PROCEEDINGS OF SCIENCE

Automation of the b-tagging calibration software at the ATLAS experiment

Marawan Barakat [a] on behalf of the ATLAS Collaboration

[a] DESY
Platanenalle 6, Zeuthen Germany
E-mail: marawan.barakat@desy.de

Particle cascades originating from quarks and gluons decays (jets) are omnipresent in proton-proton collisions at the LHC. The identification of jet flavours is essential for many physics searches at the ATLAS experiment. This is achieved using machine learning algorithms (taggers) which combine tracks and jets information to determine the flavour of the jets (b-jets, c-jets and light jets). These taggers are trained with simulated Monte Carlo events and, due to simulations imperfections, their performance need to be measured in data in order to extract correction factors for the simulation predictions. ATLAS developed a set of calibration techniques for different jets flavours to correct, then the correction factors need to be re-derived every time a new tagger is deployed. While reproducing the calibration results is a complex task that requires some expertise, automating the calibration workflow significantly accelerates the calibration cycle and makes it less prone to manual mistakes by offering a straightforward solution for results reproducibility. We present the first automated calibration framework in ATLAS using REANA platform. The results are compared with the official results using 36.2 fb^{-1} of 13 TeV collisions data from ATLAS, and a new set of calibration results with a customised setup is also included. The same method can be applied in other contexts to reduce the amount of time and resources needed to achieve the scientific goals.

The European Physical Society Conference on High Energy Physics (EPS-HEP2023)
21-25 August 2023
Hamburg, Germany

1 b-tagging calibration

The multi-jet production is the most dominant process at the in pp collisions LHC [1]. Heavy-flavour quarks are a characteristic signature of multiple interesting processes, ranging from the observation of the Higgs boson decay into b-quarks $H \rightarrow b\bar{b}$ to many Standard Model (SM) precision measurements and beyond the Standard Model (BSM) searches involving b and t quarks in the final state. Colour confinement restricts quarks from propagating freely, leading to their hadronization, then the decay products of these hadrons are reconstructed jets. Therefore, the identification of the of the jets coming from the hadronization of heavy-quarks has a major importance for the physics program of the ATLAS experiment.
Jets originating from b (c) quark decay are known as b-jets (c-jets), while quarks originating from u,d or s quarks decays are known as light-jets.

The performance of the algorithms is defined by the probability of correctly identifying a b-jet (b-jet tagging efficiency, ϵ_b) and the probability of mis-identifying a c-jet (ϵ_c) or a light jet (ϵ_l). b-tagging algorithms are trained using Monte Carlo simulation (MC). Imperfect description of the detector response and the physics modeling effects in these MC, causes a discrepancy between data and MC predictions, therefore ϵ_b is measured in data and correction factors or scale-factors (SF) are extracted depending on the jet p_{T} to correct the MC predictions, this process is referred to as b-tagging calibration [2]. Each of the ϵ_b, ϵ_c and ϵ_l has a dedicated calibration procedure and in this document we will focus on ϵ_b calibration.

The b-tagging calibration is performed by a code based on the ATLAS Athena software [3]. The first stage of the calibration code is to prepare the input file for the analysis based on ATLAS data and MC simulations. At this stage, a dedicated software package is used to produce *ntuples* (Events tabular format created by extracting relevant physics variables from raw detector data). It incorporates various techniques and algorithms to reconstruct and identify top quark events accurately. These techniques involve reconstructing the decay products of the top quarks, such as jets of particles and missing energy, and distinguishing them from other background processes. Afterwards, the final selection stage. This part of the code runs on the ntuples and apply the di-leptonic $t\bar{t}$ selection criteria and event categorization, then histograms are then filled with selected events. Finally, a likelihood fit is performed (mentioned in details in [2]) and ϵ_b and SFs are measured and plotted. The previous software machinery is intricate and requires knowledge and expertise to be used, which makes it challenging to non experts to reproduce calibration results. In this document the reproducibility crisis of scientific results is discussed, and the use of software automation is proposed as a solution to this crisis.

2 Reproducibility

The reproducibility of scientific research results is an major challenge in a wide range of scientific disciplines. According to a recent survey carried out by Nature [4], over 50% of 1,500 scientists across diverse scientific fields expressed their inability to replicate their own results.

Typically, the assurance of reproducibility is achieved through comprehensive documentation of how the research is conducted. In the case of data analysis research, this involves capturing complete information about the input data, parameters, analysis software, operating system environment, and the specific steps and methods used by the researcher to perform the analysis and obtain the original results.

Additionally, in particle physics experimental data is expensive to take, and in order to test new theoretical models in an efficient way, it is necessary to preserve information about the data, environment and the

different versions of the software in order to reuse already established analyses and datasets to test the new theories predictions and compare the results with data. Beside the search for new physics and precise measurements, the reconstruction of particles in collider experiment is a tedious task achieved by specific algorithms. These algorithms requires to be calibrated using dedicated analyses, therefore assuring the preservation and the reproducibility of the results is essential for the calibration task.

During the Run 2 of the LHC, the DL1r [5] algorithm served as the standard b-tagging tool within the ATLAS experiment. As the LHC enters Run 3, b-tagging algorithms will be deployed, necessitating calibration using the Run 3 data. Given the extensive duration typically required for such calibrations, the development of an automated calibration workflow would be advantageous, enabling more efficient validation and optimization of these new taggers.

3 Automation workflow

The automatiobn workflow (shown in Figure 1) is a series of automated steps that are performed to execute a computational task. It relies on four key components :

- Computing environments used to run the analysis code, e.g : Python version, ATLAS Software version, special libraries, etc...
- The code used to analyse the data including all the required scripts to run the algorithm, in the case of *b*-tagging, the algorithm has a source code in C++ executed by python scripts.
- Inputs to the code such as the running parameters and location of the data and simulation samples.
- Computational steps taken to achieve the results.

Having these components preserved, manual software manipulation steps can be avoided, while keeping track and documentation of all the intermediate steps to reproduce results and document the exact version used in the analysis.

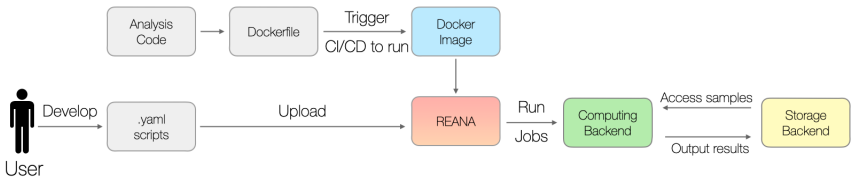

Figure 1: Scheme illustrating how REANA is executing an automation workflow.

The first automation tool is **Docker** [6], to preserve the environment used to run this code . Docker is an open-source platform that enables developers to create, deploy, and run applications inside software containers using a domain-specific language (DSL). Then, **Gitlab** [7] is used to host code at the European Organization for Nuclear Research (CERN), it is also bundled with a built in continuous integration (CI) / continuous development (CD) system. While docker containers can be run on local machines, in this document Gitlab CI/CD is used to automate the task of running the docker containers to keep the analysis environment up-to-date i.e each time that new commits are pushed. **REANA** [8] (standing for REproducible ANAlyses) : is a free and open-source platform developed by CERN designed to simplify

the management of scientific workflows for reproducible research. It provides an easy-to-use interface for creating, executing, and sharing computational workflows, enabling researchers to automate repetitive tasks and focus on data analysis.

Based on REANA and yadage language three scripts are created for the workflow : The first script **reana.yaml**, a top level script specifying the inputs to the code, simplifying the setup process and allowing for a more straightforward execution for the user. Next, the **workflow.yaml** file outlines each step of the process, also showing how these steps are linked e.g the Selection step runs in parallel on different samples, then the Fit step starts. Finally **steps.yaml**, in this file, First, the Command Line Interface (CLI) commands of each stage are specified. Second the link to the Docker images running on Gitlab is associated to each stage.

4 Validation

REANA is used to run the calibration code to provide results for the b-jet calibration and compare with the official ATLAS b-tagging SF. Good agreement is obtained from the automated workflow as shown in Figure 2. Additionally, the workflow is improved by adding more features allowing for a higher flexibility of the code. For instance easily testing new taggers, other working points(WPs), or use a coarser/finer jet p_{T} to derive scale factors are desired features, allows an easier use of the calibration code. To give an example, early Run 3 analyses in ATLAS require the calibration of the taggers with the early Run 3 dataset which is statistacally limited. Therefore having the previously mentioned features facilitates the adaptation of the calibration code setup to achieve the calibration in such special cases. In the Figure 3 results for the calibration of the DL1r with new working points different than the standard WPs defined by the flavour tagging group in ATLAS.

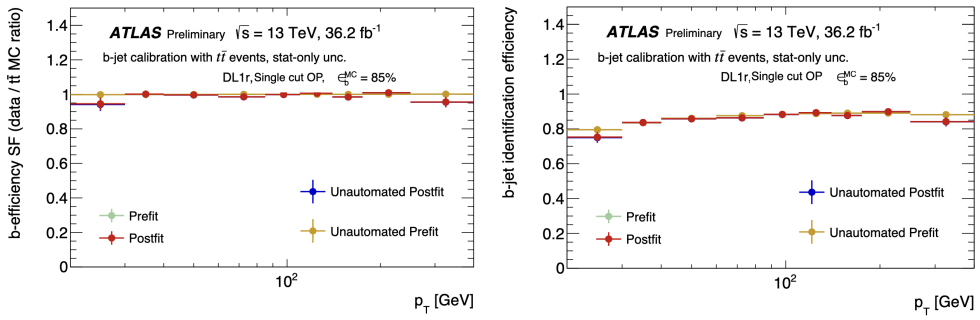

Figure 2: Comparison of the results from the automated and the unautomated code showing perfect agreement between the two results.

5 Conclusion

In this document, the automation of the analysis code for b-tagging calibration is shown. While reproducibility and time efficiency is a key point for modern research, automation presents a great potential to ensure the reproducibility of the results and allow researchers to use sophisticated codes without being experts of, keeping the focus on the inputs and outputs rather than the technical details and debugging. The

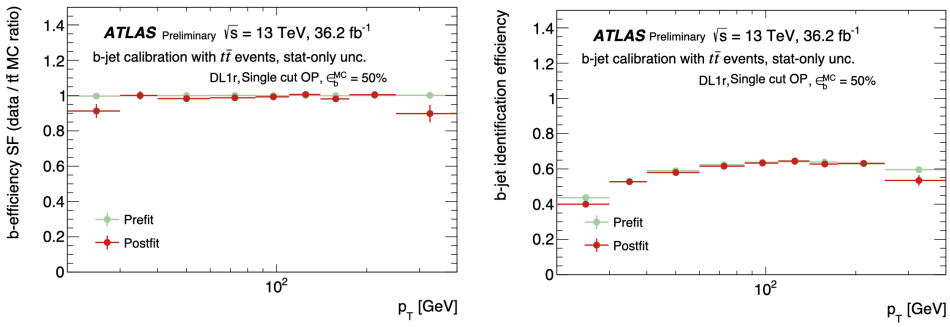

Figure 3: Results from the automation workflow with the new feature of calibrating the DL1r tagger with any custom WP.

automation workflow shown in this document is applicable to any other calibration analysis in ATLAS for better preservation and faster reinterpretation.

References

[1] *LHC Machine*, JINST **3** (2008) S08001, ed. by L. Evans and P. Bryant (cit. on p. 2).

[2] ATLAS Collaboration, *ATLAS b-jet identification performance and efficiency measurement with $t\bar{t}$ events in pp collisions at $\sqrt{s}$ = 13 TeV*, (2019), arXiv: 1907.05120 [hep-ex] (cit. on p. 2).

[3] ATLAS Collaboration, *Athena Software*, URL: %7Bhttps://atlas.cern/athena%7D (cit. on p. 2).

[4] M. Baker, *1,500 scientists lift the lid on reproducibility*, Nature News **533** (2016) 452, URL: https://www.nature.com/news/1-500-scientists-lift-the-lid-on-reproducibility-1.19970 (cit. on p. 2).

[5] ATLAS Collaboration, *ATLAS flavour-tagging algorithms for the LHC Run 2 pp collision dataset*, (2022), arXiv: 2211.16345 [physics.data-an] (cit. on p. 3).

[6] *Docker documentations*, https://www.docker.com/, Accessed: June 26, 2023 (cit. on p. 3).

[7] *CERN's GitLab*, https://gitlab.cern.ch/, Accessed: June 26, 2023 (cit. on p. 3).

[8] CERN IT, *REANA Platform*, https://reana.io/, Accessed: June 26, 2023 (cit. on p. 3).