

Tune your SWAT+ model with **SWATtunR**



Svajunas Plunge^{a*}, *Christoph Schürz*^b, *Michael Strauch*^b,
Mikolaj Piniewski^a

*Contact: svajunas_plunge@sggw.edu.pl

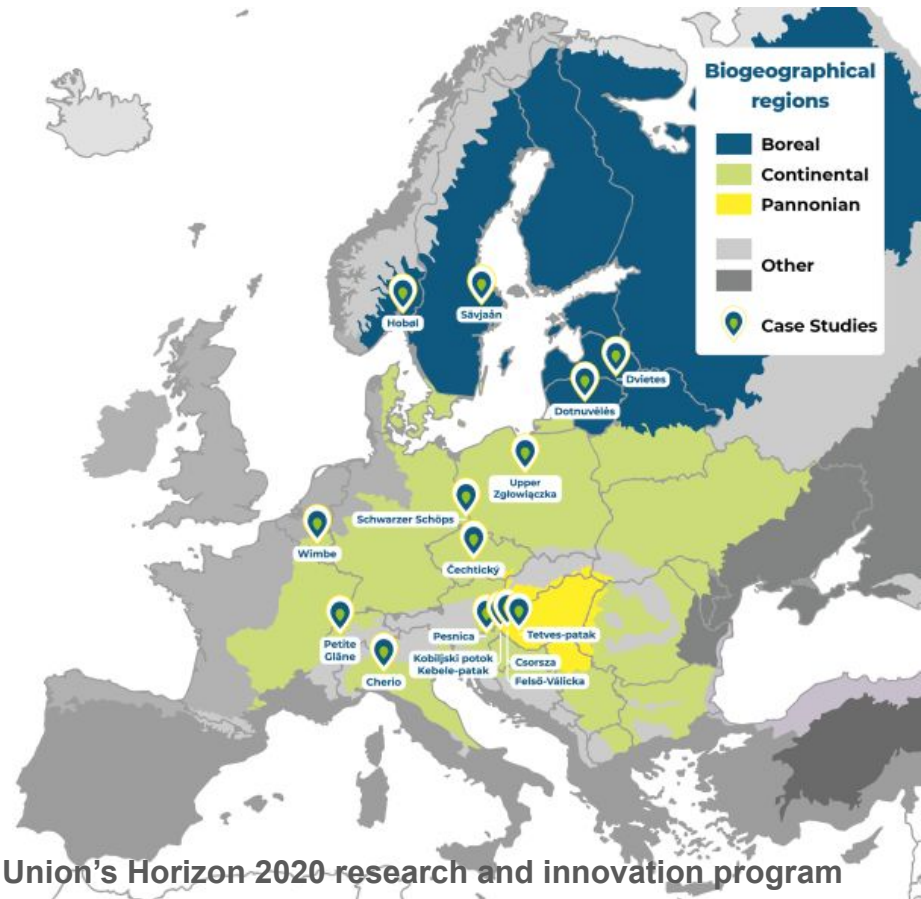
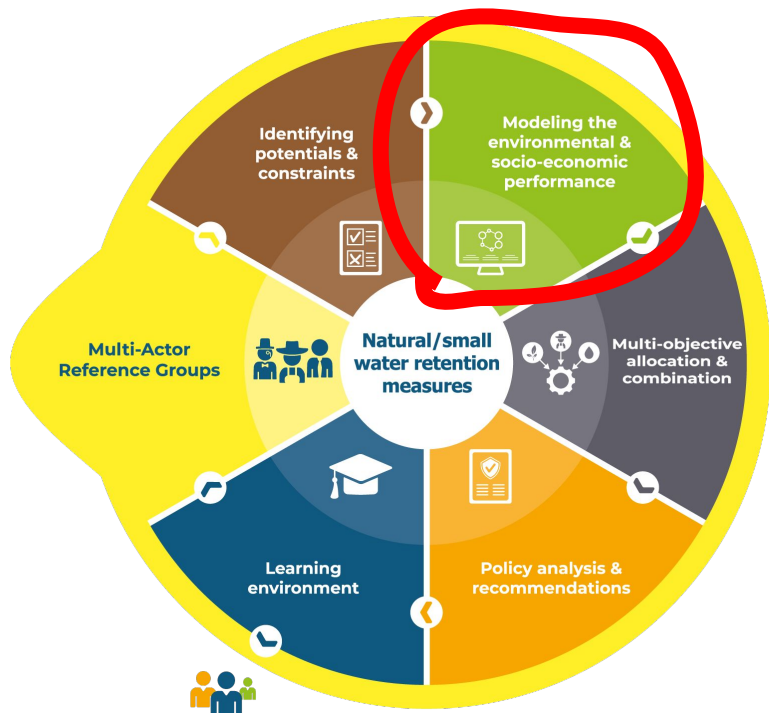
^a*Warsaw University of Life Sciences*

^b*Helmholtz Centre for Environmental Research GmbH – UFZ*





OPTAIN



This project has received funding from the European Union's Horizon-2020 research and innovation program under grant agreement No. 862756.

Workflow in R



Dr. Gu Ru

SWATprepR
SWAT+ input data preparation

SWATdoctR
Model diagnostics tool
for SWAT+ model setups

SWATrunR
Running SWAT simulations in R

SWATmeasR
Implementation of NSWRMs in
SWATbuildR model setups



SWATbuildR
An object connectivity
based SWAT+ model builder

SWATfarmR
Simple rule based management
operation scheduling

SWAttunR
Tuning SWAT+ model parameters

**Vision: SWAT+ modelling process
fully scriptable in R**

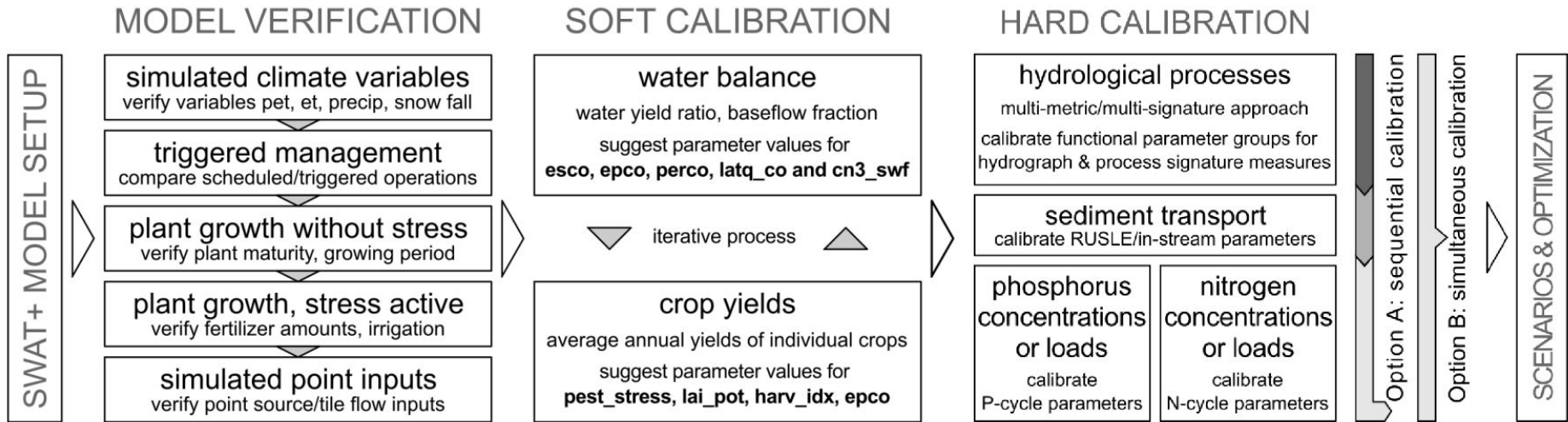
Life calibrating SWAT models could be hard 🙄



Proposed calibration workflow in OPTAIN (all scripted)

SWATdoctr

SWATtunR





Introduction to SWATtunR



SWATtunR

[devel version](#)
[0.0.1.9015](#)
[last commit](#)
[today](#)
[lifecycle](#)
[stable](#)
[repo status](#)
[Active](#)
[code size](#)
[104 kB](#)
[license](#)
[MIT](#)

The goal of SWATtunR is to help with the [SWAT+ model](#) calibration and validation. These functions were developed and tested for the implementation of modeling tasks in the [OPTAIN project](#), [Nordbalt-Ecosafe](#) and [LIFE SIP Vanduo](#). Functions are initially developed by [Christoph Schuerz](#), which added important capability on top of other R tools designed for the SWAT/SWAT+ models. Therefore, we highly recommend trying and using these tools:

- [SWATbuildR](#)¹ - R tool for building SWAT+ setups;
- [SWATprepR](#) - SWAT+ model input data preparation helper. The package is presented in the article Plunge, Szabó, et al. (2024);
- [SWATfarmR](#) - R tool for preparing management schedules for SWAT model;
- [SWATdoctr](#) - A collection of functions in R and routines for SWAT model calibration and model diagnostics. The package is presented in the article Plunge, Schürz, et al. (2024);
- [SWATrunR](#) - R tool for running SWAT models for different parameters and scenarios. Please install branch names *remove_legacy*. It could be done using line like this `remotes::install_github("chrisschuerz/SWATrunR@remove_legacy")`
- [SWATmeasR](#) - R tool for implementing Natural/Small Water Retention Measures (NSWRMs) in the SWAT+ models and running scenarios.

SWATprepR
SWAT+ input data preparation



SWATdoctr
Model diagnostics tool
for SWAT+ model setups

SWATrunR
Running SWAT simulations in R

SWATmeasR
Implementation of NSWRMs in
SWATbuildR model setups

SWATbuildR
An object connectivity
based SWAT+ model builder

SWATfarmR
Simple rule based management
operation scheduling

SWATtunR
Tuning SWAT+ model parameters

Links

[Browse source code](#)

[Report a bug](#)

License

[MIT](#) + file [LICENSE](#)

Citation

[Citing SWATtunR](#)

Developers

Svajunas Plunge

Author, maintainer

Christoph Schuerz

Author

Michel Strauch

Author

Mikołaj Piniewski

Author

Website for the package

<https://biopsichas.github.io/SWATtunR/>

Soft calibration

Crop Yields

Water Yield

Hard calibration

Run calibration

Plot results

Extend calibration

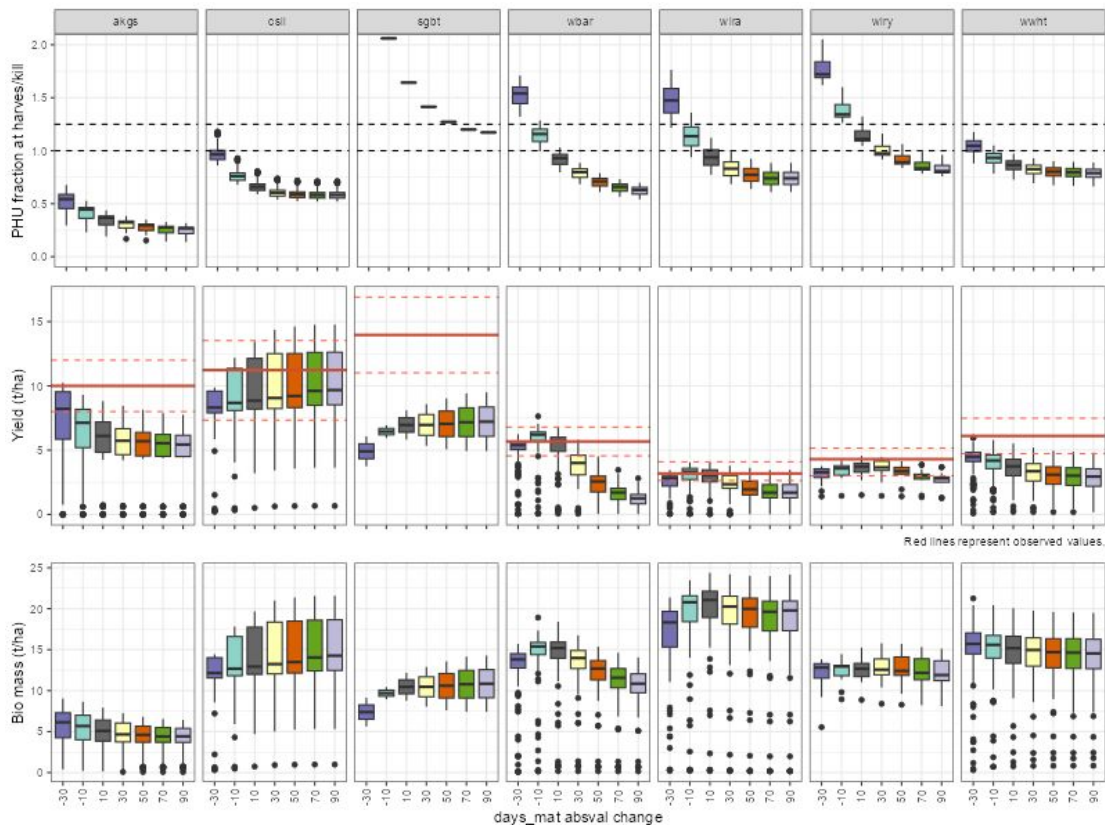
Validation

What do you get with **SWATtunR**?

New

- Definition of parameters and model runs done with SWATrunR (former SWATplusR), but there is a workflow and guidance on SWATtunR website
- New functions to calculate model performance metrics on one or multiple variables
- Result visualisation options:
 - Dotty plots with multiple customization options
 - Interactive time series plots
 - Parameter identifiability plots
 - OAT analysis plots
 - Cal/Val comparison plots
- 'calibration.cal' file export function
- QA workflow with SWATdoctR (in process)

Step 1. Soft-cal for crop yields (days_mat)



SWAT2012
Plant heat units (PHU)

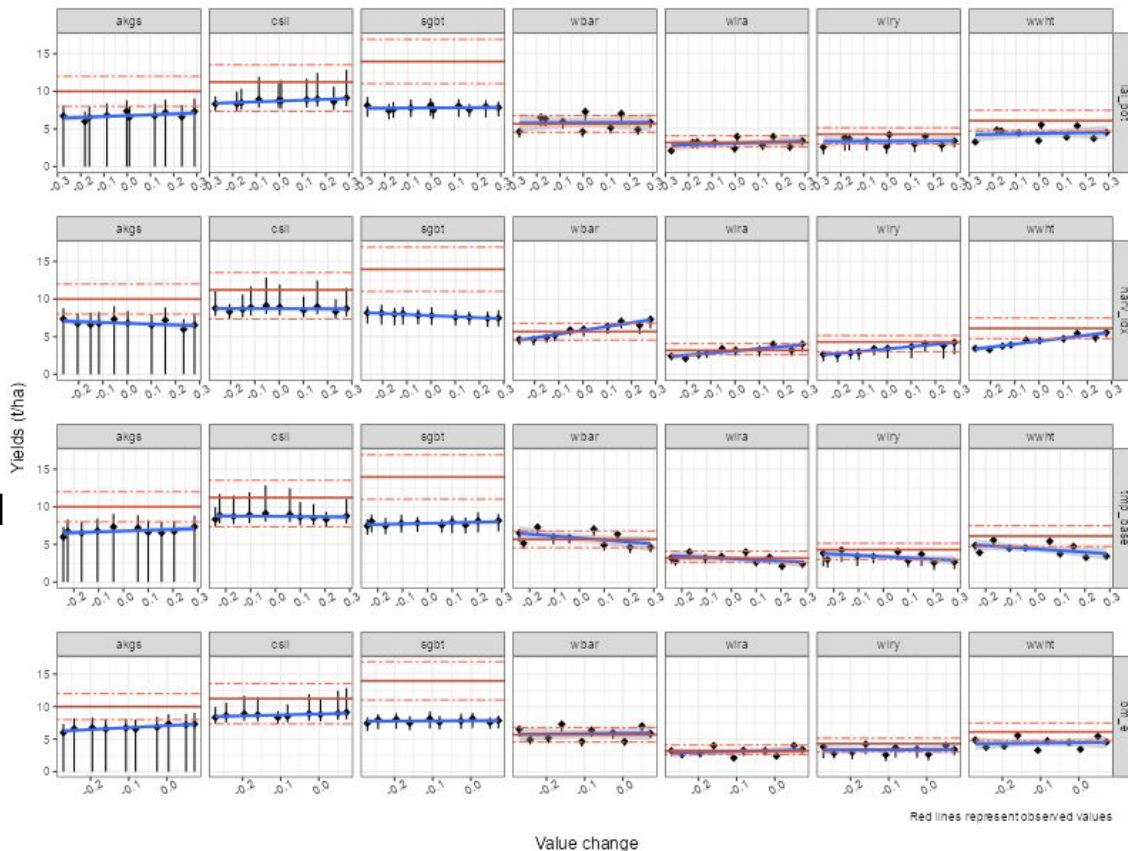


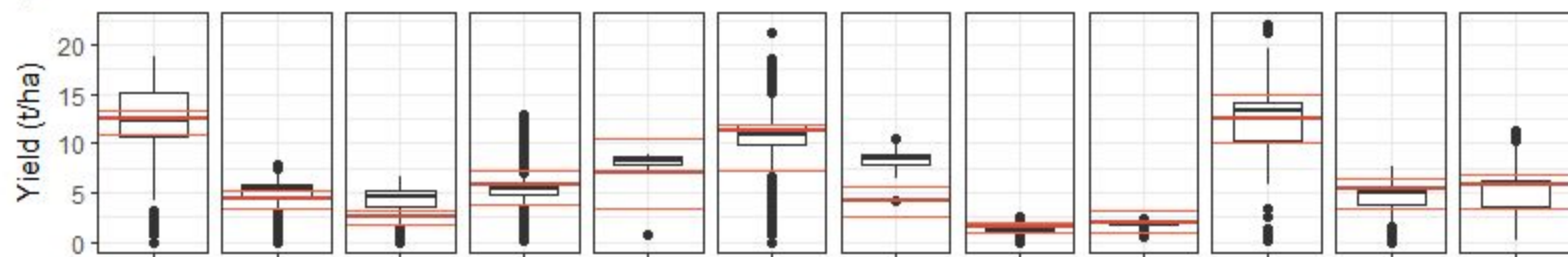
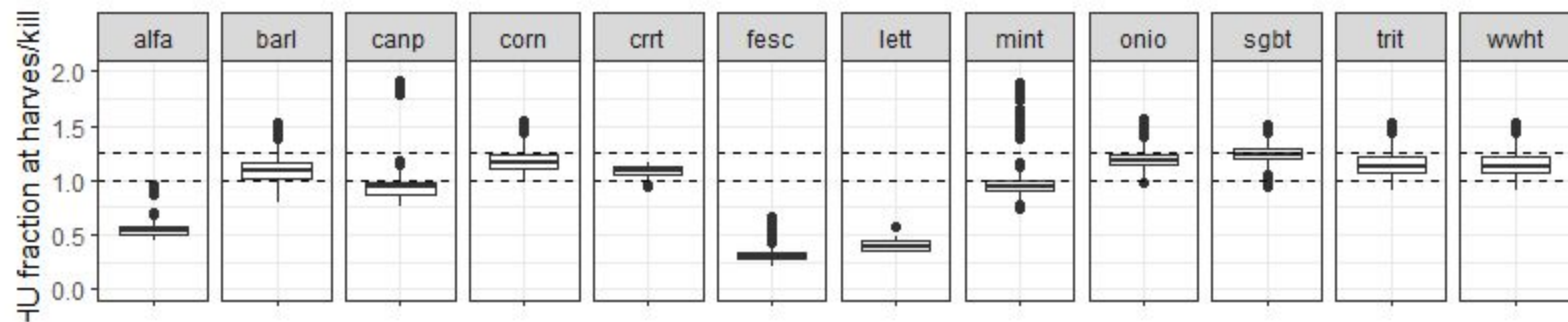
SWAT+
Days to maturity
(days_mat)

Red lines are observed yields

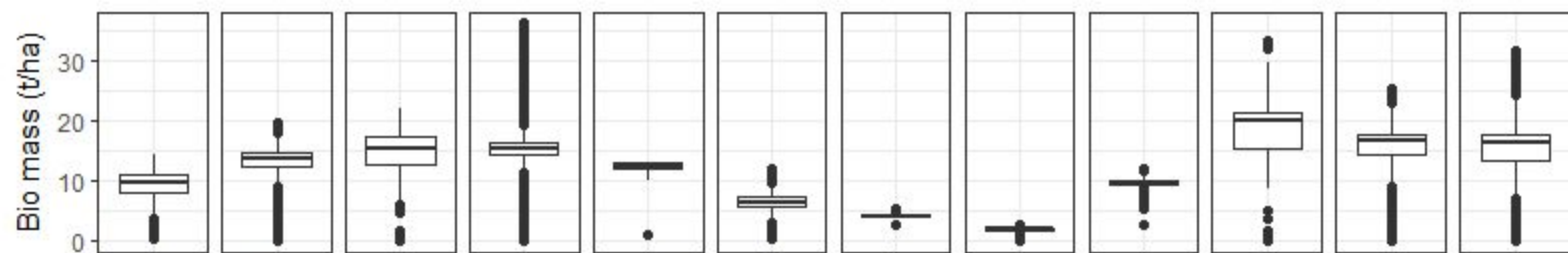
Step 2. Soft-cal for crop yields (4 additional parameters)

- **lai_pot** - maximum potential leaf area index
- **harv_idx** - Harvest index for optimal growth conditions
- **tmp_base** - Minimum temperature for plant growth
- **bm_e** - Biomass-energy ratio corresponding to the 2nd point on the radiation use efficiency curve





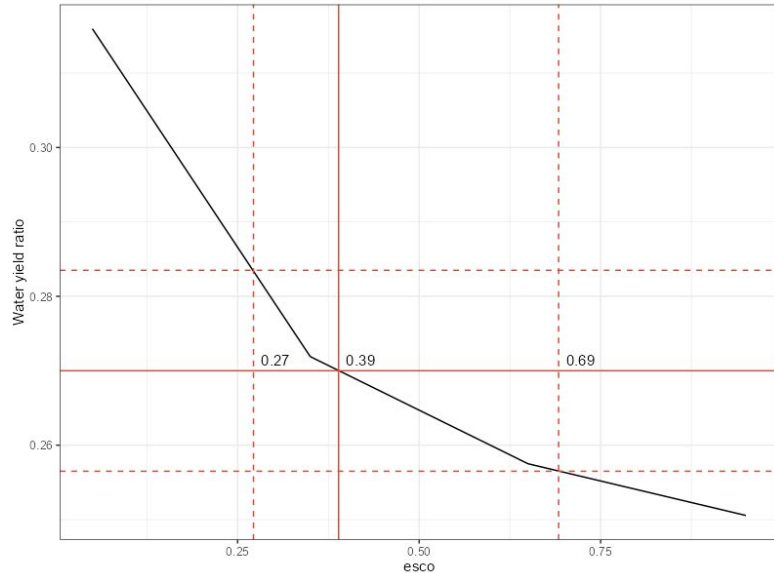
Red lines represent observed values.



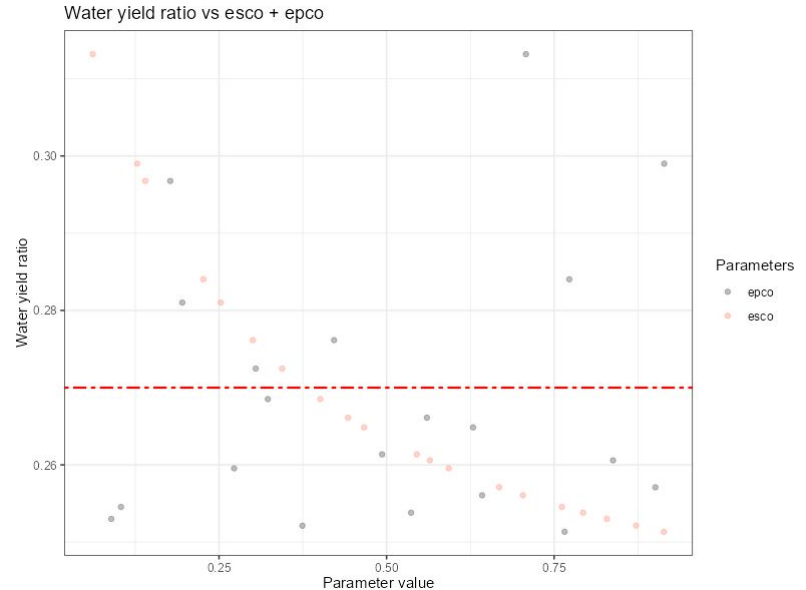
Step 3. Soft-cal for water yield

Fitting ESCO to match simulated and observed water yield ratio

Option A: ESCO



Option B: ESCO + EPCO



Step 4. Hard calibration workflows

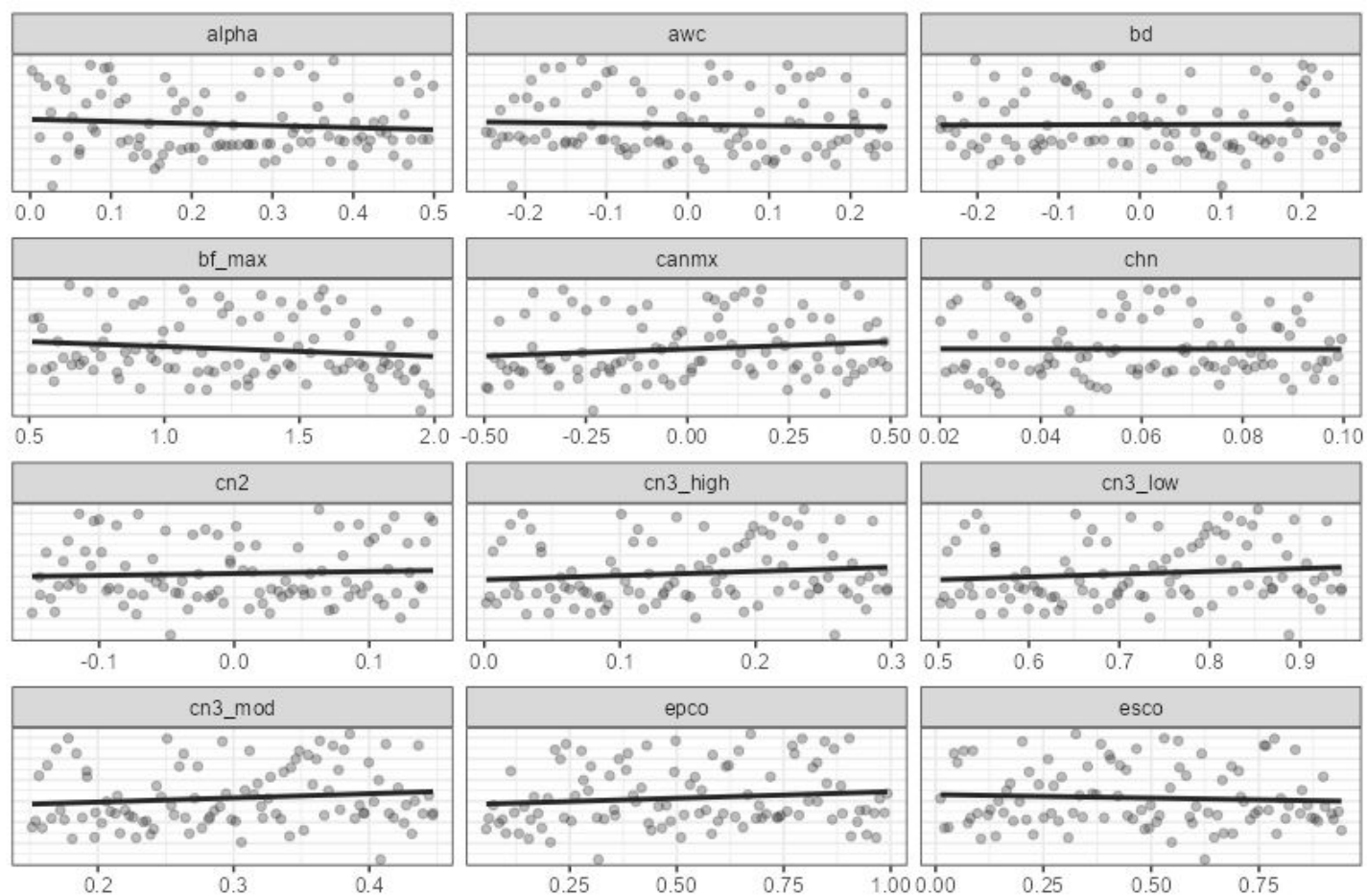
Option A: Sequential calibration

- Define **flow** parameters
- Run model
- Calculate performance metrics
- Select parameter set/s
- **Add N, P, sediment parameters**
- **Run model**
- **Calculate performance metrics**
- **Select best parameter set/s**
- Prepare 'calibration.cal'

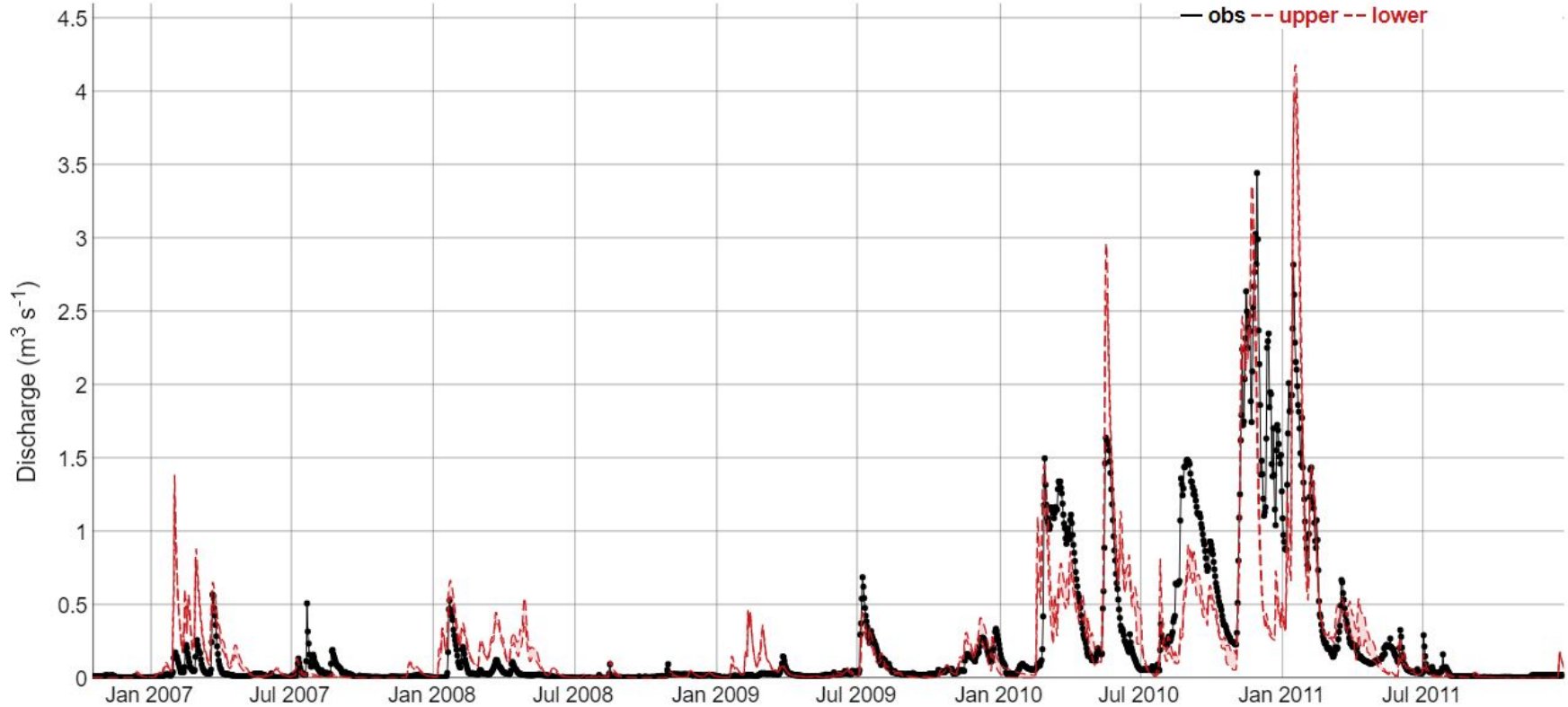
Option B: Simultaneous calibration

- Define **all** parameters
- Run model
- Calculate performance metrics
- Select parameters set/s
- Prepare 'calibration.cal'

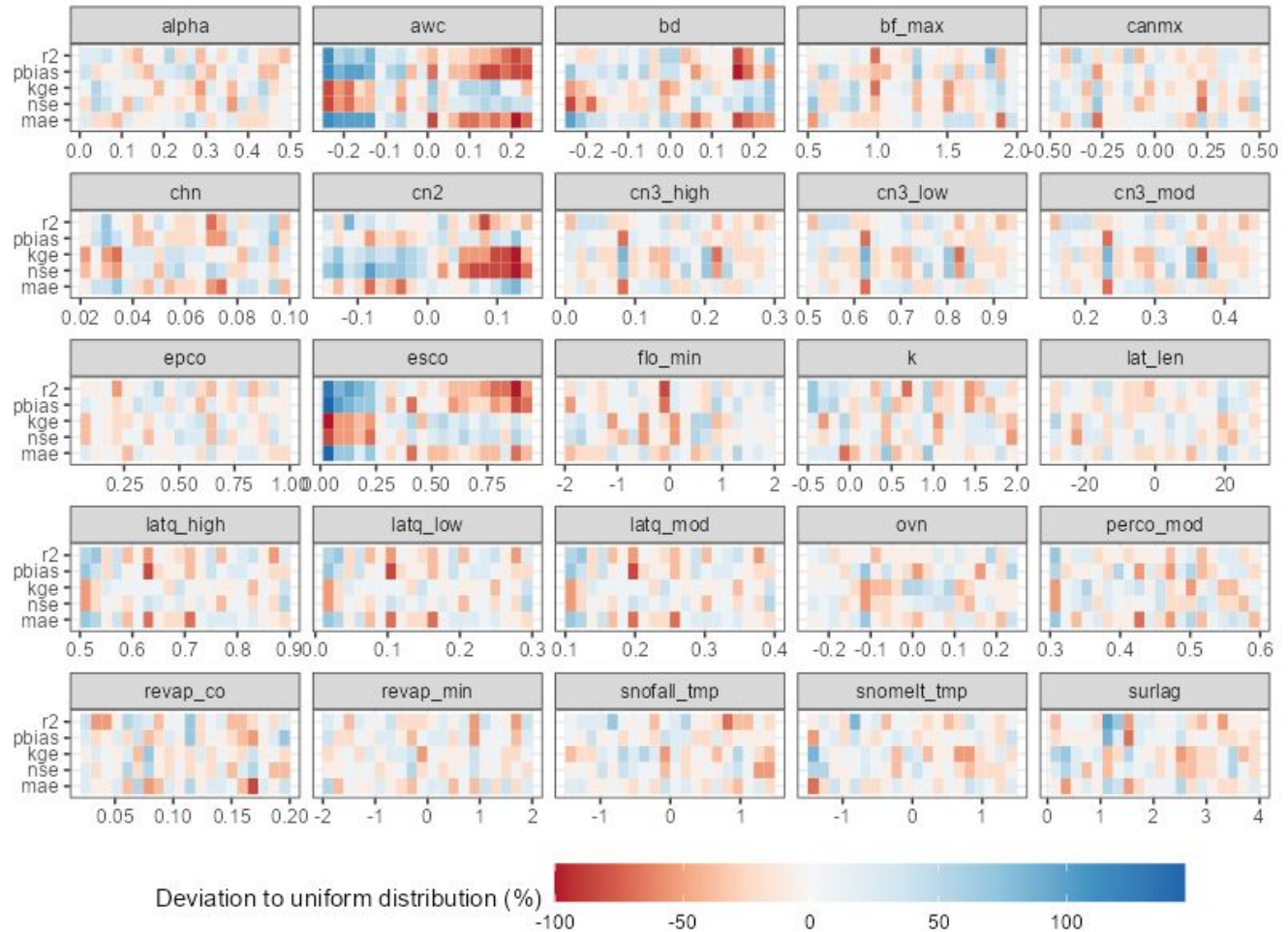
Example (dotty plot)



Example (time series plot)

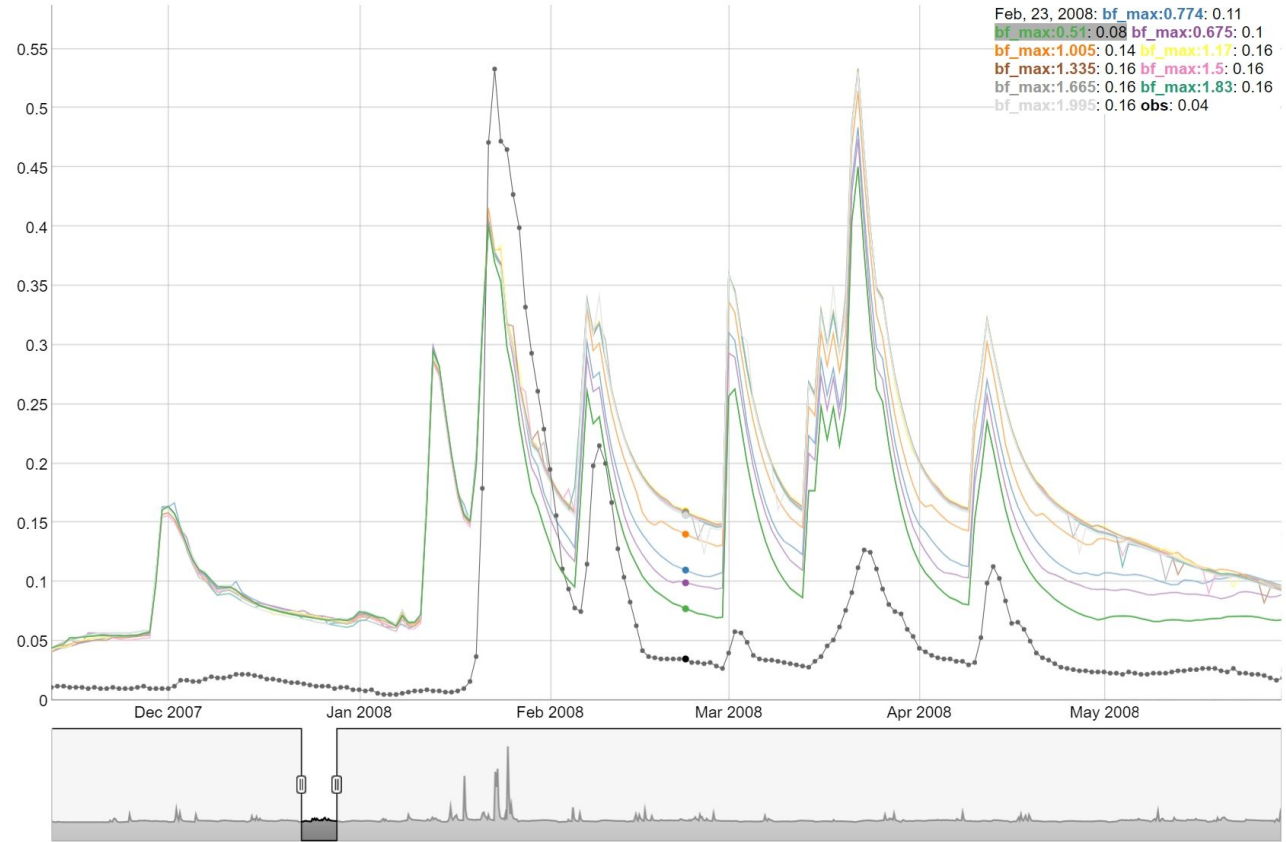


Example (identifiability plot)

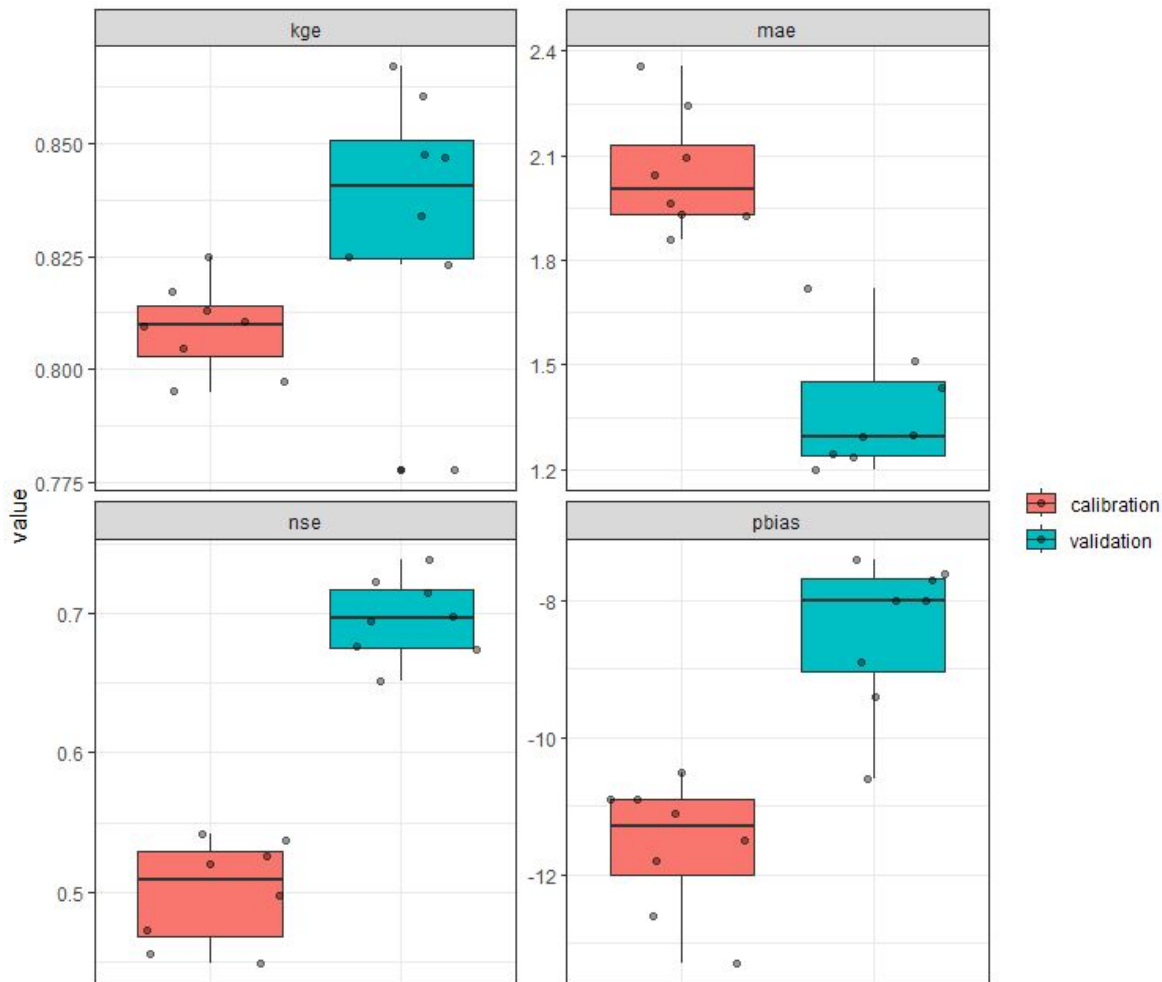


Adapted from Guse, Björn, Jens Kiesel, Matthias Pfannerstill, and Nicola Fohrer. 2020. "Assessing parameter identifiability for multiple performance criteria to constrain model parameters." *Hydrological Sciences Journal* 65 (7): 1158–72. <https://doi.org/10.1080/02626667.2020.1734204>

Example (OAT plot)



Example (cal/val comparison)



Summary

- SWAT+ soft & hard calibration, validation is fully scriptable in R.
- Saving time, documenting and corrected.
- Easily extended for multiple objectives and fully parallelized.
- Website and SWATtunR package provide easily applied tools and guidance.
- Scripted workflows already tested in two projects. Others are in line.
- Plan is to extend additional capabilities and prepare article.

Could it be any more fun?

