PID

Ale Gaz

Belle II Physics Week
October 9th 2025

Outline

- Other material;
- Introduction: why PID;
- Sources of PID information;
- Combining PID information;
- Measuring PID performance using:
  - tag & probe;
  - resonances;
- The Systematics Framework;
- Backup:
  - frequently asked questions.

Other material

Time is limited, so I will not cover in detail all the topics. For further information, please take a look at:

1) Stefan's presentation at last year’s Belle II Physics Week, for a more detailed coverage on the sources of PID information in Belle II;

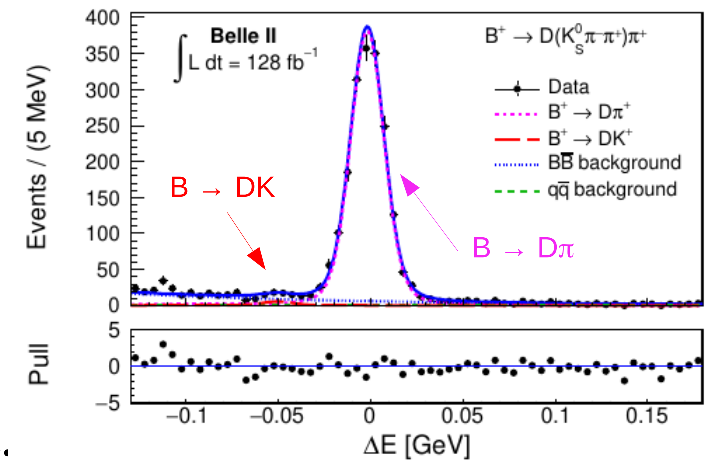
2) my presentation at the 2022 Belle II Physics Week for a (quite pedantic) description of how PID likelihoods are (or can be) combined.

Introduction

What is PID?

- At Belle II we produce 6 types of **stable** charged particles: electrons, muons, pions, kaons, protons, and deuterons (which I will mostly disregard in this talk);

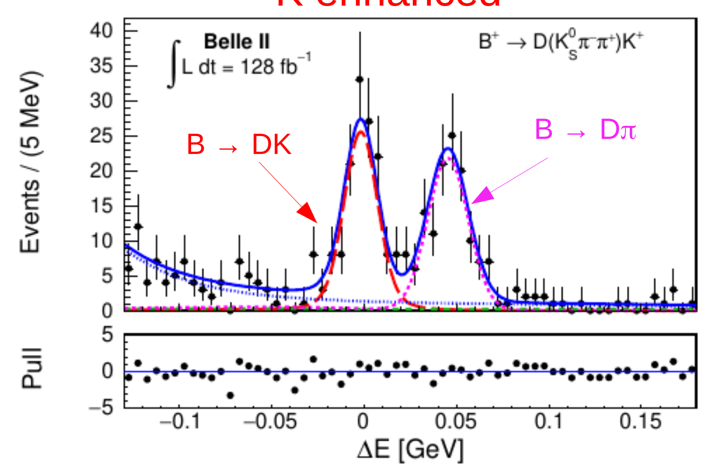
π enhanced



- The task of PID is to distinguish among these different kinds of particles:

- In practice, the identification will never be *perfect*:
  - the **efficiency** (probability that the particle species I want to select is actually selected) will be < 100%;
  - the **mis-identification (or fake) rate** (probability that a particle species that I do NOT want is actually selected) will be > 0%.

K enhanced



JHEP **2022**, 63 (2022)

Is PID relevant for Belle II?

- **Yes, it is!** Flavor physics is (almost) all about distinguishing among final states that are accessible to the same mother particle;

- Sometimes PID is the only handle that allows you to separate between very similar final states, e.g.:
  - ➔ $B^0 \rightarrow K^{*0}\gamma\,/\,\rho^0\gamma$;
  - ➔ $D^0 \rightarrow \pi^+\pi^-\,/\,\mu^+\mu^-$;
  - ➔ $\tau^+ \rightarrow e^+\,\nu\;\bar{\nu}$, $\mu^+\;\nu\;\;\nu$, $\pi^+\;\bar{\nu}$;

- Very precise LFUV measurements require a very good control over the efficiencies and the background contaminations (e's and μ's are easy to distinguish, but π's can fake both!);

- PID plays a very important role in the B and Charm Flavor Taggers. A drop or improvement in the PID performance will have a sizable impact on the tagging performance.

Is PID relevant for you?

- **Not necessarily!** It ultimately depends on your analysis;

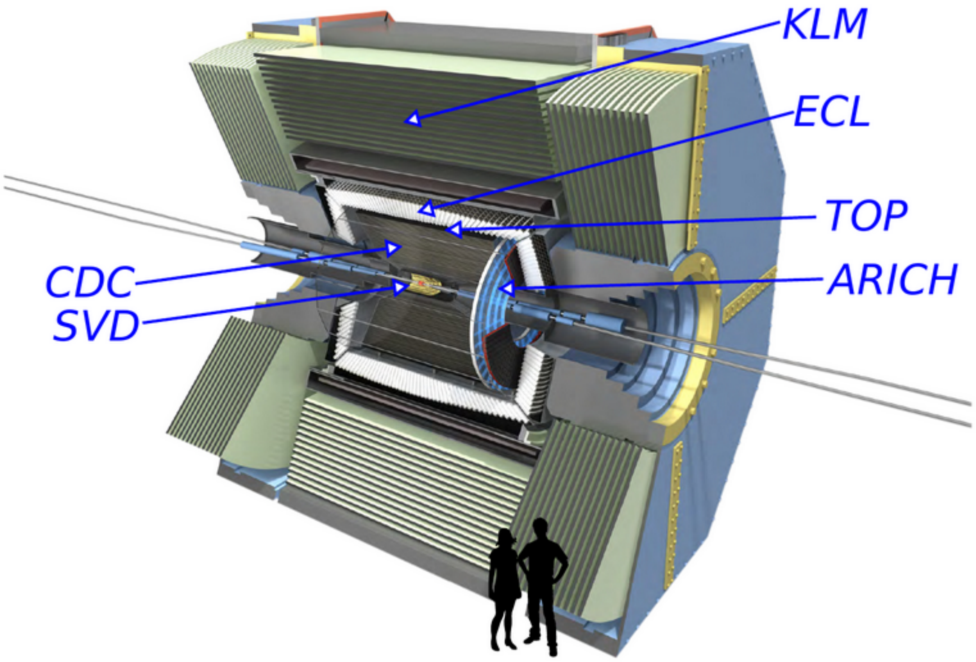
- There are quite a few cases in which you don't want to use PID:
  1) your S/N ratio is already very high: applying PID will lower your efficiency without any significant benefit from background reduction.
     Example: we can select clean samples of $K_S \rightarrow \pi^+\pi^-$ by cutting on the flight length significance, applying PID won't help much against a background that comes mostly from random combinations of π's;
  2) your main backgrounds have the same final state particles as your signal.
     Examples: PID won't help you distinguishing $\phi \rightarrow K^+K^-$ from $f_0 \rightarrow K^+K^-$ (you always have the same particles in the final state);
  3) Your backgrounds are easy to deal with (because of their different shape and/or they can be studied using data control samples);
     Example: continuum background in many $B \rightarrow$ hadronic analyses;

- You should also be aware that PID comes with systematic uncertainties: the benefits from using it should at the very least outweigh its drawbacks.

Sources of PID information at Belle II

PID @ Belle II

At Belle II, six subdetectors (all but the PXD) contribute to PID:



A large part of our job is to ensure that we combine this information in the most effective way (i.e. we maximize the efficiency and minimize the fake rate).

Measuring the velocity of particles

- From the PID point of view, SVD, CDC, ARICH, and TOP have one thing in common: they measure the velocity of the chargedparticles that pass through them;

- Keeping in mind that the tracking provides a measurement of the momentum of a particle and that

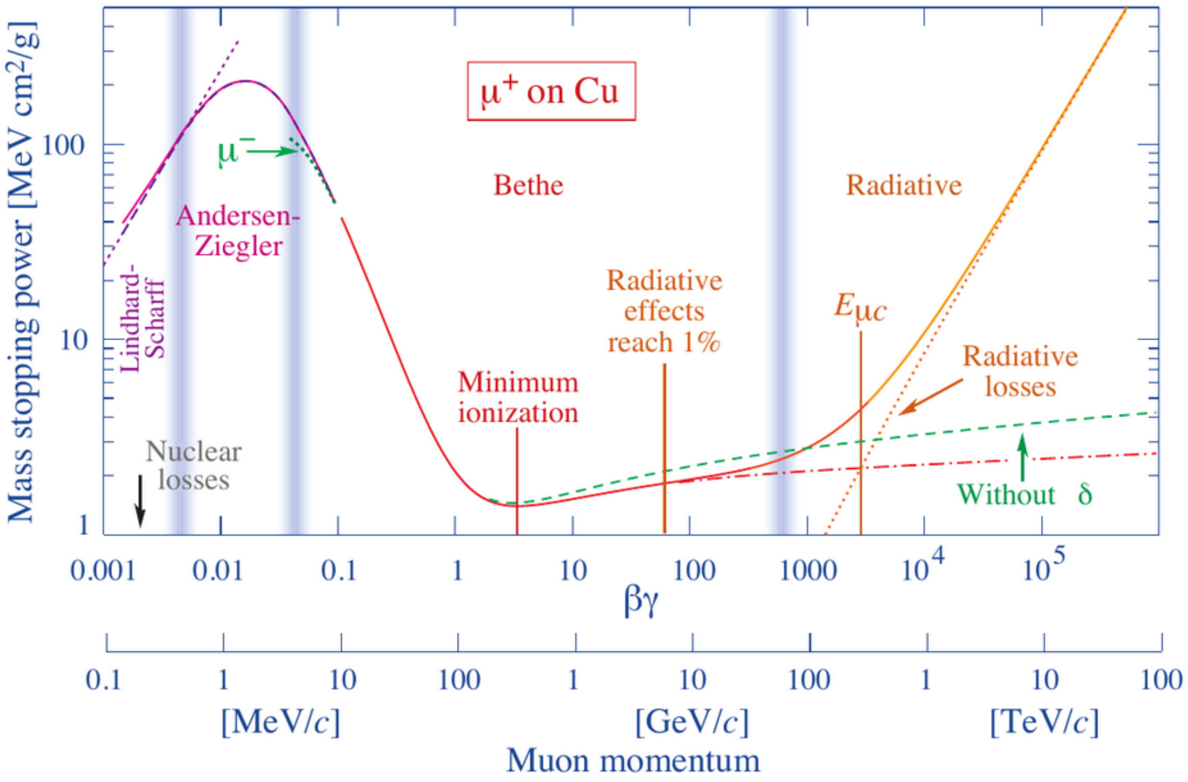
$$p = mc\beta\gamma$$

  if we measure the velocity (and thus β and γ) of that particle, we can derive its mass (which is equivalent to identifying it);

- Physics principles:
  - SVD and CDC derive β from the measurement of the specific energy loss of the particle through ionization;
  - ARICH and TOP derive β from the measurement of the Cherenkov angle emitted by the particle when traversin the aerogel / quartz.

dE/dx from SVD and CDC

As long as we are away from nuclear effects and radiative losses (which is almost awlways our case), the specific energy loss depends only on β.



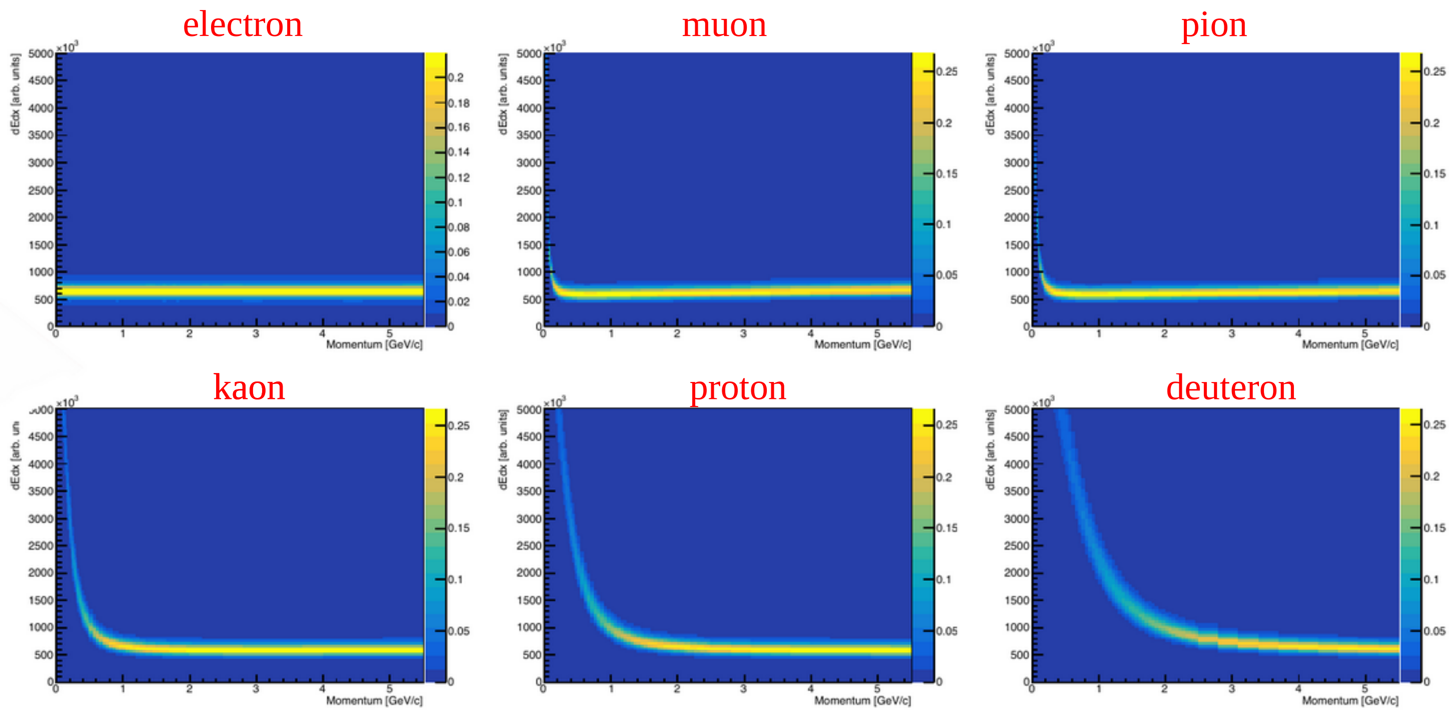
“Bethe equation, valid with good approximation for 0.1 < βγ < 1000”

$$\left\langle -\frac{dE}{dx} \right\rangle = K z^2 \frac{Z}{A} \frac{1}{\beta^2} \left[\frac{1}{2} \ln \frac{2 m_e c^2 \beta^2 \gamma^2 W_{\max}}{I^2} - \beta^2 - \frac{\delta(\beta\gamma)}{2} \right]$$

dE/dx from SVD and CDC

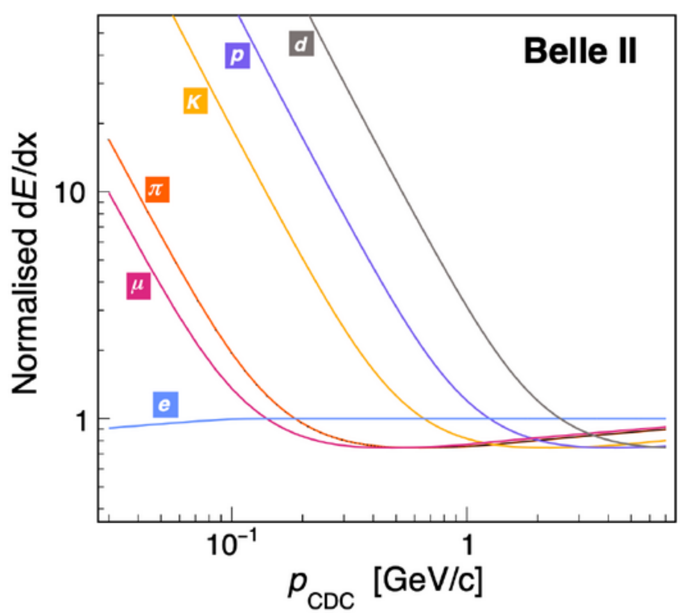
SVD

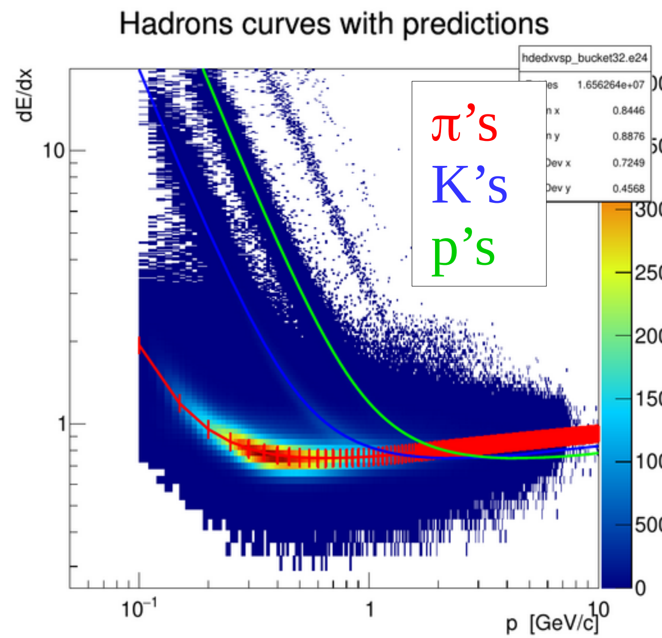
dE/dx vs p



CDC

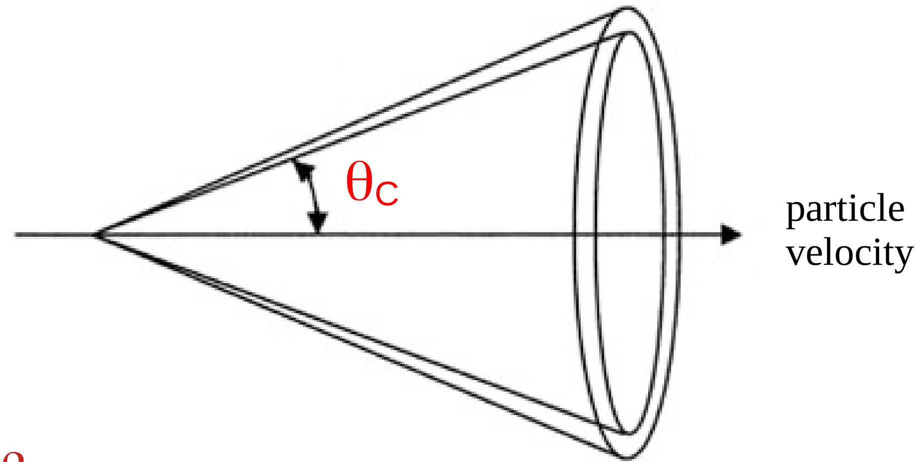
dE/dx vs p





Cherenkov angle from ARICH and TOP

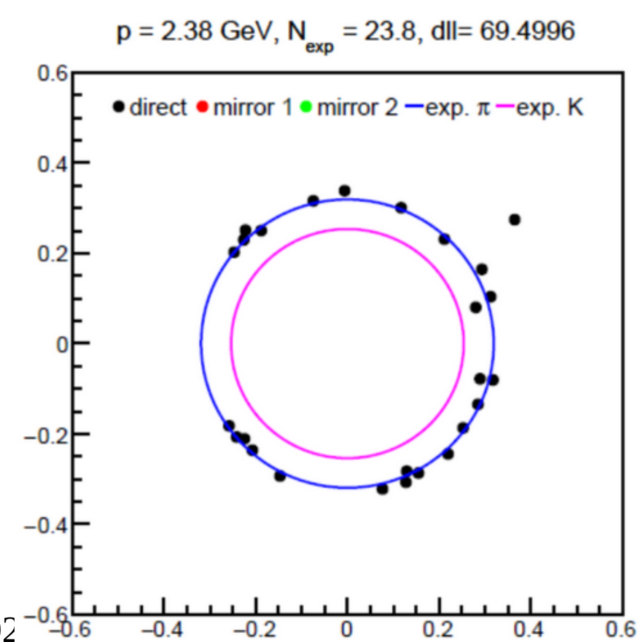
When a particle traverses a (transparent) medium with a speed higher than the speed of light in that medium (c/n), it emits Cherenkov light, at the characteristic angle θ_C, such that:



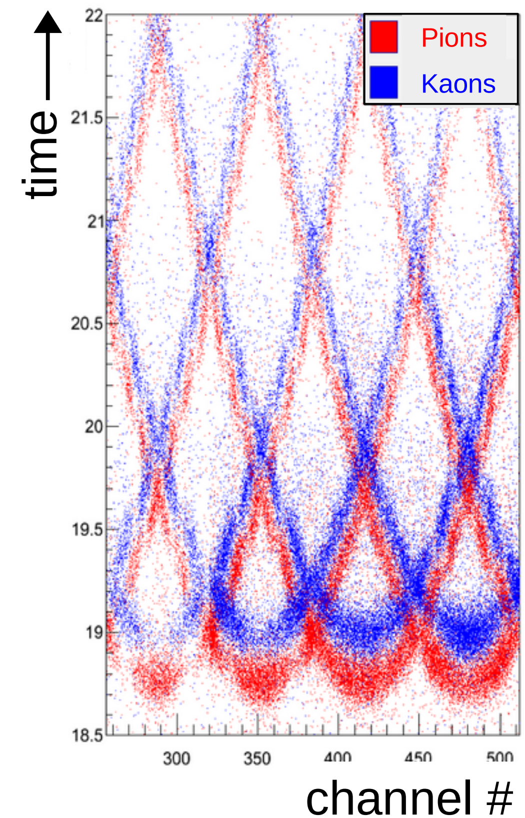
$$\cos\theta_C = \frac{1}{n\beta}$$

Measuring θ_C, we can derive β

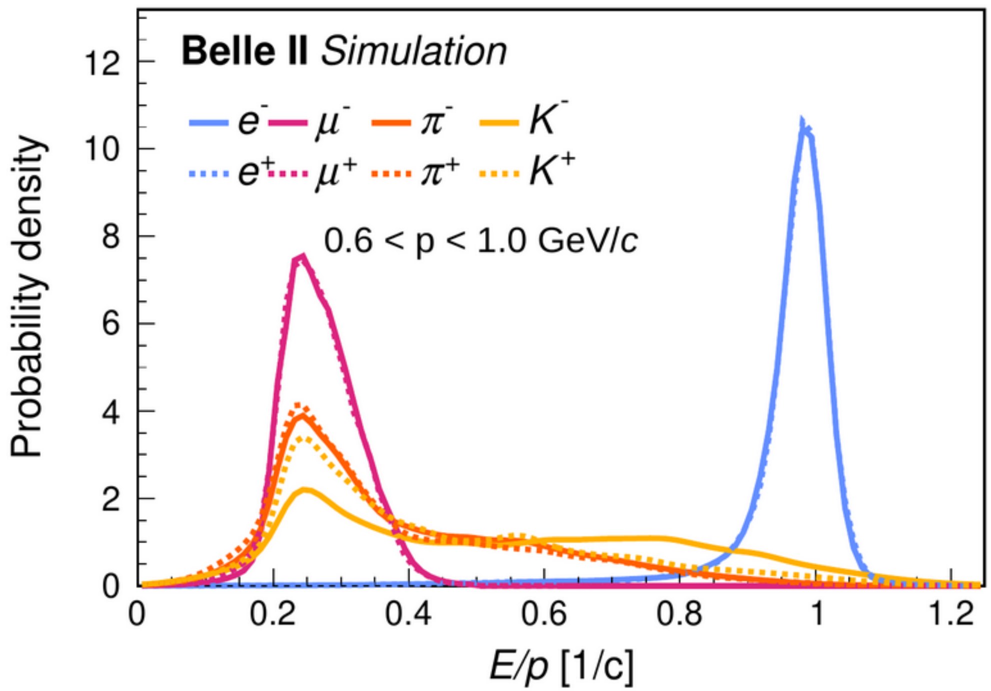
ARICH
event



TOP
working
principle



Energy deposition on the ECL



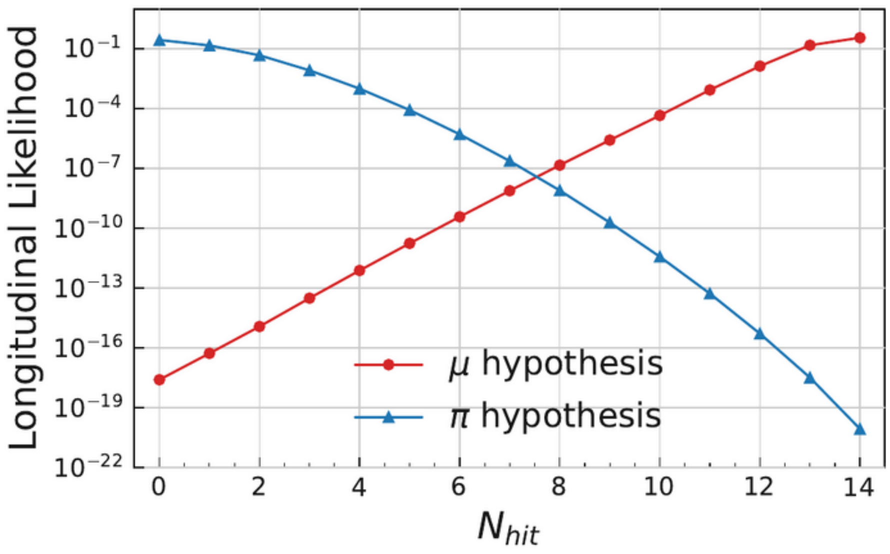
- electrons always leave ~all their energy to the calorimeter;
- muons always behave as minimum ionizing particles;
- pions (and kaons) have an intermediate behavior (and can mimic both e's and μ's)

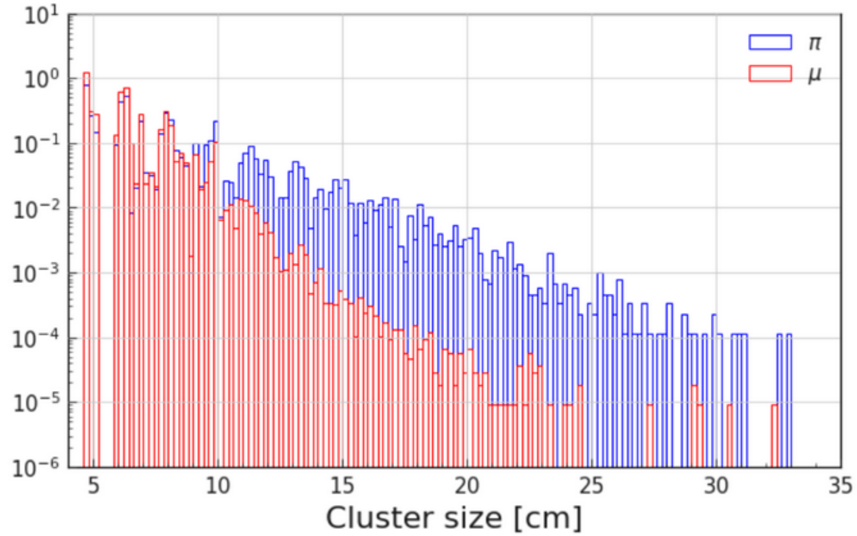
More information (and discrimination power) can be derived from the shape of the shower in the ECL.

KLM

The PID principle of the KLM is simple:

- muons (losing energy only through ionization) tend to traverse a large fraction (or possibly all) of the KLM and be only a little deflected from the
- hadrons (also undergoing hadronic interactions) are typically stopped within the first layer of the KLM and can have large deflections in the transverse direction;





Z. Wang et al. Nucl.Instrum.Meth.A 1081 (2026) 170814 and BELLE2-NOTE-TE-2024-014

PID likelihoods

- I will skip the details on how each subdetector determines it, but:
- Each of the 6 subdetector determines a **likelihood** for each of the 6 stable particle hypotheses:

$$\mathcal{L}_\alpha^d$$

d ∈ {SVD, CDC, TOP, ARICH, ECL, KLM}

α ∈ {e, μ, π, K, p, d}

- For computational reasons, we work with the logarithm of likelihoods, which you can access from basf2 via e.g.:

Lund code of the muon

$$\log(\mathcal{L}_\mu^{KLM}) = LL_\mu^{KLM} =$$
$$= \texttt{pidLogLikelihoodValueExpert(13, KLM)}$$

- The numerical values by themselves are ~meaningless, there are large arbitrary offsets between the detectors. What we care about are the differences in the log likelihoods between different particle hypotheses.

Combining PID information

Putting everything together

- For each track in the detector, we get up to 36 (6 subdetectors x 6 particle hypotheses) numbers;
- Actually, we ~never get 36 valid likelihoods because e.g. the TOP and ARICH are mutually exclusive, a track might not reach the KLM, … ;

- The picture might not be 100% consistent, e.g. for a certain track:
  - the CDC might favor (i.e. assign the highest likelihood) to the π hypothesis;
  - the TOP might favor the K hypothesis;
  - …

How do we reach a consensus on what is the most probable hypothesis for a given track, given all the available information?

The likelihood approach

- We can *multiply* all the individual subdetector likelihoods and compute the likelihood for particles hypothesis α:

$$\mathcal{L}_\alpha = \mathcal{L}_\alpha^{SVD} \cdot \mathcal{L}_\alpha^{CDC} \cdot \mathcal{L}_\alpha^{TOP} \cdot \mathcal{L}_\alpha^{ARICH} \cdot \mathcal{L}_\alpha^{ECL} \cdot \mathcal{L}_\alpha^{KLM}$$

- In practice, we sum the log likelihoods to get the log likelihood of α:

$$\log(\mathcal{L}_\alpha) = LL_\alpha = LL_\alpha^{SVD} + LL_\alpha^{CDC} + LL_\alpha^{TOP} + LL_\alpha^{ARICH} + LL_\alpha^{ECL} + LL_\alpha^{KLM}$$

- If a track is not in the acceptance of a certain subdetector, a default value is assigned to all particle hypotheses, so $\Delta LL = 0$ for all pairs of hypotheses and that subdetector practially does not contribute to the overall likelihood.

Probabilities (likelihood ratios)

- Now that we have a likelihood that combines the information from all dubdetectors, we want to compare the different particle hypotheses;
- If we want to compare the K vs π hypotheses, we can compute:

$$P(K \text{ vs } \pi) = \frac{\mathcal{L}_K}{\mathcal{L}_K + \mathcal{L}_\pi}$$

or equivalently:

$$P(K \text{ vs } \pi) = \frac{1}{1 + e^{[\log(\mathcal{L}_\pi) - \log(\mathcal{L}_K)]}} = \frac{1}{1 + e^{\Delta LL_{\pi K}}}$$

- P (K vs π) is bound to be in [0, 1], and satisfies the Kolmogorov axioms, so it can be interpreted as a **probability**.

Binary PID

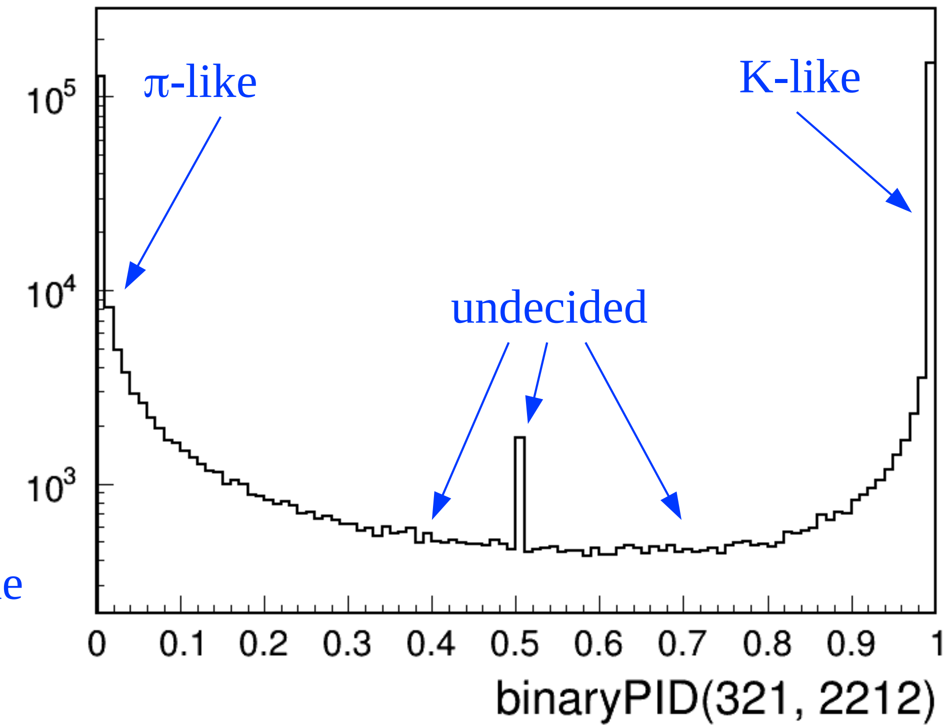
- Obviously you are not expected to compute all this by hand, `basf2` will immediately provide the ***binary*** likelihood ratio we have just defined:

$$P(K \text{ vs } \pi) = \texttt{binaryPID(321, 211)}$$

mcPDG of the particle whose L goes in the numerator

mcPDG of the particle whose L goes in the denominator

- If we plot P(K vs π), we obtain something like:



(this is a ~random sample of tracks, that contains also e's, μ's and p's)

Global PID

- Binary PID is great if you have a specific source of background that you want to suppress;
- In more generic cases, you want to compare a particle hypothesis (e.g. kaon) against all others, thus switching to the ***global*** likelihood ratio:

$$P(K) = \frac{\mathcal{L}_K}{\mathcal{L}_K + (\mathcal{L}_e + \mathcal{L}_\mu + \mathcal{L}_\pi + \mathcal{L}_p + \mathcal{L}_d)}$$

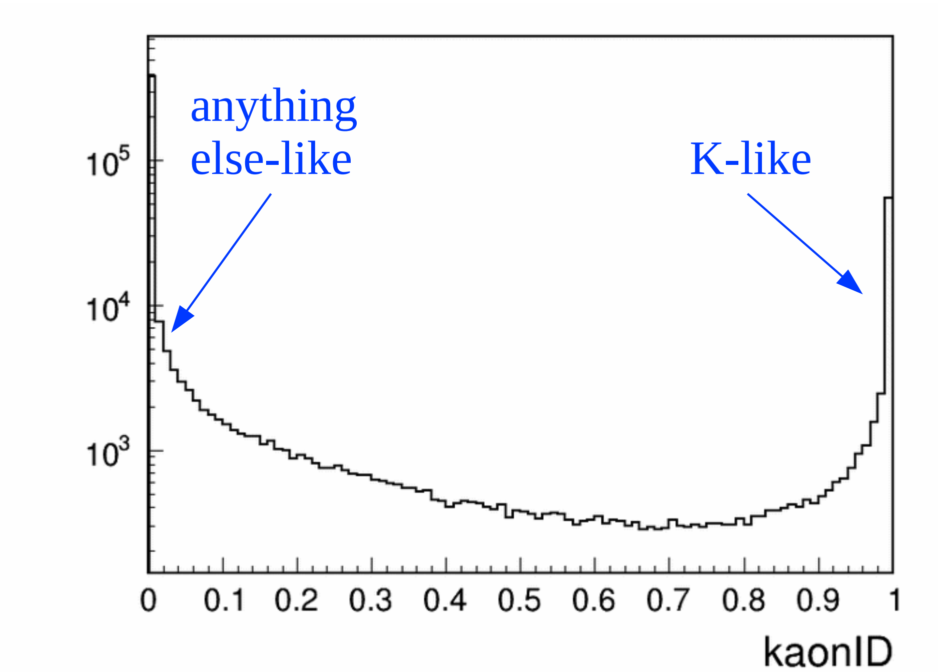
Global PID

- The global likelihood ratios I just defined are the **default PID variables**, that you can access from basf2.

  In other words, by default all sub-detectors are considered, and a particle hypothesis is checked against all others;

```
electronID
muonID
pionID
kaonID
protonID
deuteronID
```

- The distribution of e.g. kaonID is like:



(this is the same sample of tracks I used for the binary K vs p example)

Customizing likelihoods

- Given that you can get the individual likelihoods from basf2, it is easy to build your own probabilities;
- For example, you can build a *ternary* lilelihood ratio:

$$P(K\ vs\ p,\pi) = \frac{\mathcal{L}_K}{\mathcal{L}_K + \mathcal{L}_p + \mathcal{L}_\pi}$$

- Or you can compute the likelihood for each particle hypothesis excluding one or more subdetectors (in some cases the variables are already available, e.g. `muonID_noSVD`);
- In general, for release-08, it is NOT recommended to exclude subdetectors, but there might good reasons to do so for your analysis;

- The Systematics Framework will allow you to compute corrections and systematic uncertainties for any combinations of likelihoods. It takes a bit of work, but it is definitely feasible.

Is this the best we can do?

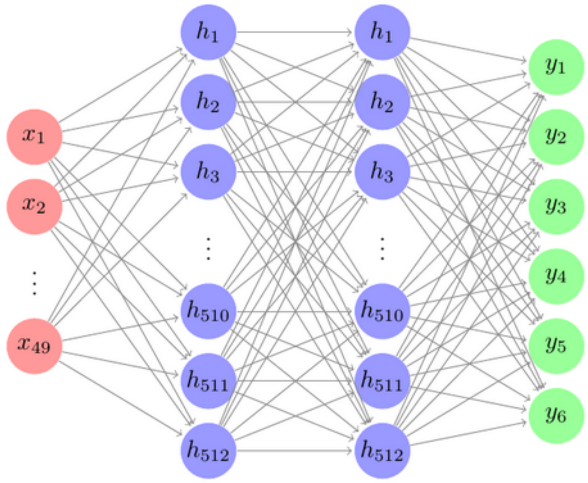
- Given that we are combining the information from all subdetectors, we might ask ourselves whether the (binary or global) approach is the best we can do;

- The answer is no, for different reasons:
  - the combination of the likelihoods assumes that the inputs are **uncorrelated** (typically a good assumption, but …);
  - subdetectors can have "**blind spots**": the global PID ignores the fact that there are regions of the phase space for which a sub-detector are not very reliable;
  - $\Delta LL\alpha = x$ does not correspond to the same separation power for all subdetectors;
  - background conditions affect sub-detectors in different ways;
  - ...

Our first attempt to overcome the limitations of the standard likelihood approach was to develop the "re-weighted" likelihood approach. I will not describe it today, but it is mentioned in the hadronID paper and described in detail in BELLE2-NOTE-TE-2021-027.

The ML approach

- To overcome the drawbacks mentioned above, we moved to the ML approach utilizing Neural Networks to perform the combination of PID information;
  - con: “black box”, not clear to see where the improvement comes from;
  - pro: clear advantage in the performance;



- The training is performed in the MC (but we might move to data at some point) using as input:
  - the 36 individual PID likelihoods;
  - some vaiables related to the ECL cluster shape / energy;
  - charge, p, cosθ, ϕ (to optimize the discrimination in different regions of the phase space).

NN PID at Belle II

- From **release-08**, the NN PID variables are available for all species:

  `electronIDNN, muonIDNN, pionIDNN, kaonIDNN, protonIDNN, deuteronIDNN`

- These are global (i.e. comparing among all 6 hypotheses) variables. If you are interested in binary (e.g. K vs π) or other combinations, you can build your own ratios, like:

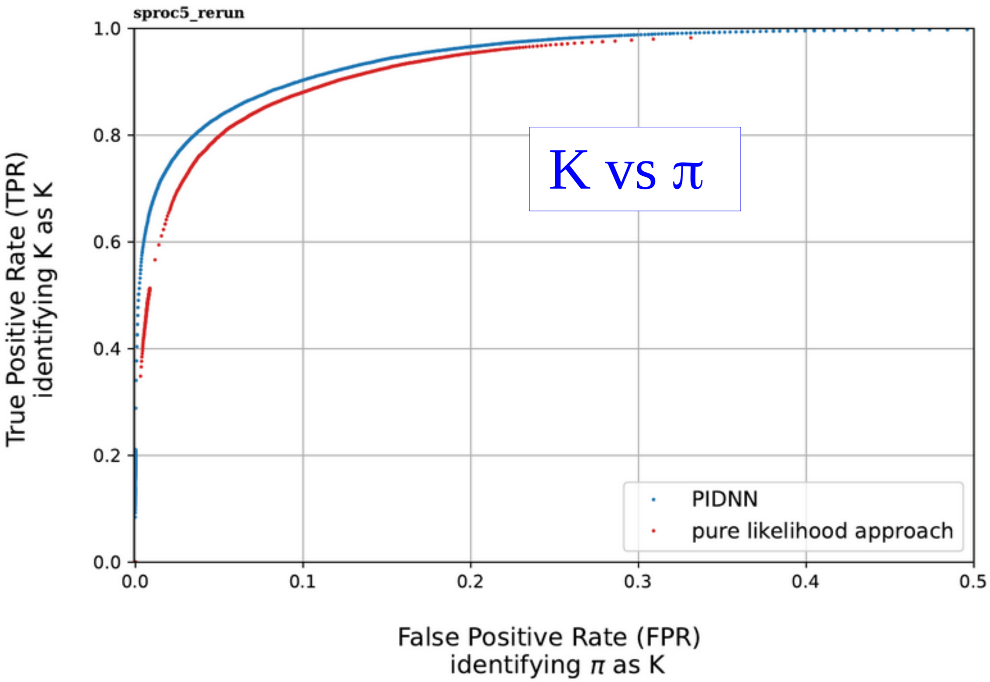
  `kaonIDNN_Kpi = kaonIDNN/(kaonIDNN + pionIDNN)`

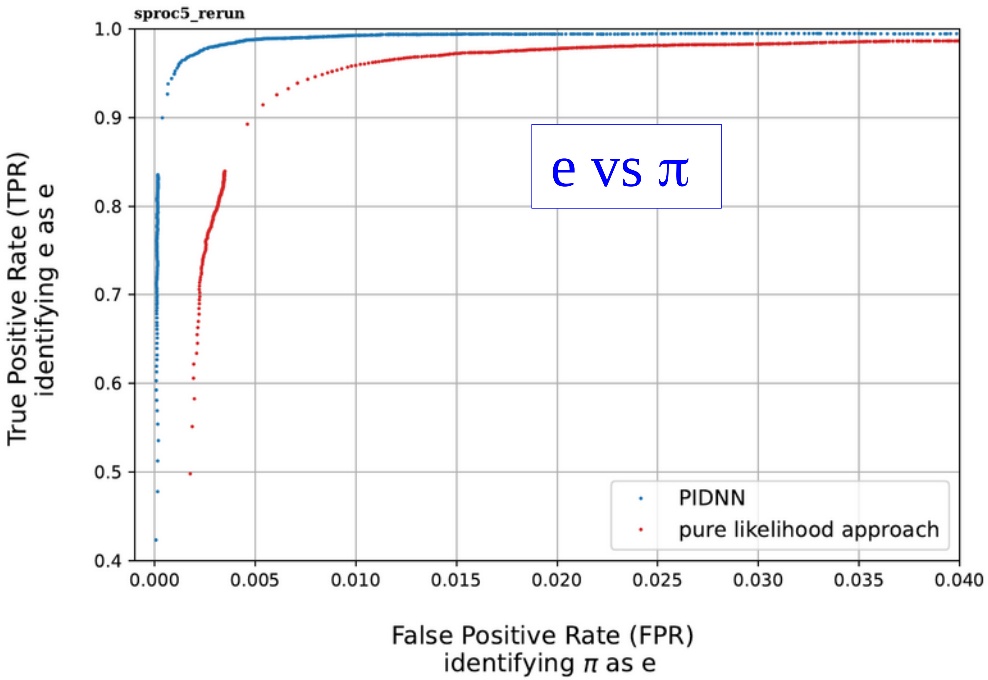
- You will have to specify the global tag: **`pid_nn_release08_v1`**;

If you are working with **release-06**:
- Only the binary pionIDNN and kaonIDNN are available;
- You will have to specify the global tag: **`pid_nn_release06_Kpi`**

Performance

PIDNN vs **likelihood**





- In *general* PIDNN variables give **clear advantage** over the likelihood;
- However, it is not guaranteed that this is true *everywhere*, there are some corners of the phase space where the likelihood works somewhat better. Please spend some time and check what works best for your analysis!

Measuring PID performance

with

- ➔ Tag & probe
- ➔ Resonances

Efficiency and fake rates

Definitions:

- **Efficiency**: probability that a particle of species α is correctly identified by the PID selection (intended to select α and reject the others);

- **β mis-ID probability (or fake rate)**: probability that a particle of species β passes the criteria meant to select α particles (you have to specify which particle is "faking" which);

Knowing your efficiencies and fake rates is key to understand your overall signal (backgrounds) selection efficiencies.

General principles

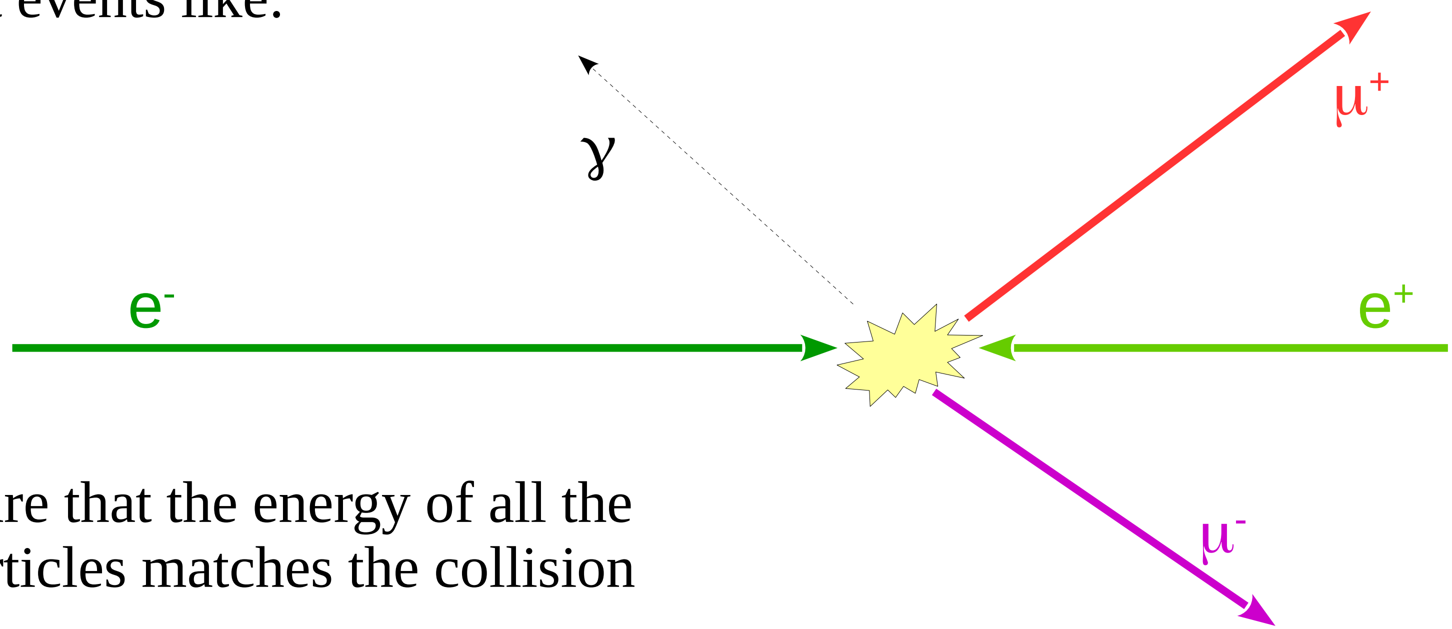
- To measure the performance of PID we need samples of tracks of a given species that are **unbiased** from the PID point of view;
- This means that we have to know the particle species we are dealing with **without applying any PID selection** to these tracks (although we can apply PID to the other tracks in the event);
- The cleaner the sample the better, but we can deal with backgrounds.

An example

- Let's say we want a sample of high-momentum muons. The natural choice is the process

$$e^+e^- \to \mu^+\mu^-(\gamma)$$

so we select events like:



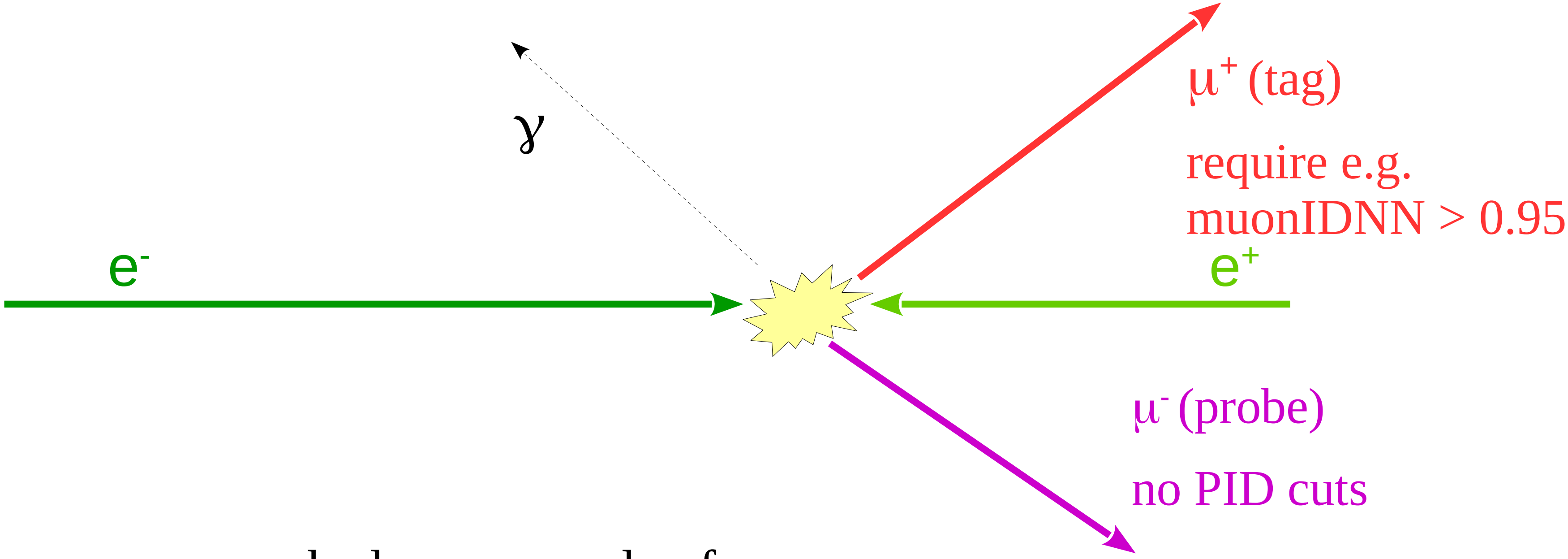
- We make sure that the energy of all the detected particles matches the collision energy… ;
- Problem: also the processes

$$e^+e^- \to e^+e^-(\gamma) \quad , \quad e^+e^- \to \pi^+\pi^-(\gamma) \quad , \ldots$$

pass our selection, we will have a very large background!

Tag & probe

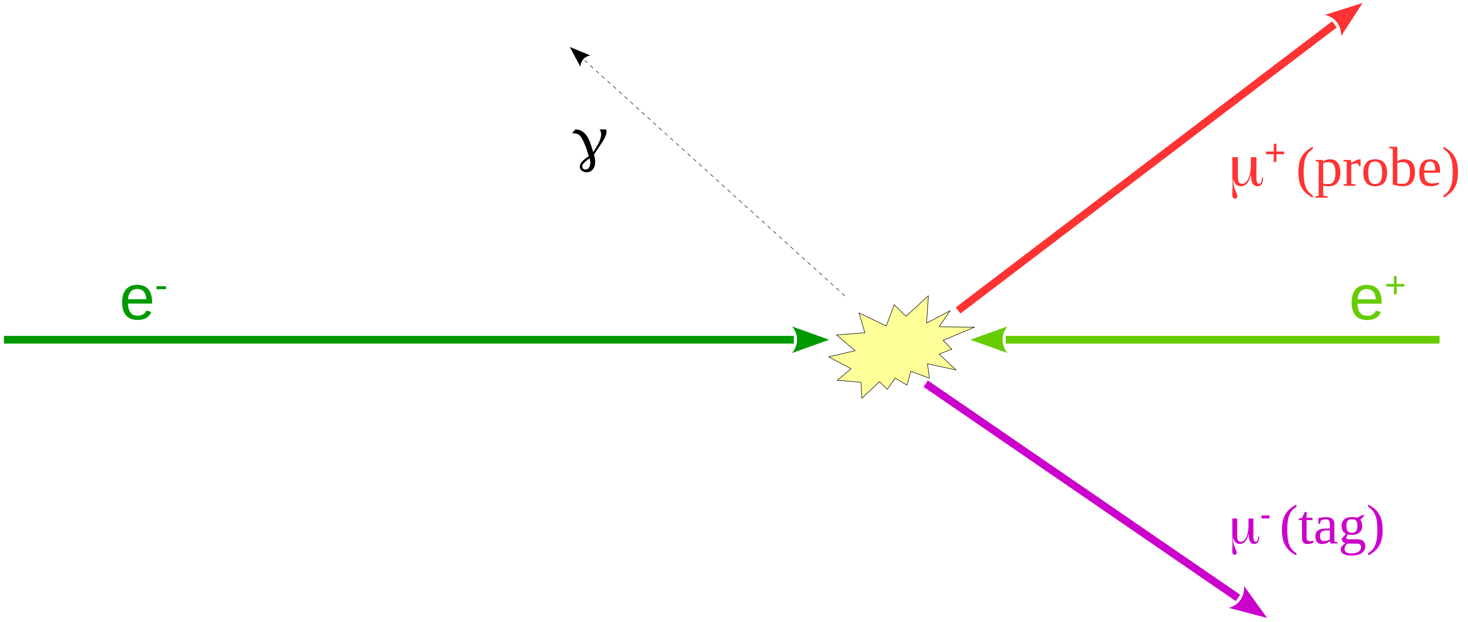
- We would like to use PID to reject backgrounds, but in this way we will bias our sample!
- Solution: apply PID to only one of the tracks (tag) and leave the other unbiased (probe):



- We have now a much cleaner sample of μ^- (the small residual background can be subtracted using the MC);
- We can apply a PID selection to this sample and measure its efficiency.

Tag & probe

- To get a clean sample of μ^+, we do the other way around:



Resonances

Another strategy employs resonances whose decay products we can recognize:

$K_S \rightarrow \pi^+\pi^-$

(when a K_S decays to two charged particles, it is always pions)

$D^{*+} \rightarrow D^0 (\rightarrow K^- \pi^+) \pi^+_{soft}$

(D^* tagged D^0's, the charge of the soft pion determines which one is the K and which one is the π)

$\Lambda \rightarrow p\, \pi^-$

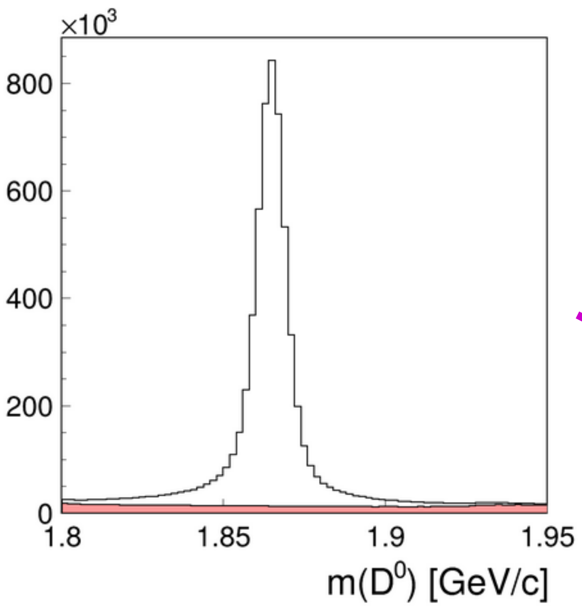
(the proton is the particle carrying most of the momentum)

$J/\psi \rightarrow e^+e^-, \mu^+\mu^-$

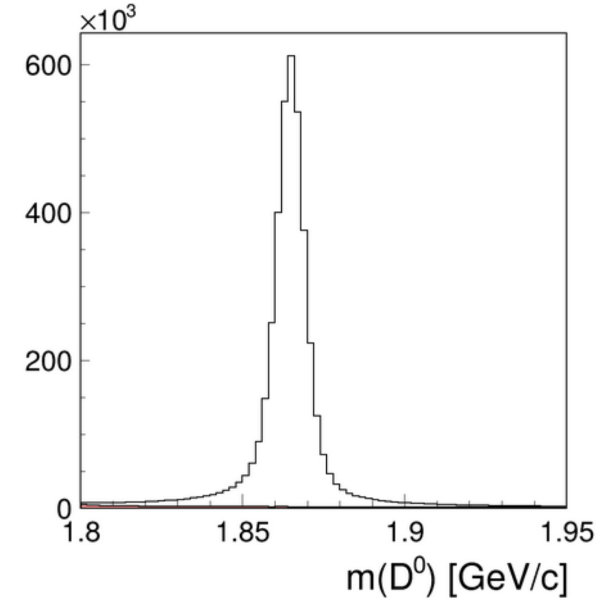
(we use tag & probe to separate the two samples and also reduce the backgrounds)

Example

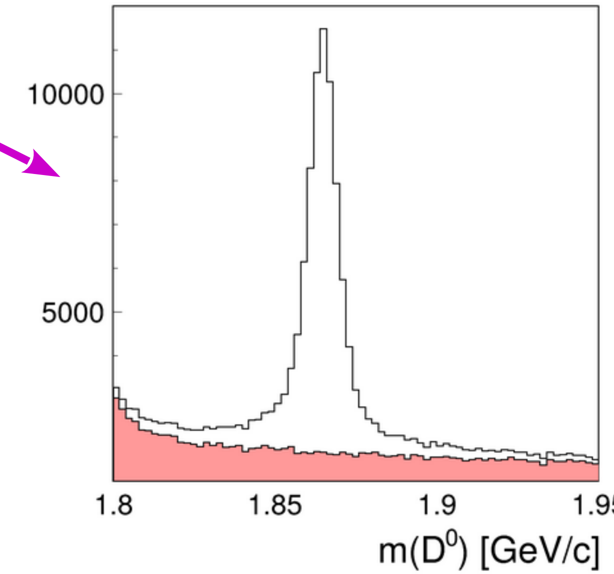
Measuring K efficiencies and $\pi \to$ K mis-ID probability with the D^0 sample:



Applying
kaonIDNN > 0.8
to the K track



Applying
kaonIDNN > 0.8
to the π track



Example

- Fitting the m(D^0) distributions shown above, we find:

$$
\begin{aligned}
N(D^0) &= 5.48 \times 10^6 \\
N(K \text{ pass}) &= 3.92 \times 10^6 \\
N(\pi \text{ fake}) &= 6.19 \times 10^4
\end{aligned}
$$

- We can then compute the K efficiency and the $\pi \to$ K fake rate:

$$\varepsilon(K \to K) = \frac{N(K \text{ pass})}{N(D^0)} = 71.6\% \qquad \varepsilon(\pi \to K) = \frac{N(\pi \text{ fake})}{N(D^0)} = 1.13\%$$

- Here we did the calculation for the whole available sample, but efficiencies and fake rates depend on p and θ. We can repeat the procedure restricting the K/π tracks to specified ranges (bins) of (p, θ);
- Of course we should also compute the statistical and systematic uncertainties of these efficiencies (I will say something later).

Why is this relevant for you?

- You might think that measuring PID efficiencies is a matter that concerns only PID and detector experts…
- Well, this is not only the case: if you are using PID in your analysis, it concerns also you!
- Suppose you want to measure a branching fraction: you will likely get your selection (that includes PID) efficiency for signal and backgrounds from the Monte Carlo;
- PID is simulated well in the MC, but not perfectly, so you will have to:
  - ➔ measure the PID efficiencies and fake rates in data and in MC;
  - ➔ compute the ratios (and their uncertainties);
  - ➔ correct the MC by these ratios so that it matches the data;

- But this (selecting control samples, applying tag selection / fitting resonances, computing efficiencies, …) is a lot of work!
- Luckily for you, this has already been done by ...

The Systematics Framework

The Systematics Framework (SF)

- Documentation:

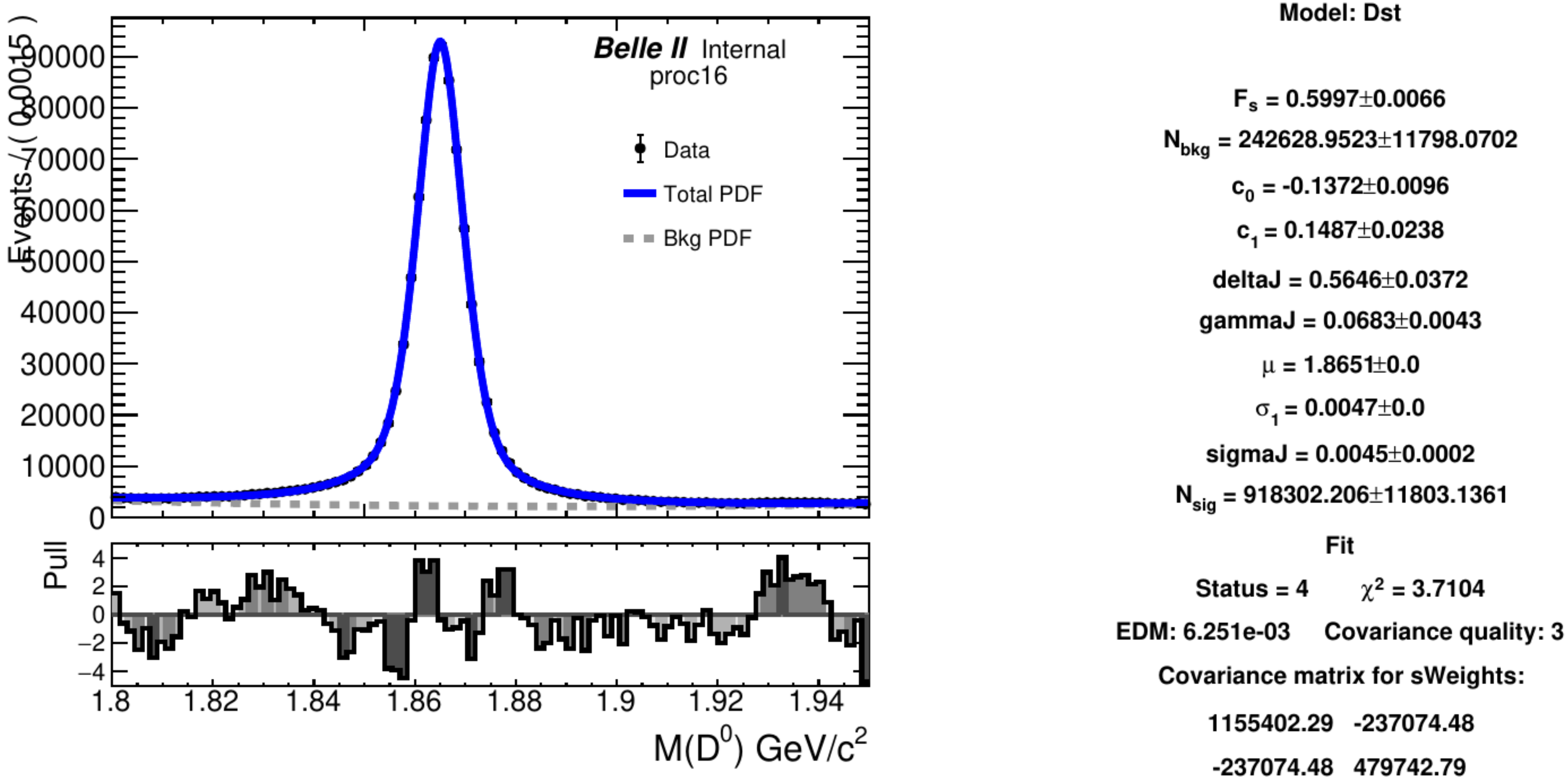
  https://syscorrfw.readthedocs.io/en/latest/

- The Systematics Framework is (going to be more and more) a central tool for Physics Performance studies;

- For the PID it provides:
  - centrally produced ntuples for all the samples needed for PID studies. The resonances are fitted already, so the ntuples are ready to be used for performance studies;
  - a suite of scripts that allow the general user to get performance plots, efficiencies, data/MC corrections, systematics … ;

- In the past we used standardized correction tables;
- Compared to the tables, the SF offers *a lot more flexibility* in terms of selection, binning, data sets.

The sWeights approach

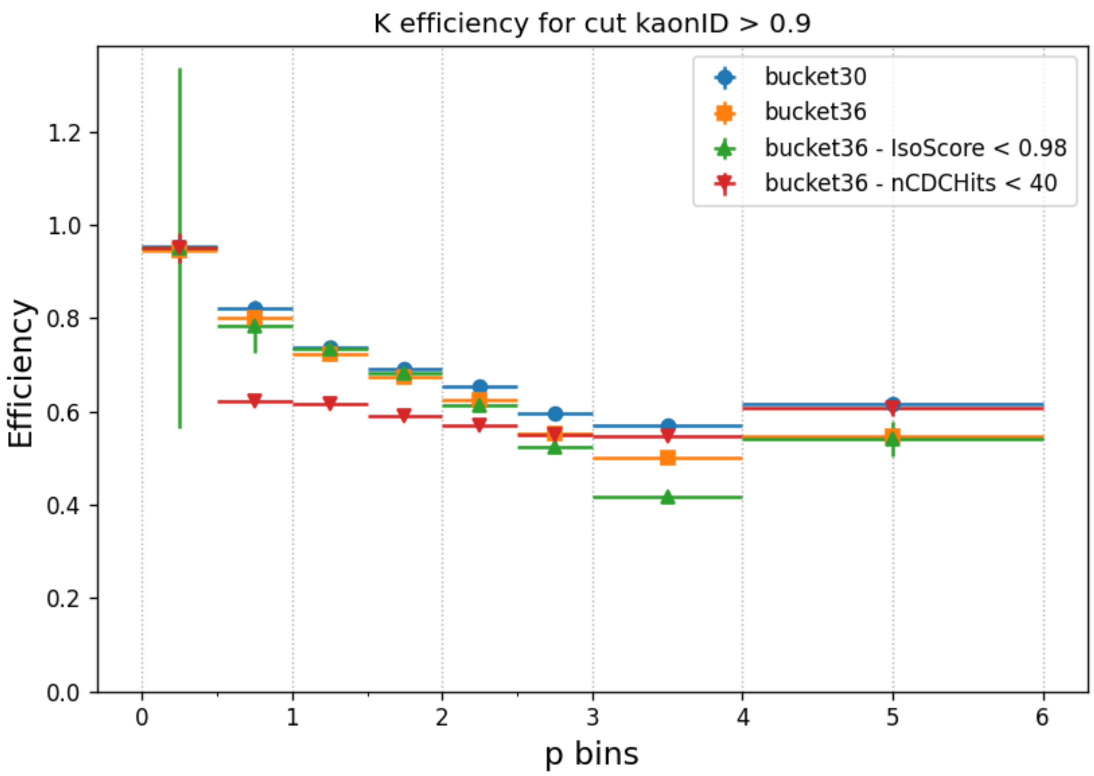


We fit the resonances mass distributions for the whole phase space for each chunk of the data set.

In other words, we are assuming that the shapes of signal and background are constant across the phase space (this is not always a good assumption)

Customizing the selection

- You can taylor the selection on your tracks so that they match your analysis (as long as the variables are in the ntuples):



- You can also study the efficiency of a custom (likelihood based) variable that you defined (e.g. a ternary likelihood ratio).

Technicalities

- You can have your local installation of the SF:

```
git clone
git@gitlab.desy.de:belle2/performance/systematic_corrections_framework.git
```

- Or you can use the scripts that are readily available at KEKCC in:

```
/group/belle2/dataprod/Systematics/systematic_corrections_framework
```

- Today I will show the usage of:

```
scripts/weight_table.py
```

The SF ntuples are available at:

KEKCC: `/group/belle2/dataprod/Systematics/production/`
NAF: `/nfs/dust/belle2/group/dataprod/Systematics/production/`

A quick example

You will have to prepare a json configuration file that looks like:

```
{
    "particle_type": "K",
    "model_names": ["Dst"],
    "cut": "kaonIDNN > 0.8",
    "data_collection": "proc16+prompt",
    "mc_collection": "MC16rd_proc16+prompt",
    "track_variables": ["p", "cosTheta"],
    "binning": [[0.5, 1.0, 1.5, 2.0, 2.5, 3.0, 3.5, 4.0, 4.5],
        [-0.866, -0.682, -0.4226, -0.1045, 0.225, 0.5, 0.766,
        0.8829, 0.9563]],
    "precut": "nPXDHits>0",
    "output": "my_K_dataMC_ratios.csv",
    "display_plots": true,
    "save_plots": "my_K_dataMC_ratios.png"
}
```

`my_K_dataMC_ratios.json`

A quick example

For which particle do you want the corrections?

Which sample do you want to use?

Recommendations:

- use one sample at a time
- for pions, use the D* sample, unless you have special needs

What PID selection are you using in your analysis?

```
{
    "particle_type": "K",
    "model_names": ["Dst"],
    "cut": "kaonIDNN > 0.8",
    "data_collection": "proc16+prompt",
    "mc_collection": "MC16rd_proc16+prompt",
    "track_variables": ["p", "cosTheta"],
    "binning": [[0.5, 1.0, 1.5, 2.0, 2.5, 3.0, 3.
        [-0.866, -0.682, -0.4226, -0.1045, 0.225, 0.5, 0.766,
        0.8829, 0.9563]],
    "precut": "nPXDHits>0",
    "output": "my_K_dataMC_ratios.csv",
    "display_plots": true,
    "save_plots": "my_K_dataMC_ratios.png"
}
```

`my_K_dataMC_ratios.json`

A quick example

```
{
    "particle_type": "K",
    "model_names": ["Dst"],
    "cut": "kaonIDNN > 0.8",
    "data_collection": "proc16+prompt",
    "mc_collection": "MC16rd_proc16+prompt",
    "track_variables": ["p", "cosTheta"],
    "binning": [[0.5, 1.0, 1.5, 2.0, 2.5, 3.0, 3.5,
        [-0.866, -0.682, -0.4226, -0.1045, 0.225, 0.
        0.8829, 0.9563]],
    "precut": "nPXDHits>0",
    "output": "my_K_dataMC_ratios.csv",
    "display_plots": true,
    "save_plots": "my_K_dataMC_ratios.png"
}
```

Data sets:

For your convenience, proc16+prompt collections are available. These contain:

- proc16_offres
- proc16_chunk1
- proc16_chunk2
- proc16_chunk3
- proc16_chunk4
- prompt16_offres
- prompt16

(you can also use a subset of these)

my_K_dataMC_ratios.json

A quick example

```
{
    "particle_type": "K",
    "model_names": ["Dst"],
    "cut": "kaonIDNN > 0.8",
    "data_collection": "proc16+prompt",
    "mc_collection": "MC16rd_proc16+prompt",
    "track_variables": ["p", "cosTheta"],
    "binning": [[0.5, 1.0, 1.5, 2.0, 2.5, 3.0, 3.5, 4.0, 4.5],
        [-0.866, -0.682, -0.4226, -0.1045, 0.225, 0.5, 0.766,
        0.8829, 0.9563]],
    "precut": "nPXDHits>0",
    "output": "my_K_dataMC_ratios.csv",
    "display_plots": true,
    "save_plots": "my_K_dataMC_ratios.png"
}
```

Do you want to compute your ratios as a function of which variables?

Your choice of binning for those variables

Preselection cuts (that match those of your analysis

`my_K_dataMC_ratios.json`

A quick example

```
{
    "particle_type": "K",
    "model_names": ["Dst"],
    "cut": "kaonIDNN > 0.8",
    "data_collection": "proc16+prompt",
    "mc_collection": "MC16rd_proc16+prompt",
    "track_variables": ["p", "cosTheta"],
    "binning": [[0.5, 1.0, 1.5, 2.0, 2.5, 3.0, 3.5, 4.0, 4.5],
        [-0.866, -0.682, -0.4226, -0.1045, 0.225, 0.5, 0.766,
        0.8829, 0.9563]],
    "precut": "nPXDHits>0",
    "output": "my_K_dataMC_ratios.csv",
    "display_plots": true,
    "save_plots": "my_K_dataMC_ratios.png"
}
```

Name of the output file for the csv file containing efficiencies, ratios, statistical and systematic uncertainties ...

Name of the output file for the plot

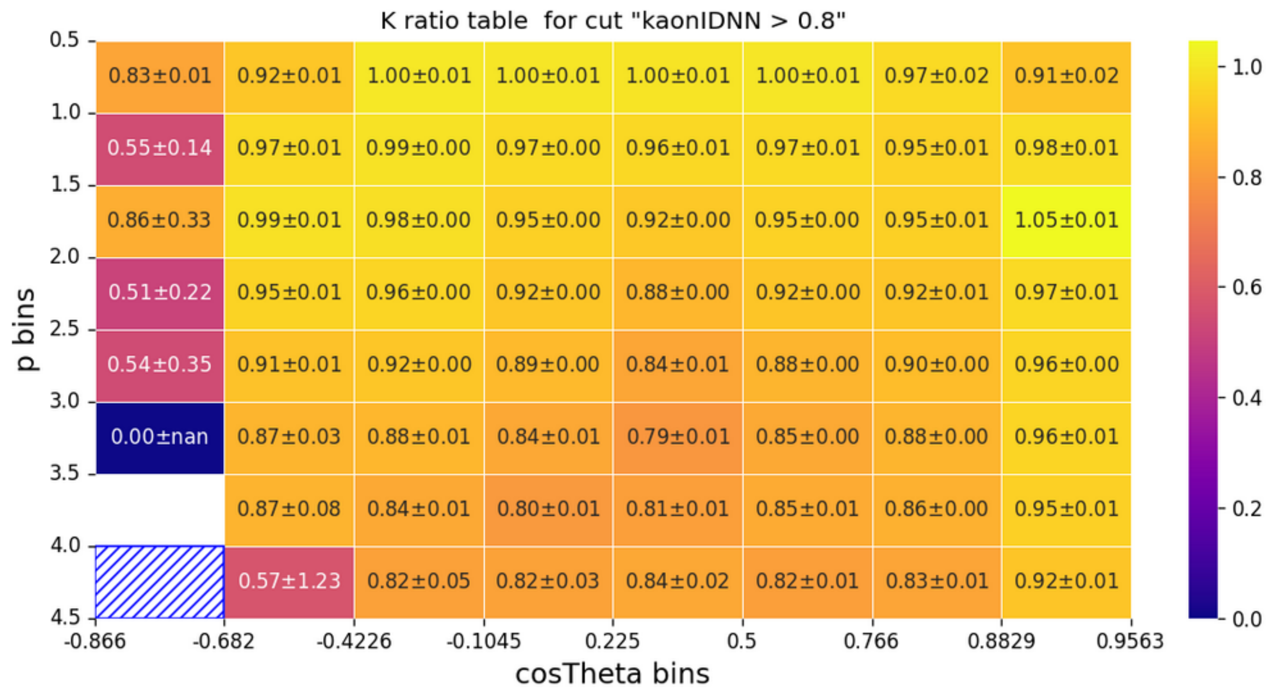
`my_K_dataMC_ratios.json`

A quick example

Run the test with:

```
python weight_table.py my_K_dataMC_ratios.json
```

and obtain something like:



plus the csv file:

```
my_K_dataMC_ratios.csv
```

Systematic uncertainties

- In the .csv file you will find the breakdown of statistical and systematic uncertainties;

- For the tag & probe samples, the systematic uncartainties are related to the background subtraction from the MC;

- For the resonance samples they come from two sources:
  1) differences in the efficiencies (in data and MC) when using two different pdf's for the background modeling;
  2) differences (evaluated in the MC) when using sWeights versus when using MCTruth;

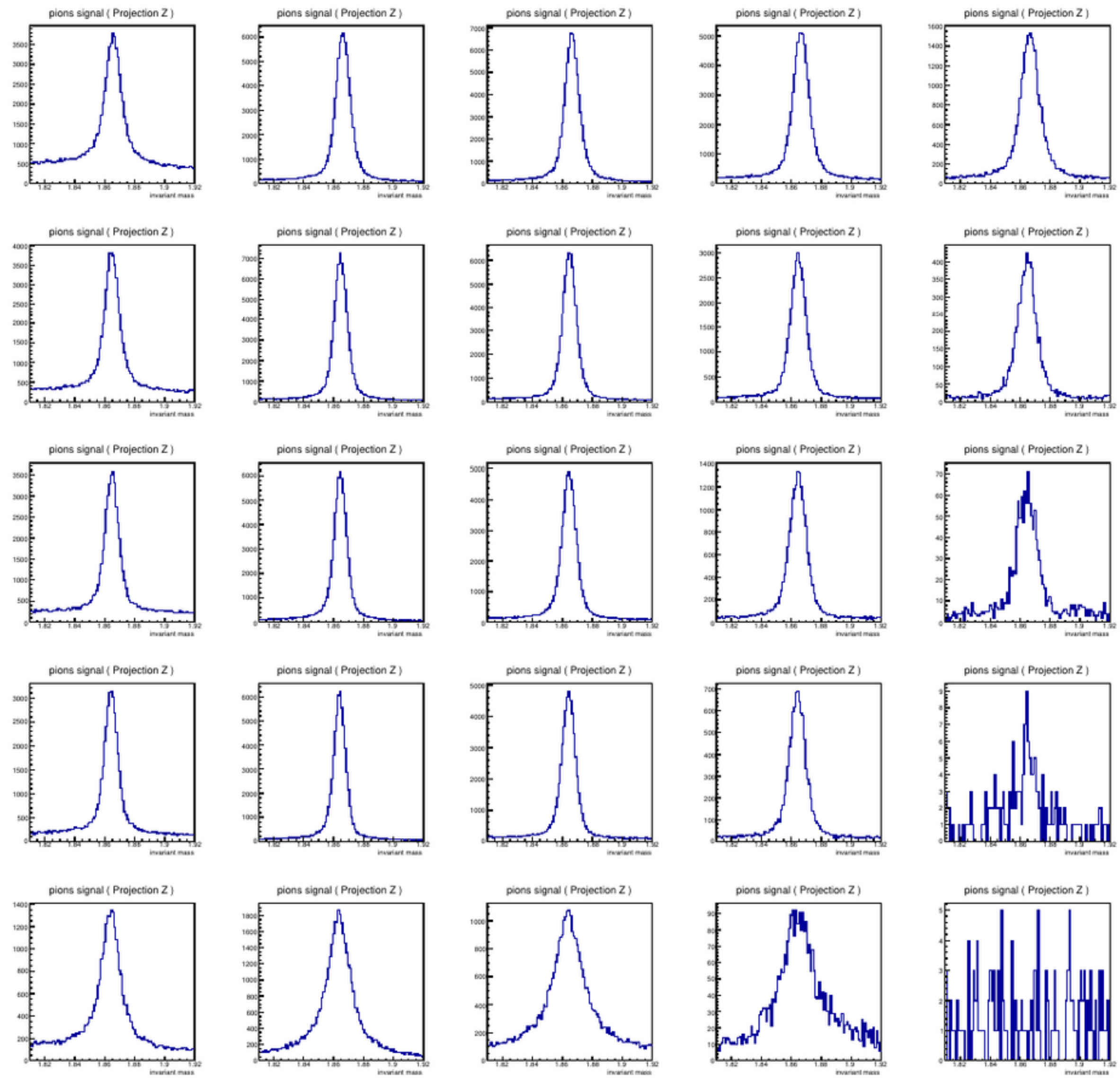
- If your systematics are large (and are a limiting factor for your analysis), it might be that the assumption that one shape (for the sWeights) fits all is showing its limitations.

Limits of the sWeights approach

- The shape of the signal and background changes based on the phase space of the tracks you are studying;

- Typically these effects are well reproduced in the MC, so you do not expect large biases;

- But the poor quality of the fit in the corner region will be reflected in the systematic uncertainties.



Marko Staric

Possible solution

A recipe to reduce the PID systematic uncertainties:

1) Make a local copy the SF ntuples that are relevant to you;
2) Reduce the ntuples to the selection that is relevant to your analysis (you can also break it into separate bins);
3) Refit the ntuples using the SF and get new (better) sWeights;
4) Create a private copy of

   `/group/belle2/dataprod/Systematics/production/prod_log_db.csv`

5) Add your refitted samples to your local db file;
6) Point the SF to your local db file, adding on your json file the line

   `"db_destination": "<path_to_your_work_area>/<your_local_db>.csv"`

   in order to use your refitted samples.

We know it's a bit complicated, but we will be happy to assist you!

Conclusions

- PID at Belle II is working well and has improved sizably from the beginning of the experiment, thanks to:
  - the dedication of detector people;
  - the ingenuity of the many people who worked on PID specific tools;
  - the feedback we received from the general users;

- If you need guidance / want to report issues, don’t be shy: attend and present at the (Neutrals and) PID meetings!

Backup Slides

FAQ: recommended PID cut(s)

- **It depends on your analysis**, we cannot give a one-fits-all recommendation;

- Very often one optimizes the selection using the figure of merit:

$$\frac{S}{\sqrt{S+B}}$$

- But this is not the only option:
  - you might have a background source that will bring in a large systematic uncertainty (because it is not well known/simulated, …) in this case you want to cut harder and increase your S/N ratio;
  - your background is harmless, so you might live with relatively low S/N ratio and enjoy a larger signal efficiency.

FAQ: bremsstrahlung recovery for e's

- Suppose you want / have to apply bremsstrahlung recovery to the electron(s) in your analysis;
- **Do you apply the PID corrections before or after the recovery?**

$$p(e) = p'(e) + p(\gamma)$$

Four-momentum of the particle you care about

Four-momentum of the particle that interacts with our detector

Four-momentum of the bremsstrahlung photon

- For us, what matters (and what we provide corrections for) is p'(e);
- When you apply bremsstrahlung recovery, in general the electron can jump from one (p, θ) bin to another, in a way that we (PID people) cannot predict;

- **Answer**: apply PID corrections **before** bremsstrahlung recovery!

FAQ: nCDCHits cut

- This is a bit of a controversial topic…

  - Tracking recommends *against* applying an nCDCHits cut, because it will make the charge asymmetry (and related systematics) higher;

  - PID is *in favor* of it, because we have seen that it makes PID better;

- Once again, we cannot give a universal recommendation: it depends on your analysis:
  - what is the impact on your efficiency if you do apply the cut?
  - will the cut reduce other systematics?
  - are you more worried about tracking systematics or PID systematics?