**READ ME: Serial incubations project - temperature data carpentry**

We deployed HOBO pendant temperature loggers in each stream, which recorded temperature data at 15-minute intervals for the duration of our study. However, due to equipment malfunctions, there were some time periods when we lost temperature data. We used linear regressions to model missing data based on either air temperature data from CS01 at Coweeta or other nearby streams. This document details the process of modeling and filling in the missing data, generating files with daily-level mean temperature data, and calculating deployment-level average temperature data.

**Modeling – Year 1**

I. In year 1 of our study, we lost temperature data in all streams for the time between 8 March – 5 April 2018 due to a shuttle malfunction. The steps we took to fill in these missing data are:

**1.** We started with all the YR1 temperature data in 2 separate CSVs per stream. The first CSV contained data from the study start date (21 September 2017) to 8 March 2018 when we began to lose data. These files have “thrumarch2018” at the end of the file name. The second CSV contained data from 5 April 2018 to the study end date (28 September 2018). These files have “Apr-Oct18” at the end of the file name. Note that “WS08” is called “WSLSF” in these file names.

**2.** We created .csv files with all the data except the 8 March – 5 April 2018 by copying and pasting the “Apr-Oct” data below the “thrumarch” data. We did this for each individual YR1 stream. The resulting files have “thrumarch\_apr-oct\_2018” at the end of the file name. We created these files so that we would be able to merge stream temperature data with air temperature data across the date ranges where we did have data, allowing us to create regressions based on as much available data as possible. Note that “WS08” is called “WSLSF” in these file names.

**3.** We downloaded daily mean air temperature data from Climate Station 1 at Coweeta, which is publicly available at this link: <https://www.srs.fs.usda.gov/coweeta/tools-and-data/>. We created a new .csv from this master file with only data for the subset of dates that overlapped with our stream temperature data. (21 September 2017 – 8 March 2018 and 5 April 2018 – 28 September 2018). This subsetted air temperature data file is called “ave.daily.temps.csv”. We also created a .csv file containing only air temperature data for 8 March 2018 - 5 April 2018 (the period for which we were missing stream temperature data). This file is called “ave.daily.temps.mar.apr.csv”.

**4.** We created an R script called “stream.air.tempmods.R”, where we created stream-specific linear regressions between stream temperature and CS01 air temperature.

A. In this script, we first read in the “thrumarch\_apr-oct\_2018” data files for each stream. Since these data were at the 15-minute interval resolution, we had to summarize it to daily means so that the resolution of the stream temperature data would match that of the CS01 air temperature data. We used the “aggregate” function to create new dataframes with daily temperature averages for each individual stream.

B. For each individual stream, we merged the air temperature and stream temperature files together and used the resulting dataframes to create linear regressions. We used the function “lm” in R to model average stream temperature as a function of average air temperature.

C. We used the slopes and y-intercepts derived from the stream-specific lm models to model stream temperature for the missing period based on the average air temperature data in “ave.daily.temps.mar.apr”. A table of stream-specific model parameter estimates and R^2 values can be found in Table 1.

D. We also repeated the process outlined above to model WS08 (Lower Shope Fork) for 31 August 2018 – 28 September 2018. We were missing temperature data for this time only for this one stream. We wrote these modeled data into a separate .csv file (“WS08\_dailies\_31Aug-29Sep\_2018.csv”).

E. We used the bind\_rows function to place the modeled data from all 12 Year 1 streams into one dataframe. This dataframe was named “all.modeled.data” and was written into a .csv file. It was later combined with the rest of the daily temperature data (see “Generating Daily Means” section below).

**Modeling – Year 2**

I. For year 2, we were missing data from both Coweeta Creek (CWCR) and Lower Ball Creek (WS09) for different time periods. Since Lower Ball Creek is a tributary of CWCR and our sampling points for these two streams are close together, we decided to generate linear regressions for these two streams based on one another to fill in the missing data. We did this in the Rmd file entitled “CWCR\_WS09\_Modeling\_YR2.R”

1. For CWCR, we were missing temperature data for 9/18/18 16:34 – 12/5/18 15:19. For WS09, we were missing temperature data for 1/7/19 15:35 - 2/6/1913:35.

2. To generate our regressions, we used a period for which we had overlapping data from both streams – 2/7/19 – 08/29/19 - these .csv files are called “CWCR\_7Feb19\_thru\_29Aug19.csv” and “WS09\_7Feb19\_thru\_29Aug19.csv”.

2. Using the regressions, we filled in the missing temperature data for both streams in Excel. The .csv files that contain the modeled data are called “WS09\_modeled.csv” and “CWCR\_modeled.csv”. There are notes in these csv’s about the process we used for filling in time and temperature data. We copied and pasted the modeled values into the master temperature files for WS09 and CWCR, respectively (“WS09\_10951961\_all.csv” and “CWCR\_10951959\_all.csv” contain these modeled values).

3. Note that, since we had 15-minute temperature data for both CWCR and WS09, we used this data resolution to generate the regression models and fill in the modeled data.

**Generating Daily Means**

I. We created an R script called “daily\_temps\_script\_SI\_BOTH\_YRS\_CSC\_16Nov2020” which we used to generate files with all the daily temperature data.

1. Since the temperature data we modeled for YR1 was in daily mean format, we aggregated all the temperature data at this level so that the data would be at a consistent resolution among months when generating deployment-level means.

2. Once all temperature data was aggregated at the daily mean level, we bound all files for a given stream (including modeled and non-modeled data) in the correct order. We then bound all these dataframes to generate master daily temperature files for both YR1 and YR2 (landscape\_dailytemp\_masterlong\_SIyr1.csv and landscape\_dailytemp\_masterlong\_SIyr2.csv).

**Generating deployment-level means**

We used the R scripts “Deployment\_averages\_SIYR1.R” and “Deployment\_averages\_SIYR2.R” to summarize the daily temperature data at the deployment level. Deployment-level temperature averages were used in subsequent analyses

(File names: “landscape\_deployment\_temp\_masterlong\_SIyr1.csv” and “landscape\_deployment\_temp\_masterlong\_SIyr2\_UPD23Feb23”