Instruction manual for

single calibration R-scripts

Updated: J. Davis 2024-05-02

# Description:

The R Project is intended to facilitate model calibration by using ‘hydroGOF’ and the ‘tidyverse’ suite of R packages (Wickham 2017, Zambrano-Bigiarini 2017) to generate common calibration plots and statistics (Moriasi 2007). Based on user-defined options in the Excel file, the scripts generate a variety of publication quality tables and figures that allow the user to apply a qualitative (e.g, time series and probability plots) and quantitative (e.g., goodness of fit statistics) approach. The user can also convert static figures into interactive HTML plots that allow figures to be displayed spatially.

This R Project is designed to accept model input data as either raw LSPC \*.OUT files, WASP \*.BMD2 or a PREPROCESSED \*.txt datafile that has already been preprocessed and extracted. When applied to LSPC \*.OUT files, the user must provide the filepath for the location of the \*.OUT files and select ‘LSPC’ on the controller tab. The scripts will then read all \*.OUT files saved within that filepath, extract the necessary segments and parameters, and write out a fully formated \*.txt of extracted data. If the user provides a raw model \*.BMD2 file and selects ‘WASP’ on the controller tab, the scripts will automatically use the ‘Extract\_BMD2.exe’ to open the raw model output (\*.BMD2), extract the necessary segments and parameters, and write out a fully formated \*.txt of extracted data. If no raw \*.BMD2 file is supplied, the R Project will bypass the data extraction step and will instead require a fully preprocessed/extracted model \*.txt as the model data source. In any workflow, model input data are subsequently paired with preprocessed observed data (i.e., field monitoring data) and linearly interpolated to calculate a simulated value at the exact time the monitoring data record was collected. Prior to its pairing with measured data, simulated model flow is aggregated to calculate an average daily flow record to improve its comparison to average daily flows reported by USGS gages. The subsequent output generated by these scripts is modular, whereby the Excel \*.xlsx spreadsheet can be used to select a range of calibration plots and statistics.

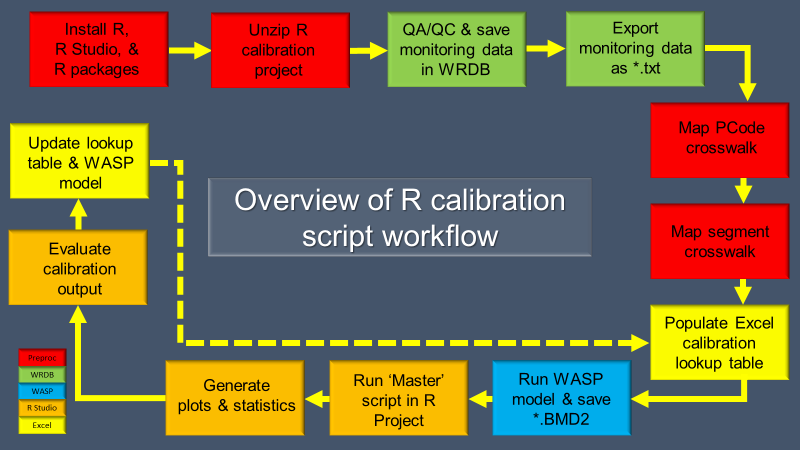
The scripts are designed as a R Project in RStudio, which provides a file organization system that allows the user to copy/paste the R Project folder into an existing modeling project. This interface also allows the user to customize options without requiring indepth understanding of R syntax. Alternatively, the master script can be run in \*.bat mode through Windows command line, which can be especially useful when processing output from multiple successive model runs. However, this more advanced option requires additional understanding of how to create .bat files and how the various folders/subfolders are organized.

Prior to running the calibration scripts, the user will need

* RStudio, R Statistical Software (v4.4.0), and required R packages to be pre-installed
* Model output: WASP \*.BMD2, LSPC \*.OUT or preprocessed model \*.txt
* Fully QA/QC’d field measured (i.e., calibration) data
* Crosswalk of model segments and calibration monitoring stations
* Crosswalk of model and measured parameters/analytes (i.e., PCodes; NOX = nitrate + nitrite)

Once completed, the scripts will generate a variety of output, including the following:

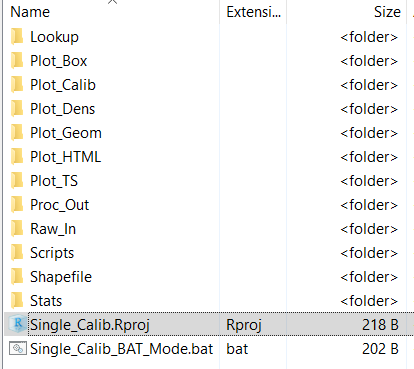
* Multipanel calibration figure w/ time series, probability, and 1:1 linear regression plots, and a summary table with common goodness of fit statistics.
* Boxplot of measured and simulated data
* Scatter plot of geometric means for chlorophyll-*a*, total nitrogen, and total phosphorus
* Data distribution density plot of measured vs. simulated data
* Interactive HTML plots of multipanel calibration and data distribution plots
* An Excel file with goodness of fit statistics, summarized by PCode and monitoring station.
* Processed output that can be used to troubleshoot crosswalk errors



# Set-up required prior to running R scripts

## Unzipping R Project folder

1. As currently designed, the calibration R Project can be unzipped into the user’s preferred model/filepath directory. It is possible to move the subfolders to other directories outside of the R Project and run the scripts in \*.bat mode (see below). However, it is highly recommended that the user initially run these scripts as an R Project with all required folder/subfolders contained in the original R Project file directory.
2. Unzip the R Project calibration file and save the entire folder in the desired location. This can be anywhere on the C-drive because the R Project automatically knows its location. When unzipped, the following folders and files should be available.



Use this to open calibration project in R Studio

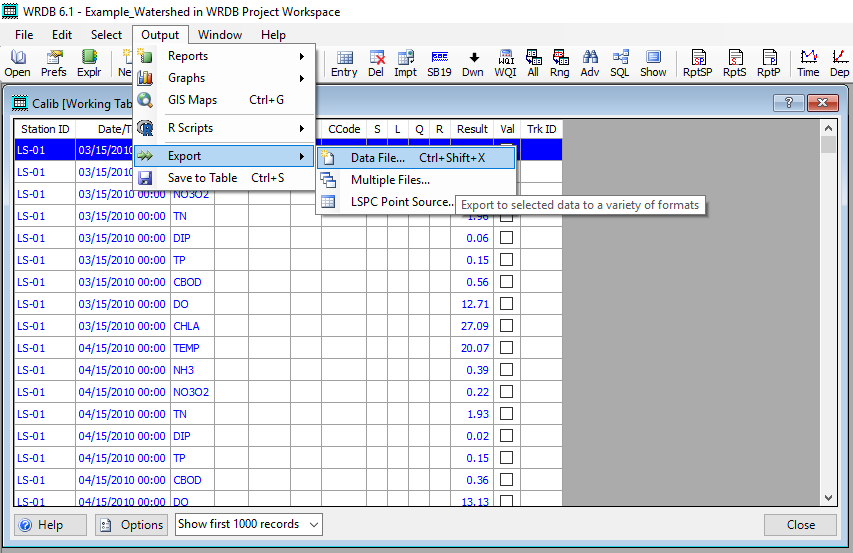
1. When necessary, the entire R Project can be copy/pasted into a new filepath to accommodate the calibration of a new model. Only the options and crosswalks in the R Calibration Lookup will need to be updated to accommodate the new model. This allows the user to create and archive multiple versions of the calibration scripts that are individually tailored to specific water quality models.

## Crosswalk monitoring/calibration stations and analytes

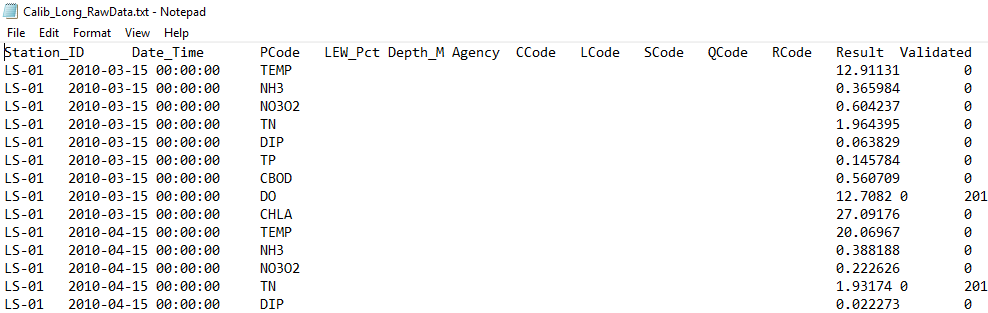
1. Before any attempt to run or set-up the calibration scripts, the user must manually determine how model parameter output translate to monitoring data parameters, and how calibration stations should be mapped to model segments. When calibrating WASP, this information will populate the ‘PCode\_Lookup’ and ‘Station\_Lookup’ tabs, respectively, in the Excel \*.xlsx lookup table (see below). The ‘LSPC\_Station’ and ‘LSPC\_PCode’ tabs are used when calibrating LSPC model output. Model and measured data need to be in the same units, and any unit conversion must be applied prior to importing into R.
   * **NOTE:** The crosswalk and arguments provided in the \*.xlsx file will also be used to populate the Control.dat file that will be used by the ‘Extract\_BMD2.exe’ to extract and preprocess the raw model output.
2. Station crosswalking can be done by overlaying the model network, watershed, and monitoring station shapefiles in a geographic information system (GIS; BASINS, ArcMap, QGIS).
   * When identifying which calibration stations to use and how to map them to model segments, consider relative position of stations in the watershed (e.g., downstream stations may be more informative than headwater stations), examine the number of data records available at each station (e.g., a station with only a 1-2 measurements may not be an effective calibration dataset), and pay particular attention to the position of monitoring stations relative to tributaries, point sources, and confluences.
3. For WASP, model segment names and model analyte names can be determined by opening the raw model \*.BMD2 in BMDUtil, which is a standalone program that is installed with WRDB or WASP. The program can be found by searching your hard drive for BMDUtil.exe
4. Monitoring station names can be determined by examining the Station\_ID used within the station support table in WRDB.

## Preprocess monitoring data for R scripts

1. This walkthrough example assumes the user has an existing WRDB project file that already contains QA/QC’d monitoring data. If monitoring data are stored in another file format (e.g., \*.xlsx or \*.csv), datafiles will need to be preprocessed into the same \*.txt file format as the example data below. At minimum, the tab-delimited \*.txt file will need the following column headings, although the cells can be empty: Station\_ID (i.e., unique monitoring station identifier), Date\_Time, PCode, Depth\_M, CCode, RCode, Result. The format of the Date\_Time column needs to be in YYYY-mm-dd hh:mm:ss format.
   1. **NOTE:** The compiled monitoring data should not include any duplicate records because these may cause issues when pairing and interpolating the model data.
2. To generate the necessary \*.txt from an existing WRDB project, open WRDB and the working table that contains the necessary calibration data records. Export all calibration records from WRDB (Output 🡪 Export 🡪DataFile). Save the file in a single \*.txt. **NOTE:** Do not save the monitoring data as a R Dataframe, only as a \*.txt.



1. Name the calibration \*.txt file. The filename will need to match the name provided in the ‘Calib\_Data\_In\_Filename’ field in the Excel calibration lookup file (i.e., Calib\_Data\_In\_Filename) and will need to be located in the directory indicated by the ‘Filepath\_Calib\_Data\_In’ argument (see below). If no filepath is provided for the ‘Filepath\_Calib\_Data\_In,’ the scripts will default to the ‘Raw\_In’ subfolder within the working directory of the RProject.
2. When viewed in a text editor, the .txt should have format similar to what is shown below.



## LSPC model \*.OUT files: Preprocess model output for R scripts

The current version of the calibration scripts can be used to assess the calibration LSPC model runs relative to the ambient monitoring data. When applying the calibration scripts to LSPC .OUT files, no preprocessing of the \*.OUT files is required; however, all of the \*.OUT files must be saved in the same filepath and must be exported from LSPC in ‘Standard’ format. The scripts also assume the \*.OUT files are named based on the LSPC subbasin identifier (i.e., 102.out). When calibrating LSPC output, select ‘LSPC’ as the ‘Selected\_Model’ on the Controller tab and use the ‘LSPC\_Station’ and ‘LSPC\_PCode’ tabs to map LSPC reaches and parameters to their appropriate values in the ambient monitoring data. See details below for how to set additional arguments.

## WASP model \*.BMD2 files: Preprocess model output for R scripts

The current version of the calibration scripts can be applied to a raw model \*.BMD2 file. If a \*.BMD2 filename is provided in the Excel lookup table, the scripts will verify that the \*.BMD2 file exists in the provided filepath. It will then use the ‘Extract\_BMD2.exe’ and its associated \*.DLLs to open the raw model \*.BMD2 file, extract the selected parameters and model segments (based on the Excel lookup table), and write out a formatted \*.txt file. The name of the raw \*.BMD2 file and extracted \*.txt file are based on the ‘Raw\_BMD2\_Filename’ and ‘Extract\_Model\_Filename’ fields reported in the Excel lookup, respectively. The extracted model results are saved to the filepath indicated by the ‘Filepath\_Model\_Data\_In’ argument, or will default to the ‘Raw\_In’ subfolder in the RProject working directory (if no filepath is provided). When calibrating WASP output, select ‘WASP’ as the ‘Selected\_Model’ on the Controller tab and use the ‘Station\_Lookup’ and ‘PCode\_Lookup’ tabs to map WASP segments and parameters to their appropriate values in the ambient monitoring data.

The .exe is called by the Master R script via a SOURCE/SHELL function. Therefore, no preprocessing of the raw model output is necessary. However, the \*.exe file requires a Control.dat that is automatically generated by the scripts based on arguments selected in the Excel lookup table. This Control.dat will be saved to the filepath of the Scripts.

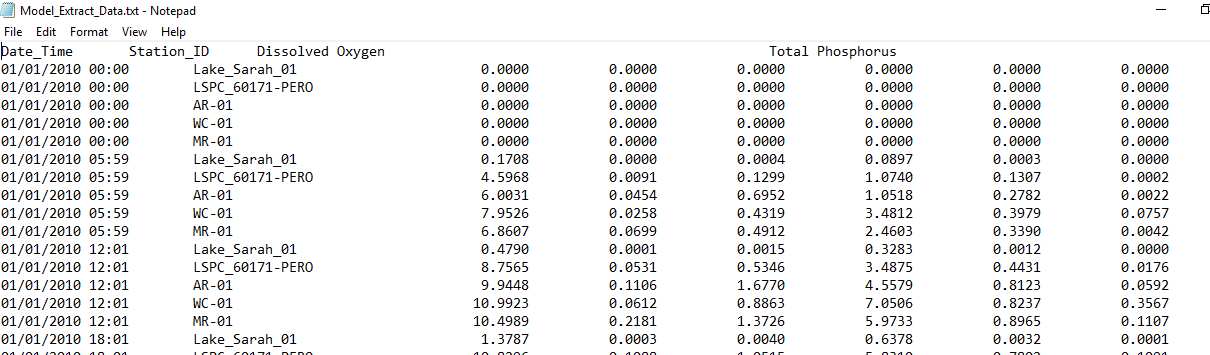
**NOTE**: All parameters selected for calibration in the Excel lookup table \*\*must\*\* be contained within the model \*.BMD2 and the monitoring data \*.txt. If a parameter or model segment is selected for calibration, but does not appear in the \*.BMD2 or \*.txt, the ‘Extract\_BMD2.exe’ may extract incorrect data for a particular parameter.

## Preprocessed model \*.TXT files: Preprocess model output for R scripts

Conversely, if ‘Preprocessed’ is selected as the model type, the user can leave the \*.BMD2 filename argument blank. The R scripts will assume the user has already extracted and preprocessed the required model input data. This allows users to apply the scripts to other model output, as long as the datafile matches the format of the \*.txt file. The R scripts will use the ‘Extract\_Model\_Filename’ and ‘Filepath\_Model\_Data\_In’ to locate the \*.txt file and import it. When the \*.txt file is used as the input data source, the input datafile must be in a ‘wide’ format data format with a Date\_Time column, unique Station\_ID column, and each parameter reported as a separate data column (see example table below). Parameter names (i.e., column headings) must exactly match the spelling of the ‘Anal\_Name’ listed in the Excel lookup table. When calibrating preprocessed data output, select ‘Preprocessed’ as the ‘Selected\_Model’ on the Controller tab and use the ‘Station\_Lookup’ and ‘PCode\_Lookup’ tabs to map WASP segments and parameters to their appropriate values in the ambient monitoring data.

**NOTE**: All parameters selected for calibration in the Excel lookup table \*\*must\*\* be contained within the monitoring data \*.txt. At least one of the selected parameters/model segments must be contained in the model \*.txt file.

### Format of the extracted model output (\*.txt)





# Create R Project

## Learning how to create and work with R projects:

Although the R calibration scripts are designed to allow a user to run the scripts with minimal understanding of R syntax, a working knowledge of the R language will facilitate the process and help troubleshoot. The implementation of the R calibration scripts as an R Project also helps to organize the various input and output files, allowing the user to copy/paste the entire project folder to new filepaths with minimal to no changes to the underlying code.

When no filepath is provided in the Excel calibration lookup table, filepaths will default to the location of the R Project \*.RProj file. However, if the user elects to move the various subfolders to filepaths outside of parent directory of the R Project, then the new filepaths will need to be indicated in the Excel lookup table. See below for the particular subfolders that are associated with a specific filepath.

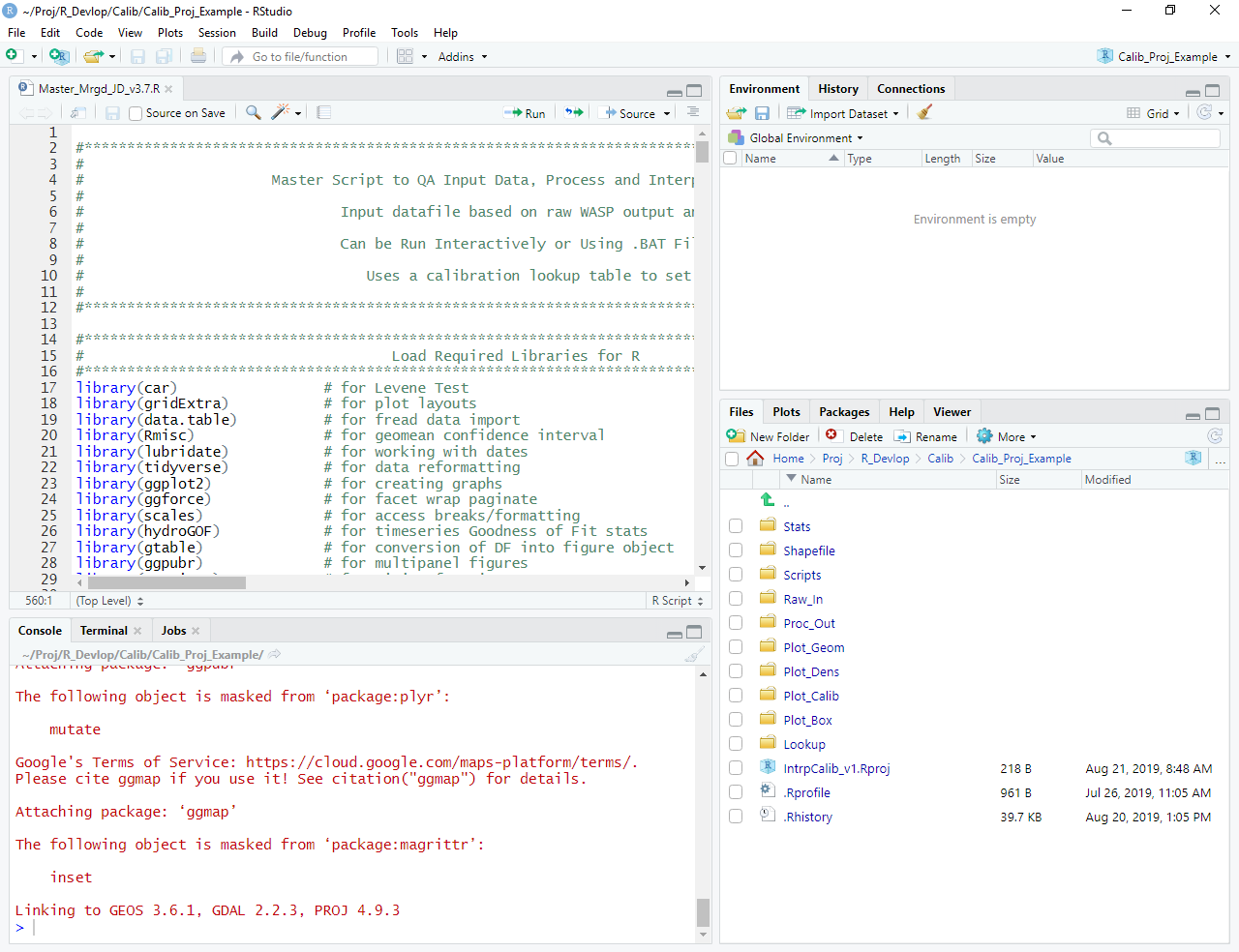
The individual scripts are designed to be run in sequence and as source codes within RStudio. Therefore, a basic knowledge of creating and working with R Projects in RStudio will be beneficial. A variety of resources and tutorials (e.g., <https://stackoverflow.com>) are available online for learning rudimentary R functions/syntax and working with RStudio (for an overview of creating R projects in Rstudio see <https://support.rstudio.com/hc/en-us/articles/200526207-Using-Projects>).

It is recommended to periodically update your version of RStudio, R Statistical Software, and the R packages to make sure you are running the most up to date versions.

## Overview of R project pane

The easiest way to run the associated calibration scripts is to run them interactively as an R Project through RStudio. The R project can be opened by double clicking the R Project file (Single\_Calib.Rproj) that is located in the parent directory of the calibration package.

This will open RStudio, which will display a variety of panels that can be customized (Tools 🡪 Global Options 🡪 Pane Layout). However, the format displayed below shows the Source pane with the currently active script (upper left panel), the list of objects that were created in the current R environment (upper right panel; no objects will be available until the scripts are run), the R console which represents the actual R Statistical Program where the scripts are executed (bottom left panel), and the folders contained within the working directory of the R Project (bottom right).



## Pre-installing R packages

The scripts included in this R Project require a variety of R packages that are listed in the table below and will need to be downloaded into your local R library. This will save the various packages locally and allow the scripts to call the required functions as necessary.

If not prompted by RStudio to download missing R packages, the following line of code can be copied and pasted into the R console (after the “>” symbol) and pressing ‘Enter.’ This will tell R to download the required packages and install them to your local R package library.

install.packages(c("car", "gridExtra", "data.table", "Rmisc", "lubridate", "tidyverse", "ggplot2", "ggforce", "scales", "hydroGOF", "gtable", "ggpubr", "magrittr", "openxlsx", "ggmap", "sf", "ggspatial", "viridis", "RColorBrewer", "pastecs", "plotly", "rlang", "rmarkdown", "leaflet", "leafpop", "htmlwidgets", "withr"))

|  |  |
| --- | --- |
| **R Package** | **Reason** |
| install.packages(‘car’) | # for Levene Test |
| install.packages(‘gridExtra’) | # for plot layouts |
| install.packages(‘data.table’) | # for fread data import |
| install.packages(‘Rmisc’) | # for geomean confidence interval |
| install.packages(‘lubridate’) | # for working with dates |
| install.packages(‘tidyverse’) | # for data reformatting |
| install.packages(‘ggplot2’) | # for creating graphs |
| install.packages(‘ggforce’) | # for facet wrap paginate |
| install.packages(‘scales’) | # for access breaks/formatting |
| install.packages(‘hydroGOF’) | # for timeseries goodness of fit stats |
| install.packages(‘gtable’) | # for converting dataframe into figure object |
| install.packages(‘ggpubr’) | # for multipanel figures |
| install.packages(‘magrittr’) | # for piping functions |
| install.packages(‘openxlsx’) | # for reading/writing Excel files |
| install.packages(‘ggmap’) | # for mapping functions |
| install.packages(‘sf’) | # for importing shapefiles |
| install.packages(‘ggspatial’) | # for mapping functions |
| install.packages(‘pastecs’) | # for summary stats |
| install.packages(‘plotly’) | # for interactive plots |
| install.packages(‘rmarkdown’) | # for HTML conversion |
| install.packages(‘leaflet’) | # for interactive HTML plots |
| install.packages(‘leafpop’) | # for interactive HTML plots |
| install.packages(‘htmlwidgets’) | # for interactive HTML plots |
| install.packages(‘withr’) | # for interactive HTML plots |

## Basic tips regarding R input datafiles

* + Although the calibration project is designed to run in either interactive or \*.bat mode, one trade-off of this flexibility is that the scripts require a strict folder/subfolder hierarchy to accommodate the range of output generated. However, certain files can be renamed and relocated as long as those changes are indicated in the calibration \*.xlsx lookup table.
  + Unless custom filepaths are provided in the Excel calibration lookup, the R Project will default to a directory based on the location the \*.RProj file.
  + The script will define the modeling time period based on the full date range in the model output.
  + When running R-scripts, spelling and capitalization matter. An object named R\_calibration is different from R\_Calibration. This is a frequent source of error.
  + Although the scripts should replace backward slashes with forward slashes, in general, R does not like filepaths that contain backwards slashes (the default in Windows).
  + R does not like spaces in column names, file names, or variable names. If possible, default to using an underscore when naming files (e.g., snake case). Similarly, avoid periods in filenames. The scripts have been written to account for some of these issues, but it is probably best to avoid them when possible.
  + If possible, use hyphens instead of underscores when naming stream segments or calibration stations. The scripts should be able to handle names that include underscores, but this can be another common source of error.

## Creating folder hierarchy for R project

Due to the large number of files required and generated by the scripts, the scripts require a strict file tree to run successfully. When unzipped, the R project should already contain all of the necessary files and folders. Verify that the folders and subfolders have been created (note the additional ‘Year’ subfolders for the Plot\_Box and Plot\_Geom).

The R Project parent directory includes the following folders and files, which largely cannot be renamed:

* **Lookup:** This is where the Excel \*.xlsx lookup calibration file is located.
* **Plot\_Box**: Where the boxplot figures will be saved. Contains a ‘Year’ subfolder.
* **Plot\_Calib:** Where the multipanel calibration figures will be saved
* **Plot\_Dens**: Where the density figures will be saved
* **Plot\_Geom:** Where the geometric mean figures will be saved. Contains a ‘Year’ subfolder.
* **Plot\_HTML:** Where the interactive HTML plots will be saved
* **Proc\_Out:** Location where the preprocessed model and paired calibration-model data .txt files are saved.
* **Raw\_In:** Location of the raw calibration data \*.txt, if the user does not provide a custom filepath. If no ‘Filepath\_Model\_Data\_In’ is provided in the Excel lookup table, the model results \*.OUT, .BMD2 or \*.txt file needs to be saved here (the scripts will look here by default when no ‘Filepath\_Model\_Data\_In’ is provided in the Excel lookup table). The ‘Extract\_BMD2.exe’ will also save the extracted data to this folder, if no ‘Filepath\_Model\_Data\_In’ is provided.
* **Scripts:** Location of the calibration scripts that will be sourced by R. The ‘Extract\_BMD2.exe’ ’ and its associated \*.DLL files also need to be saved here. Based on the Excel lookup, the scripts will automatically generate and save a Control.dat file that will be saved here and read by the \*.exe.
* **Shapefile:** Location of the watershed and model network shapefiles (optional, but required for HTML plots)
* **Stats:** Where the goodness of fit statistics will be saved
* **.RProfile:** Text file that automatically loads the necessary packages when the R Project is opened
* **Single\_Calib.Rproj:** Rstudio project file that can be used to run the scripts interactively

## Set-up script options in Excel calibration lookup table:

The R scripts require a variety of settings that can be defined within the Excel calibration lookup table (i.e., Calib\_Lookup.xlsx). This allows the user to set the necessary arguments without having to directly edit the scripts. The arguments provided in this Excel file also will be used to construct the Control.dat file that will be called by the ‘Extract\_BMD2.exe’ to extract raw model output and generated the extracted model output \*.txt.

**NOTE**: When running the scripts interactively in a Rstudio project, the name and location of this file (and associated subfolder) \*\*cannot\*\* be changed from the location of the \*.RProj file. Within this Excel lookup table are five tabs (Controller, Station\_Lookup, PCode\_Lookup, LSPC\_Station, LSPC\_PCode) that also \*\*cannot\*\* be renamed. Conversely, when the scripts are run in \*.bat mode, the name of this Excel file and its location can be changed (see below). The ‘Station\_Lookup’ and ‘PCode\_Lookup’ tabs are used when calibrating WASP or preprocessed model data, while ‘LSPC\_Station’ and ‘LSPC\_PCode’ tabs are when calibrating LSPC model data. Depending on the model type being run (WASP, LSPC, or PREPROCESSED), arguments on these tabs can be left empty (e.g., LSPC\_Station and LSPC\_PCode can be left empty if the ‘Model\_Type’ is set to ‘WASP’ on the Controller tab).

## Controller tab:

The controller tab in the Excel lookup table is where the user has the ability to set a variety of options that will customize the calibration process. It also allows the user to point the R scripts to the appropriate filename and filepath, allowing the user to customize calibration scripts for individual modeling projects. This tab is broken into several sections that are discussed separately.

### Model type setting:

The calibration scripts can be run on output generated by LSPC, WASP, or a preprocessed datafile. Depending on this setting, the scripts will activate the appropriate Station and PCode lookup tabs. When calibrating LSPC output, the ‘LSPC\_Station’ and ‘LSPC\_PCode’ tabs must be used for mapping LSPC output to their respective ambient monitoring data. In addition, a filepath to the location of the LSPC \*.OUT files must be provided on the ‘Controller’ tab and all \*.OUT files must be saved within that location. The scripts assumed the \*.OUT files were exported in ‘Standard’ format from LSPC and are named based on the subbasin identifier (i.e., 102.out). Extracted data will then saved based on the ‘Model\_Extracted\_Filename’ argument.



When calibrating EFDC output, the ‘Station\_Lookup’ and ‘PCode\_Lookup’ tabs must be used for mapping model output to their respective ambient monitoring data. The scripts assume the model output is saved in a single \*.BMD2 file and named based on the arguments set on the ‘Controller’ tab (see below). Extracted data will then saved based on the ‘Model\_Extracted\_Filename’ argument.



When calibrating WASP output, the ‘Station\_Lookup’ and ‘PCode\_Lookup’ tabs must be used for mapping model output to their respective ambient monitoring data. The scripts assume the model output is saved in a single \*.BMD2 file and named based on the arguments set on the ‘Controller’ tab (see below). Extracted data will then saved based on the ‘Model\_Extracted\_Filename’ argument.



When calibrating Preprocessed model output, the ‘Station\_Lookup’ and ‘PCode\_Lookup’ tabs must be used for mapping LSPC output to their respective ambient monitoring data. However, the model data do not need to be extracted because the scripts assume the data have already been preprocessed. When calibrating ‘Preprocessed’ model output, the ‘Model\_BMD2\_Filename’ must be blank and a filename provided for the ‘Model\_Extracted\_Filename’ argument can be used to read in the preprocessed datafile (see below).



When blank, all other filepaths will default to their respective subfolder in RProject working directory.

The RProject will automatically run data extraction .exe when BMD2 filename provided and WASP model type selected. Not required when calibrating LSPC \*.OUT files or preprocessed data.

Model\_Extracted\_Filename can be used to read in preprocessed model data \*.txt

If ‘Filepath\_Model\_Data\_In’ is left blank, scripts will default to the ‘Raw\_In’ subfolder in the RProject working directory.

### Calibration settings:



* **Model\_Run\_Code:**  This is a character string that can be used to distinguish among different model runs, especially helpful if comparing calibration of different model runs. The length of this code is limited to ≤ 5 characters.
* **Filepath\_R\_Scripts:** Filepath where the Scripts are located. NOTE: If this cell is left blank, the scripts will default to the Scripts subfolder in the .RProject file.
  + NOTE: If the user moves the ‘Scripts’ subfolder from the default filepath, all scripts, Extract\_BMD2.exe’ , and associated \*.DLLs need to be saved in that new location.
* **Filepath\_Model\_Data\_In:** Filepath where the raw model \*.OUT, \*.BMD2 or \*.txt files are located. If this field is left blank, the scripts will default to the filepath of the ‘Raw\_In’ subfolder in the RProject working directory. When calibrating LSPC \*.OUT files, all \*.OUT files must be saved in this filepath and must be named based on subbasin number (e.g., 102.out).
  1. NOTE: This is also where the extracted model output is saved and named based on ‘Model\_Extracted\_Filename’
* **Filepath\_Calib\_Data\_In:** Filepath where the raw monitoring data are located. NOTE: If this cell is left blank, the scripts will default to the ‘Raw\_In’ subfolder in the .RProject file.
* **Filepath\_ProcData\_Out:** Filepath where the calibration output will be written. Where the various Plot folders and Stats folder are located. NOTE: If this cell is left blank, the scripts will default to the relative filepath of the .RProject file.
* **Calib\_Data\_In\_Filename:** Name of the \*.txt file that contains the monitoring data exported from WRDB. Requires the \*.txt extension and filename must match exactly what is used in this field (including capitalization).
* **Model\_BMD2\_Filename:** Name of the raw model output (\*.BMD2) that is generated by WASP and will be read by the ‘Extract\_BMD2.exe’ .’ If this argument is left blank and ‘Preprocessed’ data are selected as the model type, the scripts will bypass the ‘Extract\_BMD2.exe’ and will assume the user is using a preprocessed model \*.txt file rather than a \*.BMD2 file. No filename is required if calibrating LSPC \*.OUT files. The scripts will automatically read in all \*.OUT files based on the ‘Filepath\_Model\_Data\_In’ argument.
  1. NOTE**:** Any parameter or segment selected for calibration \*\*must\*\* be included in this \*.BMD2. Otherwise, errors in the data extraction may occur.
* **Model\_Extracted\_Filename:** Depending on the model data source, this argument is used two different ways. When calibrating WASP or LSPC model output, this ‘Model\_Extracted\_Filename’ is used to name the datafile that is extracted from the model input data. On the other hand, if ‘Preprocessed’ data are selected as the Model Type, the ‘Model\_Extracted\_Filename’ is the name of the preprocessed \*.txt file that contains the model data that will be used as the model input data. This file will be read by subsequent SOURCE scripts to further process, interpolate, and generate calibration output.
  1. NOTE:This argument must be provided regardless of whether the user is using a \*.BMD2 or \*.txt file as the model input data.

### Shapefile settings:



These arguments are optional and only required when a map is included on calibration figure

* The arguments in the shapefile settings matrix are optional. However, they are required if the user selects to add a watershed map to the multipanel calibration figures, or activates the ‘Export\_as\_HTML’ plotting option.
* **Filepath\_Shapefile:** Filepath where the shapefile subfolder is located. NOTE: If this cell is left blank the scripts will default to the ‘Shapefile’ subfolder in the .RProject file.
* **Watershed\_Shapefile\_Filename:** Name of the shapefile that designates the watershed boundary. Requires the .shp extension.
* **Stream\_Network\_Filename:** Name of the shapefile that designates the model stream network boundary. Requires the .shp extension.
* **Coord\_Ref\_Sys\_Code:** This is the EPSG (European Petroleum Survey Group) code that designates the coordinate system that your monitoring station lats/longs are in (e.g., EPSG 4326

= WGS84). Typically, lat/longs reported in .txt files are reported in a CRS of 4326.

### Filter settings:



* The arguments in the filter settings matrix are used to set various data clean-up and filter arguments.
* **Geomean\_First\_Month:** When plotting geomeans, the user has the option to set which months (in number format) are used in those calculations (e.g., 1 to 12; or only a growing season based on 3 to 10 [Mar to Oct]). This cell allows the user to set the first month used in the calculation (inclusive). NOTE: Regardless of the months used in the geomean calculation, all other plots and calculations will be based on the full date range.
* **Geomean\_Last\_Month:** This cell allows the user to set the last month (in number format) used in the geomean calculation (inclusive). NOTE: Regardless of the months used in the geomean calculation, all other plots and calculations will be based on the full date range.
* **Minimum\_Observations:** A monitoring station needs to have at least this many records (inclusive) to be included in the calculation of goodness of fit statistics or plotted. If not, this station will be skipped by the scripts. This threshold is assessed based on combined flagged and accepted records for a given station and PCode.
* **Rcode\_Flag\_Exclusion:** Any record in the monitoring data that has these characters (singly or in combination with another character) listed in the Remark Code (RCode) column will be excluded from goodness of fit statistics. The records will still be plotted, but they may be highlighted as ‘Flagged’ in the legend.However, when exclusion of flagged records results in < 2 accepted records or standard deviation equal to 0, only summary statistics will be calculated. If the exclusion of flagged records results in zero accepted records being retained, no statistics will be calculated.
  1. **NOTE:** When creating the vector, use commas to separate RCodes. If multicharacter RCodes have different definitions in combination than when reported separately, include them as separate entries. For example, if G6 is to be flagged when in combination, but G and 6 are not flagged when reported separately, include G6 as flagged Rcode (e.g., U,J,G6) and do not include G or 6 separately. Conversely, if 6 is to be flagged singly or in combination with any other RCode (e.g., A6, G6, etc), listing only 6 will flag any Rcode that includes it (e.g., U,J,6) and will not flag G when it is reported separately in the data.
* **Boxplot\_Density\_Min\_Obs:** Similar to the Minimum Observations threshold that is applied for goodness of fit calculations and the multipanel figure, this threshold is applied to determine which stations will be included in Geomean, Boxplot, and Density plots. When left blank, the scripts will plot all stations on the respective figure; otherwise, only those stations with at least this many records will be included. This can help reduce the number of stations include in the plots and is independent of the threshold used for GoF statistics and multipanel plotting.
* **Boxplot\_Statistic:** When generating boxplots, the user has the option to either plot the ‘Median’ or ‘Mean’ of the simulated data.

### Plot type settings:



* **Plot/Type:** Any cell with an ‘X’ will generate the plot type indicated in the ‘Type’ column, or activate the ‘Convert\_Flow\_to\_AvgDaily’ processing step.
  1. **Calibration / Boxplot / Geomean / Density:** Use these to activate the associated plot.
  2. **Map:** Will add a watershed map panel to the Calibration multipanel figure. If the user activates the Map plot, the Calibration plot must be active, and all shapefile information must also be entered (see above).
  3. **Annual:** Allows the user to activate annual comparisons of geometric means and boxplots. Deactivating annual comparisons can accelerate plotting.
  4. **Convert\_Flow\_to\_AvgDaily**: Allows the user to first aggregate simulated flow values by date and segment, calculate an average daily flow, and that average daily flow is subsequently used for calibration (e.g., if flow is reported out 4 times a day in the model, those 4 records will be averaged into a single daily average). If this option is not activated, the simulated flow values will be used exactly how they are reported in the model .BMD2 file (e.g., if flow is reported out 4 times a day, each of those four records will be used separately for calibration. This option can help accommodate calibration/monitoring flow data that is either reported as instantaneous data or as average daily flow. This option is only applied to model flow and will not be applied to any other model parameters.
     + **NOTE:** If flow is aggregated into daily average flow, the time stamp for the model data is changed to noon (12:00) for each daily record. **When calibrating LSPC output,** **verify that this time stamp is not causing a 12-hr phase shift when pairing the LSPC and calibration data.**
  5. **Export\_as\_HTML:** When activated, select plots will be converted into interactive HTML plots and saved in ‘Plot\_HTML’ subfolder. Otherwise, static plots are created and saved in their respective subfolders (e.g., Plot\_Calib). Option is only available for multipanel calibration figure, density, and boxplots. If activated, user must provide a shapefile for the stream network and watershed boundary. This option requires at least one of the following plotting options to also be activated: Calibration, Boxplot, or Density. Interactive multipanel plots are generated separately from the distribution plots (density and boxplot).
     + **NOTE:** Static and HTML plots cannot be generated and saved simultaneously; therefore, scripts must be run twice if the user wants to generate both types of plots.

### Time period settings:



* **Selected\_TZ/Model\_TimeZone:** The cell with an ‘X’ will use the corresponding time zone listed in the Model\_TimeZone. This allows the user to adjust, pair, and report data in the correct time. NOTE: This assumes the measured data are reported out in the selected time zone.

## WASP:

### WASP Station\_Lookup tab

When applying the scripts to either WASP \*.BMD2 data or preprocessed model output, use the Station\_Lookup table to map the WASP segments to the appropriate calibration stations.

Plot\_Title and Plot\_Order can be customized. Used to map multiple stations to same model segment

Calib\_Station\_ID must \*\*exactly\*\* match the Station\_IDs used in WRDB/monitoring data \*.txt. Cannot start with leading zero.

WASP\_Name must \*\*exactly\*\* match the model segment name used in the model output \*.BMD2



### EFDC Station\_Lookup tab



* The calibration ‘Station\_Lookup’ tab is where the user will provide the WASP Segment to Calibration Station crosswalk that needs to be known beforehand. This is necessary because the calibration data is based on the monitoring station name, while the model output reports only a model segment name. Therefore, the user must provide a lookup table that not only tells the scripts which model segment number corresponds to which model segment name, but also which calibration station name is associated with that model segment.
  + Names provided in this lookup table must match \*\*exactly\*\* names reported in model and observed data. Spelling/capitalization errors can be a common reason for scripts to crash.
  + Any segment selected for calibration (i.e., an ‘X’ is reported in the ‘Calibration’ field) \*\*must\*\* be in the WASP \*BMD2 file. Otherwise, errors during the extract of model results may occur.
  + The tab is designed in such a way that the crosswalk of all WASP segments and calibration stations can be set-up beforehand and only those segments being actively calibrated need to be activated. This allows the user to create a master list of WASP segments that can be used with minimal modification among runs.
  + When multiple monitoring/calibration stations are located within the same WASP segment, the Plot\_Title and Plot\_Order cells provide a means for merging the monitoring data (even if the monitoring station names are different). This can account for co-located monitoring stations.
  + The fields within the spreadsheet have conditional formatting to highlight cells red when values are duplicated. This can provide a quick QA when mapping multiple monitoring stations to the same WASP segment.
  + Although the scripts can account for spaces and underscores in the WASP\_Name and Calib\_Station\_ID, it is recommended to use hyphens. This can be common source of error.
* **Calibrate:** Any cell with an ‘X’ will include the model segment listed in WASP\_Segment in the calibration. When multiple monitoring stations are mapped to the same model segment, the user also has the ability to deactivate/activate individual stations.
* **WASP\_Segment:** This is the WASP stream segment number that can be found on the ‘Segments’ screen of WASP 8.
* **WASP\_Name:** This is the WASP Segment Name that can be found on the ‘Segments’ screen of WASP 8 (or by opening the \*BMD2 with BMDUtil). This can be populated by copy-and-paste directly from the WASP 8 \*.WIF. Segment names must **exactly** match what is reported in the model input data.
* **Calib\_Station\_ID:** This is the Station\_ID of calibration data reported in the WRDB Support Table.
  + **NOTE**: Station\_IDs reported in this field have to **exactly** match those used in the WRDB Support Table. It also **cannot** start with a leading zero.
* **Plot\_Title:** This field is used to map multiple monitoring stations to the same model segment. The value reported in this field can be customized—it will be used to name any calibration output and allows data from multiple monitoring stations to be merged. Accordingly, multiple Calib\_Station\_IDs can have the same Plot\_Title reported in this field.
  + **NOTE**: This value is also what allows the user to merge monitoring data from multiple calibration stations into a single WASP segment. When co-located monitoring stations are to be merged, the same \*\***exact**\*\* name must be used in their respective cells. The R scripts will then merge their data and compare them to the appropriate WASP Segment.
* **Plot\_Order:** This field allows the user to define a custom order that monitoring stations are included in the boxplot, geometric mean, and density plots. For instance, the user can set the most downstream segment to Plot\_Order = 1 and then number them into the headwaters.
  + **NOTE**: When data from multiple calibration stations are mapped to the same WASP segment, the number reported in their respective cell has to be identical.
  + The sequence can skip numbers and does not have to be in numeric order, avoiding the need to reorder the sequence when different stations are selected.
* **Lat:** Latitude of the monitoring station in decimal degrees. This cell is optional and only required if the user includes the watershed map on the multipanel calibration figure.
* **Long:** Longitude of the monitoring station in decimal degrees. This cell is optional and only required if the user includes the watershed map on the multipanel calibration figure.

### WASP PCode\_Lookup tab

When applying the scripts to either WASP \*.BMD2 data or preprocessed model output, use the PCode\_Lookup table to map the WASP segments to the appropriate calibration stations.

Anal\_Name must \*\*exactly\*\* match analyte names listed in model output \*.BMD2 or \*.txt file

PCode must \*\*exactly\*\* match parameter codes used in WRDB/monitoring data \*.txt



All PCodes selected for calibration (i.e., Calibrate = X) \*\*must\*\* be in the raw model \*.BMD2 output file

### EFDC PCode Lookup Tab



* This tab crosswalks WASP parameter names to PCodes and units reported in your monitoring data. Prior to running the scripts, the user will need to manually identify how WASP parameters and monitoring data pair. The RProject will not apply unit conversions; therefore, all unit conversions need to be applied prior to importing the data into R.
  + Any PCode/analyte selected for calibration \*\*must\*\* be in the model \*.BMD2 file. Otherwise, errors during the extraction of model results may occur.
  + Because the Anal\_Name and PCode provided in this lookup table must match \*\*exactly\*\* the names reported in the WASP output and monitoring data, errors on this tab can be a common reason for the scripts to crash. Analyte names in the \*.BMD2 can be determined by opening the file with BMD\_Util.exe.
  + Similar to the Station\_Lookup tab, a master list of analytes and PCodes can be included in this table and the user can activate only those PCodes that are currently being calibrated.
* **Calibrate:** Any field with an ‘X’ will select the corresponding parameter listed in the PCode column for calibration. NOTE: Not all parameters have to be measured at each station. The scripts will only generate output for a station when it meets the minimum number of records threshold in the \*.xlsx lookup.
* **PCode:** This is the PCode that is used in your monitoring data to designate the parameter (e.g., the PCode listed in the WRDB support table). This has to \*\*exactly\*\* match the WRDB PCode.
* **Units:** These are the units of the PCode in WRDB and in the WASP output and will be used to label calibration figures. Note: Model and monitoring data need to be in the same units prior to importing into R, as no unit conversions will be applied during calibration.
* **Anal\_Name:** This is the analyte/parameter name that is used in the WASP model output. This has to \*\*exactly\*\* match the name used in the model input data that is read into R. NOTE: You can open the WASP \*.BMD2 file with BMDUtil to verify the parameter names used in the WASP output.

## LSPC

### LSPC\_station lookup tab

When using the calibration scripts for LSPC \*.OUT files, use the ‘LSPC\_Station’ tab to map the LSPC reaches to their respective calibration stations. Many of these columns are similar to those used in the WASP ‘Station\_Lookup’ table. This assumes the LSPC \*.OUT files are saved in ‘Standard’ format, are named based on their subbasin number (i.e., 102.out), are all located in the location indicated in the ‘Filepath\_Model\_Data\_In’ argument on the ‘Controller’ tab.



* **Calibrate:** Any cell with an ‘X’ will include the model segment listed in LSPC\_Subbasin in the calibration. When multiple monitoring stations are mapped to the same model segment, the user also has the ability to deactivate/activate individual stations.
* **LSPC\_Subbasin:** This is the LSPC subbasin number that is used to name the \*.OUT files.
* **LSPC\_Name:** This is the LSPC subbasin name, including the runoff type (i.e., ‘RO’).
* **Calib\_Station\_ID:** This is the Station\_ID of calibration data reported in the WRDB Support Table.
  + **NOTE**: Station\_IDs reported in this field have to **exactly** match those used in the WRDB Support Table. It also **cannot** start with a leading zero.
* **Plot\_Title:** This field is used to map multiple monitoring stations to the same model segment. The value reported in this field can be customized—it will be used to name any calibration output and allows data from multiple monitoring stations to be merged. Accordingly, multiple Calib\_Station\_IDs can have the same Plot\_Title reported in this field.
  + **NOTE**: This value is also what allows the user to merge monitoring data from multiple calibration stations into a single LSPC subbasin. When co-located monitoring stations are to be merged, the same \*\***exact**\*\* name must be used in their respective cells. The R scripts will then merge their data and compare them to the appropriate LSPC subbasin.
* **Plot\_Order:** This field allows the user to define a custom order that monitoring stations are included in the boxplot, geometric mean, and density plots. For instance, the user can set the most downstream segment to Plot\_Order = 1 and then number them into the headwaters.
  + **NOTE**: When data from multiple calibration stations are mapped to the same LSPC subbasin, the number reported in their respective cell has to be identical.
  + The sequence can skip numbers and does not have to be in numeric order, avoiding the need to reorder the sequence when different stations are selected.
* **Lat:** Latitude of the monitoring station in decimal degrees. This cell is optional and only required if the user includes the watershed map on the multipanel calibration figure.
* **Long:** Longitude of the monitoring station in decimal degrees. This cell is optional and only required if the user includes the watershed map on the multipanel calibration figure.

### LSPC\_PCode lookup tab

When calibrating LSPC \*.OUT files, use this table to select the LSPC analytes to calibrate and to map them to their appropriate calibration data PCode. This assumes the LSPC \*.OUT files are exported from LSPC in standard format using the names listed below.



* This tab crosswalks LSPC parameter names to PCodes and units reported in your monitoring data. Prior to running the scripts, the user will need to manually identify how LSPC parameters and monitoring data pair. The RProject will not apply unit conversions; therefore, all unit conversions need to be applied prior to importing the data into R.
  + Any PCode/analyte selected for calibration \*\*must\*\* be in the model \*.OUT file. Otherwise, errors during the extraction of model results may occur.
  + Because the Anal\_Name and PCode provided in this lookup table must match \*\*exactly\*\* the names reported in the LSPC output and monitoring data, errors on this tab can be a common reason for the scripts to crash. Analyte names in the \*.OUT can be determined by opening the \*.OUT files.
  + Similar to the Station\_Lookup tab, a master list of analytes and PCodes can be included in this table and the user can activate only those PCodes that are currently being calibrated.
* **Calibrate:** Any field with an ‘X’ will select the corresponding parameter listed in the PCode column for calibration. NOTE: Not all parameters have to be measured at each station. The scripts will only generate output for a station when it meets the minimum number of records threshold in the \*.xlsx lookup.
* **PCode:** This is the PCode that is used in your monitoring data to designate the parameter (e.g., the PCode listed in the WRDB support table). This has to \*\*exactly\*\* match the WRDB PCode.
* **Units:** These are the units of the PCode in WRDB and in the LSPC output and will be used to label calibration figures. Note: Model and monitoring data need to be in the same units prior to importing into R, as no unit conversions will be applied during calibration.
* **Anal\_Name:** This is the analyte/parameter name that is used in the LSPC model output. This has to \*\*exactly\*\* match the name used in the model input data that is read into R.

# Interactively running calibration scripts as RProject:

When first setting-up and running the calibration scripts, it is recommended that the user initially interactively runs the scripts as an R Project. This will provide greater details on errors/warnings that may be inhibiting proper functioning of the scripts, and help troubleshoot possible crosswalk errors.

When running the scripts in the R Project, the only R script that needs to be run is the Master R Script, as this script will SOURCE the other subscripts based on the user defined options in the Excel Spreadsheet.

## Run interactively with default filepaths

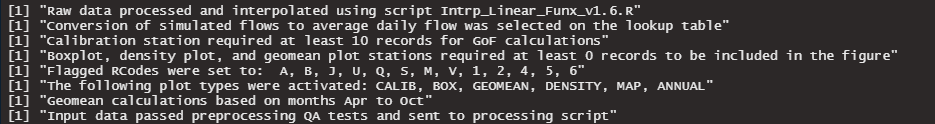
1. Assuming that the necessary R packages have been downloaded and installed (see above) and the Excel lookup table has been properly updated to reflect the locations/filenames of the necessary datafiles, double-click the R Project file (Single\_Calib.Rproj). This will open R Studio.
2. Although the R Project is open, no scripts are being sent/run in the R Console. Therefore, the user will need to open the Master script that will SOURCE the various sub-scripts as necessary.
3. To open the Master script, either navigate to the ‘Scripts’ subfolder via the ‘Files’ pane (bottom right in the example above) or File 🡪 File Open. Select the Master script (Master\_Mrgd\_v4.2.R or some variation on that name), which will open the script in the Source panel (upper left panel in the example above).
   1. The Master script is the only script that needs to be actively run by the user, as it will SOURCE other sub-scripts as needed based on user defined options in the Excel lookup.
   2. Additional scripts are located in this folder and will be called as needed

|  |  |
| --- | --- |
| **Script name** | **Script function** |
| Master\_Mrgd | Master script that QA's data and calls sub-scripts |
| Extract\_BMD2\_EXE | Calls the ‘BMD2\_Extract.exe’ and \*.DLLs to extract model output |
| Extract\_LSPC\_OUT | Reads in multiple LSPC \*.OUT files and extracts model output |
| Intrp\_Linear\_Funx | Pairs the model and measured data and interpolates |
| Proc\_Funx | Preprocesses/reformats the input data |
| GOF\_Stats | Calculates goodness of fit and summary statistics |
| Plot\_Funx\_Density | Creates data density distribution plots |
| Plot\_Funx\_Geomean | Calculates and plots geometric means |
| Plot\_Funx\_Calib | Creates multipanel calibration plot |
| Plot\_Funx\_Box | Creates boxplot |
| Cleanup\_Environ | Periodically deletes objects from R environment |

1. To execute the script and send it to the R Console, the user can highlight all text in the SOURCE panel and either press ‘Ctrl + Enter’ or by click the Run icon. A third option is to simply click the ‘Source’ icon.



1. If the scripts are running correctly, the R Console will display various QA messages to indicate which options were selected and which version of the scripts were used. Moreover, when a sub-script completes successfully, the R Console will indicate this success and whether the results were sent to the next sub-script. When an error is encountered, this can help isolate which script is having a problem.



1. As the scripts will implement a variety of QA/QC during the preprocessing and analyses, any encountered errors will also be displayed in the R Console window. This can help troubleshoot potential errors.
2. When the R Console is actively running a script, the R Console will have a stop sign displayed in the upper right corner (which can be clicked to terminate the script, if necessary).
3. Once completed, the plots and statistical output will be written to the default filepath (i.e., the location of the R Project) or the user defined filepath in the Excel lookup (i.e., Filepath\_ProcData\_Out). See Appendix A for an overview of the output.

# Running scripts in .BAT mode via command line:

In addition to running the scripts interactively as a \*.RProj, the user also has the option of running the scripts in \*.bat mode. This further automates the calibration process and can accommodate the calibration of multiple models with a single \*.bat file. However, running scripts in .bat mode requires additional knowledge of creating .bat files. Moreover, it is important to review the breadth of folders that are associated with the customizable filepaths and filenames in the Excel calibration lookup (see the ‘Controller tab’ section above). Incorrectly placed or named files are a common source of errors.

When applying the scripts to a new modeling project, it is highly recommended the user format and initially run the scripts interactively before they attempt to automate the process in \*.bat mode. Interactive mode allows for more detailed error messages that can help troubleshoot any error in the set-up of the calibration lookup. Errors are harder to troubleshoot with .bat mode.

1. When run in \*.bat mode, the lookup table should be populated as above. However, the user has additional flexibility that is not available in interactive mode. For instance, the user has the ability to rename the ‘Calib\_Lookup.xlsx’ file in \*.bat mode. The filepath and filename of the calibration lookup will be fed to the Master R script, which will open the lookup table and extract the necessary user-defined options. **NOTE:** This ability to rename the Calib\_Lookup.xlsx provides the user the option to create/save multiple copies of the lookup table that can be independently called by the \*.bat file. When this ability is paired with the ability to move various subfolders, it provides the user the added option to maintain multiple lookup tables that can be linked to different model runs. This can help compare calibration of multiple model runs.
2. Populate the calibration lookup table as before and rename the file (if desired).
3. On a single line, the following text can be written to a text editor and saved as a \*.bat file. NOTE: Only a single space is allowed between \*.bat file elements.

"C:\Program Files\R\R-4.1.2\bin\x64\Rscript.exe" --verbose ".\Scripts\Master\_Funx\_v1.8.R"

2> errorfile.txt 1> messagefile.txt ".\Lookup\Calib\_Lookup.xlsx" "C:\Program Files\RStudio\bin\pandoc" "BASE" "Y"

1. The batch file has several components that must be included in a specific order.

|  |  |
| --- | --- |
| **\*.bat file argument** | **Reason** |
| Filepath to the Rscript.exe | Executes the R terminal program |
| --verbose | Tells R terminal to output error/messages |
| Filepath and filename to the Master\_Funx R script | Tells R terminal where to find the ‘Master’ calibration script.  This can be a relative or absolute filepath, depending on the location of the \*.bat file relative to the Master R script. |
| 2> piping to create error file | Tells R terminal to create and output an error message file that can be used for troubleshooting. |
| 1. piping to create message file | Tells R terminal to create and output a message file that can be used for troubleshooting. |
| Filepath and filename to the Excel calibration lookup \*.xlsx | Tells R terminal where to find the Excel lookup. Allows output from multiple models to be created in sequence |
| Filepath to Pandoc.exe library that is installed with RStudio | When run in .BAT mode, this argument tells R where to find the Pandoc library that is required to generate interactive HTML plots. Library is should be installed within ‘bin’ subfolder of Rstudio install. |
| Optional 4-character model run code | Appends model output w/ this code.  If a model run code is provided in the \*.bat file, it will overrule the code used in the Excel lookup table. This allows multiple output files to be created in sequence. |
| Save.image() argument | Can be set to “N” or “Y”, or not included. It will tell the R scripts to export and save the data image during processing. This allows the R scripts to export the .RData file during .bat mode and can allow the file to be reopened in RStudio when the .bat file completes. Allows review of .RData in RStudio. |

1. Once the \*.bat file is saved, it can be run either by double-clicking it or via command line.
2. If no output is generated or the command prompt window immediately closes, examine the “error.txt” and “message.txt” files to determine the source of the error.
3. If the source of the error cannot be determined when running the scripts in .bat mode, update the scripts and run interactively in R Studio.
4. If the Save.image() argument is set to ‘Y,’ the \*.RData file can be opened in RStudio for review.

# Re-running scripts based on new model output

Once the lookup table and crosswalks have been set-up, the scripts can be run iteratively when new model output is generated. **NOTE:** The scripts will overwrite the existing figures and stats, unless output is routed to a different location.

1. Copy/paste the new model output to the location indicated by the ‘Filepath\_Model\_Data\_In’ in the Excel lookup table, or update the filepath so it points to the new model output.
2. Update the ‘Model\_BMD2\_Filename’ argument in the Excel lookup (optional)
3. Update the ‘Model\_Extracted\_Filename’ argument in the Excel lookup (optional)
4. Update ‘Model\_Run\_Code’ argument (optional)
5. Move any figures/stats files that you do not want overwritten (optional)
6. Re-run the Master script to regenerate figures/stats based on new model run

# Common sources of errors:

1. The calibration data cannot contain duplicate records, which are assessed based on Station\_ID and Date\_Time. When multiple samples are collected at the same location and time, this can lead to issues when pairing and interpolating the model data. If multiple samples were collected at the same time and location, these may need to be averaged prior to use in the calibration scripts.
2. Using a lookup table that doesn’t have the correct format or argument names, which can lead to missing script arguments even when they are included in the Excel lookup table. Verify that the Lookup table version number (found in the comment box on the ‘Controller’ tab) is version 5.
3. Parameters or segments selected for calibration, but are not included in the model \*.BMD2. These types of crashes will be almost immediate and will have limited error output. Determine whether the Control.dat file was created and verify that selected parameters/segments are in the raw model \*.BMD2. If no Control.dat or extracted model \*.txt were created, it suggests an mismatch in the Excel crosswalk and the data included in the model \*BMD2 file.
4. If the user is not using a raw model \*.BMD2 file and instead using a preprocessed \*.txt file, errors can occur when the model input data \*.txt has the wrong format or analyte names that do not match Anal\_Names provided in the Excel lookup table.
5. Misspelled filenames or filepaths that don’t match those reported in the Excel \*.xlsx file.
6. In the Excel \*.xlsx, including the trailing forward slash in the filepath field. Remove the trailing forward slash and/or subfolder name.
7. Not including the file extension (.BMD2, .out, .txt or .shp) in the filename field. Capitalization matters.
8. When using custom filepaths, all required input data were not moved to the custom filepath.
9. Misspelled stream segments or monitoring stations that don’t \*\***exactly**\*\* match those reported in the Excel lookup \*.xlsx file. Names provided in the station crosswalk must match what is used in the model and measured input files. Calibration Station IDs that start with leading zeros can cause errors and should be avoided.
10. Misspelled Anal\_Name or Pcodes that don’t \*\***exactly**\*\* match those reported in the Excel lookup \*.xlsx file. Anal\_Names provided in the station crosswalk must exactly match what is reported in the model \*.BMD2 and extracted data \*.txt. Pcodes must exactly match what is reported in the monitoring data \*.txt.
11. Spaces or capitalization errors in filenames, filepaths, stream segment names, or monitoring station identifiers.
12. Using a model segment name or Anal\_Name or WASP\_Name in the Excel station lookup that doesn’t \*\*exactly\*\* match what is used in the raw model \*.BMD2 file. Use BMDUtil to examine raw \*BMD2 to verify the spelling of model segment and parameter names.
13. Unit conversions not applied prior to importing data into R.
14. If the data interpolation scripts are not completing, it is possible that the filenames and filepaths are not correct in the Excel \*.xlsx file. In addition, any required subfolders may not have been properly moved when custom filepaths are used in the \*.xlsx file.
15. If the interpolation and processing scripts are completing, but no data are being plotted or calibrated. This suggests an error in the stream segment and monitoring station crosswalk, or in the analyte name to PCode crosswalk.
16. When attempting to run in .bat mode, including more than one space in between arguments or putting arguments in the wrong order. Also, verify that arguments are set-off with straight double-quotes.
17. When run in .bat mode, not providing all required arguments in their specific order.
18. When run in .bat mode, providing an incorrect filepath to the Pandoc library, which is required even when HTML plots are not activated.

# References

Moriasi, D. N., Arnold, J. G., Van Liew, M. W., Bingner, R. L., Harmel, R. D., and T. L. Veith. 2007. Model evaluation guidelines for systematic quantification of accuracy in watershed simulations. Transactions of ASABE. 50: 885−900.

R Core Team 2018. R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. https://www.R-project.org/

Wickham, H. 2017. tidyverse: Easily install and load the 'Tidyverse'. https://CRAN.R-project.org/package=tidyverse

Zambrano-Bigiarini, M. 2017. hydroGOF: Goodness-of-fit functions for comparison of simulated and observed hydrological time series. R package version 0.3-10. URL http://hzambran.github.io/hydroGOF/

Appendix A:

Model Output

# Script output: Datafiles

Once completed, the calibration scripts will output several tab delimited \*.txt files that contain the original model data, paired measured and interpolated simulated data, and a QA/QC file. Each file will be appended with the model run code. These files can be archived or used for comparing output from multiple model runs.

|  |  |
| --- | --- |
| **Filename** | **Description** |
| Extracted Model Output.txt | When calibrating LSPC or WASP model data, this file contains the model results that were extracted by the ‘Extract\_BMD2.exe’ or ‘Extract\_LSPC\_Out’ and saved in the filepath indicated by ‘Filepath\_Model\_Data\_In.’ The name is based on the name provided in the ‘Extract\_Model\_Filename’ field in the Excel lookup table.  When run on Preprocessed model data, this is the user-defined preprocessed model input datafile \*.txt that can be read into the scripts, bypassing the model extraction step. |
| Proc\_Model\_XXXX.txt | Contains the full extracted dataset for the simulated model data. It has been reformatted to include additional crosswalk columns  (e.g., Monitoring Station\_ID). Saved in the ‘Proc\_Out’ subfolder. These are the model data that were used for data interpolation and can be read into other software packages for additional analyses. |
| Proc\_CalibModel \_XXXX.txt | Contains the observed data and its paired simulated data that were linearly interpolated to the exact time of the measured data record.  Additional crosswalk columns were added. Saved in the ‘Proc\_Out’ subfolder. These data only contain model data that have a paired calibration data record. |
| Proc\_ QACheck\_XXXX.txt | This is a QA/QC file that reports out observed and simulated values (pre and post the measured data record) that were used for interpolation at the exact time the measured data were collected. This allows the user to verify that the simulated data were correctly paired and interpolated. Saved in the ‘Proc\_Out’ subfolder. |
| Proc\_Rcode\_Crosswalk\_XXXX.txt | This is a QA/QC file that can be reviewed to verify which Rcodes were flagged and which were accepted when processing the data. |

# Script output: Goodness of fit statistics

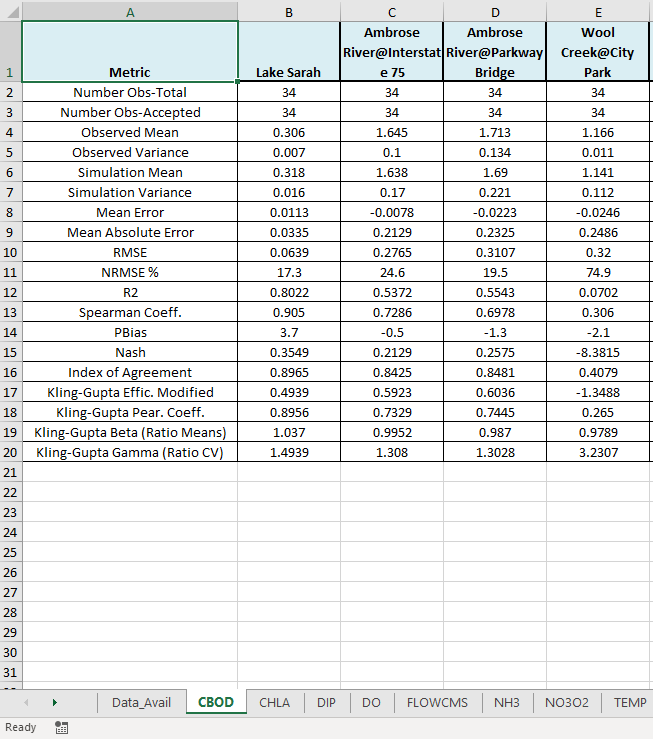
The R Project will automatically generate a data availability table that summarizes the total number of observed data available. In addition, it will generate several versions of the goodness of fit statistics that are calculated for each monitoring station and parameter selected on the Excel calibration lookup (flagged records are excluded based on the RCode\_Flag\_Exclusion field). The ‘Number Obs-Accepted’ indicates how many records were used for goodness of fit calculations. Monitoring stations that did not meet the minimum number of records threshold will be left blank. Stations are named based on the names provided in the ‘Plot\_Title’ field. See Appendix B for additional details on the interpretation of the statistical metrics.

NOTE: The initial minimum number of records threshold is applied before flagged records are excluded. However, when the exclusion of flagged records results in < 2 accepted records or a standard deviation equal to 0, this situation can violate assumptions of several goodness of fit statistics. In this case, the goodness of fit statistics will not be calculated and only summary statistics will be calculated. When the exclusion of flagged records results in zero accepted records, neither the goodness of fit statistics nor summary statistics will be calculated. When a metric is not calculated, it is represented by a blank cell or NA.

## Data availability table



## Goodness of fit statistics v1 (Multitab): Metric (rows), Station (columns) & PCode (tabs)



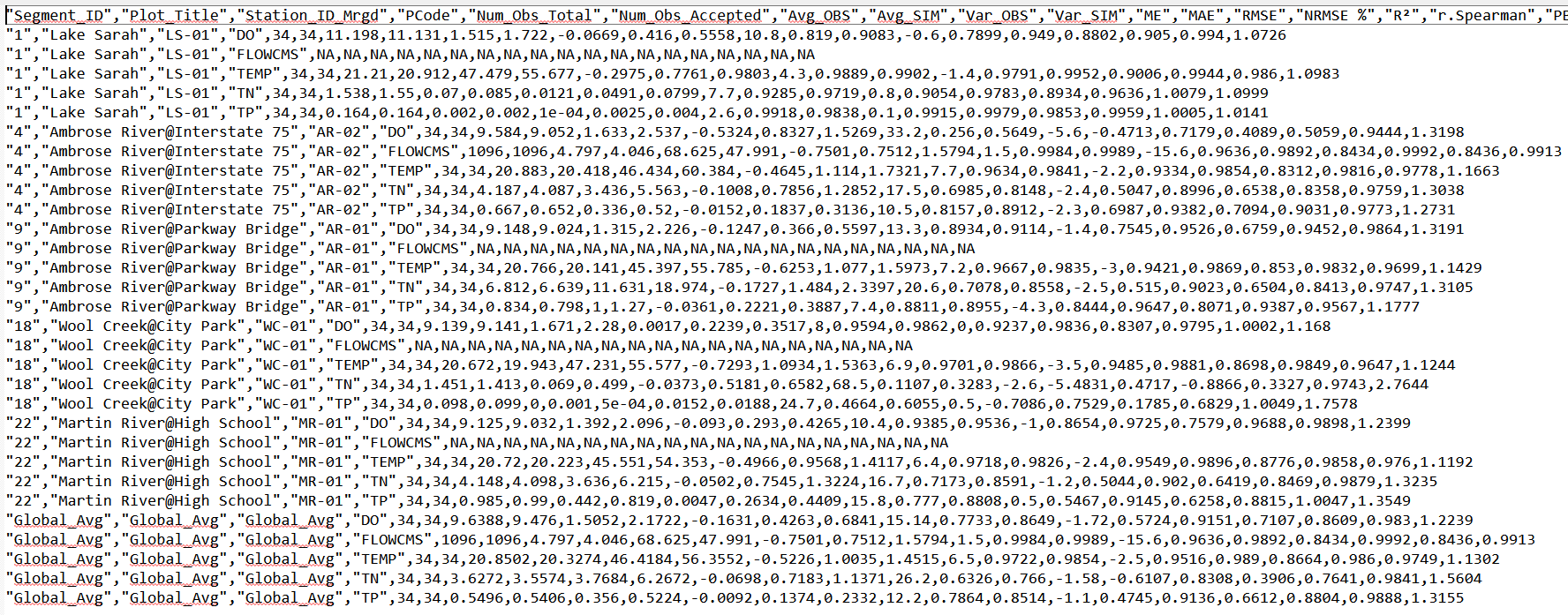
## Goodness of fit statistics v2 (Wide): Station/PCode (rows) and Metric (columns)



## Goodness of fit statistics v3 (Long): Station/Metric (rows) and PCode (columns)



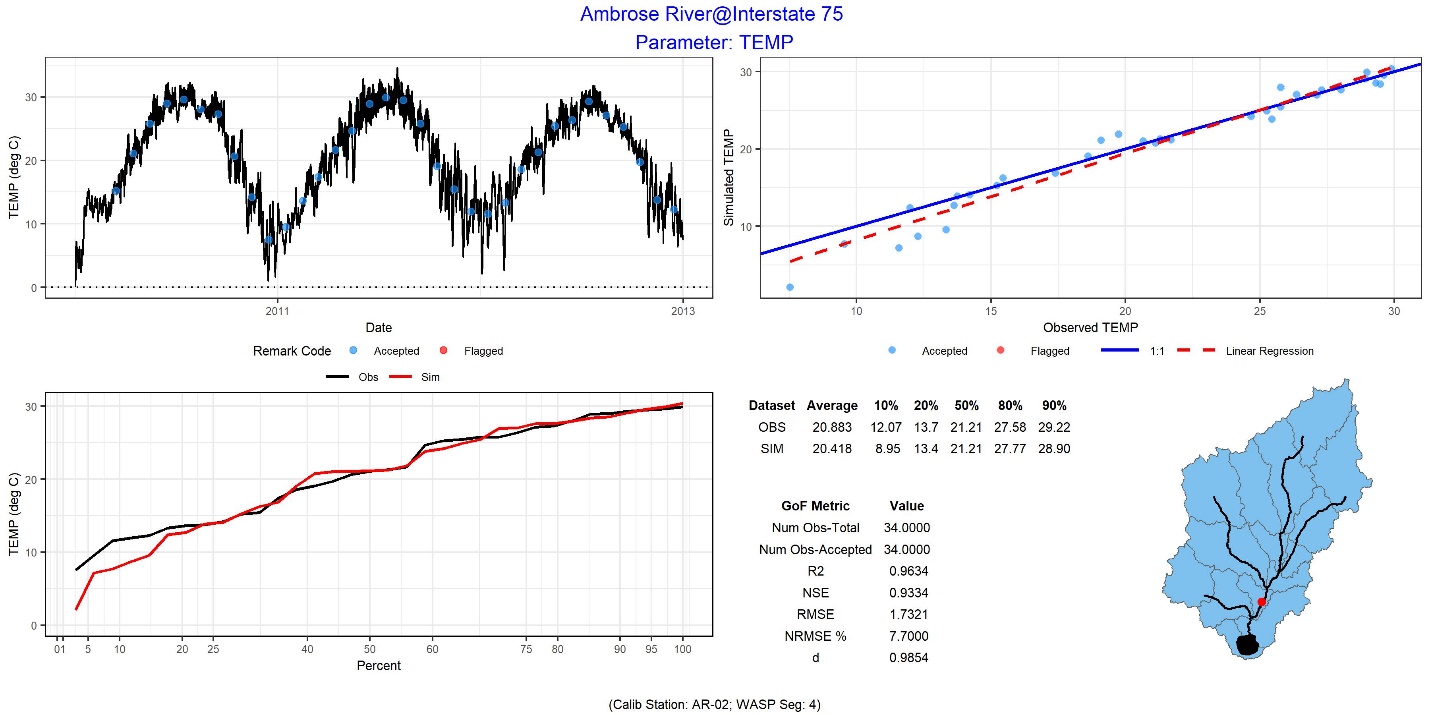
## Goodness of fit statistics v4 (.csv file in Wide format)



# Script output: Figures

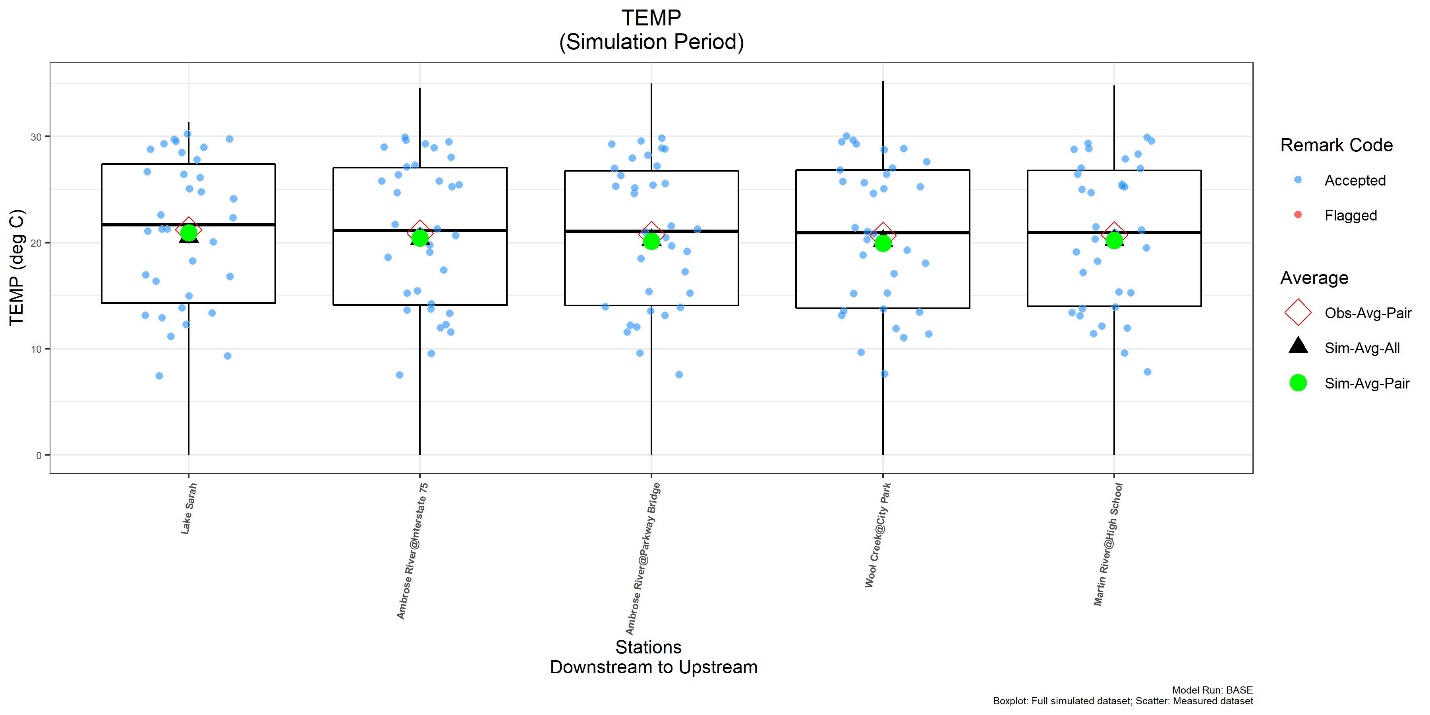
Depending on the plots selected in the Excel calibration lookup table, the scripts will generate a variety of figures that are commonly used for model calibration. These are summarized below.

## Multipanel calibration figure



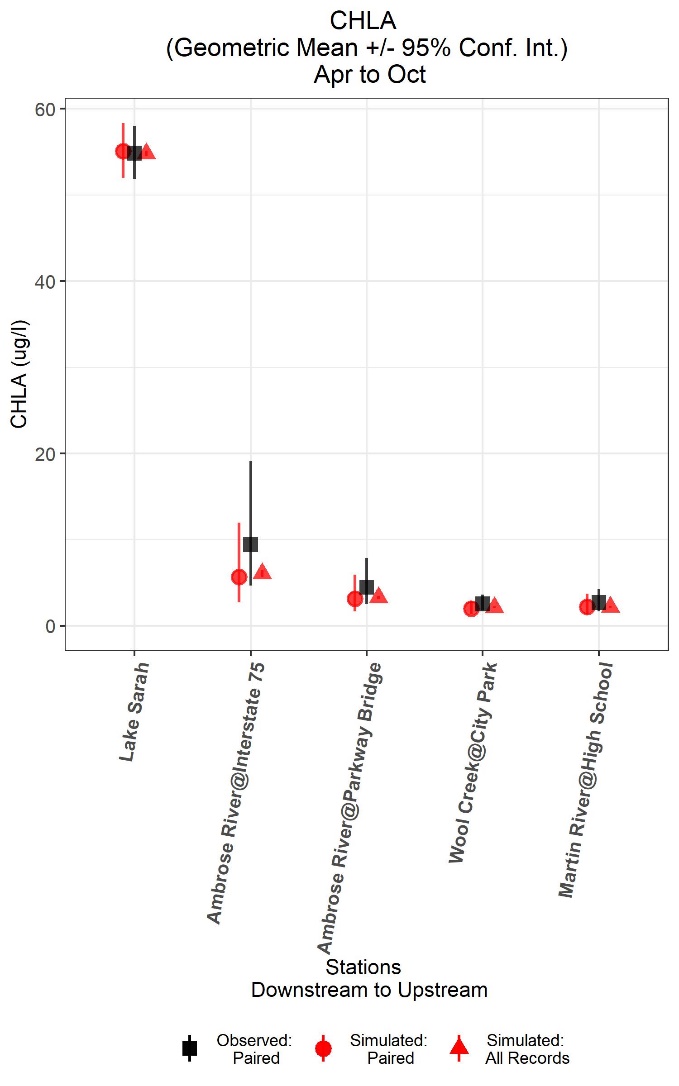
Several commonly accepted calibration plots comprise the multipanel calibration figure, with a separate figure generated for each monitoring station and model parameter. The plot title is derived from the Plot\_Title option in the Excel calibration lookup .xlsx. The top left panel displays a time series of all simulated data, with the observed data overlaid as individual data points. Approved and flagged observed data points are distinguished as blue or red circles, respectively. The top right panel displays a scatter plot comparing the paired simulated and measured data points, with a best-fitting linear regression line (dashed red line) and 1:1 line (blue line). The bottom left panel displays the empirical cumulative distribution curve for the paired simulated (red line) and observed (black line) data (when plotting flow, this plot uses a log10 y-scale). The final panel reports several common summary and goodness of fit statistics, including the total number of observations available, the number of accepted observations (used for GoF calculations), coefficient of determination (R2), Nash-Sutcliffe efficiency coefficient (NSE), root mean square error (RMSE), normalized root mean square error (NRMSE%), and the index of agreement (d). Flagged observed records are excluded from the goodness of fit calculations. When a particular statistics was not calculated, the value is reported as missing (i.e., NA). The optional map can be included by selecting the ‘Map’ option in the Excel lookup table. Lastly, a footer reports the model segment number and monitoring station identifiers that are associated with the displayed data.

## Boxplot



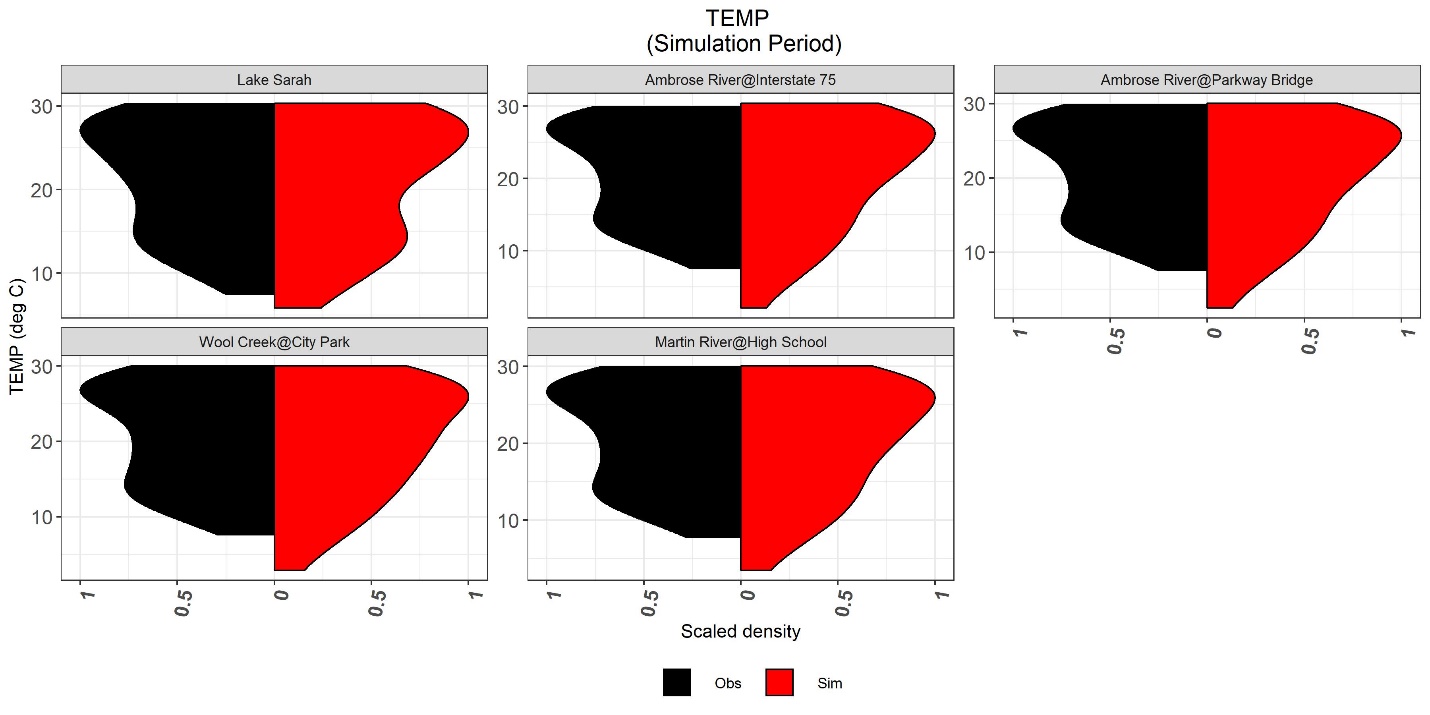
For each model parameter, a grouped boxplot is generated based on the full simulated dataset and the observed data. Observed data are plotted as individual data points (blue circles are ‘accepted’ values, while red circles are ‘flagged’ values), with their respective average represented by the open red diamond. The blue box and whisker plots indicate the data distribution of the full simulated dataset (i.e., median, min, max, and 25th and 75th percentiles), with the mean of all simulated data indicated by the black triangle. The green circle represents the mean based only on those simulated data records that had a paired calibration data point. Based on the setting on the ‘Controller’ tab, the median of the simulated data can be plotted instead of the model mean. A separate box-and-whisker is created for each monitoring station that is based on the Plot\_Title field from the Excel calibration lookup, with the order based on the Plot\_Order. When ‘Annual’ plots are activated on the Excel lookup table, additional boxplots are generated on an annual basis and saved within the ‘Year’ subfolder.

## Geometric means



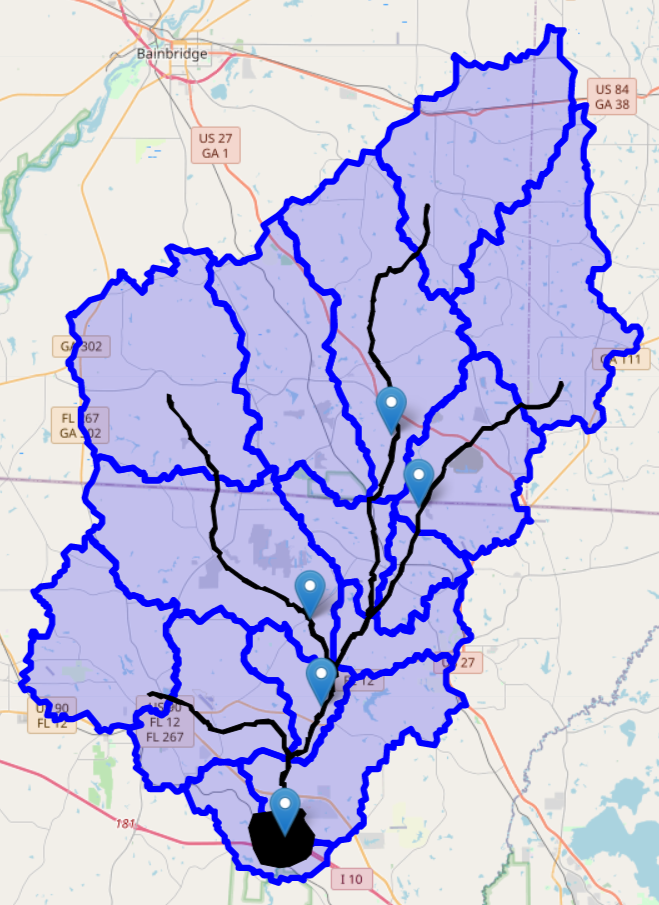
To help assess model calibration based on geometric means, this scatter plot reports the geometric mean (+/- the 95% confidence interval) of the paired simulated (red circles) and observed data (black squares). An additional geometric mean based on all simulated records across the entire modeling period may be included as a red triangle. Geometric means are reported for each monitoring station based on the Plot\_Title field in the Excel calibration lookup, with the order defined by the Plot\_Order field. The data records can be filtered by Date\_Time based on the date range (inclusive) that is defined in the Excel calibration lookup table. The presence of negative or zeros in a particular dataset may preclude the ability to calculate the geometric mean. Plots are limited to chlorophyll-*a*, total nitrogen and total phosphorus. When ‘Annual’ plots are activated on the Excel lookup table, additional plots (without confidence intervals) are generated on a yearly basis and saved in the ‘Year’ subfolder.

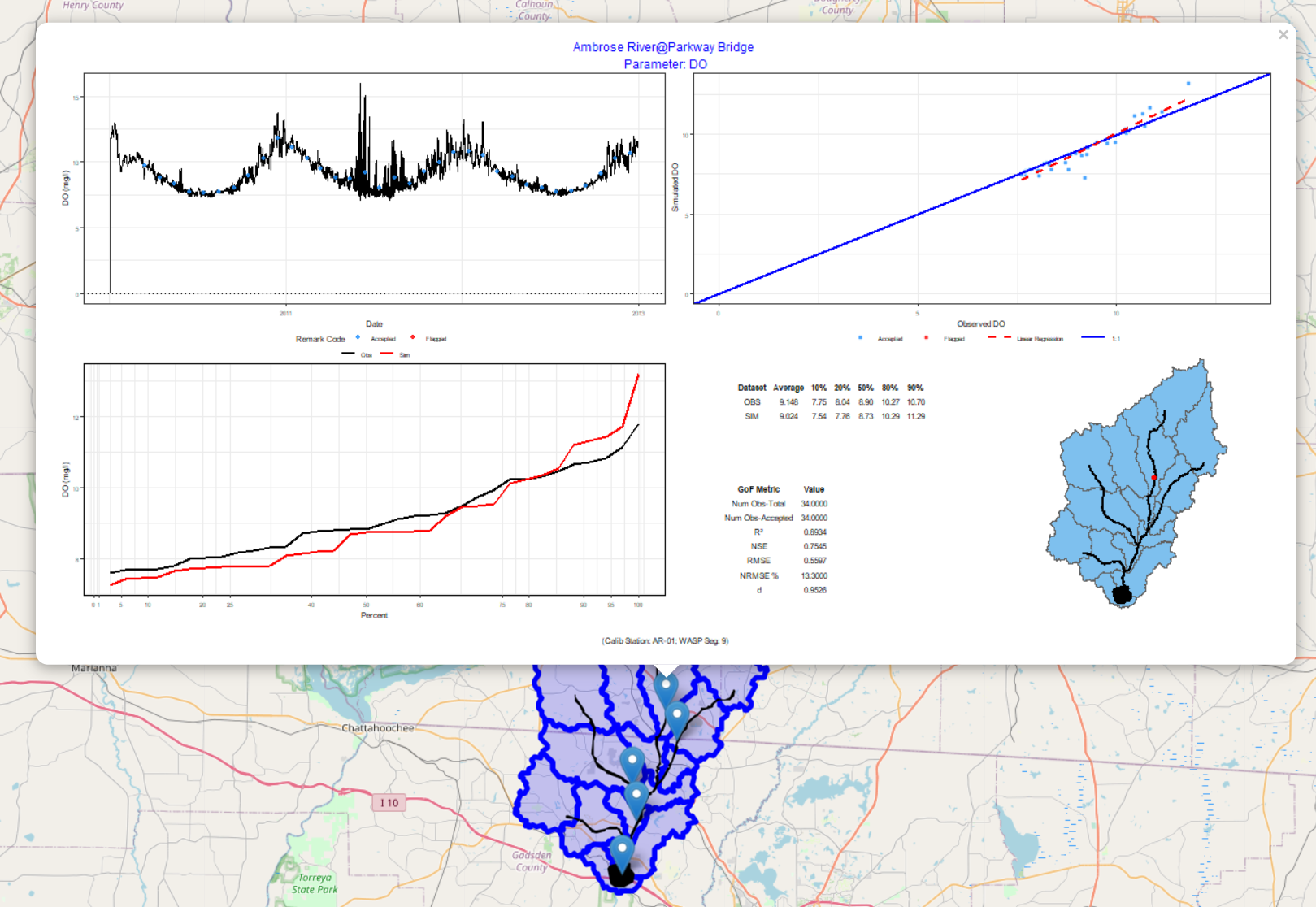
## Density distribution plot

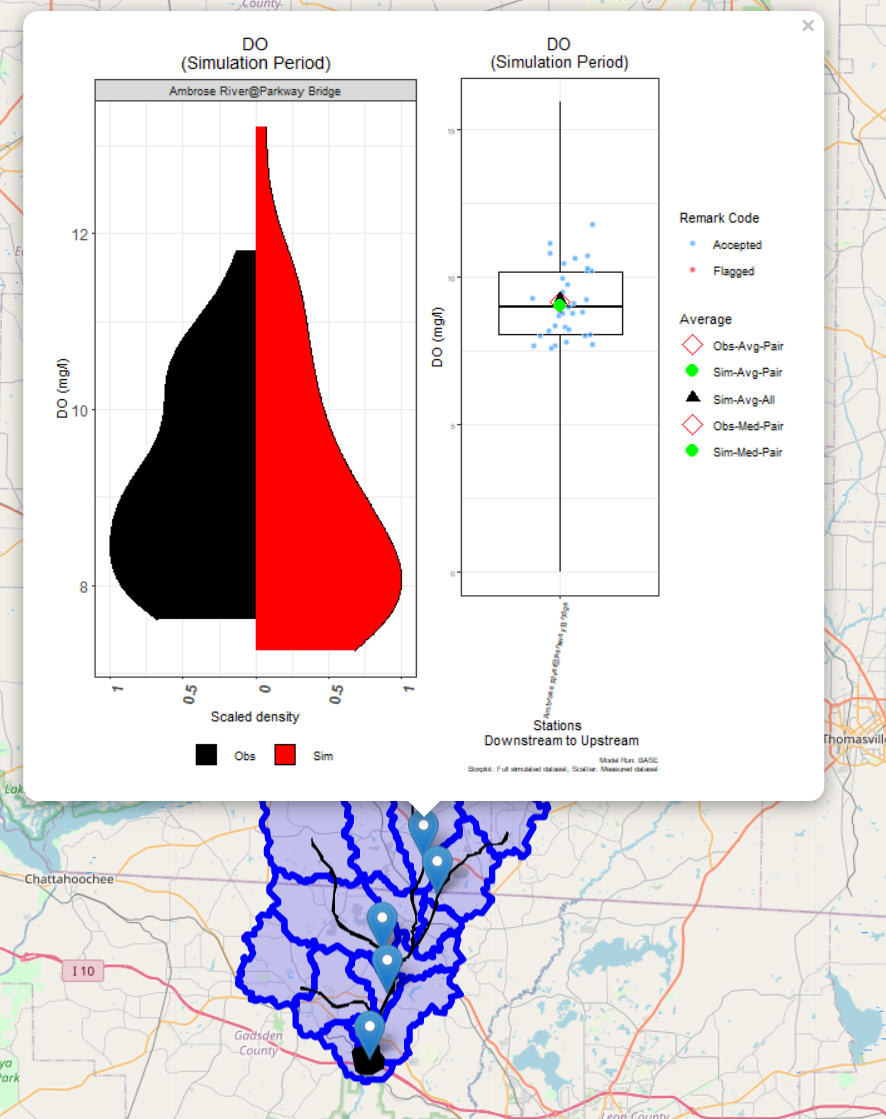


Similar to the cumulative probability plot and boxplot, this plot illustrates the degree of overlap in the range and distribution of simulated and observed data. These plots are generated based only on those simulated data records (red fill) that have a corresponding paired observed value (black fill). The distributions are scaled to 1 to allow for better comparison.

## Interactive HTML plots







When ‘Export\_as\_HTML’ option is activated in the Calib\_Lookup.xlsx file, the multipanel and/or distribution plots will be converted into HTML plots that allow the user to view the various plots spatially. When a map pin is clicked, it will open either the multipanel plot or distribution plots depending on the types of static plots selected as well. To utilize this functionality, the user must provide a watershed shapefile, stream network shapefile, and the latitude/longitudes of each monitoring station. Static and interactive plots cannot be generated simultaneously, so multiple runs of the scripts are required if the user wants to generate both types of plots.

Appendix B:

Primer on interpreting goodness of fit statistics

Quantitative analysis of model fit can be assessed using several widely used goodness of fit statistics (Moriasi et al. 2007) that were calculated using the ‘hydroGOF’ package (Zambrano-Bigiarini 2017). An overview of the various statistical comparisons is provided below.

* Arithmetic Mean () – On average, assesses how well the simulated values represent observed values. For both the observed and simulated dataset, an arithmetic mean is calculated for each parameter across the entire model simulation period.
* Percentiles – This verifies the model is reasonably predicting extreme values in the observed data. Entails inspection of the 10th and 90th percentiles of both observed and simulated data.
* Mean Error (ME) – For each pair of measured and simulated values, measures the average difference between observed and simulated data. Primarily assesses average model bias, which is one reason Mean Error is also known as Mean Bias Error. Does not indicate if the simulated value is over or underpredicting the observed value and does not consider the natural variation in the observed data. For each paired observed and simulated record, the difference of the observed and simulated value is calculated, and subsequently averaged.
* Mean Absolute Error (MAE) – Measures the average magnitude of the difference (i.e., error) between observed and simulated data. It does not consider the direction of those differences (i.e., whether the model is over or underpredicting) or natural variation in the observed data. Calculated similarly to Mean Error, but the absolute value of the difference is taken. All errors are given equal weight, which can make it more robust to outliers.
* Root Mean Square Error (RMSE) – Measures the difference (i.e., error) between observed and simulated data. This metric provides assurance that the model is matching the frequency, magnitude, and duration of water quality changes. However, it does not account for natural variability in observed data. Squared term gives more weight to large errors, which can make it relatively more sensitive to outliers. Values of RMSE range from 0 to infinity, with RMSE = 0 indicating a perfect match between observed and simulated data.
* Normalized Root Mean Square Error (NRMSE %) – Similar to RMSE; however, error is standardized relative to the range of the observed data. Other variants of the calculation standardize the error relative to the average or standard deviation of the observed data. NRMSE is reported as a percent. Values of NRMSE range from -100% to 100%, with NRMSE = 0% indicating a perfect match.
* Coefficient of determination (R2) – Assesses the strength of the relationship between observed and simulated data based on a simple linear regression. It describes the proportion of variation in the observed data that is explained by a simple linear regression relating observed and simulated data. This can be an indicator of how well the simulated data are capturing the temporal dynamics of the measured data. Values of R2 range from 0 to 1, with better fitting models possessing higher R2 values.
* Spearman Rank Correlation Coefficient (r) – Conceptually, similar to simple linear regression, but the relationship between simulated and observed values is assessed based on their rank value (i.e., highest value given a rank of 1). The correlation coefficient indicates the strength and direction of the monotonic relationship between paired simulated and observed data. This can be an indicator of how well the simulated data are capturing the temporal dynamics of the measured data. As the comparison is nonparametric, data do not need to meet assumptions of normality or equal variance. Values range from -1 to 1, with r = -1 indicating a perfect negative relationship between simulated and observed data and r =1 indicating a perfect positive relationship.
* Percent Bias (PBIAS) – Provides a measure of whether a model, on average, tends to over- or underestimate observed values. The magnitude of the difference in observed and simulated data is calculated relative to the sum of observed data. Values range from -100% to 100%, with more accurate models exhibiting PBIAS that approach 0%. Values of PBIAS > 0% indicates that the model is overestimating observed values, while PBIAS < 0% indicates the model is underestimating them.
* Nash-Sutcliffe Coefficient (NSE) – This metric is closely related to mean square error (MSE) and root mean square error (RMSE), as NSE is the ratio of MSE relative to natural variability of the observed data. Using the mean of the observed data as a baseline, it assesses the magnitude of the difference in observed and simulated data relative to residual variance (i.e., natural variation) of observed data. This unitless metric indicates how well the linear fit of observed versus simulated data fits a 1:1 line. Its use of a square term can make this metric sensitive to outliers. Values range from -Infinity to 1, whereby NSE = 1 represents a perfect match of simulated and observed data, NSE = 0 indicates that model predictions are as accurate as the mean of observed data, while NSE = -Infinity indicates that the mean of observed values is a better predictor than simulated data.
* Index of Agreement (d) – Provides a measure of model error relative to natural variability (i.e., error). It is the ratio of MSE relative to overall potential error. Similar to NSE, its use of a squared term can make this metric sensitive to outliers, although variants using different power terms can reduce this sensitivity (see Willmot et al. Int. J. Clim. 2012). Values range from 0 to 1, with an index of agreement = 1 indicating a perfect fit of simulated and observed data, and a value of 0 indicating no agreement between them.
* Modified Kling-Gupta Efficiency (KGE') – This unitless metric is similar to the Nash-Sutcliffe Coefficient (NSE) but attempts to assess fit by assigning equal weight to correlation, bias, and variability metrics. In contrast, optimal models indicated by NSE have the potential to overemphasize the linear correlation component and can exhibit reduced effectiveness when assessing highly seasonal parameters.

Accordingly, KGE' is decomposed into three components, which are simultaneously reported out with the KGE'. It uses a Pearson Correlation Coefficient (r) to assess the strength and direction of the linear correlation between measured and simulated values, the ratio of simulated means vs. observed means (β) to assess model bias, and the ratio of coefficient of variations (𝛾) to assess model variance. This decomposition helps to indicate whether a model is reproducing temporal dynamics (as represented by r) as well as capturing the measured data’s average (β) and variability (𝛾). Values range from -Inf to 1, whereby an ideal fitting model is indicated by KGE' = 1.

* + Kling-Gupta Pearson Correlation Coefficient (r) – The Pearson correlation coefficient between simulated and observed values. In contrast to the nonparametric Spearman rank correlation coefficient, the parametric Pearson correlation coefficent is calculated based on the actual paired values rather than their ranked values. This can be an indicator of how well the simulated data are capturing the temporal dynamics of the measured data. Values range from -1 to 1 and a perfectly fit model is indicated by r =1.
  + Kling-Gupta Beta (β: Ratio of means) – The ratio of the simulated mean to the observed mean. An ideal model has β = 1.
  + Kling-Gupta Gamma (𝛾 : Ratio of coefficient of variation) – The ratio of the simulated coefficient of variation relative to the observed coefficient of variation. An ideal model has 𝛾 = 1.

# Additional resources

* Duda et al. 2012. Transactions of ASABE 55:1 523-1547
* Gupta et al. 2009. Journal of Hydrology 377: 80-91
* Gupta et al. 2012. Journal of Hydrology 424-425: 264-277
* Jackson et al. 2019. Environmental Modelling and Software 119: 32-48
* Krause et al. 2005. Advances in Geosciences 5: 89-97
* Legates and McCabe 1999. Water Resources Research 35: 233-241
* Moriasi et al. 2007. Transactions of ASABE 50: 885-900
* Nash and Sutcliffe 1970. Journal of Hydrology 10: 282-290
* Nash and Sutcliffe 1970. Journal of Hydrology 10: 317-329
* Roberts et al. 2018. Hydrology. 5: 66 (Hydrostats package)
* Willmott et al. 2012. International Journal of Climatology 32: 2088-2094
* Zambrano-Bigiarini. 2017. (hydroGOF: goodness-of-fit functions) URL https://github.com/hzambran/hydroGOF