

Analysis Tutorials from the ALICE perspective: experience and review of the material

Dario Berzano · Matteo Concas

ROOT

Data Analysis Framework

<https://root.cern>



ALICE: **since 2014**, with focus on physics analysis

- Mostly students with previous experience, some veteran members
 - we expect/require "**academic**" C++, commonly included in most study plans
 - no ROOT nor Python notions
- **Thematic lectures** on specific analysis topics 3 times a year
 - 2h long frontal lectures. 1 teacher, ~100 students (40 live + 60 remotely attending)
 - We record the webcasts
- **Starterkit** with LHCb and SHiP once a year, since 2017, at CERN
 - 5 days of interactive lectures, 80 students in 3 rooms each with 1 teacher and 3 helpers
 - 3 days of Bash, Git, Python (advanced and basic) + 2 experiment-specific days



- ▶ We piggyback on **existing ROOT documentation** for teaching ROOT itself
 - [The ROOT Primer](#) (a bit old but still does its job)
- ▶ Single entry point for all documentation on Gitbook
 - **ALICE-specific:** alice-doc.github.io/alice-analysis-tutorial/
 - **Shared** with LHCb/SHiP (HSF): hsf-training.github.io/analysis-essentials/
 - Contribute via PRs (Markdown on GitHub): [ALICE](#), [HSF](#)
 - Examples of ROOT-relevant topics from our doc:
 - How to run an ALICE analysis from the grounds up (has many ROOT notions, such as streamers)
 - How to migrate your analysis from ROOT 5 to ROOT 6
 - How to debug C++ and ROOT code



Tools & Environment



We use a mixture of online notebooks and “local” software

- Online notebooks simply use SWAN
- Software runs locally (*i.e.* works on a plane too) with [alidock](#)
 - Just type [alidock](#): it seamlessly takes you inside a containerised env
 - It takes care of the Docker quirks transparently on Linux, macOS, Win:
 - ◆ X forwarding
 - ◆ multiple shells in the same container
 - ◆ persistent directory for user data
 - Simple to install/run, it autoupdates: <https://github.com/alidock/alidock/>
 - Very popular in ALICE: **zero time wasted** on tutorials, prebuilt sw fetched