Example of CWatM calibration on the Rhine basin

We followed the example “calibration fast 2” from exercice 9.

Overview

To run the simulation, we need to run “*calibration_single.py*” with the calibration settings file “*settings_fast_2*”.
This calibration settings file defines main pathnames, where are the observed data to compare, the starting and ending dates of these data, the CWatM settings file used as template, the runpy command, and the initial parameters.

The CWatM settings file is “*templates/settings_fast.py*”.
This settings file will be used as a template for all simulations, only parameters (‘CALIBRATION’ section) noted with a “%” before will be calibrated.

Results from the simulation will be saved in the folder ”*calibrationRuns_fast2*” as indicated in “*settings_fast_2*”. It will generate one folder per simulation. In this folder, “*calibration_single.py*” will also generates other files containing information about the calibration.

Main settings

We did the same calibration than presented in exercice 9 (two years of simulation). However we reduced the number of parameters to calibrate to make it easier to understand. Other parameters are fixed with standard values.

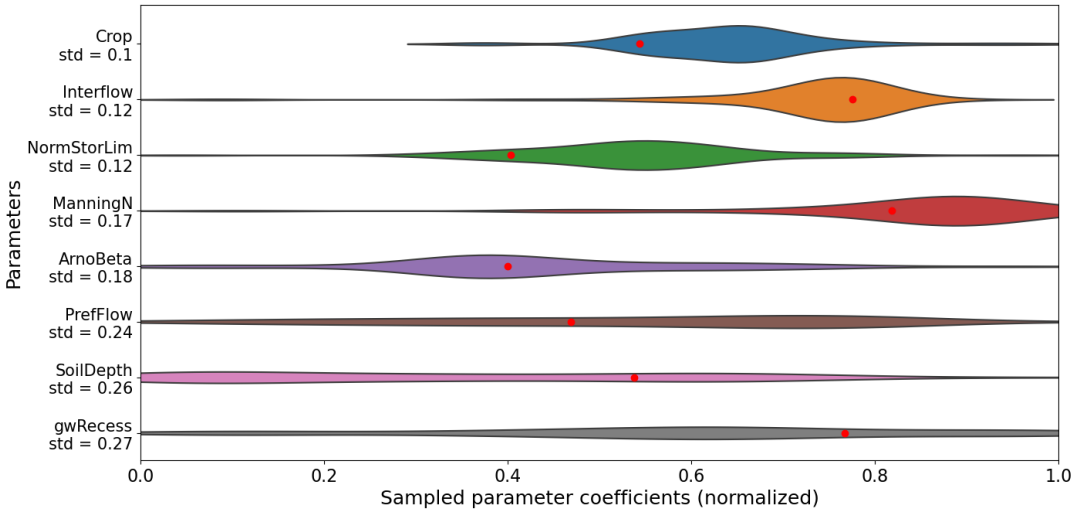
We tried to calibrate 7 parameters:
crop KC, soil depth, pref. flow, arno beta, interflow factor, groundwater recession, manningN and normalStorageLimit.
For most of them we are multiplying the parameter map (soil depth for example, but it can be just one value for the whole study area) by a calibration coefficient applied to the whole study area. Thus, if soildepth_factor = 2, the soil depth map will twice thicker than the initial map provided as input.
For some parameters, like normalStorageLimit or ArnobetaAdd, it does not correspond to a coefficient to calibrate but to a parameter value.

Extracting results from the calibration

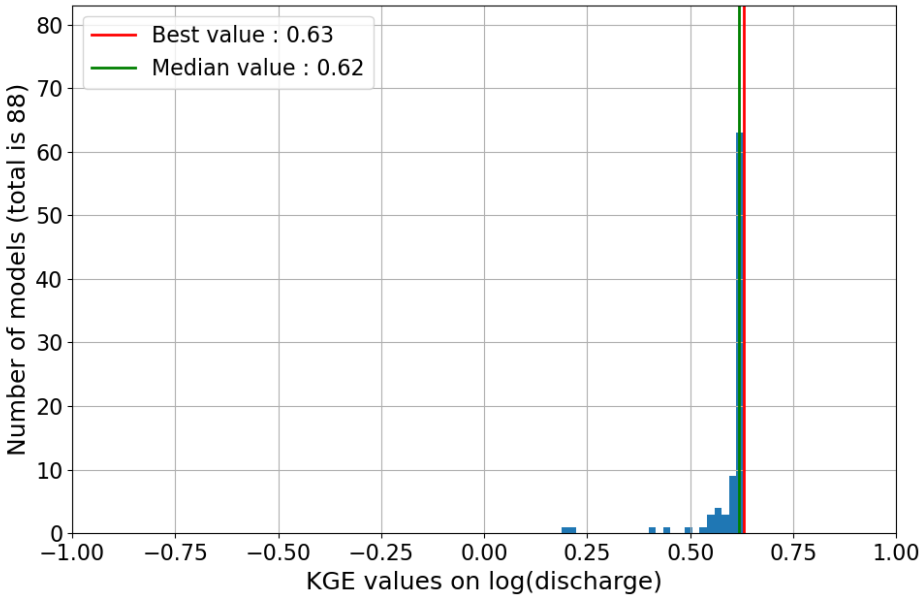
As defined in exercice 9 from the tutorial, the calibration is done on the KGE efficiency criteria comparing observed and simulated discharge at one discharge station (location defined in “*templates/settings_fast.py*”).

Results can be extracted by running *Violin_plots.py* to display the optimal value of each parameter (normalized in the plot) and to display KGE values of simulation as well.

The graph indicates how each parameter converges toward an optimal value (red dots) by showing the distribution of the sampled values. If the calibrated parameter is very well constrained, the standard deviation should be small, and the optimal value should be where there are more samples.

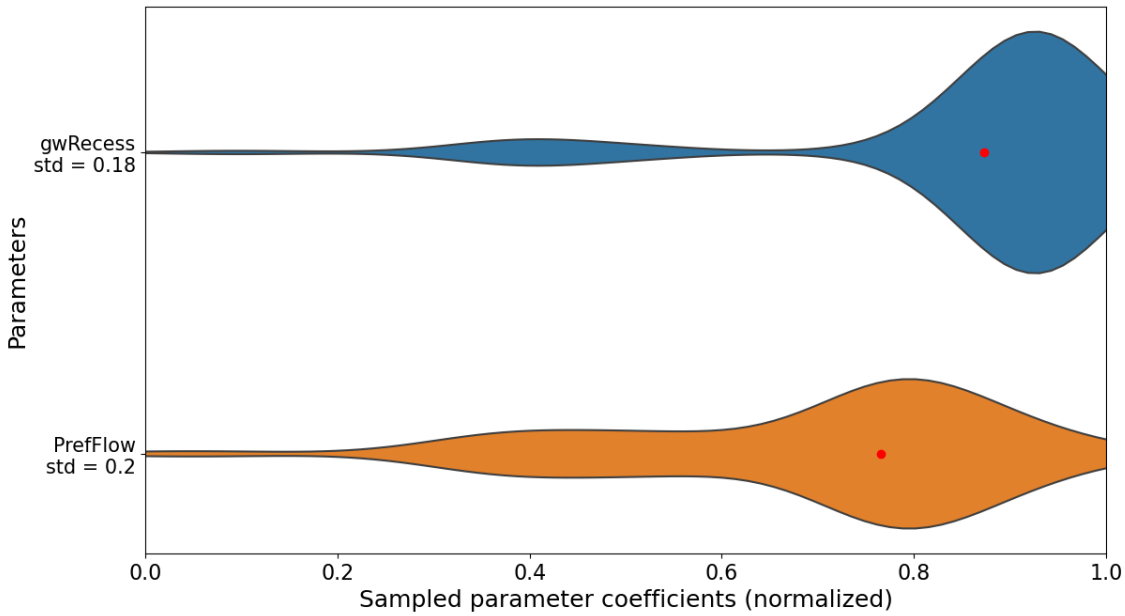


This second graph highlights the distribution of the KGE values:



Simpler calibration

We also did the same exercice but with only two parameters to give an idea of the sensitivity of the calibration. All required are indicated with a "_V2" at the end.
We only calibrate preferential flow and groundwater recession coefficients. All the other parameters are fixed. Note how these two parameters are well constrained now. However, we can expect to converge toward different preferential and recession coefficients if the fixed parameters are different.


gwRecess
std = 0.18
PrefFlow
std = 0.2
Parameters
0.0
0.2
0.4
0.6
0.8
1.0
Sampled parameter coefficients (normalized)