LHCb-PUB-2016-021
September 8, 2016

The PIDCalib package

L. Anderlini[1], A. Contu[2], C.R. Jones[3], S. Malde[4], D. Müller[5], S. Ogilvy[6],
J.M. Otalora Goicochea[7], A. Pearce[5,8], I. Polyakov[9], W. Qian[4], B. Sciascia[6],
R. Vazquez Gomez[6], Y. Zhang[10]

[1] *Sezione INFN di Firenze, Firenze, Italy*
[2] *European Organization for Nuclear Research (CERN), Geneva, Switzerland*
[3] *Cavendish Laboratory, University of Cambridge, Cambridge, United Kingdom*
[4] *Department of Physics, University of Oxford, Oxford, United Kingdom*
[5] *School of Physics and Astronomy, University of Manchester, Manchester, United Kingdom*
[6] *Laboratori Nazionali dell'INFN di Frascati, Frascati, Italy*
[7] *Universidade Federal do Rio de Janeiro (UFRJ), Rio de Janeiro, Brazil*
[8] *STFC Rutherford Appleton Laboratory, Didcot, United Kingdom*
[9] *Institute of Theoretical and Experimental Physics (ITEP), Moscow, Russia*
[10] *LAL, Universit Paris-Sud, CNRS/IN2P3, Orsay, France*

Abstract

The PIDCalib package is a tool, widely used within the LHCb Collaboration, which provides access to the calibration samples of electrons, muons, pions, kaons and protons. This note covers both theoretical aspects related to the measurement of the efficiency of particle identification requirements, and more technical issues such as the selection of the calibration samples, the background subtraction procedure, and the storage of the data sets in the new data-processing scheme adopted by the LHCb experiment during the second run of the LHC.

Contents

1 Introduction 1

1.1 Purpose of the package and intent of the note . . . 1

1.2 A historical perspective . . . 2

1.3 Towards Run 2 . . . 3

2 Computing the PID efficiency 4

2.1 General procedure . . . 4

2.2 PID requirements in reference preselections . . . 6

2.3 Implementation in PIDCalib . . . 6

2.4 Reference sample operations . . . 7

2.4.1 “Classic” PIDCalib . . . 8

2.4.2 “All data” PIDCalib . . . 8

2.4.3 MCResampling tool . . . 8

2.5 Summary . . . 9

3 Uncertainties estimation 9

3.1 Statistical uncertainties . . . 10

3.1.1 Calibration sample size . . . 10

3.1.2 Reference sample size . . . 11

3.2 Systematic uncertainties . . . 11

3.2.1 Differences between reference and signal sample . . . 11

3.2.2 Choice of binning . . . 12

3.2.3 ${}_s\mathcal{W}$eight in calibration sample . . . 13

4 Processing the calibration samples 13

4.1 Selection strategy . . . 14

4.2 Workflow in Run 2 . . . 14

4.3 Fit procedure and sWeight computation . . . 15

5 Conclusions 16

References 18

1 Introduction

1.1 Purpose of the package and intent of the note

Particle identification (PID) is the area of the LHCb experiment, including detectors, reconstruction software, and data analysis, developed with the purpose of distinguishing charged final state particles - including pions, kaons, protons, muons and electrons - from one another.

The sub-detectors used for particle identification are the two Ring Imaging CHerenkov (RICH) detectors, the muon system, and the calorimeter system composed of a Scintillating Pad Detector (SPD), a preshower, a shashlik-type electromagnetic calorimeter and a hadronic calorimeter. A discussion on the technical aspects of the detectors can be found in Ref. [1]; the performance of the RICH, Calorimeter, and Muon systems are described in Refs. [2], [3], and [4], respectively.

The reconstruction software is the set of algorithms transforming the electronic response of the detectors into raw variables that are combined in several ways to provide discriminants used in physics analyses. During the LHC Run 1 the use of PID discriminants in the trigger selection was advantageous to reduce the retention rate for some channels. In Run 2 the use of PID discriminants in the trigger selection has become much more common. The LHCb trigger system consists of two levels: a first level implemented in hardware followed by a software trigger (HLT). In Run 1 the HLT ran a simplified version of the offline event reconstruction to accommodate the more stringent CPU time requirements. For Run 2, thanks to both the additional computing resources and a re-optimisation of the code, the full event reconstruction can now be utilised [5].

The simulation of the detectors devoted to PID is non-trivial. Indeed, computing the response of these detectors to a traversing particle requires modelling of the kinematics of the particle, the occupancy of the detectors (which may be different from event to event and sensitive to beam conditions), and the experimental conditions such as alignments, temperature, and gas pressure (which may modify the response of detectors from run to run). These considerations have motivated the use of data-driven techniques to measure the efficiency of selections involving particle identification.

The PIDCalib package, standing for *Particle IDentification Calibration*, is a set of tools to help analysts compute the efficiency of particle identification selection requirements. The package has evolved greatly since its first version and remains under active development. At the beginning of Run 2 it is being developed into a more robust package, better exploiting the computing technologies made available by the LHCb collaboration. There are two main reasons behind this change:

1. Particle identification is used extensively in the selections defined in the second stage of the high-level trigger (HLT2). Discriminating variables computed online (in the trigger) may be different to those computed offline (after the trigger). The latter are subject to modifications and improvements by the introduction of new algorithms, while the former are computed once in the trigger and cannot be changed in future reprocessings.

2. The PID calibration sample size is larger than it was during the Run 1 of the LHC. The systematic uncertainties on physics analyses due to the limited size of the calibration samples were dominant in some fringe regions of the PID efficiency parameter space, but the larger sample size expected for Run 2 will reduce the size of this contribution and others. This imposes requirements on the data management and processing strategy of PIDCalib, to ensure fast processing and flexibility to meet the ever-changing needs of the analysts.

This note describes the current status of PIDCalib at the beginning of the second run of the LHC. The note is organized as follows. The next two sections summarize the evolution of the PIDCalib package within the collaboration from a historical perspective (1.2) and the main changes needed in passing from Run 1 to Run 2 (1.3). Section 2 reviews the theoretical aspects of the efficiency determination for a reference sample through calibration samples, while Section 3 is devoted to the discussion of the statistical and systematic uncertainties on the efficiencies predicted within PIDCalib. Section 4 summarizes the technical issues related to the data-processing, selection and signal extraction of the calibration samples. A summary and outlook conclude the note in Section 5.

1.2 A historical perspective

The first version of the PIDCalib package was developed by the Oxford group in 2009. At that time it was comprised of a loose set of scripts to exploit the $D^{*+} \to D^0\pi^+$ decay to tag the flavour of the D^0 meson unambiguously, and therefore distinguish the kaon and the pion in the subsequent decay $D^0 \to K^-\pi^+$ without the use of PID information. The distributions of the PID variables of these PID-unbiased samples were used as input for the analyses studying *CP* violation in the decay $B^\pm \to D^0K^\pm$ in order to measure the CKM angle γ [6–8]. The reconstruction and selection efficiency of kaons and pions, and any possible charge asymmetry, had to be known very precisely in order not to bias the measurement of the charge asymmetry of the decay. The $_s\mathcal{P}$lot technique [9] was used to statistically subtract the background contribution from the D^{*+} sample, which was then weighted to reproduce the kinematic variables of pions and kaons in the $B^\pm$ sample.

Later in 2010 the package was extended to include calibration samples for protons. One candidate decay mode was $\mathit{\Lambda} \to p\pi^-$, principally because of the abundance of $\mathit{\Lambda}$ baryons produced at the LHC, but also because of the good background discrimination provided by the long $\mathit{\Lambda}$ lifetime, which results in its decay vertex being significantly displaced from the primary pp interaction vertex.

At the same time, in Frascati, an alternative technique was used to determine the efficiency of PID requirements [10] in a search for the rare decay $B^0_s \to \mu^-\mu^+$. The procedure relied on the technique whereby one of the J/ψ child candidates is required to pass strict muon PID requirements, allowing the other muon to be selected without the use of PID information (thus being ‘PID-unbiased’) [4]. In 2011, this calibration of muon PID selection efficiencies was first integrated into the PIDCalib package, again using the $_s\mathcal{P}$lot technique to subtract the background contribution from the calibration sample.

Close attention was paid to the possible bias of the online selection (trigger) to the PID variables of the muon candidates.

In 2014, the calibration of PID requirements on electrons was made available in PIDCalib, using a comparatively small sample of $B^+ \to J/\psi\,(\to e^+e^-)K^+$ decays. Electrons were the last particle type to be added to PIDCalib, making it a comprehensive package, able to cope with the needs of most physics analyses.

During Run 1, several analyses requiring proton PID calibration noted that the $\mathit{\Lambda}$ calibration sample suffered from a low sample size for high-momentum protons. Although many $\mathit{\Lambda}$ decays are collected, the softer spectrum with respect to protons from heavy flavour decays led to large statistical uncertainties on the calibration of high-momentum protons. Two strategies were put in place during Run 1: firstly a set of two $\mathit{\Lambda}$ selections were made, one which was artificially scaled down to select low-momentum protons at an acceptable rate, and another which required protons with a momentum of a least 40 GeV/c; and secondly the addition of calibration samples made from protons from charm and beauty decays was considered, such as $\Lambda_b^0 \to \Lambda_c^+\pi^-$, followed by $\Lambda_c^+ \to pK^-\pi^+$. Decays of $\Sigma_c^0 \to \Lambda_c^+\pi^-$ and $\Sigma_c^{++} \to \Lambda_c^+\pi^+$ were also studied, and were included in the calibration samples in 2015.

1.3 Towards Run 2

Before the start of Run 2, it was already clear that for some decay channels it would be necessary to include some PID requirements in the online (trigger) selections. In contrast to Run 1, the online reconstruction was expected to be identical to the offline one, allowing analysts to perform their measurements directly on the output of the online reconstruction. Online reconstruction objects can then be saved directly into a specific stream of data, called Turbo, which is made available to analysts through a specially developed application [11]. This new trigger scheme, along with the need to evaluate the performance of PID variables both online and offline, required a revaluation of the data flow of the PID calibration samples [12]. The new data flow simplifies the management of the larger samples that are expected in Run 2, and allows for more detailed studies of the systematic uncertainties associated with PID calibration.

In between Run 1 and Run 2, several other calibration modes were studied in order to provide a cross-check of the correctness of the package, to assess the systematic uncertainties related to the choice of the calibration sample, and to cover regions of the kinematic phase space where the primary samples are limited in statistics. Details of the suite of samples now selected in the trigger and made available through the PIDCalib package, are available in Ref. [13].

The recent update of the PIDCalib package aims to offer tools to help develop and test new algorithms on real data. This is crucial in a data flow that, in order to save disk space, tends to reduce the detector information available to analysts. Storing all available information for only a few calibration modes allows the continuation of the work of improving the reconstruction and PID, fundamental in studies for the LHCb upgrade, while keeping disk space usage to a minimum.

2 Computing the PID efficiency

To compute any efficiency it is necessary to know the number of signal events before and after the selection. For a majority of analyses some PID selection is applied at a point where it cannot be changed, such as in the trigger or in the centralized selection, and the number of signal events before the selection is lost. It may be the case that the number of signal events before the selection is inaccessible; either due to a high level of background or that no events are observed after the selection, as in the case of a search for a rare decay The use of Monte Carlo simulation can only give an approximate evaluation of the PID efficiency since the PID variables used in the selection are not necessarily well described. To obtain the required accuracy it is necessary to use samples taken from data, in conjunction with calibration techniques. Using these, a proxy for the signal sample (the *calibration sample*) is obtained without the use of PID information, then arbitrary PID requirements may be applied comparing the number of candidates before and after the selection.

A complication arises from the non-uniform PID response for particles of a given species. The efficiency is known to vary as a function of several variables, the most important among which are the particle momentum, the particle pseudorapidity, and the number of tracks in the event. If the distribution of these variables is different between the calibration sample and those of the signal under scrutiny, the average PID efficiencies will differ.

This section discusses how these challenges can be overcome, presenting first the theory of the technique, and then covering the implementation. Note that the discussion of the computation of the associated uncertainties is postponed until Section 3.

2.1 General procedure

We assume that a data sample is available, with an infinite number of the tracks of known particle type, obtained without the use of any particle identification information, which are denoted *calibration tracks*. We further assume that the response of a PID variable is fully parameterised by some known set of variables, such as the track momentum p or the track multiplicity (such as the number of reconstructed tracks traversing the whole detector or the number of hits in the SPD detector). By partitioning the sample with sufficient granularity in the parameterising variables, the probability distribution function (PDF) of the PID variable does not vary significantly within each subset, such that the efficiency of a selection requirement on that variable is single valued within each subset.

Because the PID efficiency is fully parameterised, we can deduce the efficiency of a PID requirement on *any* track by looking up the efficiency of the subset that the track would belong to, regardless of whether it originates from the set of calibration tracks or not. In the ith subset, the efficiency ε_i of a PID cut is

$$\varepsilon_i = \frac{N_i'}{N_i} \,, \tag{1}$$

where N_i is the number of tracks in the subset before the PID cut, and N_i' is the number after. The average efficiency, across the entire space of the parameterising variables, is then

$$\bar{\varepsilon} = \frac{\sum_i \varepsilon_i N_i}{\sum_i N_i} \ . \tag{2}$$

In the trivial case that the events are from the calibration sample there is no need to compute per-subset efficiencies and the average efficiency is simply

$$\bar{\varepsilon} = \frac{N'}{N} \ , \tag{3}$$

which one obtains when substituting Equation 1 into Equation 2.

To compute the PID efficiency on a sample other than the calibration sample, denoted hereafter as the *reference sample*, the parameterising variables in the calibration sample can be weighted to look like those in the reference. The PID efficiency can then be computed as above using the per-subset weights. The weights are defined as the normalised ratio of reference to calibration tracks

$$w_i = \frac{R_i}{C_i} \times \frac{C}{R} \ , \tag{4}$$

where R_i (C_i) is the number of reference (calibration) tracks in the ith subset, and R (C) is the total number of reference (calibration) tracks in the sample.

After applying the PID cut to the weighted calibration sample, the per-subset efficiency of which is computed using Equation 1, the average efficiency of the PID requirement on the weighted calibration sample is then

$$\bar{\varepsilon} = \frac{\sum_i \varepsilon_i w_i C_i}{\sum_i w_i C_i} \ . \tag{5}$$

Because the calibration sample has been weighted in the parameterising variables to match the reference sample, and because the PID efficiency is assumed to be *fully* parameterised, $\bar{\varepsilon}$ is also the average efficiency of the PID requirement on the reference sample.

By substituting the weights defined in Equation 4 into Equation 5, we obtain

$$\bar{\varepsilon} = \frac{\sum_i R_i \varepsilon_i}{\sum_i R_i} = \frac{1}{R} \sum_i R_i \varepsilon_i \ . \tag{6}$$

This shows that the computation of the PID efficiency can be thought of in two ways: either as the reweighting of the calibration sample to match the reference, as in Equation 5, or as the assignment of efficiencies to reference tracks based on the subset they belong to, as in Equation 6.

This can be extended to reference samples where PID requirements have been imposed on multiple tracks. One then requires the efficiency of an ensemble of cuts. The individual efficiency ε_i for each track is calculated, and the efficiency for the ensemble $\varepsilon_{\text{ensemble}}$ is the product of the N_{trk} individual efficiencies

$$\varepsilon_{\text{ensemble}} = \prod_i^{N_{trk}} \varepsilon_i \ . \tag{7}$$

2.2 PID requirements in reference preselections

Obtaining a reference sample from real data, rather than Monte Carlo simulations, would be ideal because it avoids the harmful effects of possible modelling defects in the parameterising variables. However, sometimes data for the reference sample is only available after PID requirements have been made, such as in a trigger selection.

We can still try to apply the method defined in Section 2.1, weighting the calibration sample to match the reference sample, after whatever PID-based preselection has been applied to it

$$w'_i = \frac{R'_i}{C'_i} \times \frac{C'}{R'},$$

where R'_i (C'_i) is the number of reference (calibration) tracks in the ith subset and R' (C') is the total number of reference (calibration) tracks in the sample, all quantities being *after* the PID requirements in the preselection have been applied. In principle, it is not guaranteed that the new average efficiency

$$\overline{\varepsilon}' = \frac{\sum \varepsilon_i w'_i C_i}{\sum w'_i C_i} \ , \tag{8}$$

is still the same average PID efficiency as defined in Equation 5. However, we note that the ratio between the two weights is constant across all subsets

$$\frac{w_i}{w'_i} = \frac{R_i}{C_i}\frac{C'_i}{R'_i} \times \overbrace{\frac{R}{C}\frac{C'}{R'}}^{A} = \frac{\varepsilon_i^C}{\varepsilon_i^R} A = A,$$

where ε_i^R and ε_i^C are as defined in Equation 1 for the reference and calibration samples respectively. The final equality then follows from the PID efficiency in a subset being source-independent. Therefore

$$w'_i = w_i/A,$$

and so it follows that equations 5 and 8 are equivalent

$$\overline{\varepsilon}' = \frac{\sum w'_i \varepsilon_i C_i}{\sum w'_i C_i} = \frac{A^{-1} \sum w_i C_i \varepsilon_i}{A^{-1} \sum w_i C_i} = \overline{\varepsilon} \ . \tag{9}$$

Therefore, we can use a reference sample with PID cuts applied to calculate the efficiency of those PID cuts, assuming the cuts do not completely remove all events from any phase space subset. Notice that this argument does not hold when the selection is applied on the poorly modelled simulation distributions.

2.3 Implementation in PIDCalib

In practice, the sample of calibration tracks is not pure. Kaon and pion tracks, for example, are obtained from selecting $D^0 \to K^-\pi^+$ decays, but the sample also contains combinatorial

D^0 candidates (vertices formed from random tracks that happen to pass the selection criteria). This means we cannot just count the number of candidates before and after the PID selection. Instead, we could perform maximum likelihood fits, but performing one fit in every subset would be slow and requires a lot of manual checking.

We instead compute ${}_s\mathcal{W}$eights from maximum likelihood fits to the full sample without any PID selection applied, as described in Section 4.2. These can then be used to compute the number of signal and background candidates in each subset, on a statistical basis, before and after the PID selection is applied. The number of signal candidates in the ith subset, as in Equation 4, is then defined as

$$C_i = \sum_{\text{Candidates}} {}_s\mathcal{W}_i \ , \tag{10}$$

where ${}_s\mathcal{W}_i$ is the signal ${}_s\mathcal{W}$eight for candidate i.

The use of ${}_s\mathcal{W}$eights relies on the assumptions of the ${}_s\mathcal{P}$lot formalism [9], namely that the parameterising variables are uncorrelated to the fit variables (usually the mass of the vertex). The possible bias introduced by the correlation between the two kinds of variables, e.g. the dependency of the resolution of the mass peak on the momentum and pseudorapidity of the probed track, might be relevant and should be considered case by case as a source of systematic uncertainty (see also Sect. 3.2.3).

Another source of systematic uncertainty is the presence of only few events in one or more the chosen subsets. There it may happen that the sum of ${}_s\mathcal{W}$eights before and/or after the PID selection is negative. The partitioning should be chosen to avoid this situation which is often hinted by unphysical efficiencies, lying outside the $0 \le \varepsilon \le 1$ interval. The partitioning in PIDCalib is implemented as the binning schema of a one-, two-, or three-dimensional histogram. ‘Partitions’ then become ‘bin boundaries’, and ‘subsets’ become ‘bins’. The use of histograms is just an implementation detail; they act as lookup-tables when assigning efficiencies to the reference sample, as in Equation 6.

For the analyst, the main use-case for PIDCalib is generating these objects, known as *performance histograms*. What they are used for is largely dependent on the analysis, but PIDCalib can provide a table of per-event efficiencies for the reference sample, as a ROOT `TTree` [14], and can also compute the average efficiency of the user-specified PID selection on the reference sample.

2.4 Reference sample operations

There are a number of ways in which the calibration samples can be used to extract PID efficiencies, and PIDCalib is able to perform each of these in intuitive ways. Three broad strategies have been commonly implemented at LHCb in the past and a brief description of the implementation of these methods is given in the following sections. Depending on a number of circumstances, including the analysis selection, the modelling of the signal kinematics and the ability to cleanly extract the real data kinematics, the users may find that one particular method is optimal for them in extracting a PID selection efficiency.

2.4.1 "Classic" PIDCalib

The original approach uses a simulated reference sample to provide the kinematics of the signal tracks under consideration. The simulated sample should have no PID requirements asserted (the argument outlined in Section 2.2 does not hold when the selection is applied on the poorly modelled simulation distributions). PIDCalib will then provide the user with a ROOT `TTree` of per-event efficiencies for these simulated events. The average efficiency for the given PID requirements on that signal sample is then simply the average of the per-event PID efficiencies. This is an ideal approach to use when the kinematics of the signal tracks are understood to be well modelled.

2.4.2 "All data" PIDCalib

There are some cases where the signal kinematics are known not to be well modelled, and a re-weighting of these kinematics is not trivial. If the signal in the real data can be reliably separated from the other species in the sample, such that some background subtraction can be used to extract the signal kinematics, another approach to the PIDCalib reference sample can be used. The real data in the analysis can be used to provide the signal kinematics instead of a simulated sample. This holds even where PID selection has been used in the data in the trigger selections or in the centralized pre-selection, and the distributions of the data before PID selection is not accessible, as per Section 2.2.

The technique uses the sum of signal candidates surviving PID selection in local regions of the phase space, and the efficiency of the selection on that region of phase space taken from the calibration sample, to extract the number of signal candidates in that region of the phase space before selection. Say the $_s\mathcal{P}$lot technique is used to extract the kinematics of the signal candidates - the per-event efficiencies given to the user by PIDCalib can then be used in the following expression to extract an average efficiency of the PID requirements on the signal:

$$\frac{1}{\overline{\varepsilon}} = \frac{\sum w_i/\varepsilon_i}{\sum w_i} \,, \tag{11}$$

where the sum is over the signal candidates, ϵ_i is the ensemble PID per-event efficiency for that candidate and w_i is the $_s\mathcal{W}$eight for the signal candidate in the reference sample.

An example of an analysis which has used this procedure in a published result is the 13 TeV charm cross-section analysis, produced as part of the 2015 early measurements programme [15].

2.4.3 MCResampling tool

Another approach in PIDCalib is the explicit re-weighting of a MC PID distribution, which requires no information from the signal data, but simply a simulated signal sample with no PID selection applied. Using a sub set of the variables η, p, p_T, and the track multiplicity in the event, the calibration sample is split into regions, and in each region a distribution of the PID variable in question is constructed. Using the same sub set of variables, each track in the signal simulation is matched to one of the regions defined for the calibration sample. The distribution of the PID variable from the matched region is then used as a PDF to randomly draw a new PID value. In this way, the efficiency of the selection can then be extracted from the signal simulation itself. This procedure has the effect of improving the agreement of the PID distributions between simulation and data. However, it only preserves the correlations of the sampled variables (up to three in the present implementation in PIDCalib) and breaks any others, the effects of which have to be carefully evaluated on a case-by-case basis.

2.5 Summary

PIDCalib can be used to produce performance histograms and tables of efficiencies for a user-specified set of PID requirements. These can be parameterised by up to three variables, chosen among track momentum, transverse momentum, pseudorapidity, and track multiplicity. If a reference sample is provided, PIDCalib can also produce tables of per-event efficiencies and the average efficiency over the sample.

The choice of binning for the performance histograms is important. If it is too coarse, the efficiency will not be constant within a bin, and so the hypothesis of full parametrization of the efficiency will not be satisfied and Equation 5 will not hold. Conversely, using many variables or too fine a binning will lead to large statistical uncertainties on the computed efficiencies. A compromise must be found between satisfying the assumptions of the theory and decreasing the statistical uncertainty on the efficiencies, and one must consider the associated systematic effects. Similar arguments hold for the choice of parameterising variables.

The reference sample used to compute the PID efficiency for a signal sample can be from real data or from simulation, and in the former case it may already have PID cuts applied to it. The important points are that the distributions in the parameterising variables in the reference sample must match those of the signal, and that any PID cuts that are applied to the reference sample must not have already removed the entire contents of a kinematic bin of the calibration sample.

3 Uncertainties estimation

Section 2 presented the calibration procedure used to estimate the efficiency $\bar{\varepsilon}$ of the reference sample. In the following section, different uncertainties affecting this procedure are discussed, giving rise to an uncertainty on the quoted efficiency. Although any uncertainty on the efficiency of a particle identification requirement is most likely treated as a systematic uncertainty in the respective analysis using the PIDCalib procedure, the discussion is split into statistical and systematic uncertainties on $\bar{\varepsilon}$. For most of the uncertainties presented here, no scripts exist in PIDCalib to compute their effect and hence the discussion focuses on describing their source and possible treatments. The actual treatment of systematic uncertainties on PID efficiencies may vary depending on the actual analysis as potential correlations to other systematic uncertainties may arise.

Furthermore, detailed studies prepared individually for the specific analyses are advised if the PID efficiency is expected to be a dominant uncertainty.

3.1 Statistical uncertainties

Statistical uncertainties on $\bar{\varepsilon}$ arise from finite statistics in the input samples to the calibration procedure, namely the calibration and reference sample.

3.1.1 Calibration sample size

The efficiency ε_i in each bin i of the efficiency tables produced by PIDCalib is the result of dividing the number of events passing the selection by the total number of events in the calibration sample falling into this bin. However, the resulting uncertainty cannot be computed using the well-known equation for binomial uncertainties:

$$\sigma_{\varepsilon_i}^2 = \frac{\varepsilon_i\,(1-\varepsilon_i)}{R_i}\ , \tag{12}$$

as this equation assumes unweighted data while ${}_s\mathcal{W}$eights are used to estimate the number of signal decays in the calibration sample. Toy studies reveal that the uncertainty is underestimated by the above equation. Internally, PIDCalib uses ROOT to divide two histograms with bin-by-bin variance defined by

$$\sigma_{\varepsilon_i}^2 = \frac{1}{C_i^2}\left[\left(1-2\frac{C_i'}{C_i}\right)\sum_{j\in C_i'} {}_s\mathcal{W}_j^2 + \frac{C_i'^2}{C_i^2}\sum_{j\in C_i} {}_s\mathcal{W}_j^2\right]\ . \tag{13}$$

This equation, derived following Chapter 8.5 in [16], is found to reproduce the true uncertainty in toy studies and reduces to Equation 12 in case of unit weights.

When computing the efficiency of the reference sample $\bar{\varepsilon}$ using Equation 6, the uncertainties on the individual ε_i have to be propagated to the final uncertainty on $\bar{\varepsilon}$. While an analytical solution is possible if only one track per event has a PID selection requirement, the problem becomes highly complicated in case of multiple tracks. In this case, events can be partly correlated if some of the tracks share the same bin in the efficiency tables. Therefore, it is recommended to use Monte Carlo error propagation to obtain the correct uncertainty taking all correlations into account. In this approach, the PIDCalib calibration procedure should be repeated n times. In each iteration, a new efficiency table is generated by drawing a new random number for each ε_i from a Gaussian distribution with a mean equal to the value in the nominal table (obtained using the technique described in Section 2) and sigma given by Eq. 13. This leads to n results for $\bar{\varepsilon}$ whose distribution follows a Gaussian distribution as it is obtained from a linear combination of Gaussian distributed variables. Therefore, the uncertainty on $\bar{\varepsilon}$ due to the size of the calibration sample is given by the standard deviation of the n results for $\bar{\varepsilon}$.

3.1.2 Reference sample size

In the calculation of the $\bar{\varepsilon}$ in Equation 6, an average over all events in the reference sample is taken. The limited size of the reference sample may introduce a statistical uncertainty on $\bar{\varepsilon}$. However, several points should be considered before this uncertainty is included. First of all, if the reference sample is much larger than the signal sample, its contribution to the overall uncertainty is negligible. Furthermore, in case one does not include this uncertainty, the obtained $\bar{\varepsilon}$ can be seen as being the efficiency for exactly this reference sample, and not as the expected efficiency covering all possible classes of similar but statistically independent reference samples. This might be desired in certain cases, *e.g.* if the reference sample is a simulation sample that is also used to estimate other efficiencies in order to avoid double-counting of the uncertainty due to the size of this sample (otherwise, a careful evaluation of the correlation, for example using bootstrapping [17], might be necessary). In case one has to include this uncertainty explicitly, it can be calculated using the well-known standard error of the mean formula

$$\sigma_{\bar{\varepsilon}}^2 = \frac{\sum a_e^2}{\left(\sum a_e\right)^2} \cdot \frac{\sum a_e}{\left(\sum a_e\right)^2 - \sum a_e^2} \cdot \sum \left(\varepsilon_e - \bar{\varepsilon}\right)^2 a_e \ , \qquad (14)$$

where ε_e and a_e are the efficiency and weight of the e-th event in the reference sample and all sums are taken over all events in the reference sample. Alternatively, a resampling of the reference sample using bootstrapping might be used, which can be combined with the evaluation of the uncertainty due to the calibration sample size.

3.2 Systematic uncertainties

Several systematic uncertainties related to assumptions made in the calibration procedure are discussed in the following. Generally, those uncertainties substantially exceed the statistical sources discussed in Sect. 3.1.

3.2.1 Differences between reference and signal sample

In the calibration procedure, the reference sample is used to reweight the calibration sample to estimate the efficiency of the signal sample. This implies that the kinematics of all tracks and their correlations found in the signal sample are correctly described by the reference sample. The evaluation of the arising uncertainty is highly dependent on the actual analysis. It can even be avoided entirely by using the actual signal sample as the reference sample. In case Monte Carlo samples are used as reference samples, studies on the agreement of MC and data are necessary, also including careful evaluation of potential correlations to other components of the analysis that use the same simulated data sample. For example, for the result in Ref. [18], for each kinematic bin the p_T and η distribution in simulation is weighted to the one in background subtracted data, and the efficiency is evaluated again with the weighted kinematic distribution in each bin; the variation of the efficiencies is quoted as systematic uncertainty.

3.2.2 Choice of binning

The choice of how the calibration sample is partitioned to compute the entries in the efficiency table should balance the competing requirements of keeping as small as possible both the variation of the efficiency over the bin and the statistical uncertainty on that efficiency. To illustrate how possible uncertainties are introduced by this choice, let x be an observable on which the efficiency $\varepsilon(x)$ depends. Let $R(x)$ and $C(x)$ be the probability density function of x in the reference and calibration sample respectively and let x_i and x_{i+1} be the chosen boundaries of the i-th bin in the efficiency table. When computing the efficiency for this bin $\varepsilon_i = C'_i/C_i$, one provides an estimate whose expectation value is given by

$$\langle\varepsilon_i\rangle_C = \frac{\int_{x_i}^{x_{i+1}} \varepsilon(x) \cdot C(x) \mathrm{d}x}{\int_{x_i}^{x_{i+1}} C(x) \mathrm{d}x} ,$$

where the subscript C indicates that the expectation is with respect to $C(x)$. The same expectation value for the reference sample is hence given by

$$\langle\varepsilon_i\rangle_R = \frac{\int_{x_i}^{x_{i+1}} \varepsilon(x) \cdot R(x) \mathrm{d}x}{\int_{x_i}^{x_{i+1}} R(x) \mathrm{d}x} .$$

When applying the values in the efficiency tables to the reference sample to obtain $\bar{\varepsilon}$, the obtained ε_i must be an estimate of $\langle\varepsilon_i\rangle_R$ in order to yield an unbiased estimate of $\bar{\varepsilon}$:

$$\begin{aligned} \frac{\sum \langle\varepsilon_i\rangle_R R_i}{R} &= \frac{\sum \frac{\int_{x_i}^{x_{i+1}} \varepsilon(x) R(x) \mathrm{d}x}{\int_{x_i}^{x_{i+1}} R(x) \mathrm{d}x} R \frac{\int_{x_i}^{x_{i+1}} R(x) \mathrm{d}x}{\int_X R(x) \mathrm{d}x}}{R} \\ &= \frac{\sum \int_{x_i}^{x_{i+1}} \varepsilon(x) R(x) \mathrm{d}x}{\int_X R(x) \mathrm{d}x} = \frac{\int_X \varepsilon(x) R(x) \mathrm{d}x}{\int_X R(x) \mathrm{d}x} = \langle\bar{\varepsilon}\rangle . \end{aligned}$$

However, the expectation value of ε_i is $\langle\varepsilon_i\rangle_C$ which in general is not identical to $\langle\varepsilon_i\rangle_R$, thus introducing a bias to $\bar{\varepsilon}$. The magnitude of the bias (and thus the systematic uncertainty) significantly depends on the chosen binning scheme's approximation of the assumption $\langle\varepsilon_i\rangle_R = \langle\varepsilon_i\rangle_C$. Based on the definitions above, two extreme cases can be identified which satisfy the assumption:

1. The efficiency within a bin does not depend on x ($\varepsilon(x) \approx \varepsilon$ for all x in the bin), which cancels the dependence on the distribution of x.

2. The distributions $R(x)$ and $C(x)$ are identical within a bin ($R(x) = a \cdot C(x)$ for all x in the bin with a being constant), resulting in the same expectation value for this bin independent of the distribution.

A binning scheme should be chosen keeping this in mind. To assess the actual uncertainty on $\bar{\varepsilon}$, a first test should be a variation of the used binning scheme. While a small or even negligible change of $\bar{\varepsilon}$ does not necessarily indicate a good binning scheme (the tested binning schemes might just violate the assumption equally but still substantially), a large change indicates a major problem with the chosen binning scheme. Careful studies are therefore necessary, as the PIDCalib procedure might not be applicable to the specific analysis in terms of obtainable precision. The size of the uncertainty could be examined by performing studies using toy models resembling the kinematics and efficiency dependence on these kinematics as closely as possible. With increasing number of bins in the efficiency tables, a converging behaviour is seen and the difference between the efficiency in the nominal binning and the converged value in the toys can be assigned as the systematic uncertainty.

3.2.3 ${}_s\mathcal{W}$eight in calibration sample

In order to obtain the signal contribution in each bin of the efficiency tables, the sum of ${}_s\mathcal{W}$eights is computed within those bins. These ${}_s\mathcal{W}$eights have been obtained from a fit to the entire calibration sample and hence imply that all assumptions stated in Ref. [9] are fulfilled. However, studies show a discrepancy between the sum of ${}_s\mathcal{W}$eights in a bin and the yield resulting from a fit performed only in the specific bin. This is caused by a dependence of the fit to the invariant mass – namely the resolution of the mass peak – on the momentum and pseudorapidity of the probed track. Preliminary studies indicate a bias of the values in the efficiency tables of about 0.1% per track (absolute, not relative, and in the case where multiple tracks are involved, it has to be added linearly). The actual impact of this bias depends on the kind of the analysis and on the accuracy required. Analyses not relying too much on an accurate PID evaluation (on its absolute correction and/or on its kinematic dependence) can safely use the 0.1% absolute uncertainty. Other analyses should study in detail the impact of this bias in their final result. Particular attention has to be paid to the level of mis-identification: $\mathcal{O}(0.1\%)$ rates, e.g. in the evaluation of $p \rightarrow \mu$, are quite common and in such cases the systematic effect from the kinematic dependence of the ${}_s\mathcal{W}$eights would introduce an $\mathcal{O}(100\%)$ uncertainty. These analyses require specific - often time consuming - studies, as it has been done for the $B^0_s, B^0 \rightarrow \mu^+\mu^-$ results, e.g. Ref. [19], from which the ~0.1% estimate comes.

4 Processing the calibration samples

Defining and selecting the calibration samples, applying the ${}_s\mathcal{P}$lot technique to subtract the residual background, and storing properly the combined information is a set of tasks that are not directly related to individual physics analyses, and which set scalability challenges to satisfy the increasing need for precision of many of the LHCb measurements. Therefore a collaboration-wide strategy has been implemented to select calibration samples directly in the trigger. A *general-purpose* fit strategy and ${}_s\mathcal{W}$eight computation is also implemented, but, as explained in Section 3, analyses heavily relying on PID may need more accurate studies to reduce the uncertainties in particular regions of the phase space. This section briefly describes the data-processing of the calibration samples.

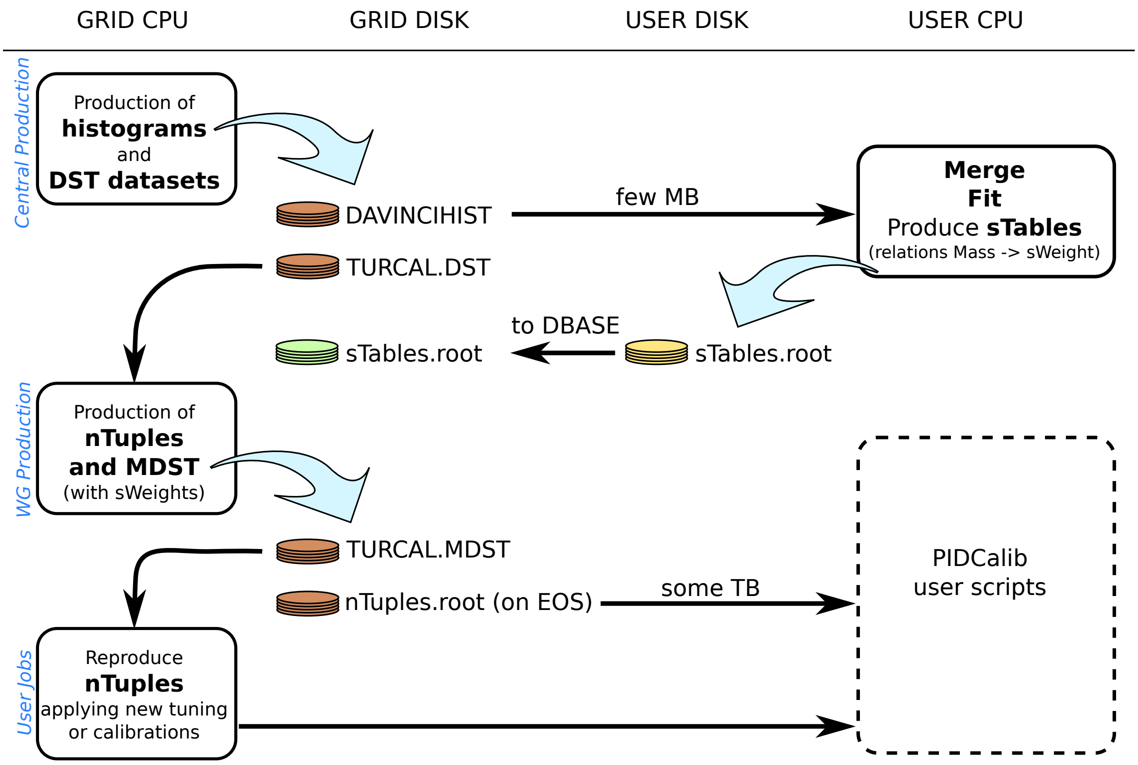

Figure 1: Schematic of the data flow and processing in the *expert-side* of PIDCalib.

4.1 Selection strategy

The selection of PID calibration samples is mainly implemented in a set of HLT2 lines grouped in the TurboCalib stream of data [12]. At least two samples per particle type (with the exception of electrons) have been identified with the purpose of comparing the performance, in order to assess systematic uncertainties due to the choice of the sample. In general, an attempt was made to have the two samples cover different regions of the phase space, in order to enrich bins otherwise poorly populated. At the same time the samples are required to have sufficient overlap to permit the comparison of their performance. The list of the trigger lines developed and commissioned in 2015 and the final selection for the Run 2 are described in detail elsewhere [13].

4.2 Workflow in Run 2

The processing of the calibration samples in Run 2 required a completely new data flow that has been described in detail elsewhere [12]. A schematic representation of the workflow is depicted in Figure 1. For the present discussion the only relevant point is that in the processing a few histograms are produced for each run. These are mainly the input for the first step of the PIDCalib processing, but are used also to monitor the quality of the selected PID samples. The obtained histograms are then fitted using the RooFit package [20] to obtain the ${}_s\mathcal{W}$eights (details are given in the following Sect. 4.3). The latter are stored in a database to be accessible to users as well as to Dirac jobs [21]. A main feature of the workflow in Run 2 is that the full calibration sample files are processed through a centralized production that pairs each candidate with its corresponding ${}_s\mathcal{W}$eight (read from the database) and saves all this information into both ROOT `TTree` (*nTuples*) and MicroDST [1] files. A number of PID studies and algorithm developments can be achieved simply using these output files. First, the low-level detector-related variables can be attached to the single candidate together with the ${}_s\mathcal{W}$eight, allowing for the study of systematic effects, such as the asymmetry depending on the entry point of a particle in a particular subdetector. Second, the output files contain both the online-reconstructed decay chain and the matched tracks as reconstructed offline [12]. This feature is intended to allow a possible modification of the offline reconstructions measuring the change in performance of requirements combining the new selection strategy with respect to the online selection based on the reconstruction available during data-taking.

4.3 Fit procedure and sWeight computation

Each physics channel, corresponding to one or more trigger lines, is fitted with a different fit model. The description of the fit model is beyond the purpose of this document and additional information can be found for example in Ref. [13]. Every fit model defines the components that are expected to contribute, their functional forms and associated shape parameters as well as their respective normalisation parameters. The components are named arbitrarily, but one and only one “Signal” component has to be defined. The “Signal” component is the one for which the PID performance will be computed. All other components are considered backgrounds and are statistically subtracted from the sample through the ${}_s\mathcal{P}$lot technique. The fitting procedure is split into different phases. First, the fit lets all parameters free including masses and widths of the signal components and shape parameters of the background components. This first part may include subsequent fits to initialize some parameters in a smart order. The last iteration of the fit fixes all parameters constant, except the normalizations of the various components. The covariance matrix produced in this last iteration is used to define a relation between the mass m of the parent particle and the signal ${}_s\mathcal{W}$eight to be assigned to the child candidate. After the fit, the invariant-mass distribution is divided into many fine bins, for each of which the ${}_s\mathcal{W}$eight of the signal component is defined as

$${}_s\mathcal{W}_{\mathrm{sig}}(m) = \frac{\displaystyle\sum_{i\in\mathcal{S}} \mathcal{V}_{\mathrm{sig},i} f_i(m)}{\displaystyle\sum_{j\in\mathcal{S}} f_j(m)\mathcal{N}_j} \,, \tag{15}$$

[1] files in a similar format to the full calibration samples but storing only a sub-set of the information

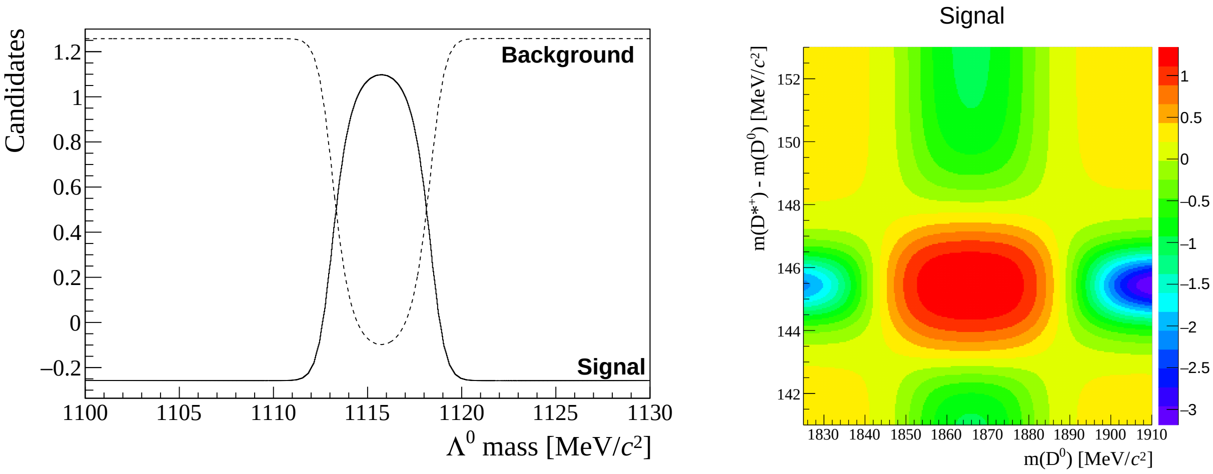

Figure 2: Two examples of ${}_s\mathcal{T}$ables. The plot on the left represents, as an example of one-dimensional ${}_s\mathcal{T}$able, the relation between the invariant mass of a $\mathit{\Lambda}$ candidate and the associated signal and background ${}_s\mathcal{W}$eights. The plot on the right represent the relation between the invariant mass of D^{*+} and D^0 candidates and the signal ${}_s\mathcal{W}$eight associated to the daughter kaon and pion.

where i and j run over the categories $\mathcal{S}$ including a signal component and one or more background components, $f_i(m)$ is the probability density function PDF of the i-th component computed for mass m (the value at the bin centre), and $\mathcal{N}_i$ the normalization of that component as found from the last iteration of the fit procedure. Finally, $\mathcal{V}_{\text{sig},i}$ is the element of the covariance matrix obtained from the fit defining the covariance between the signal and the i-th component. The ${}_s\mathcal{W}_{\text{sig}}(m)$ values are stored (as a map relating each fine bin in m to an ${}_s\mathcal{W}$eight to be assigned to the daughter particles) in ${}_s\mathcal{T}$ables that can be accessed for subsequent use through a release to a distributed database. It is worth to note here that the values stored in the ${}_s\mathcal{T}$ables may suffer from the systematic effect described in Sect. 3.2.3. Figure 2 presents two ${}_s\mathcal{T}$ables that are examples of the 1D and 2D cases. The 1D ${}_s\mathcal{T}$able is taken from the fit to the $\mathit{\Lambda} \to p\pi^-$ decay channel, while the 2D case is taken from the decay $D^{*+}(\to K^-\pi^+) \to D^0\pi^+$. According to Ref. [9], the sum of the ${}_s\mathcal{W}$eights of the different categories applied to a single candidate, or to a single value of its mass, is unity. This means that the sum of the ${}_s\mathcal{T}$ables must produce, by construction, a constant function, equal to one for every value of the mass. This property is used to perform consistency checks of a valid construction of the ${}_s\mathcal{T}$ables.

5 Conclusions

The PIDCalib package is the main tool used within the LHCb Collaboration to determine PID performance. It provides background-subtracted calibration samples of electrons, muons, pions, kaons, and protons. The tool was first developed within a small group of analyses [6–8] and it has grown over the years including many other PID-related studies and tools. This note has described the historical evolution of the tool and how it has been modified to cope with the increasing size of PID calibration samples collected during Run 1, and even more so in Run 2. The main characteristics of the performance evaluation, namely the definition of the provided quantities such as the efficiency and the evaluation of both the statistical and systematic errors, have been described. Finally, the new approach of selecting and processing the calibration samples, developed in between Run 1 and Run 2, will allow the PIDCalib package to also be used to test the performance of new PID algorithms. This feature is a key ingredient for the high quality of Run 2 physics results, and also for the prospective LHCb upgrade.

Acknowledgements

We express our gratitude to our former colleagues T. Bird (at the School of Physics and Astronomy, University of Manchester, Manchester, United Kingdom), P. Hunt and A. Powell (both at the Department of Physics, University of Oxford, Oxford, United Kingdom) who developed the early versions of the PIDCalib package.

References

[1] LHCb collaboration, A. A. Alves Jr. *et al.*, *The LHCb detector at the LHC*, JINST **3** (2008) S08005.

[2] M. Adinolfi *et al.*, *Performance of the LHCb RICH detector at the LHC*, Eur. Phys. J. **C73** (2013) 2431, `arXiv:1211.6759`.

[3] R. Aaij *et al.*, *Performance of the LHCb calorimeters*, LHCb-DP-2013-004, in preparation.

[4] F. Archilli *et al.*, *Performance of the muon identification at LHCb*, JINST **8** (2013) P10020, `arXiv:1306.0249`.

[5] B. Sciascia, *LHCb Run 2 Trigger Performance*, LHCb-PROC-2016-020.

[6] LHCb collaboration, R. Aaij *et al.*, *Observation of the suppressed ADS modes $B^\pm \to [\pi^\pm K^\mp \pi^+\pi^-]_D K^\pm$ and $B^\pm \to [\pi^\pm K^\mp \pi^+\pi^-]_D \pi^\pm$*, Phys. Lett. **B723** (2013) 44, `arXiv:1303.4646`.

[7] LHCb collaboration, R. Aaij *et al.*, *Observation of CP violation in $B^\pm \to DK^\pm$ decays*, Phys. Lett. **B712** (2012) 203, Erratum ibid. **B713** (2012) 351, `arXiv:1203.3662`.

[8] LHCb collaboration, R. Aaij *et al.*, *A model-independent Dalitz plot analysis of $B^\pm \to DK^\pm$ with $D \to K^0_S h^+h^-$ $(h = \pi, K)$ decays and constraints on the CKM angle γ*, Phys. Lett. **B718** (2012) 43, `arXiv:1209.5869`.

[9] M. Pivk and F. R. Le Diberder, *sPlot: A statistical tool to unfold data distributions*, Nucl. Instrum. Meth. **A555** (2005) 356, `arXiv:physics/0402083`.

[10] LHCb collaboration, R. Aaij *et al.*, *First evidence for the decay $B^0_s \to \mu^+\mu^-$*, Phys. Rev. Lett. **110** (2013) 021801, `arXiv:1211.2674`.

[11] R. Aaij *et al.*, *Tesla : an application for real-time data analysis in High Energy Physics*, `arXiv:1604.05596`.

[12] L. Anderlini *et al.*, *Computing strategy for PID calibration samples for LHCb Run 2*, LHCb-PUB-2016-020.

[13] O. Lupton, L. Anderlini, V. V. Gligorov, and B. Sciascia, *Calibration samples for particle identification at LHCb in Run 2*, LHCb-PUB-2016-005.

[14] R. Brun and R. Rademakers, *ROOT - An Object Oriented Data Analysis Framework*, Proceedings AIHENP'96 Workshop, Lausanne, Sep. 1996, Nucl. Inst. and Meth. in Phys. Res. A **389** (1997).

[15] LHCb collaboration, R. Aaij *et al.*, *Measurements of prompt charm production cross-sections in pp collisions at $\sqrt{s} = 13\ TeV$*, JHEP **03** (2016) 159, `arXiv:1510.01707`.

[16] F. E. James, *Statistical Methods in Experimental Physics; 2nd ed.*, World Scientific, Singapore, 2006.

[17] B. Efron, *An introduction to the bootstrap*, Chapman & Hall, New York, 1994.

[18] LHCb collaboration, R. Aaij *et al.*, *Measurement of forward J/ψ production cross-sections in pp collisions at $\sqrt{s} = 13$ TeV*, JHEP **10** (2015) 172, `arXiv:1509.00771`.

[19] LHCb collaboration, R. Aaij *et al.*, *Measurement of the $B^0_s \to \mu^+\mu^-$ branching fraction and search for $B^0 \to \mu^+\mu^-$ decays at the LHCb experiment*, Phys. Rev. Lett. **111** (2013) 101805, `arXiv:1307.5024`.

[20] W. Verkerke and D. Kirkby, *The roofit toolkit for data modeling*, tech. rep.

[21] A. Tsaregorodtsev *et al.*, *DIRAC: Distributed Infrastructure with Remote Agent Control*, eConf **C0303241** (2003) TUAT006, `arXiv:cs/0306060`.