Thermosalinograph Codebook

Yanique Campbell

April 26 2022

Contents

1 Introduction

1.1 Description

This codebook describes the script used in the intermediate stage to process Thermosalinograph (TSG) data without date and time measurements. It reads the CNV files produced by the SBE Data processing software, extracts the metadata and data and creates a date-time column based on the start date/time and interval. It then saves them as CSV files with the appropriate headers.

2 Setup

Visit https://docs.python.org/3/library/ to learn more about these packages

2.1 Script-Specific options or packages required

1. datefinder-
   A python module for locating dates inside text.
   Installation: pip install datefinder
2. pandas-
   Pandas is a module used to extract data from input files, and also for data analysis and manipulation.
3. datetime-
   Datetime is a module that supplies classes for manipulating dates and times.
4. os-
   The Operating system(os) package is a package used to provide a means to access certain operating system specific functionality. Using the Os package, we can define certain operating system specific behaviour for different operating systems.
5. time-
   This module provides various time-related functions.

3 Script Description

Script name(s): cnv2csv.py, ini.py

3.1 ini.py

This script contains the path to the input directory for the raw files and the output directory for the processed files. The user can set the path to whatever they choose here. For example:

```
#Enter the directories for input and output files
input_directory_TSG='/Users/yaniquecampbell/Desktop/TSG'
output_directory_TSG='/Users/yaniquecampbell/Desktop/TSG/Output'
```

Ensure that the raw files are in the input folder as specified in this file, and the output folder is created.

NOTE: Paths are written differently for different operating systems. For macOS and Linux systems, a forward-slash is used, while for Windows a backslash to separate directories in file paths.

3.2 cnv2csv.py

3.2.1 Steps

1. Import the user-defined directories from the ini.py script
2. Create a loop to read all the CNV files. The subsequent steps are performed for each file.
3. Begin reading the CNV file line by line
4. Extract first group of metadata into a dictionary. These metadata are preceded by * in the CNV file. The script grabs this information, and creates a dictionary for the metadata, with keys and values separated by either '=' or ':'. For example:

   `Ship:William Kennedy`

   A while loop is used here:

   ```
   while allData[i][0] == '*':
   ```

5. Extract second group of metadata preceded by #. This metadata is categorized by:
   - Metadata containing the variable names and units
   - Metadata containing the span (range) of the data

- Metadata containing `< metadata >` which is saved as XML metadata
- The remaining metadata

Again, a while loop is used to get all the metadata starting with #:

```
while allData[i][0] == '#':
```

6. Check for the start date/time tag: `start_time` and extract the date
7. Check for the time interval (tag containing the word `interval`)
8. Extract the variable names
9. Extract the actual data. This data starts at the end of all the metadata in the CNV files. Therefore, after both while loops mentioned above are fulfilled.
10. Create a data frame from the variable names (column headers) and the data
11. Calculate a range of dates starting with the start date and interval.
12. Add as a column in the data frame
13. Save as a CSV file

3.3 Run cnv2csv.py from a command-line interpreter

To run this script on command-line you will require a version of python2.7 or higher installed. If you do not already have python visit https://www.python.org/downloads/ to download python. The package `datefinder` must also be installed. To install:

pip install datefinder

If you do not have pip installed on with your version of python, instructions on how to install pip can be found here

3.3.1 Windows

1. Open search by pressing the windows button and type "python" into the search bar. If you do have python installed you will find a folder named pythonxx (xx being the version number. eg. python37). Within this folder, you will find a python.exe file. You will require the path to this folder in step 3.
2. To open command line, type "cmd" into the search bar.

3. After opening terminal type:

   `C:\pythonxx\python.exe C:\path\script_name.py`

   For example, if cnv2csv.py is located on the desktop in a folder called **code** then you would type:

   `C:\pythonxx\python.exe C:\Desktop\code\cnv2csv.py`

3.3.2 Mac OS

1. To open command line, use keyboard shortcut "Command + Space" to open Spotlight search, type in "Terminal"
2. After opening Terminal, type:
   `Python path/script_name.py.`
   For example, if cnv2csv.py is located on the desktop in a folder called **code** then you would type:
   `Python /Users/yaniquecampbell/Desktop/code/cnv2csv.py`

3.3.3 Linux

1. To open command line, use keyboard shortcut "Ctrl + Alt + T" to open terminal. You also could use the search option and type in Terminal
2. After opening terminal, type:
   `python path/script_name.py`.
   Example, if cnv2csv.pyis located on the desktop in a folder called **code** then you would type:
   `python Desktop/code/cnv2csv.py`.

4 Log

For more information on python visit https://https://www.python.org/