100
Median (QR1,QR3)	96.43 (83.09, 99.59)	91.15 (87.56, 94.87)	96.13 (84.80, 99.57)	99.90 (97.92, 100.00)	96.55 (94.53, 98.20)	96.75 (90.94, 99.63)	99.26 (92.11, 99.79)	67.27 (51.43, 77.58)	64.28 (50.57, 74.44)	97.93 (93.41, 99.87)
IQR	16.5	7.31	14.78	2.08	3.68	8.69	7.68	26.15	23.87	6.47
Mean (SD)	88.07 ± 16.79	88.75 ± 10.02	90.88 ± 10.97	98.48 ± 2.79	95.43 ± 5.72	92.85 ± 10.13	91.49 ± 16.08	65.32 ± 17.77	63.78 ± 17.11	94.59 ± 8.89

OUTLIERS

2000

- *Central & Southern Asia*: Afghanistan (27.07), Tajikistan (57.18)
- *Eastern & Southeastern Asia*: NONE
- *Europe & Northern America*: Republic of Moldova (84.06), Albania (87.58), Lithuania (90.05), Serbia (91.82)
- *Latin America & the Caribbean*: Haiti (56.42)
- *Northern Africa & Western Asia*: Yemen (42.71), Sudan (43.43), Morocco (64.13)
- *Oceania*: Papua New Guinea (36.71), Kiribati (60.79), Solomon Islands (80.23), Vanuatu (81.61)
- *Sub-Saharan Africa*: NONE

2015

- *Central & Southern Asia*: Afghanistan (62.98), Tajikistan (74.14)

- *Eastern & Southeastern Asia*: NONE
- *Europe & Northern America*: Republic of Moldova (86.69), Serbia (91.18), Albania (91.39), Saint Pierre and Miquelon (91.4), Channel Islands (94.15)
- *Latin America & the Caribbean*: Haiti (64.17), Nicaragua (82.26)
- *Northern Africa & Western Asia*: Sudan (58.93), Yemen (70.36)
- *Oceania*: Papua New Guinea (36.6), Solomon Islands (64.03), Kiribati (64.39), Marshall Islands (78.16)
- *Sub-Saharan Africa*: NONE

Rural

Rural: At Least Basic	World	Central & Southern Asia	Eastern & Southeastern Asia	Europe & Northern America	Latin America & Caribbean	Northern Africa & Western Asia	Oceania	Sub-Saharan Africa	Least Developed Countries	Not LDC
2000										
Minimum	6.18	20.92	32.16	74.50	40.65	29.79	29.48	6.18	6.18	19.24
Maximum	100	93.66	99.96	100.00	93.68	100.00	100.00	98.93	93.66	100
Median (QR1,QR3)	77.17 (48.52, 96.64)	76.70 (73.96, 83.30)	65.86 (47.29, 86.30)	99.56 (96.72, 100.00)	74.20 (70.19, 81.97)	84.10 (58.77, 97.40)	92.10 (78.31, 98.83)	40.27 (27.23, 51.39)	40.65 (26.87, 49.75)	85.95 (71.95, 98.96)
IQR	48.13	9.33	39.01	3.28	11.78	38.62	20.52	24.17	22.88	27.02
Mean (SD)	70.26 ± 27.17	73.31 ± 18.81	66.94 ± 23.83	96.77 ± 6.26	72.63 ± 13.18	75.81 ± 25.11	83.64 ± 21.77	40.51 ± 19.99	41.39 ± 21.39	81.08 ± 20.37

Rural: At Least Basic	World	Central & Southern Asia	Eastern & Southeastern Asia	Europe & Northern America	Latin America & Caribbean	Northern Africa & Western Asia	Oceania	Sub-Saharan Africa	Least Developed Countries	Not LDC
2015										
Minimum	5.54	53.48	56.36	79.39	40.29	51.71	29.48	5.54	5.54	29.48
Maximum	100	99.70	99.35	100.00	100.00	100.00	100.00	99.83	98.78	100
Median (QR1,QR3)	87.65 (60.38, 99.04)	87.46 (83.97, 96.96)	85.82 (69.55, 96.18)	99.78 (98.30, 100.00)	87.57 (83.60, 93.69)	91.19 (77.97, 98.31)	96.71 (86.68, 99.54)	51.51 (36.94, 61.40)	52.60 (39.36, 63.32)	95.70 (84.49, 99.69)
IQR	38.66	12.99	26.63	1.70	10.08	20.34	12.86	24.46	23.95	15.2
Mean (SD)	78.77 ± 23.29	86.06 ± 13.06	81.38 ± 15.84	98.17 ± 3.88	86.06 ± 13.03	85.88 ± 15.08	85.36 ± 22.82	49.84 ± 17.40	53.09 ± 20.88	88.55 ± 15.40

OUTLIERS

2000

- *Central & Southern Asia*: Afghanistan (20.92), Tajikistan (44.57)
- *Eastern & Southeastern Asia*: NONE

- *Europe & Northern America*: Republic of Moldova (74.5), Lithuania (77.17), Albania (81.83), Russian Federation (85.78)
- *Latin America & the Caribbean*: Haiti (40.65), Peru (50.85), Paraguay (51.89)
- *Northern Africa & Western Asia*: NONE
- *Oceania*: Papua New Guinea (29.48)
- *Sub-Saharan Africa*: NONE

2015

- *Central & Southern Asia*: Afghanistan (53.48)
- *Eastern & Southeastern Asia*: NONE
- *Europe & Northern America*: Republic of Moldova (79.39), Albania (89.88), Russian Federation (90.22), Lithuania (92.6), Serbia (95.08)
- *Latin America & the Caribbean*: Haiti (40.29), Nicaragua (60.69)
- *Northern Africa & Western Asia*: NONE
- *Oceania*: Papua New Guinea (29.48), Kiribati (44.22), Solomon Islands (56.44)
- *Sub-Saharan Africa*: NONE

Urban

Urban: At Least Basic	World	Central & Southern Asia	Eastern & Southeastern Asia	Europe & Northern America	Latin America & Caribbean	Northern Africa & Western Asia	Oceania	Sub-Saharan Africa	Least Developed Countries	Not LDC
2000										
Minimum	48.37	49.81	74.92	91.11	84.94	60.32	77.09	48.37	48.37	74.02
Maximum	100	98.18	100.00	100.00	100.00	100.00	100.00	99.75	97.85	100
Median (QR1,QR3)	95.38 (86.22, 99.46)	94.94 (92.42, 97.47)	98.31 (89.12, 99.60)	100.00 (98.72, 100.00)	95.57 (93.64, 97.80)	97.67 (93.15, 99.50)	95.34 (90.11, 97.75)	79.32 (71.95, 85.65)	75.69 (71.20, 83.95)	97.71 (94.40, 99.88)
IQR	13.25	5.05	10.48	1.28	4.16	6.35	7.64	13.70	12.75	5.48
Mean (SD)	91.03 ± 11.06	91.85 ± 12.34	92.19 ± 9.47	99.01 ± 1.86	95.31 ± 3.34	93.85 ± 10.16	93.21 ± 6.92	78.92 ± 10.69	76.46 ± 11.02	96.12 ± 4.77

Urban: At Least Basic	World	Central & Southern Asia	Eastern & Southeastern Asia	Europe & Northern America	Latin America & Caribbean	Northern Africa & Western Asia	Oceania	Sub-Saharan Africa	Least Developed Countries	Not LDC
2015										
Minimum	59.56	88.98	82.40	88.06	81.02	73.08	70.33	59.56	59.56	70.33
Maximum	100	98.67	100.00	100.00	100.00	100.00	100.00	99.92	99.58	100
Median (QR1,QR3)	97.04 (89.58, 99.70)	96.07 (91.68, 97.27)	96.50 (93.23, 99.69)	99.97 (98.88, 100.00)	98.82 (97.47, 99.53)	98.52 (94.71, 99.26)	99.38 (91.97, 99.92)	84.78 (78.29, 89.47)	84.83 (77.97, 89.72)	98.89 (96.37, 99.97)
IQR	10.13	5.59	6.46	1.12	2.06	4.55	7.95	11.17	11.75	3.59

Urban: At Least Basic	World	Central & Southern Asia	Eastern & Southeastern Asia	Europe & Northern America	Latin America & Caribbean	Northern Africa & Western Asia	Oceania	Sub-Saharan Africa	Least Developed Countries	Not LDC
Mean (SD)	93.40 ± 8.62	94.52 ± 3.41	95.76 ± 4.82	98.92 ± 2.29	97.87 ± 3.56	94.78 ± 7.22	94.80 ± 8.62	83.50 ± 9.05	83.40 ± 9.22	97.02 ± 4.63

OUTLIERS

2000

- *Central & Southern Asia*: Afghanistan (49.81)
- *Eastern & Southeastern Asia*: NONE
- *Europe & Northern America*: Serbia (91.11), Republic of Moldova (95.38), Albania (95.62), Ireland (95.71), Lithuania (96.39), Ukraine (96.73)
- *Latin America & the Caribbean*: Haiti (84.94)
- *Northern Africa & Western Asia*: Sudan (60.32), Yemen (78.96)
- *Oceania*: Kiribati (77.09)
- *Sub-Saharan Africa*: Somalia (48.37)

2015

- *Central & Southern Asia*: NONE
- *Eastern & Southeastern Asia*: Myanmar (82.4)
- *Europe & Northern America*: Serbia (88.06), Albania (92.52), Republic of Moldova (95.62), The former Yugoslav Republic of Macedonia (95.79), Montenegro (96.7), Ukraine (96.89), Bosnia and Herzegovina (97.04)
- *Latin America & the Caribbean*: Haiti (81.02)
- *Northern Africa & Western Asia*: Sudan (73.08), Yemen (84.66), West Bank and Gaza Strip (85.66)
- *Oceania*: Marshall Islands (70.33)
- *Sub-Saharan Africa*: South Sudan (59.56)

Analysis

- The 2000 minimum was in Ethiopia (16.73), a sub-saharan african and least developed country.
- The next four lowest percentage countries were Eritrea (16.83), Somalia, (20.68), Mozambique (22.21), and Afghanistan (27.07).
- The 2015 minimum was in Eritrea (19.29), a sub-saharan african and least developed country.
- The next four lowest percentage countries were Papua New Guinea (36.6), Uganda, (38.92), Ethiopia (39.12), and Somalia (40).
- In 2000, only 52 of the 207 countries were below the 25th percentile (74.94), and 39 of those countries were from the sub-saharan african region.
- In 2015, only 57 of the 227 countries were below the 25th percentile (83.09), and 42 of those countries were from the sub-saharan african region.

Conclusions

I ended up not completing this project due to the amount of time in between finishing this project and doing school. I wanted to move on and look at other datasets that interested me. However, I'm still proud of the progress I made on this project and still decided it was worth posting.