**Folder Structure for Hard-to-REACH**

The folders contain the following:

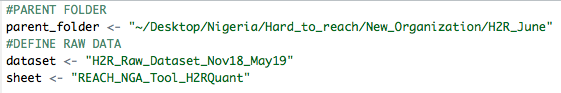
* **Raw\_to\_Cleaning\_Process**: The script that applies the cleaning log: “H2R\_Profiling\_one\_to\_many\_Cleaning.R”
  + **Raw\_Data:** The ***raw data file***and***cleaning log*** (same or different excel sheets).
  + **HQ\_Cleaning**: “H2R\_Profiling\_HQ\_Cleaning.R” and the outputs from the script in the folder:
  + **updated\_cleaning\_log**: The updated cleaning log output from “H2R\_Profiling\_one\_to\_many\_Cleaning.R”
  + **updated\_data**: The updated/cleaned dataset output from “H2R\_Profiling\_one\_to\_many\_Cleaning.R”
* **Analysis**: “H2R\_all\_in\_one.R” and the outputs from that script.
  + **Settlements\_Merged**: Folder where the settlement-level aggregated data will be stored.
  + **Global\_Results**: Folder where the global-level aggregated data will be stored.
  + **LGA\_Results**: Folder where the LGA-level aggregated data will be stored.
  + **Ward\_Results**: Folder where the Ward-level aggregated data will be stored.
  + **GIS\_Settlement\_list**: Folder where the GIS data (master list of settlements) is stored.
  + **Indicator\_Ranking:** Folder where “H2R\_Indicator\_Ranking.R” is located along with the necessary “INDICATORS\_TO\_RANK.xlsx” file.
    - **processed\_ranked\_outputs:** Folder where the rankings for each indicator are stored after processing.

**STEP #1.** Find outliers using “H2R\_Profiling\_HQ\_Cleaning.R”

**FIRST:** Run the entire “FIRST\_H2R\_PROFILE\_FUNCTIONS.R” script to load functions.

**Step 1a.**

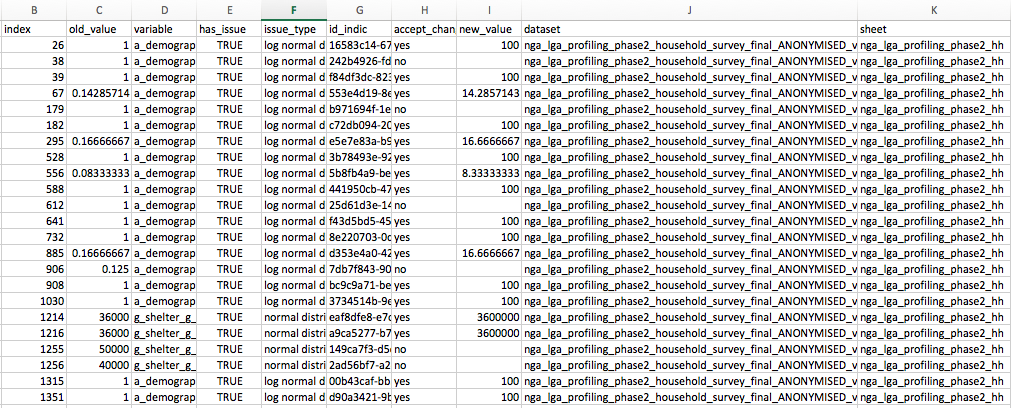
1. Ensure the working directory is set to the “Raw\_to\_Cleaning\_Process” folder.
2. Copy the name of the raw data excel file.
   1. NOTE: In this case the “folder” in the cleaning log should be defined as “Raw\_Data” (this is where the KoBo output is located).
3. Copy the name of the excel sheet where the raw data is located.
4. Check the parameters of the “hq\_cleaning\_checks” function.
5. Run the whole script.



The following error will occur:



However, a .csv file ending in “outlier\_id” is produced. Open this file.



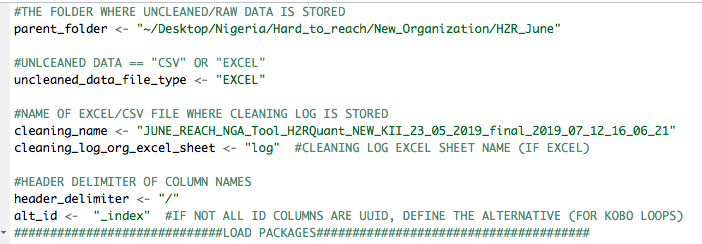
**Step 1b.**

1. Examine the proposed edits and write “yes” or “no” in the “accept\_change” column.
2. Then place the corrected value in the “new\_value” column.
   1. If the entry (survey) must be removed, write NULL (all caps) in the “new\_value” column.
3. Leave the “new\_value” cell blank if the change was not accepted.
4. Save this file as an “.xlsx” file using the exact same name in the “HQ\_Cleaning” folder.
5. Run the script again.
6. Open the “.csv” file ending in “hq\_changes” and copy and paste the contents of that file into the cleaning log.

**Step 2:** Apply the cleaning log to the raw data using “H2R\_Profiling\_one\_to\_many\_Cleaning.R”

1. Set the working directory to the “Raw\_to\_Cleaning\_Process” folder.
2. Copy the name of the cleaning sheet excel (or .csv) file.
3. Copy the name of the excel sheet where the cleaning log is located.
4. Define how the column names are separated—typically usuing “/”.
5. Define an alternative ID column header, other than “uuid” column.
   1. Example: “\_index” for KoBo loops edits—This is the default and generally should not be changed.

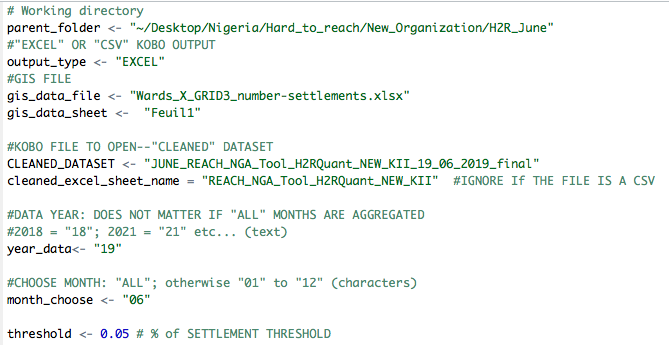
Run the whole script and find the updated data in the “updated\_data” folder and the updated cleaning log in the “updated\_cleaning\_log” folder.

****

**Step 3.** Settlement-level aggregation using “**Settlements\_Merged.R**”

**Step 1.**

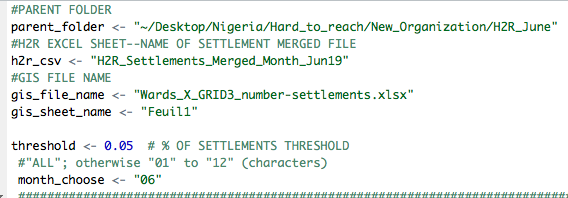
1. Set the working directory to the “parent” folder (“H2R\_June”).
2. Define the name of the excel file and sheet of the GIS settlement list data.
3. Define the name of the file (excel or .csv) file that contains the cleaned KoBo data.
4. Define the year of the dataset (2019= “19”); write “ALL” if data combining multiple months.
5. Define the month of the dataset (June = “06”); write “ALL” if data combining multiple months.
6. Define the threshold (percentage) of settlements per-ward to qualify for “enough” to be flagged in the settlement-level aggregation (5% threshold = 0.05).

****

**Step 4.** Ward & LGA-level aggregation using “**Ward\_LGA\_Results.R**”

**Step 1.**

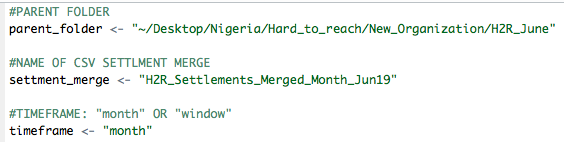
1. Set the working directory to the “parent” folder (“H2R\_June”).
2. Define the name of the aggregated settlement-level data.
3. Define the name of the excel file and sheet of the GIS settlement list data.
4. Define the threshold (percentage) of settlements per-ward and per-LGA to qualify for “enough” to be retained in the aggregated output (5% threshold = 0.05).
5. Define the month of the dataset (June = “06”); write “ALL” if data combining multiple months.

****

**Step 5.** Global-level aggregation using “**Global\_Results.R**”

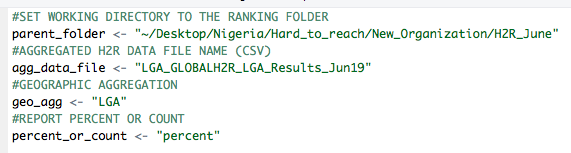
**Step 1.**

1. Set the working directory to the “parent” folder (“H2R\_June”).
2. Define the name of the aggregated settlement-level data.
3. Define the timeframe of the data (month or window [multiple months]).

****

**Step 6.** Rank multiple option indicators using “**Indicator\_Ranking.R**”**.**

1. Define the parent folder of the H2R edition.
2. Define the name of .csv file with the aggregated data.
3. Define the column name with the geographic aggregation units.
4. Define whether to report counts of percentages.
5. The remaining parameters of the function will be defined *as long as the folder structure is exactly followed.*



**Select-all and run.**

***NOTE****: Indicator names must not be sub-strings other indicators in the list to be ranked. For example, if “N\_info\_source” and “N\_info\_source\_who” are included in the list, then “N\_info\_source” will not rank correctly because it includes the “N\_info\_source\_who” values as well because ““N\_info\_source\_who” contains the string “N\_info\_source.” In short, the script will not be able to identify the difference between “N\_info\_source” and “N\_info\_source\_who.”*

*Avoid these indicators (if possible) or create a distinct column header for troublesome indicators; for example, “N\_info\_source” becomes “N\_info\_source\_main.”*