2nd Belle II Academy, 1–5 Aug 2022
University of Bonn

Improving your workflow by organizing your codes

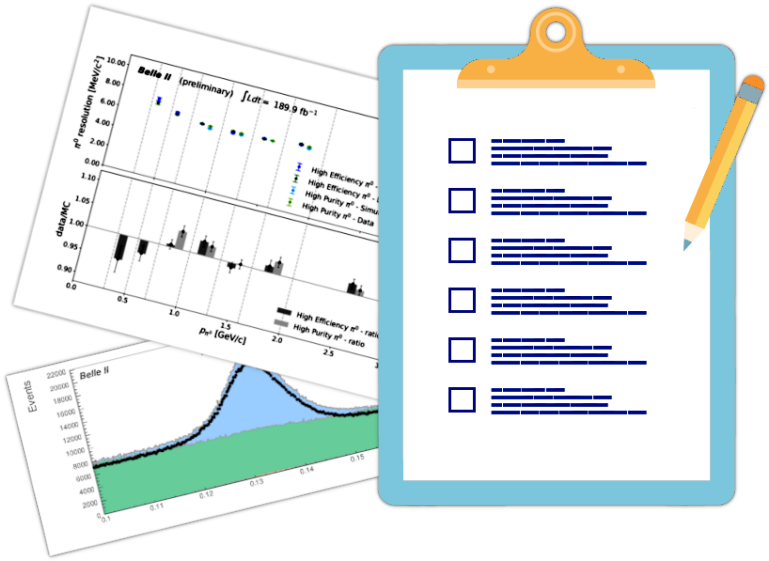
Zuzana Gruberová

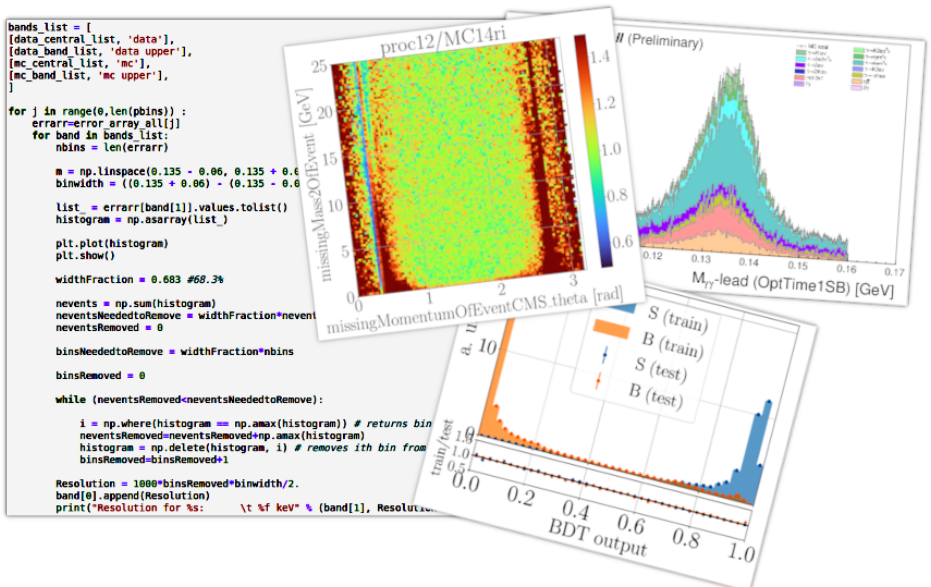
The goal

Write a code for a data analysis

» **The analysis**

- MC simulation and data (input)
- prepare the sample (signal definition, background suppression, cuts, MVA)
- do some checks (variable distributions, data/MC agreement)
- perform the measurement (fitting, signal yield, efficiency, purity)
- get the results (numbers, plots, limits)





» **The code**

- load the input (switch between samples)
- change the signal selection (adjust cuts, MVA)
- view various checks (distributions, ROC curve, FOM)
- do the computations (fitting, calculations)
- produce the results (numbers, plots, limits)

Tips and tricks

Simple but very very handy

» **Bad habits**

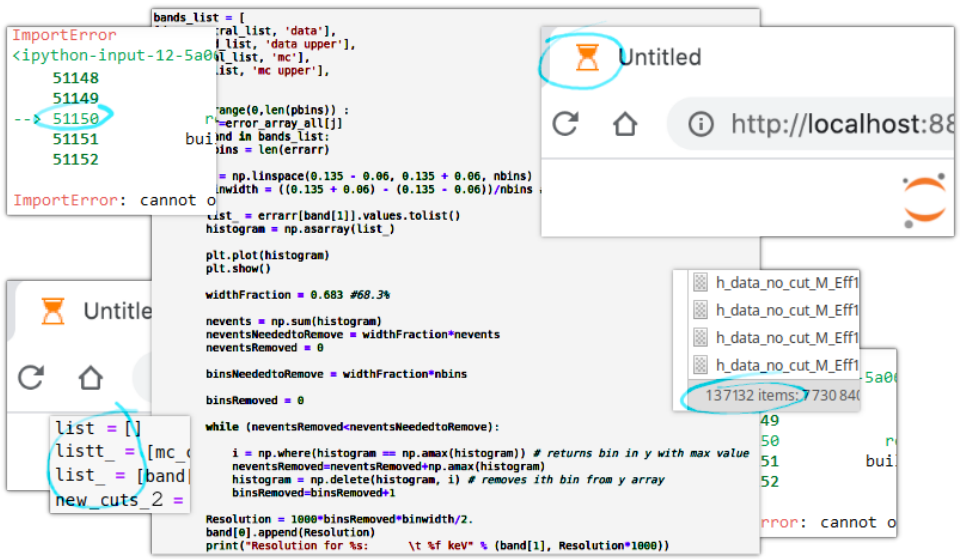
✗ one huge code for everything

✗ absence of comments and headlines

✗ ambiguous variable names

✗ one output folder for all plots and other files

✗ no intermediate steps saved



» **Organized approach**

✓ separate code files for different steps
e.g. function definitions, applying corrections, applying cuts, fitting, final plots

✓ inline comments, headlines separating different steps/checks
in this case more is better than less

✓ clear variable names
e.g. BDT_variables_list,
pi_sample_MC_hist_LIDcorrection_no_cuts

✓ organized output folders
e.g. data/MC distributions, histograms, CSV files, final plots

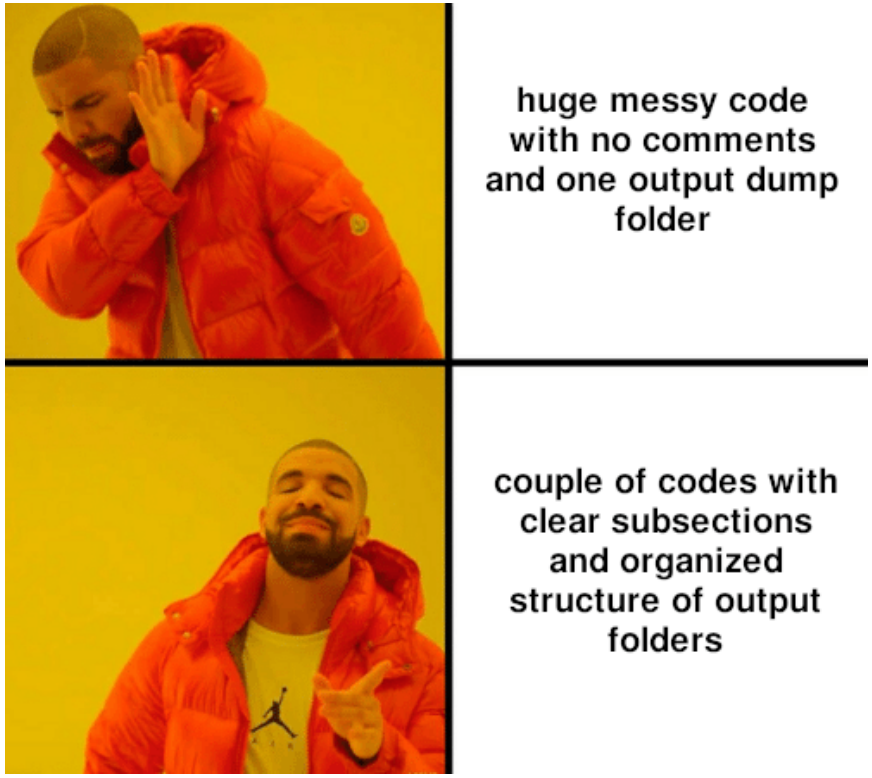
✓ save intermediate steps
e.g. save histograms after applying cuts to use in the fitting code later

Conclusion

» **Keeping your code and folders organized pays off!**
- saves time
- prevents unnecessary moments of despair
- makes it easier to share the code with others

» **There is no "one and only correct way"**
- try what works for you...
- ...but do not be afraid to adopt others' habits and tricks

No need to learn all thing the hard way



Good luck with your codes!