Date Created: 2021 June 15

Last Updated: 2021 July 21

README for Obertegger Tovel data

This document details the data cleaning and harmonization procedure performed for the Zooplankton as Indicators working group (ZIG). It is meant to highlight major cleaning steps taken to make the dataset compatible with other submitted dataset. Any questions about the larger ZIG project can be submitted to:

- Steph Figary (sef92@cornell.edu)
- Michael Meyer (michael.f.meyer@wsu.edu)
- Warren Currie (warren.currie@dfo-mpo.gc.ca)

This submitted dataset was cleaned by
Michael F. Meyer (michael.f.meyer@wsu.edu).

Cleaning Script and Session Info

The script `000a_oberteggger_tovel_meyer.R` was used to check data values, correct values formats, and generally assess the data for outliers. This scripts requires:

- **Inputs**: This script requires the submitted Obertegger input data file `CORRECTED_ZIG_data_Obertegger_Tovel_vs2 - Ulrike Obertegger.xlsx`.

- **Outputs**: This script outputs the following files to the directory `data/derived_products/obertegger_toverl_disaggregated`:

    - `additional_data_obertegger.csv`
    - `complete_lake_station_water_obertegger.csv`
    - `complete_lake_station_zooplankton.csv`
    - `equipment_clean_obertegger.csv`
    - `lake_information_obertegger.csv`
    - `lake_timeline_obertegger.csv`
    - `station_information_obertegger.csv`
    - `taxa_list_obertegger.csv`
    - `complete_lake_station_zooplankton.csv`
    - `water_parameters_obertegger.csv`
    - `zooplankton_abundance_obertegger.csv`
    - `zooplankton_length_obertegger.csv`

  The script requires the following directory tree:

```
|───data
|   ├───derived_products
|   |   └───obertegger_tovel_disaggregated
|   └───inputs
├───figures
```

```
│   └───obertegger_tovel_qc
└───scripts
```

This dataset was cleaned using the R Statistical Environment (R Core Team 2019). The following session information describes the computational environment, in which the cleaning procedure occurred.

```
- Session info ----------------------------------------
 setting  value
 version  R version 3.6.2 (2019-12-12)
 os       Windows 10 x64
 system   x86_64, mingw32
 ui       RStudio
 language (EN)
 collate  English_United States.1252
 ctype    English_United States.1252
 tz       America/Los_Angeles
 date     2021-06-15

- Packages --------------------------------------------
 package     * version date       lib source
 assertthat    0.2.1   2019-03-21 [1] CRAN (R 3.6.2)
 backports     1.1.8   2020-06-17 [1] CRAN (R 3.6.2)
 broom         0.5.3   2019-12-14 [1] CRAN (R 3.6.2)
 cellranger    1.1.0   2016-07-27 [1] CRAN (R 3.6.2)
 cli           2.3.1   2021-02-23 [1] CRAN (R 3.6.3)
 colorspace    2.0-0   2020-11-11 [1] CRAN (R 3.6.3)
 crayon        1.4.1   2021-02-08 [1] CRAN (R 3.6.2)
 DBI           1.1.0   2019-12-15 [1] CRAN (R 3.6.2)
 dbplyr        1.4.2   2019-06-17 [1] CRAN (R 3.6.2)
 dplyr       * 1.0.4   2021-02-02 [1] CRAN (R 3.6.2)
 ellipsis      0.3.1   2020-05-15 [1] CRAN (R 3.6.3)
 fansi         0.4.2   2021-01-15 [1] CRAN (R 3.6.3)
 forcats     * 0.4.0   2019-02-17 [1] CRAN (R 3.6.2)
 fs            1.3.1   2019-05-06 [1] CRAN (R 3.6.2)
 generics      0.1.0   2020-10-31 [1] CRAN (R 3.6.3)
 ggplot2     * 3.3.3   2020-12-30 [1] CRAN (R 3.6.3)
 glue          1.4.2   2020-08-27 [1] CRAN (R 3.6.3)
 gtable        0.3.0   2019-03-25 [1] CRAN (R 3.6.2)
 haven         2.2.0   2019-11-08 [1] CRAN (R 3.6.2)
 hms           0.5.3   2020-01-08 [1] CRAN (R 3.6.2)
 httr          1.4.1   2019-08-05 [1] CRAN (R 3.6.3)
 jsonlite      1.7.2   2020-12-09 [1] CRAN (R 3.6.3)
 lattice       0.20-38 2018-11-04 [2] CRAN (R 3.6.2)
 lifecycle     1.0.0   2021-02-15 [1] CRAN (R 3.6.3)
 lubridate     1.7.4   2018-04-11 [1] CRAN (R 3.6.2)
 magrittr      2.0.1   2020-11-17 [1] CRAN (R 3.6.3)
 modelr        0.1.5   2019-08-08 [1] CRAN (R 3.6.2)
 munsell       0.5.0   2018-06-12 [1] CRAN (R 3.6.2)
 nlme          3.1-152 2021-02-04 [1] CRAN (R 3.6.3)
 pillar        1.5.1   2021-03-05 [1] CRAN (R 3.6.3)
 pkgconfig     2.0.3   2019-09-22 [1] CRAN (R 3.6.2)
 purrr       * 0.3.4   2020-04-17 [1] CRAN (R 3.6.3)
 R6            2.5.0   2020-10-28 [1] CRAN (R 3.6.3)
 Rcpp          1.0.6   2021-01-15 [1] CRAN (R 3.6.3)
 readr       * 1.3.1   2018-12-21 [1] CRAN (R 3.6.2)
 readxl      * 1.3.1   2019-03-13 [1] CRAN (R 3.6.2)
 reprex        0.3.0   2019-05-16 [1] CRAN (R 3.6.2)
 rlang         0.4.10  2020-12-30 [1] CRAN (R 3.6.3)
 rstudioapi    0.13    2020-11-12 [1] CRAN (R 3.6.3)
 rvest         0.3.5   2019-11-08 [1] CRAN (R 3.6.2)
 scales        1.1.1   2020-05-11 [1] CRAN (R 3.6.3)
 sessioninfo   1.1.1   2018-11-05 [1] CRAN (R 3.6.2)
 stringi       1.4.6   2020-02-17 [1] CRAN (R 3.6.2)
 stringr     * 1.4.0   2019-02-10 [1] CRAN (R 3.6.2)
 tibble      * 3.1.0   2021-02-25 [1] CRAN (R 3.6.3)
 tidyr       * 1.1.2   2020-08-27 [1] CRAN (R 3.6.3)
 tidyselect    1.1.0   2020-05-11 [1] CRAN (R 3.6.3)
 tidyverse   * 1.3.0   2019-11-21 [1] CRAN (R 3.6.3)
 utf8          1.2.1   2021-03-12 [1] CRAN (R 3.6.3)
 vctrs         0.3.6   2020-12-17 [1] CRAN (R 3.6.3)
 withr         2.4.1   2021-01-26 [1] CRAN (R 3.6.3)
 xml2          1.3.2   2020-04-23 [1] CRAN (R 3.6.3)
```

# Notes from MFM while cleaning Obertegger data:

- When cleaning water parameters, NAs were coded as characters and not left blank. This meant that I performed a `grep` statement to find and replace NA character strings with coded NA values.
- When cleaning water parameters, dataset owner used `<` to indicate a value less than a minimal detection limit. Like NAs, I performed a `grep` statement to flag these values, and then used a `gsub` statement to remove the less than symbol and divided the numeric value in half.
- The main issue with the Obertegger data is that there is a discrepancy with two sampling time points, leading to the dataset not being entirely interoperable based on lake, station, year, month, day of month. After a data harmonization team meeting on 17 June 2021, the team decided to think on potential solutions, and we can return to the topic for later. As for now, the dataset remains in two separate CSV files, which future users can decide to harmonize based off their preferred merging schema.