fluxtools: interactive Shiny tool for QA/QC and code generation of Ameriflux eddy-covariance data

18 June 2025

Summary

Eddy covariance data processing requires extensive quality control (QA/QC) to identify and remove implausible or erroneous half-hourly flux data before submission to public data repositories such as Ameriflux (AmeriFlux Management Project 2025). *Fluxtools* (Key 2025) is an R (4.5.0) *Shiny* (Chang et al. 2024) application built with *Plotly* (Sievert et al. 2024) and *dplyr* (Wickham et al. 2023) packages designed to streamline this workflow by providing interactive visualization, year-based filtering, and on-the-fly R code generation for specified data removal. Users can visually flag anomalous data points (i.e., sensors went down, physically implausible data), accumulate multiple cleaning steps, inspect before/after R^2 values via base R's *lm()* function, and export both a cleaned .csv file and a full R script that records every decision. *Fluxtools* significantly accelerates the QA/QC workflow, ensuring transparent, reproducible, and shareable data cleaning suitable for final dataset preparation and repository submission.

Figure 1: *Fluxtools* hex logo

Key features:

Interactive Plotly Scatterplots: Plot any numeric or time variable; hover mouse over data points to see timestamps and values; export plots as .png directly from the app

Flexible point selection: Select data points via box, lasso, or by standard-deviation ( ) cutoffs. *See Fig 2* for interface and data selection example

On-the-fly R code generation: The *Preview* pane shows selected timestamps and values; ready-to-copy R code using *dplyr's case_when(… ~ NA_real)* snippets generate in the *current* code box automatically; *add current selection* adds code to the *accumulated* code box for easy and continuous data selection

Before/after R^2 diagnostics: When numeric variables are compared against each other, a linear regression generates a R^2 value. Automatically computes post-removal R^2 value where selected data points are dropped to see step comparisons. *See Fig 3* for an example of the *Fluxtools* interfacing using the $\pm$ outliers selection tool. The top (red) R^2 is for all data points and the bottom R^2 (orange) is when selected points are dropped from the linear regression model

Export a cleaned CSV: *Apply removals* in-app (converting data points into NAs for selected timestamps) and download both a cleaned .csv and a comprehensive R script documenting each data removal step

Statement of need

High-frequency (10 Hz; data recorded 10 times per second) eddy covariance measurements produce large datasets that must be aggregated carefully into half-hourly fluxes, requiring meticulous quality assurance and quality control (Burba 2021). Tools like *EddyPro* (LI-COR Biosciences 2021) converts raw 10Hz data into half-hourly fluxes, while R packages like *REddyProc* (Wutzler et al. 2024), and Python tools like PyFluxPro (Isaac 2021), automate u*-threshold filtering, gap-filling, and flux partitioning. These tools excel at bulk data processing but offer no interactive means to inspect or carefully remove outliers that slip through their algorithms.

In practice, data managers resort to custom scripts, extensive manual visualization, fragmented documentation, and significant effort to detect and remove erroneous data points caused by sensor drift, malfunction, or calibration issues. These procedures are labor-intensive, prone to errors, challenging to reproduce, and lack transparency. *Fluxtools* fills this gap by pairing an interactive scatterplot-based interface with on-the-fly R code generation. Users can visually flag implausible half-hourly data points, automatically generate the exact *case_when(… ~ NA_real_) dplyr* code snippets (or apply removes automatically in the app) and export a .zip file containing a cleaned CSV plus a comprehensive R script documenting each data removal step that captures every user-made QA/QC decision.

By combining interactivity with code-based reproducibility, *Fluxtools* significantly streamlines and clarifies the QA/QC workflow. It promotes transparent documentation of decisions, reduces manual effort, and

accelerates the preparation of flux data for repository uploads such as Ameriflux (e.g., sites like US-VT1 and US-VT2). Ultimately, Fluxtools lowers the barriers to robust and reproducible QA/QC workflows, enabling researchers to devote less time to manual anomaly detection and more time to scientific analysis.

Code Example

Fluxtools can be installed from Github.

```
library(fluxtools)

#Set your site's UTC offset (e.g., -5 for Eastern Standard Time)
fluxtools::run_flux_qaqc(-5)
```

Acknowledgments

Fluxtools was developed in my role as Co-Principal Investigator for the AmeriFlux US-VT1 (Key and Novick 2025a) and US-VT2 (Key and Novick 2025b) sites to streamline our QA/QC pipeline. I thank Ameriflux for maintaining clear, up-to-date data-formatting specifications (see Ameriflux Data Variables) and I acknowlege the broader R and Shiny communities for their efforts in advancing interactive data-visualization tools.

I am grateful to Dr. Kim Novick (Indiana University), Housen Chu (Ameriflux), and Benju Baniya (Texas A&M University) for their invaluable expertise in assisting with the Ameriflux data processing workflow. Without their support, this data pipeline would not have been possible. I also thank Dr. Mallory Barnes, Dr. Daniel Beverly, and Dr. Xian Wang (all at Indiana University) for their support during development and testing. Special thanks goes to George Burba (LI-COR Biosciences) for his encouragement and for insightful discussions that have helped shape the future of *Fluxtools*.

This work was supported by NSF's Division of Environmental Biology (DEB) Dynamics of Socio-Environmental Systems Program (Award 22-06086) and NASA's Carbon Monitoring System (Award 80NSSC23K1254).

References

AmeriFlux Management Project. 2025. "AmeriFlux Data Portal." 2025. https://ameriflux.lbl.gov/.

Burba, George. 2021. *Eddy Covariance Method - for Scientific, Regulatory, and Commercial Applications*. Lincoln, Nebraska: LI-COR Biosciences. https://www.licor.com/products/eddy-covariance/ec-book.

Chang, Winston, Joe Cheng, JJ Allaire, Carson Sievert, Barret Schloerke, Yihui Xie, Jeff Allen, Jonathan McPherson, Alan Dipert, and Barbara Borges. 2024. *Shiny: Web Application Framework for r*. https://doi.org/10.32614/CRAN.package.shiny.

Isaac, Peter. 2021. "PyFluxPro." https://github.com/OzFlux/PyFluxPro.

Key, Kesondra. 2025. "Fluxtools: Interactive QA/QC for AmeriFlux Data." Zenodo. https://doi.org/10.5281/zenodo.15626645.

Key, Kesondra, and Kim Novick. 2025a. "AmeriFlux BASE US-VT1 Vermillion Tributary Paired Cropland – Site 1 (Corn/Soy; No Cover Crops)." AmeriFlux Network, Lawrence Berkeley National Laboratory. https://doi.org/10.17190/AMF/2567994.

———. 2025b. "AmeriFlux BASE US-VT2 Vermillion Tributary Paired Cropland – Site 2 (Corn/Soy; Cover Crops)." AmeriFlux Network, Lawrence Berkeley National Laboratory. https://doi.org/10.17190/AMF/2567995.

LI-COR Biosciences. 2021. "EddyCovariance Processing Software (EddyPro)." https://www.licor.com/env/products/eddy-covariance/eddypro.

Sievert, Carson, Chris Parmer, Toby Hocking, Scott Chamberlain, Karthik Ram, Marianne Corvellec, Pedro Despouy, Salim Brüggemann, and Plotly Technologies Inc. 2024. "Plotly: Create Interactive Web Graphics via 'Plotly.js'." https://doi.org/10.32614/CRAN.package.plotly.

Wickham, Hadley, Romain François, Lionel Henry, Kirill Müller, and Davis Vaughan. 2023. *Dplyr: A Grammar of Data Manipulation*. https://doi.org/10.32614/CRAN.package.dplyr.

Wutzler, Thomas, Markus Reichstein, Antje Maria Moffat, and Mirco Migliavacca. 2024. *REddyProc: Post Processing of (Half-)hourly Eddy-Covariance Measurements*. https://doi.org/10.32614/CRAN.package.REddyProc.