Post-process eddy covariance data with ease using R package openeddy

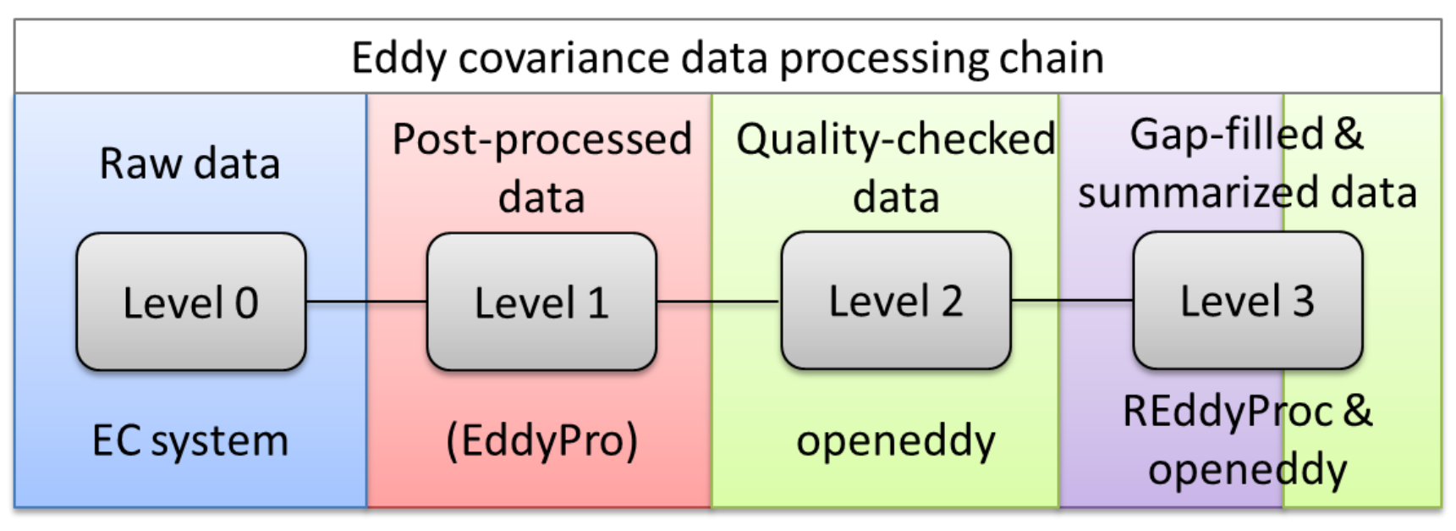
Ladislav Šigut (sigut.l@czechglobe.cz) – Global Change Research Institute CAS, Bělidla 4a, 603 00 Brno, CZ

Motivation

Eddy covariance provides rich data sets that can be inspected and utilized in many ways.

Data processing consists of multiple dependent steps that should be ideally streamlined.

Generalized tools, especially for designing custom quality control schemes, are still lacking.



openeddy

github.com/lsigut/openeddy

The openeddy R package is a software infrastructure for eddy covariance data post-processing. It provides tools for handling data with attached metadata, setting up quality control scheme and summarizing and plotting the results. It is aligned with REddyProc package that provides methods for uStar-filtering, gap-filling, and flux-partitioning. Thus the combined use of openeddy and REddyProc allows to run the whole eddy covariance post-processing chain.

Tutorials

To learn how to use openeddy, go to:

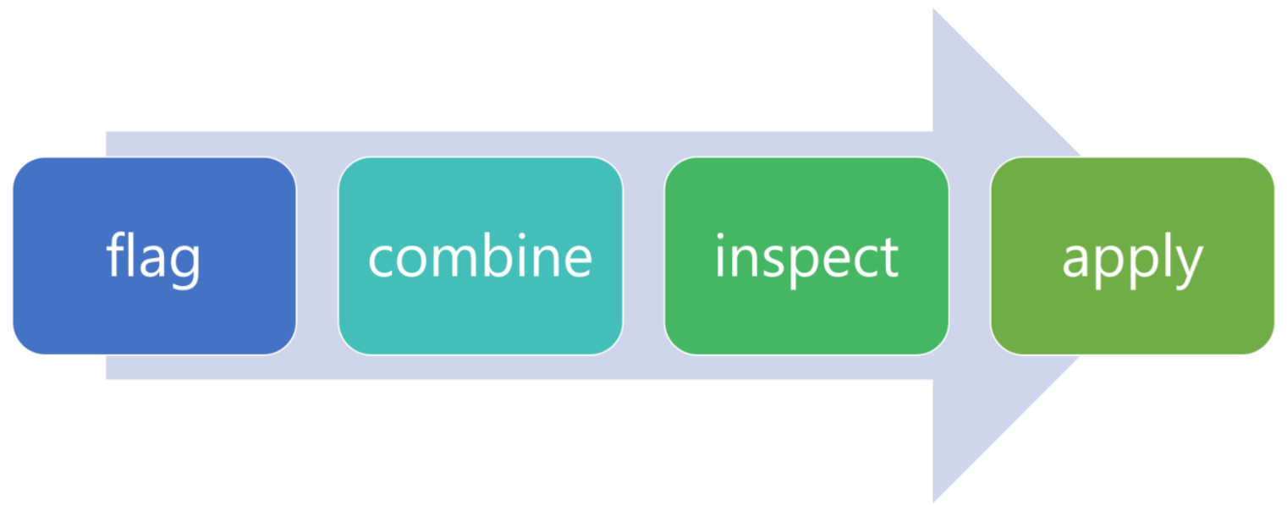
https://github.com/lsigut

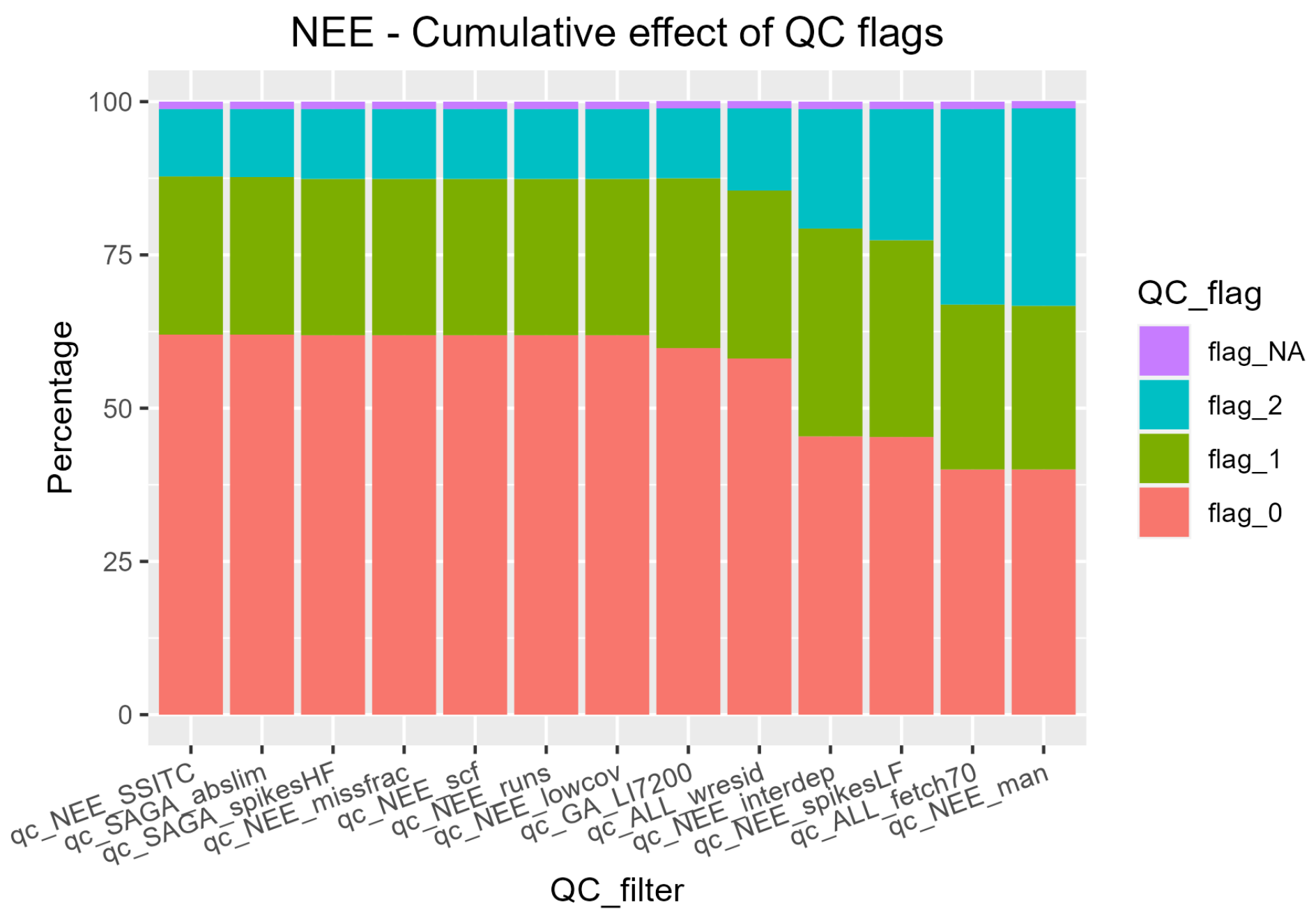
/openeddy – all functions are fully documented with small examples

/openeddy_tutorials – see isolated use-cases of small set of functions

/ROI_boundary – fetch_filter() setup

/EC_workflow – full processing chain





Example of utilization of summary_QC()

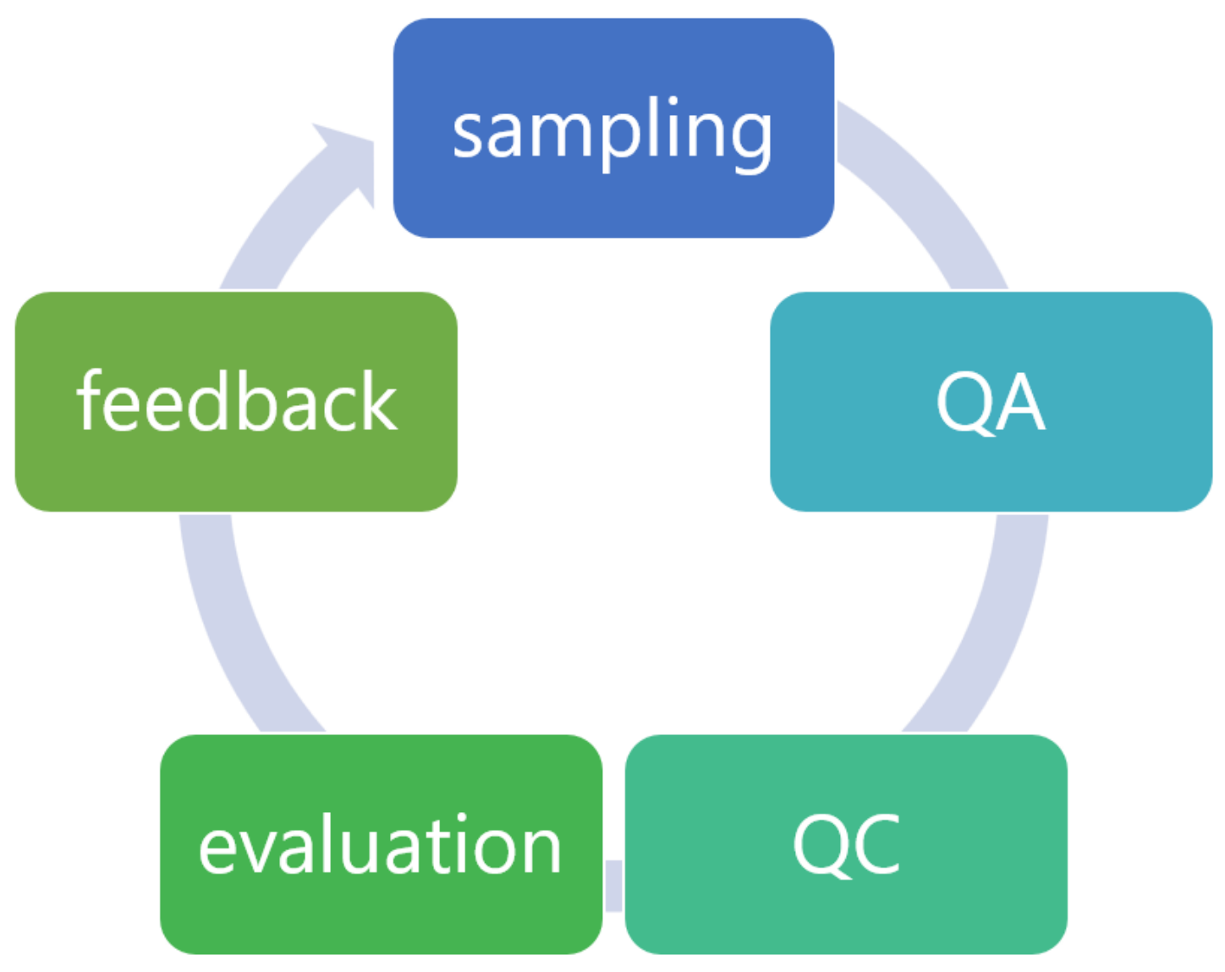
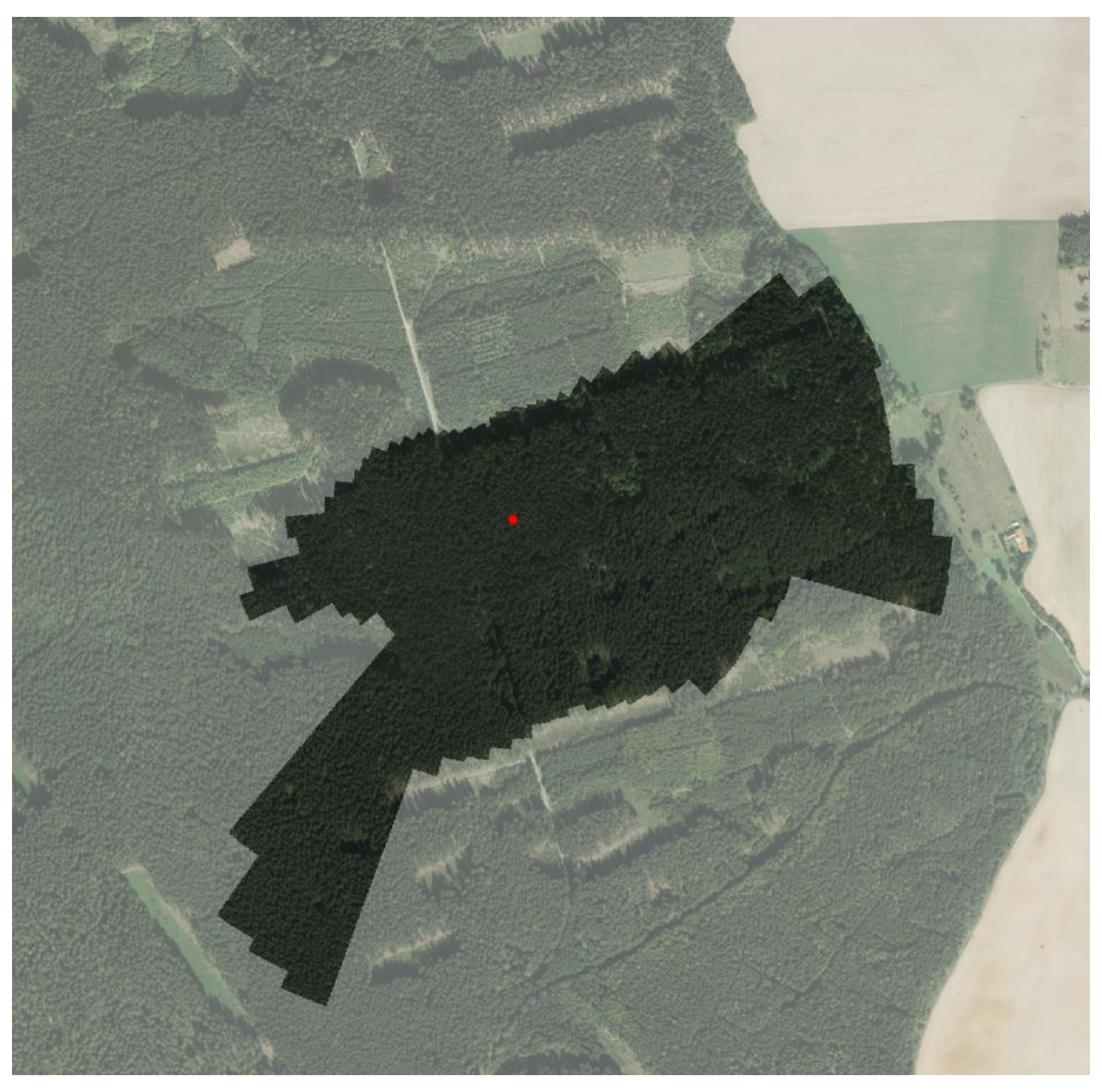


Diagram of quality assessment and quality control process



Visualization of fetch filter for site Rajec

Typical use-cases

Read and write data with units

read_eddy() – read tabular data with units.
read_EddyPro() – read EddyPro files.
write_eddy() – write tabular data with units.
ex() – extract without loss of units.

Merging data with timestamp

strptime_eddy() – convert and validate timestamp.
merge_eddy(list(df1, df2)) – merge multiple data frames along timestamp.

Gap-filling regular timestamp

strptime_eddy(..., allow_gaps = TRUE) – relaxed validation.
merge_eddy(list(df1)) – timestamp filled if regular.

Plotting of time series data

plot_hh(), plot_precheck() – single variable overview.
plot_eddy() – monthly and weekly plots of fluxes with their QC and corresponding meteo data.

Quality control filters

apply_thr() – flag according to threshold.
flag_periods() – exclude time periods.
extract_QC() – get filters from EddyPro output.
interdep() – consider flux interdependency.
despikeLF() – low frequency data despiking.
fetch_filter() – utilize 1D footprint modelling.
check_manually() – visual inspection and flagging.

QC evaluation

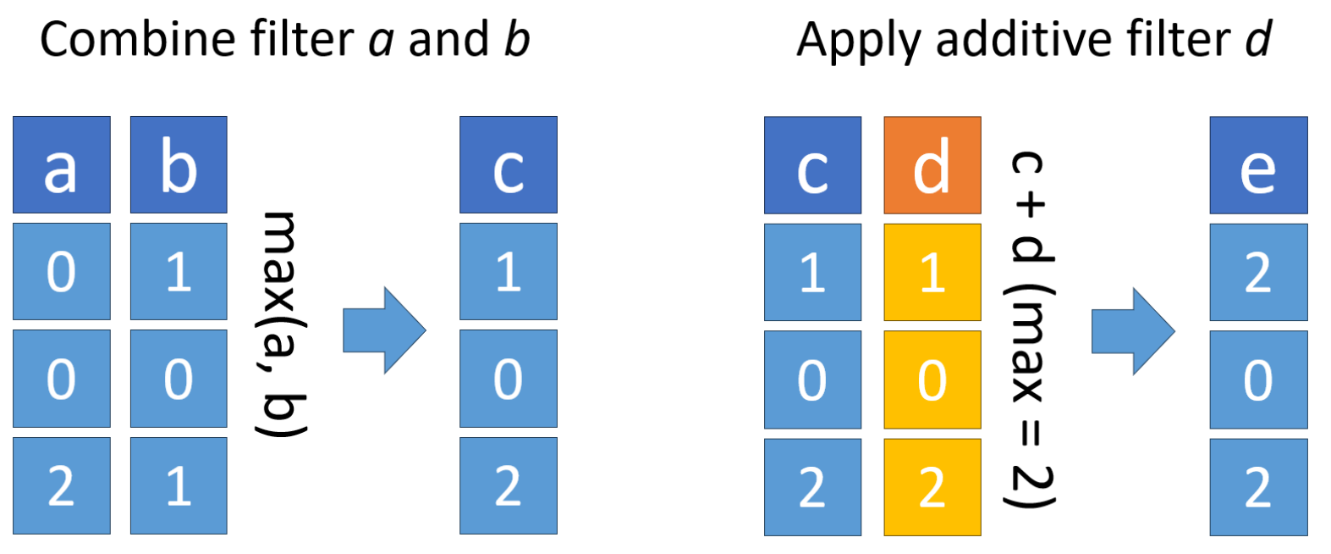
combn_QC() – combine regular and additive QC flags.
apply_QC() – remove fluxes based on flags.
summary_QC() – flag statistics for filters (table/plot).

Other plotting

ggplot_stats() – scatterplot with additional statistics.
barplot_agg() – show aggregated data in set intervals.

Aggregation and uncertainty evaluation

agg_mean(), agg_sum() – aggregate data across set period.
agg_fsd(), agg_DT_SD() – uncertainty of filled (partitioned) data.



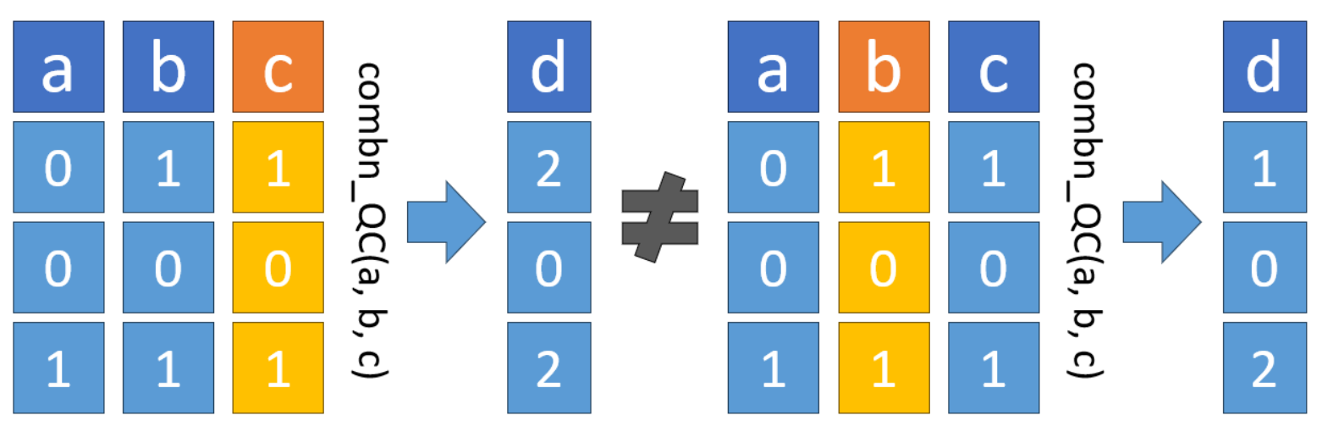
Combine filter *a* and *b*

a	b	c
0	1	1
0	0	0
2	1	2

Apply additive filter *d*

c	d	e
1	1	2
0	0	0
2	2	2

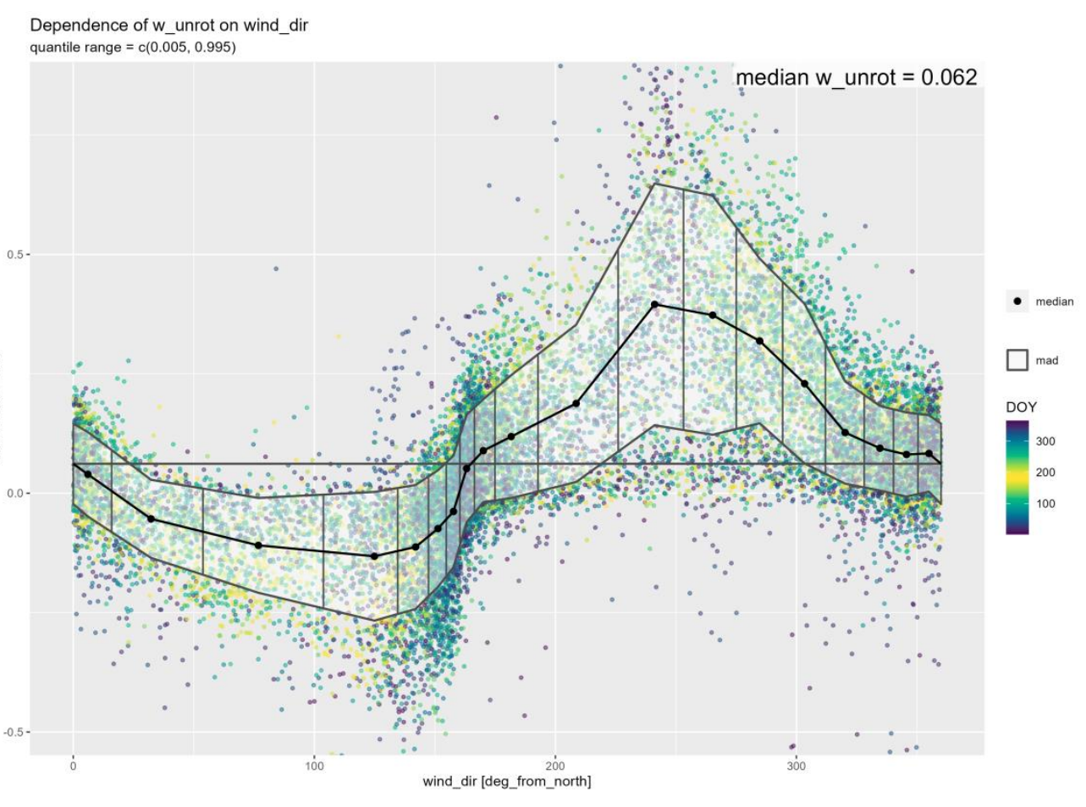
Introduction of additive filter makes filtering order-dependent



a	b	c	d
0	1	1	2
0	0	0	0
1	1	1	2

≠

a	b	c	d
0	1	1	1
0	0	0	0
1	1	1	2



Example of utilization of ggplot_stats()

Acknowledgement: This work was supported by the Ministry of Education, Youth and Sports of CR within the CzeCOS program, grant number LM2023048.