Matching QA developments

MFT Software & Physics Meeting - October 20th, 2025

andrea.ferrero@cern.ch

Outline

- Draft task for matching QA:
  - Purities & efficiencies
  - Candidates ranking
  - Alternative chi2 matching functions and ML models
- Tagging of matching candidates in real data
- Possible improvements for the MCH tracks extrapolation to the matching plane

Definitions

Metrics for matching characterization (MC)

$$\textbf{Pairing purity}: \quad \text{purity} = \frac{N_{\text{GM true}}}{N_{\text{GM rec}}}.$$

"How many correct matches do we have"

$$\textbf{Pairing efficiency}: \quad \text{pairing eff} = \frac{N_{\text{GM rec (pairable)}}}{N_{\text{pairable}}}.$$

"how often we **attempt** a match **when it's possible to be correct.**"

$$\textbf{True efficiency}: \quad \text{true eff} = \frac{N_{\text{GM true (pairable)}}}{N_{\text{pairable}}}.$$

correct picks among pairable tracks

$$\textbf{Fake efficiency}: \quad \text{fake eff} = \frac{N_{\text{GM fake (pairable)}}}{N_{\text{pairable}}}.$$

wrong picks among pairable tracks

From S.A. Barday (2025-09-08)

Definition of “Good Matches”

- Global Muon track selection:
  - MCH track chi2 < 5
  - MCH p_T > 0.7 GeV/c
  - $-3.6 < eta_{MCH} < -2.5$
  - $17.6 < R_{ABS} < 89.5$
  - $sigma_{pDCA} < 6$

- MFT track selection:
  - MFT track chi2 < 999
  - MFT nClusters >= 7

- Matched track selection:
  - Match score > 0.5 (equiv. to chi2 < 50)

- Proposed conversion from chi2 to score value: $score = \frac{1}{\chi^2/_{50}+1}$
  - Score=1 when chi2=0
  - Score = 0.5 when chi2=50

Definition of “pairable” tracks

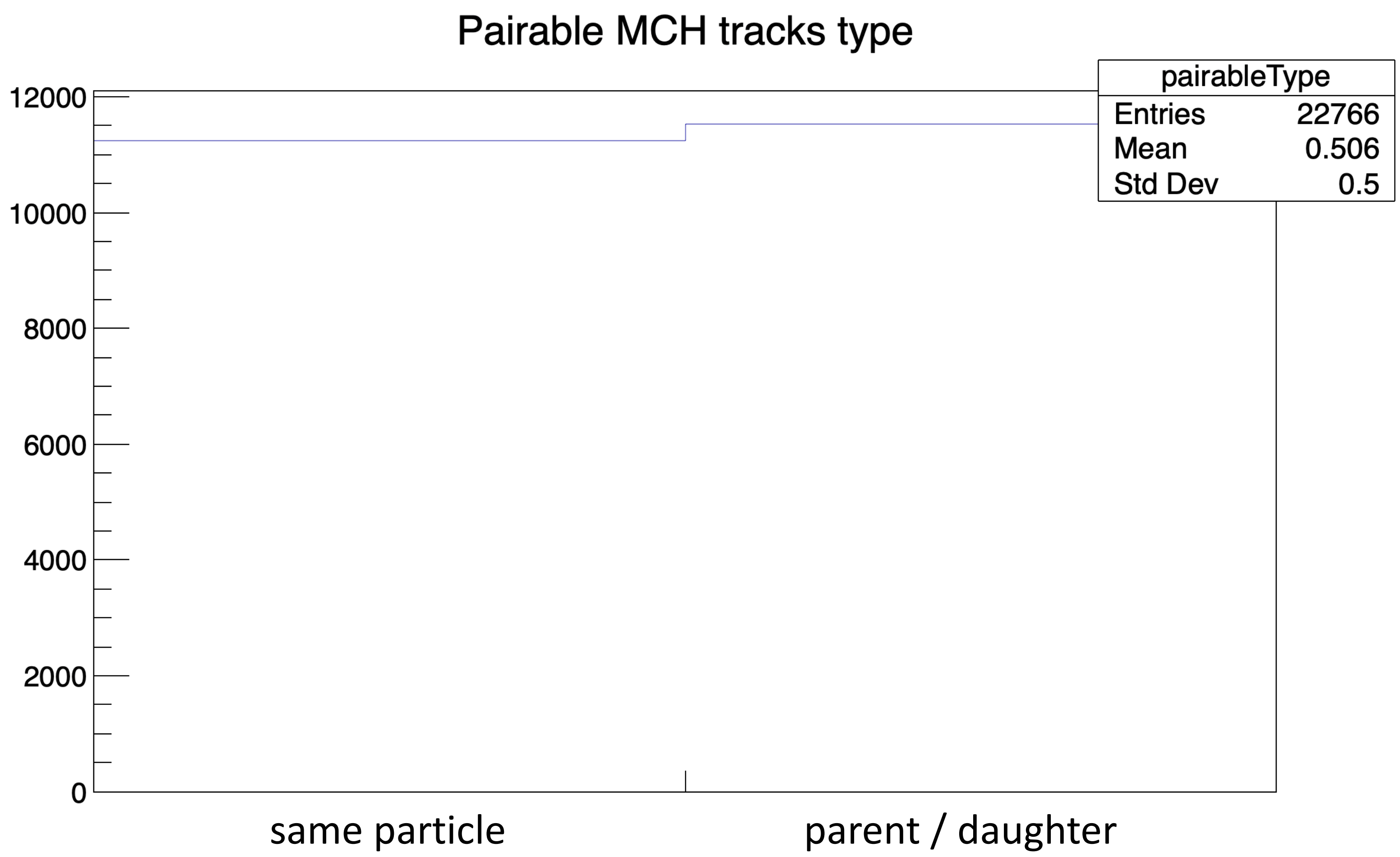
- The “pairable” tracks identify the pairs of MFT-MCH tracks that are along the same particle trajectory
- The pairable tracks are defined as follows:
  - The MFT and MCH standalone tracks are associated to the **same MC particle**

    OR

    The MFT track is associated to the **parent particle** of the MCH track
  - The MFT and MCH tracks both pass the quality cuts (**“good” tracks**)
- In the ideal case, we expect to have a reconstructed match for each of the pairable tracks

"Pairable" tracks and particle decays

- In about half of the cases, the particles associated to the MFT and MCH tracks are different:



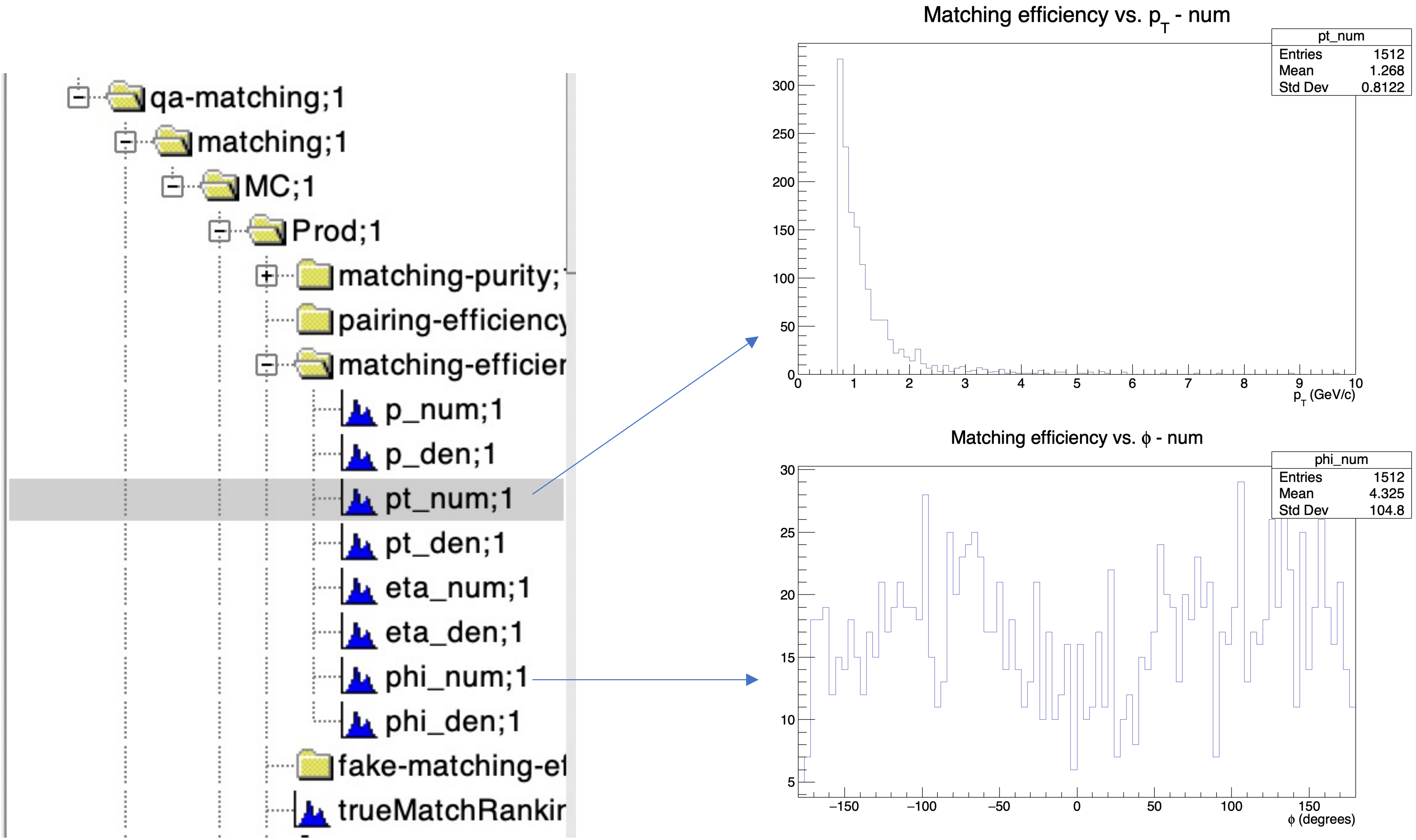
Matching QA Task

What the matching QA task provides (so far)

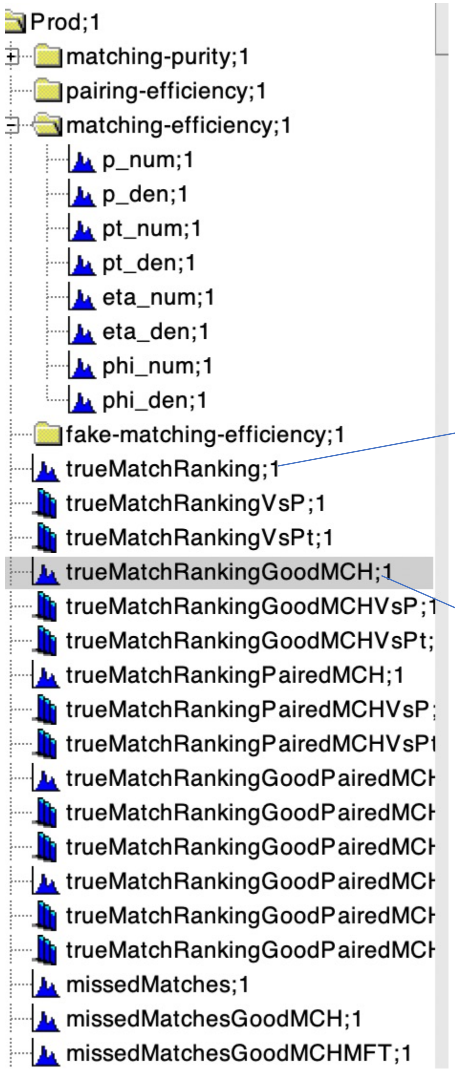
- Matching performance characterization:
  - Matching and pairing efficiency, matching purity
  - Plotted as function of MCH momentum, pT, eta and phi
- Correctness of the matching candidate ranking
- Matching score distribution for true and fake matches

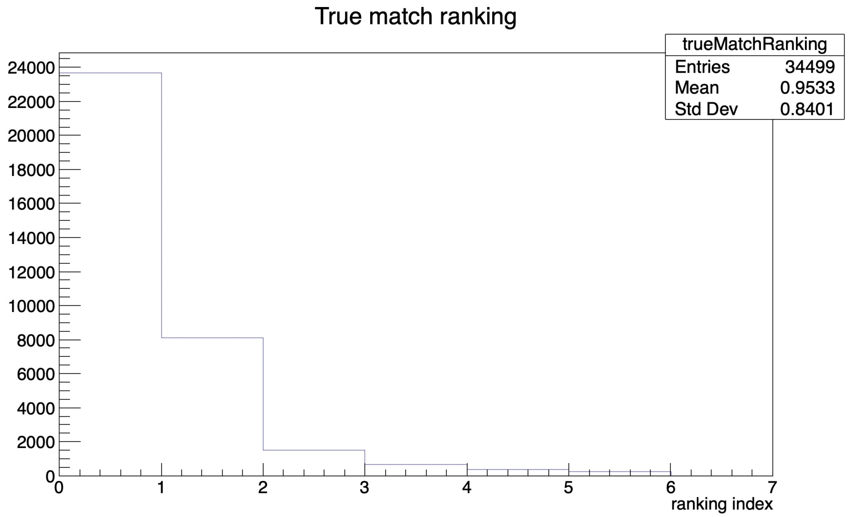
- Can load and run several ML models via the `MlResponseMFTMuonMatch` class (M. Coquet) as well as several chi2-based matching functions
  - Matching plane and minimum score value can be defined for each model and function
  - The same matching characterization is performed for each model and function

Matching characterization: efficiency & purity

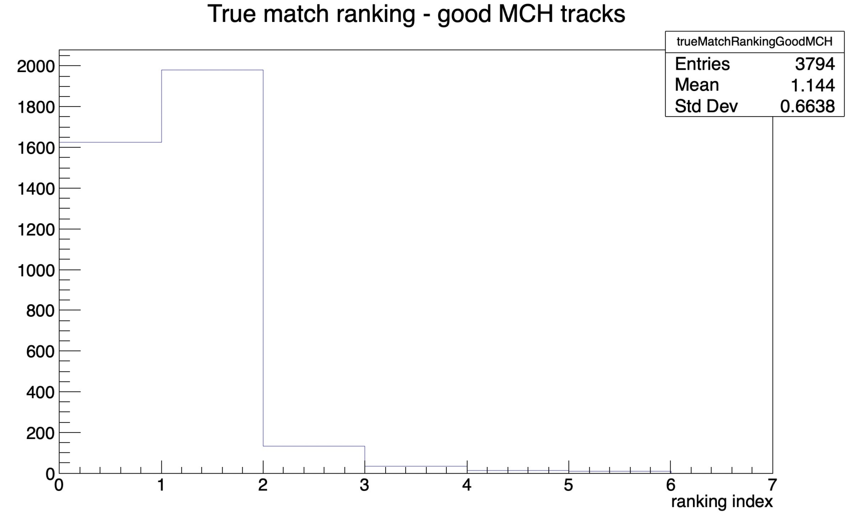


Matching characterization: candidates ranking



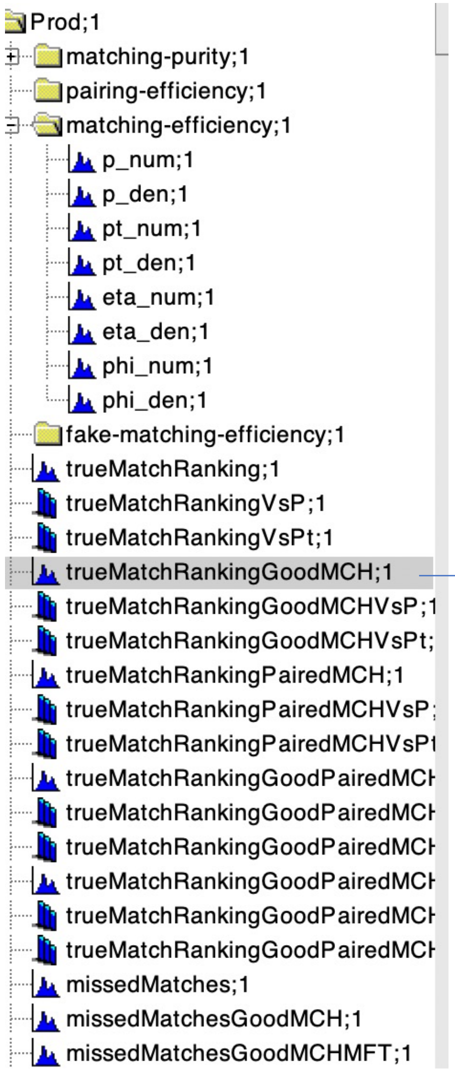


All matched tracks

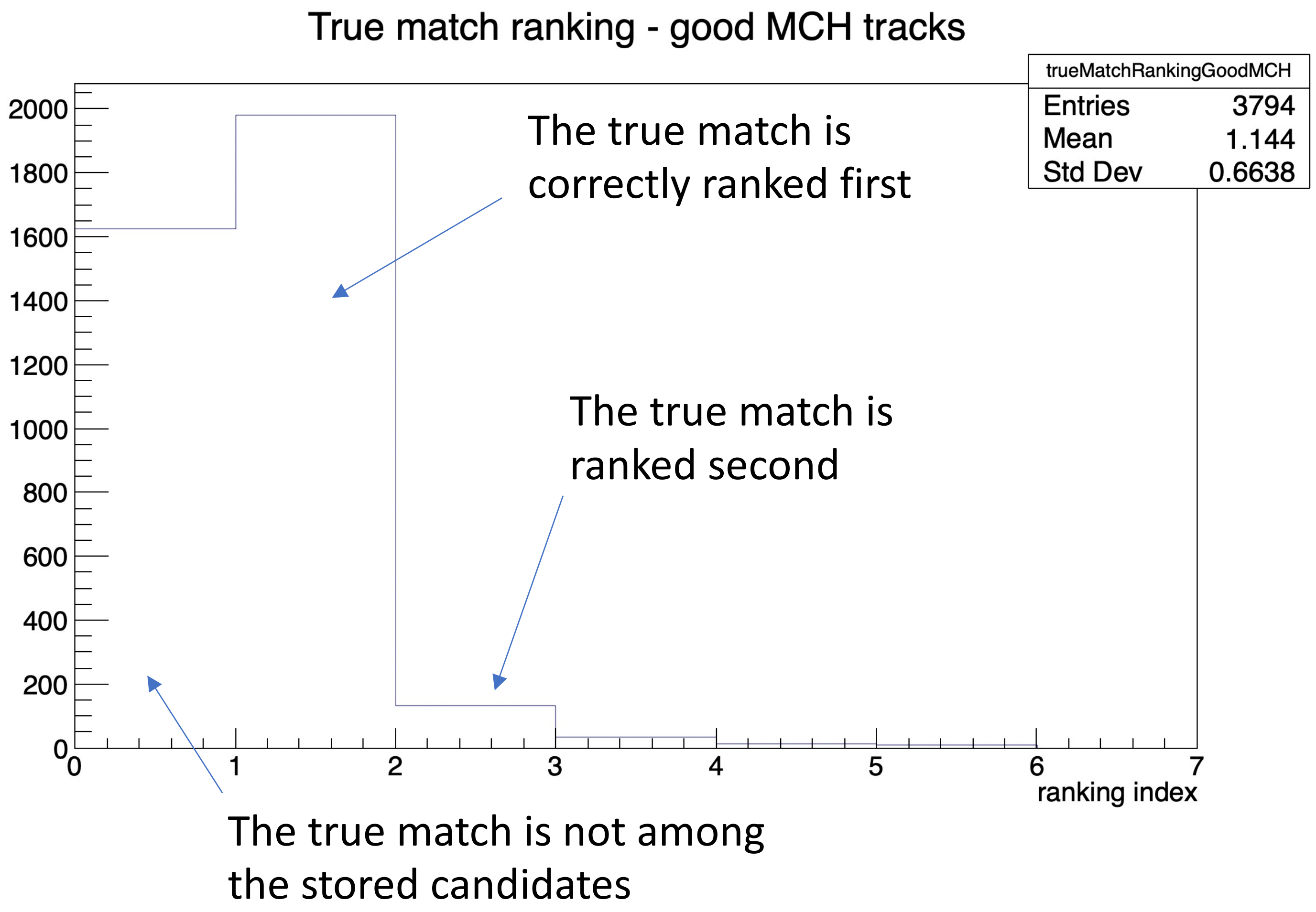


Matches with good MCH tracks

Matching characterization: candidates ranking

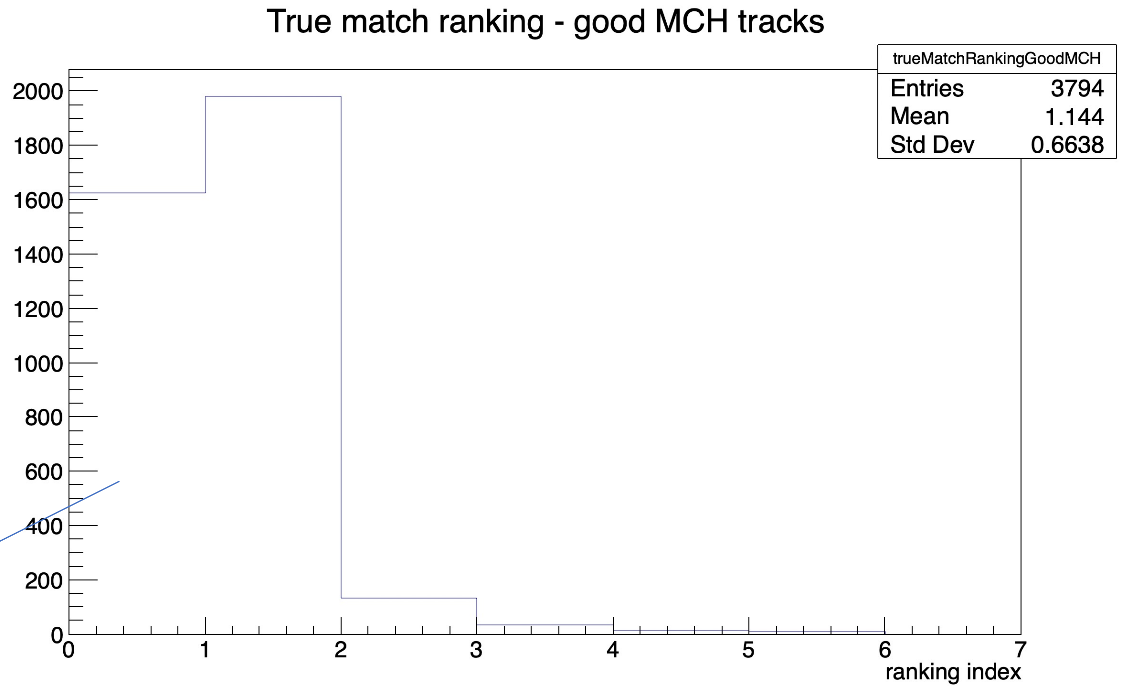


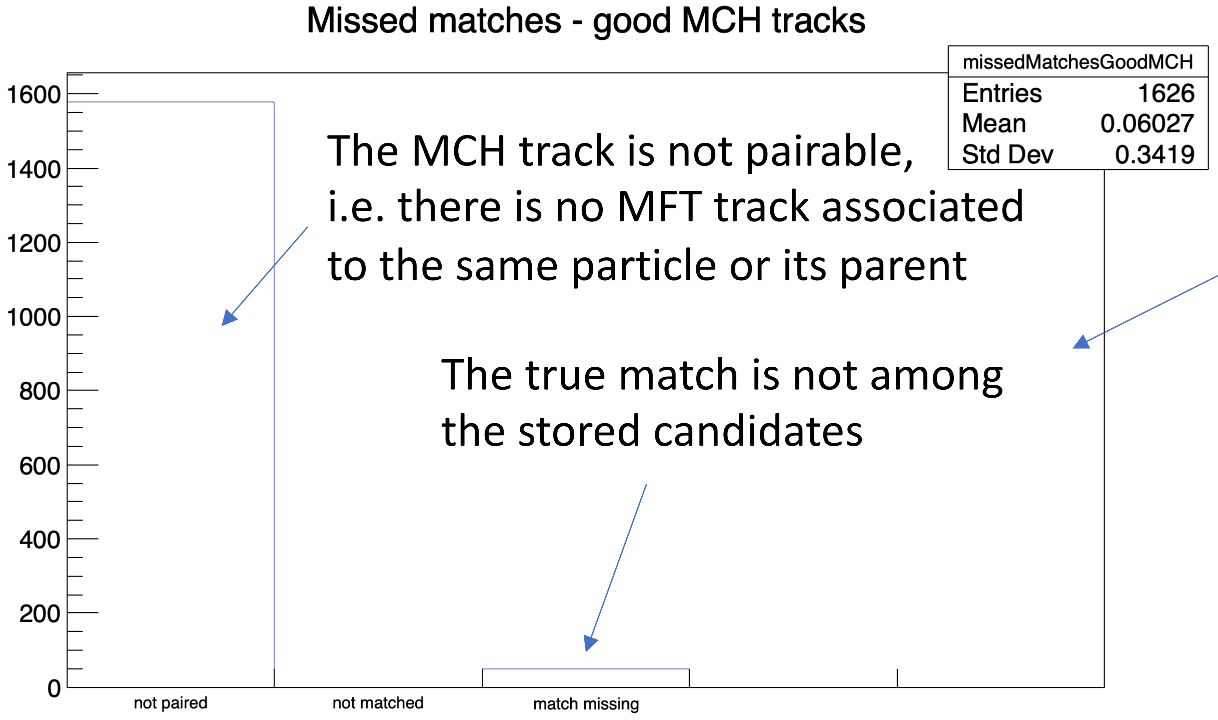
- Ranking of the true match among the stored candidates



Matching characterization: candidates ranking

- Details of the missing true matches

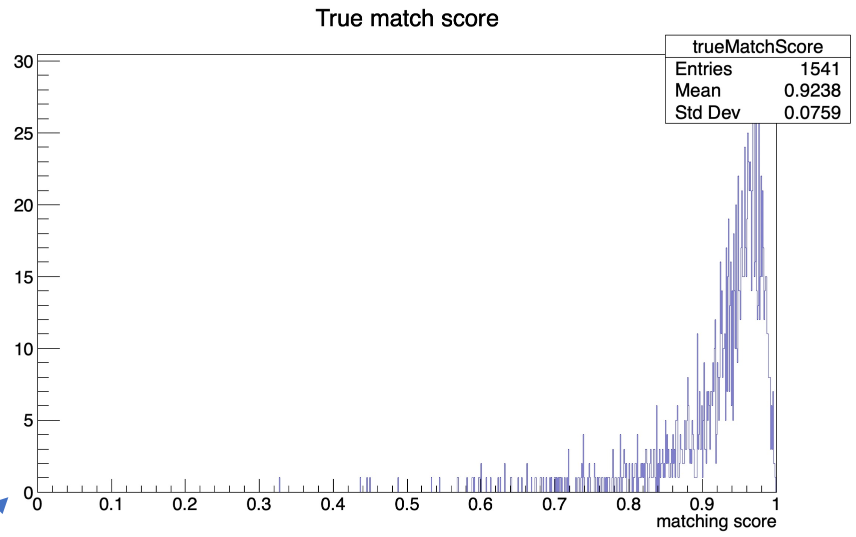




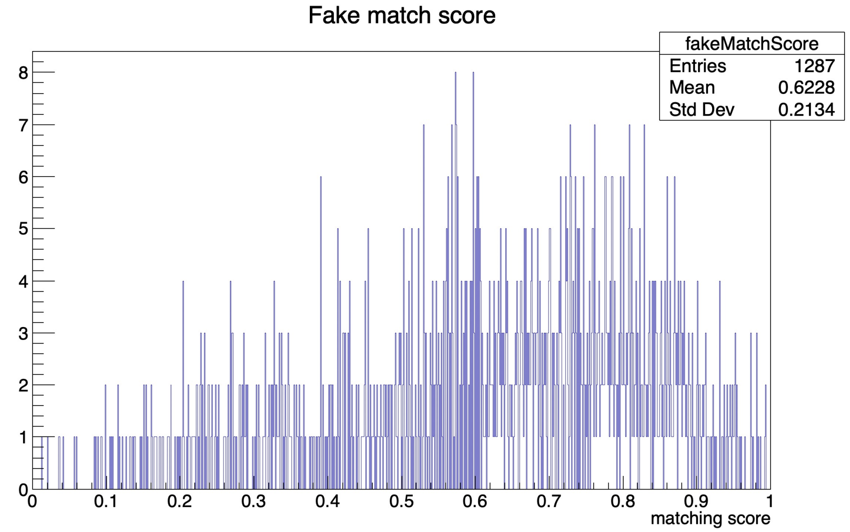
The MCH track is not pairable, i.e. there is no MFT track associated to the same particle or its parent

The true match is not among the stored candidates

Matching characterization: match score

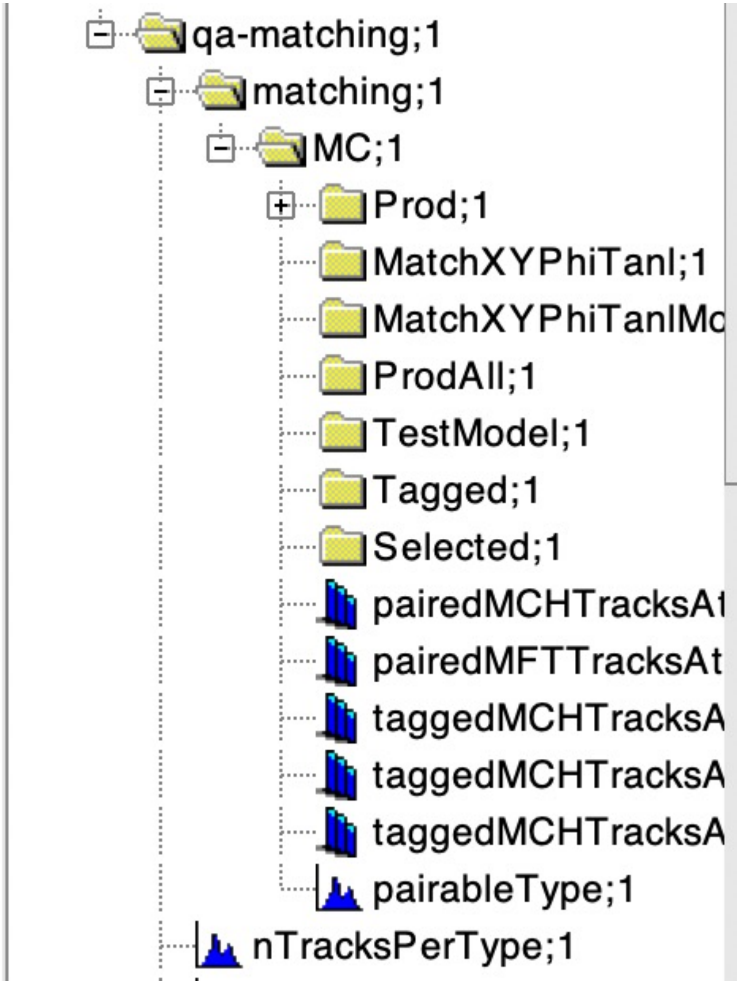


Score distribution for true matches



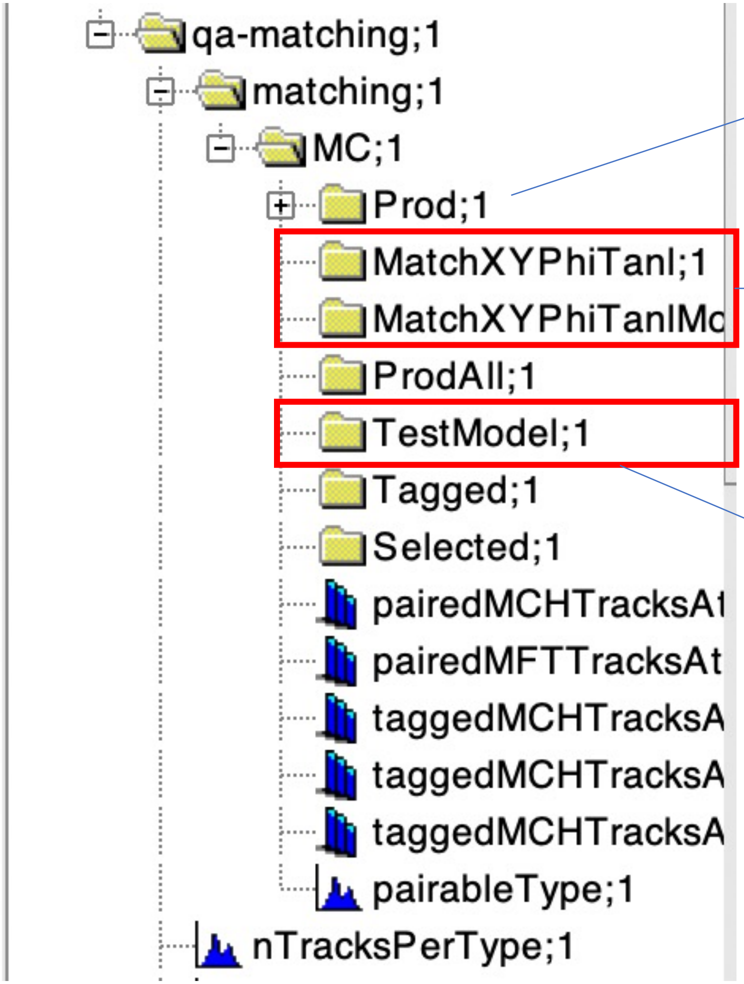
Score distribution for fake matches

Custom functions and ML models



The characterization is repeated for the production matches and the output of the custom ML models or chi2-based functions

Custom functions and ML models



Matched candidates in the AO2D

Chi2-based functions

ML model loaded via `MlResponseMFTMuonMatch`

Status of matching QA task

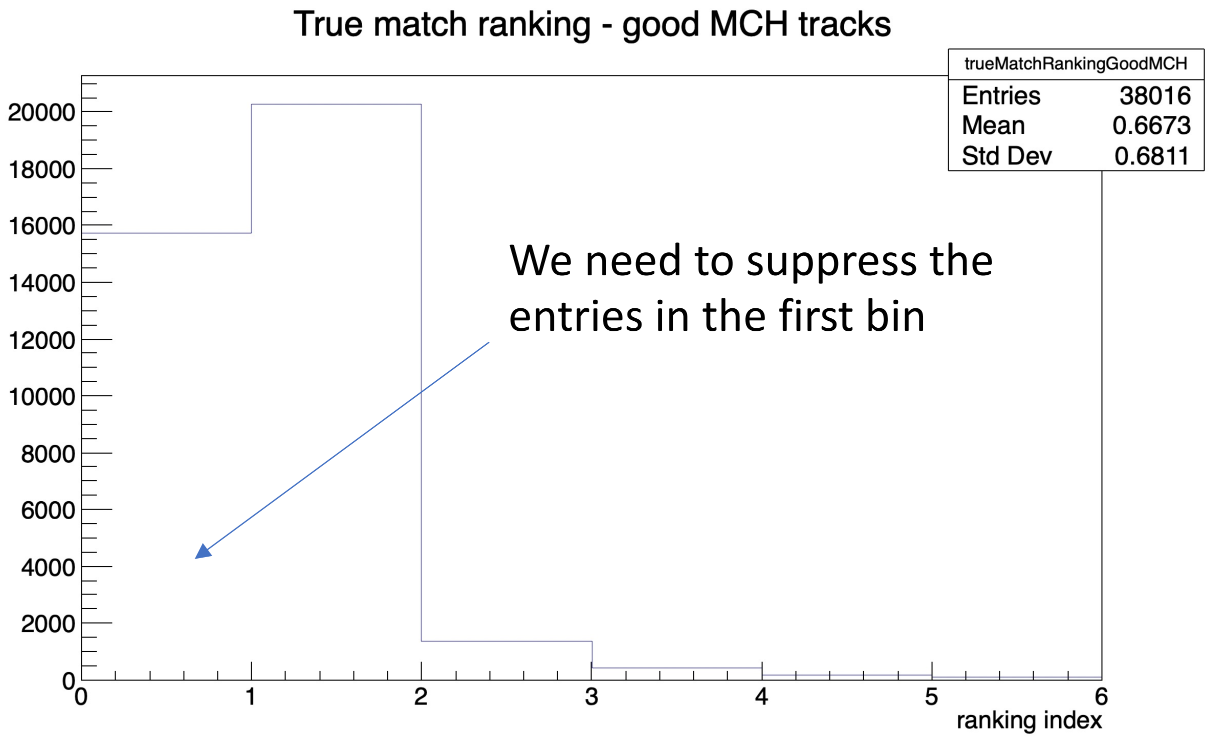
- A draft version of the task is available on GitHub:
  - https://github.com/AliceO2Group/O2Physics/pull/13245
- The code still needs a little bit of polishing, but is already usable
- Currently tested only on O-O simulations (LHC25h3 production)
- ML interface only tested with the dummy ONNX model provided by Maurice
  - We urgently need to migrate existing models to ONNX!

- Can be used as starting point for systematic studies of matching performances
- Guarantees a uniform and documented definition of quality observables (purities, efficiencies, etc…) and matching computation (track extrapolation, chi2 functions, ML models, etc…)

Tagging of muon tracks

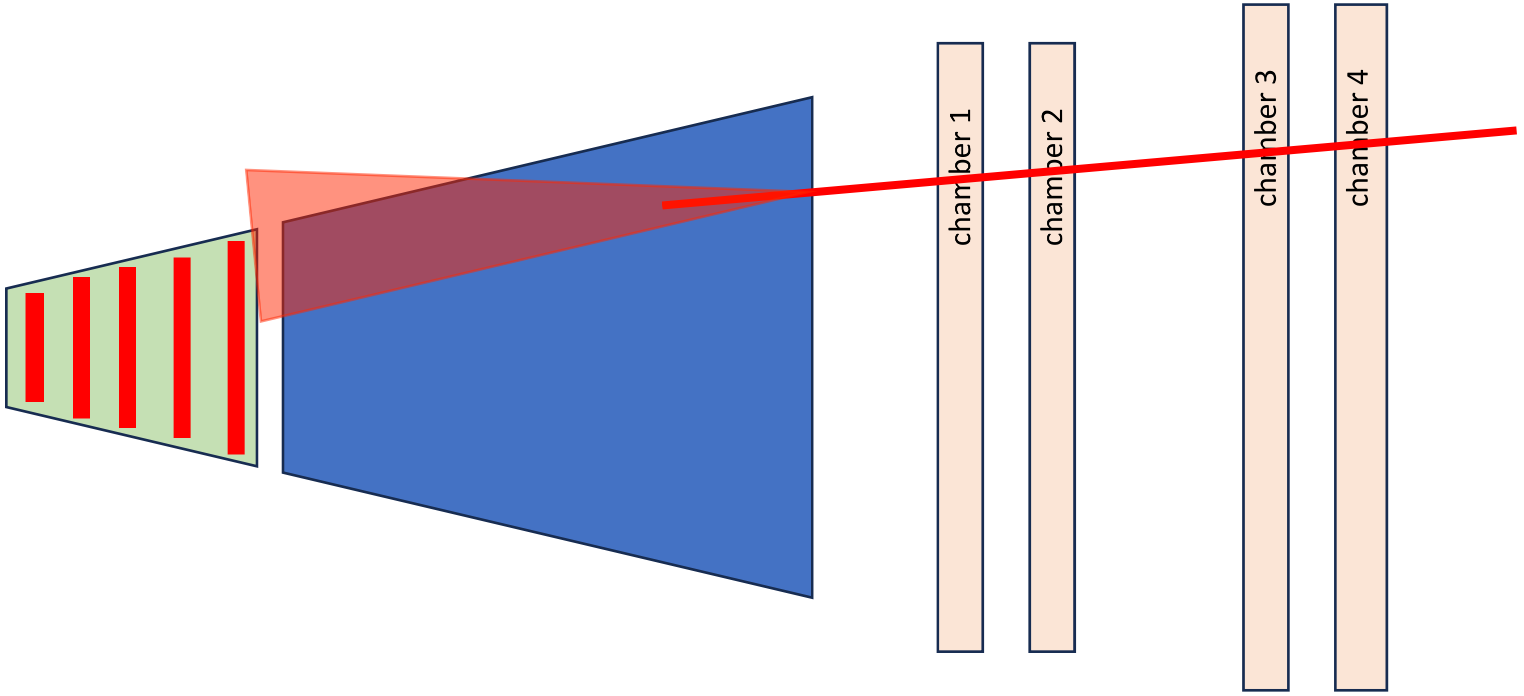
Tagging MUON tracks for matching characterization

- The matching characterization in real data requires to select a "clean" sample of matching candidates
- We need to identify MCH tracks that are very likely paired to an MFT track
- Here is the proposed strategy:
  1. Apply strict quality cuts to the MCH tracks
     - Stricter eta, chi2 or pDCA cuts?
  2. Extrapolate the MCH track to the front and back of MFT, and check that the track in both the MFT active areas



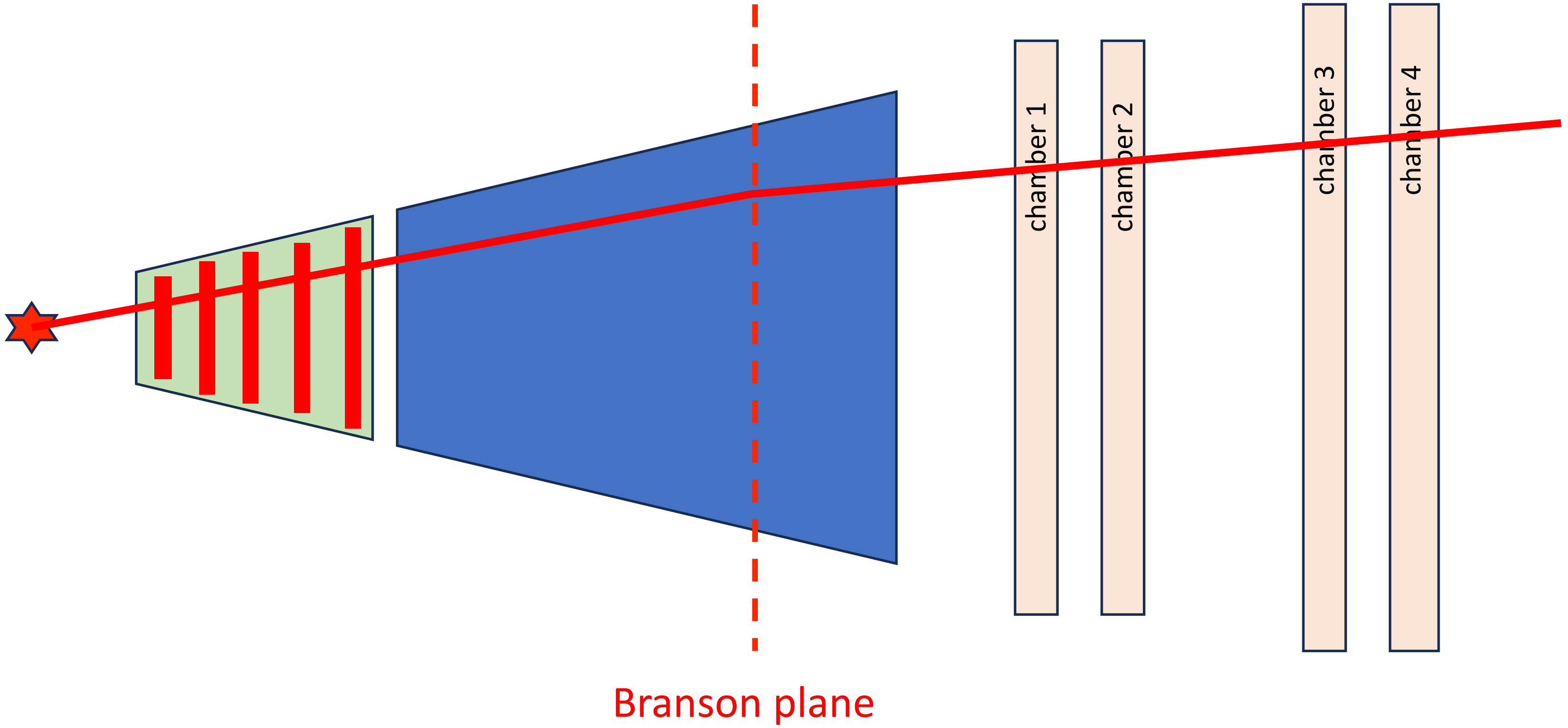
Improved MCH extrapolation to the MFT

- The standard MCH track extrapolation gives a rather imprecise estimation of the track trajectory through the MFT planes, particularly at low momenta, due to the multiple scattering



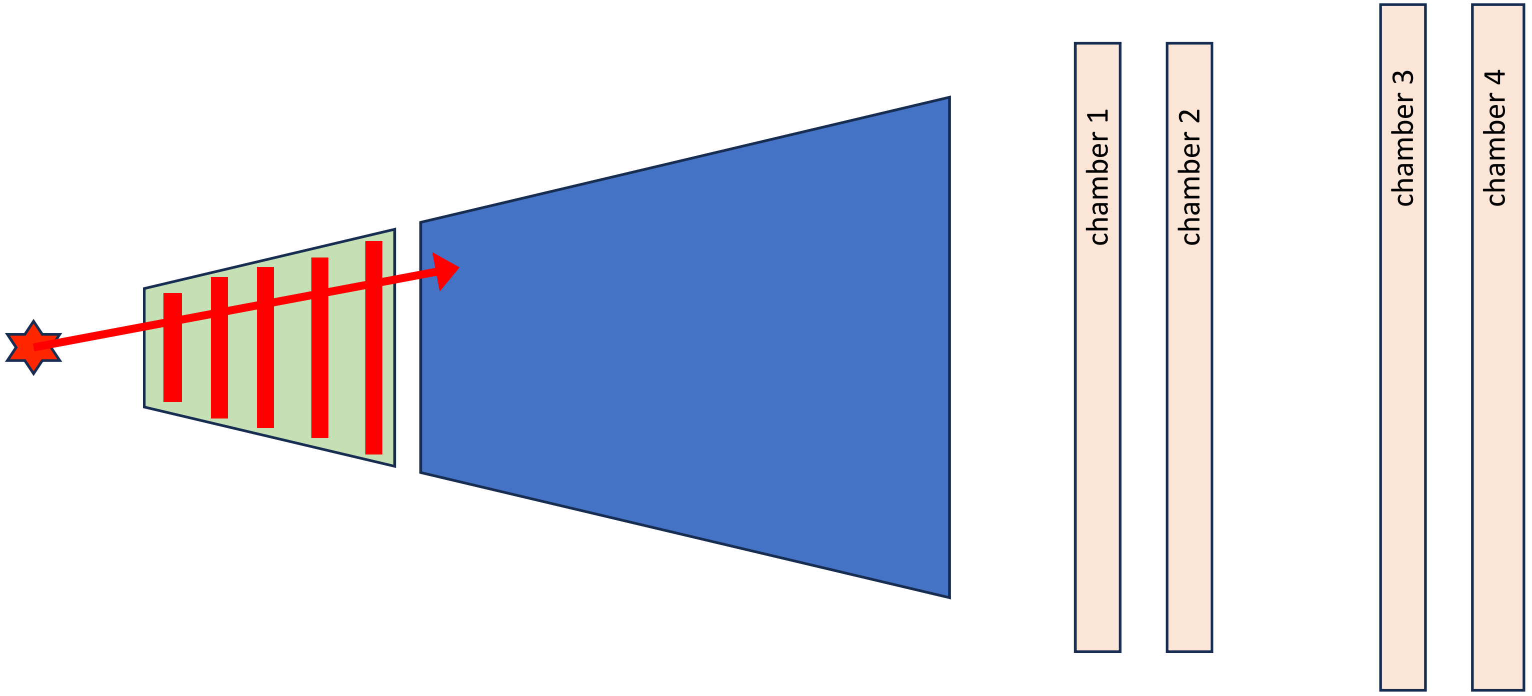
Improved MCH extrapolation to the MFT

- What if we include the vertex constraint in the track extrapolation?
- The extrapolation to the collision point models the MS as a single kick at the Branson plane
- The collision point provides a strong constraint to the track direction upstream of the absorber



Improved MCH extrapolation to the MFT

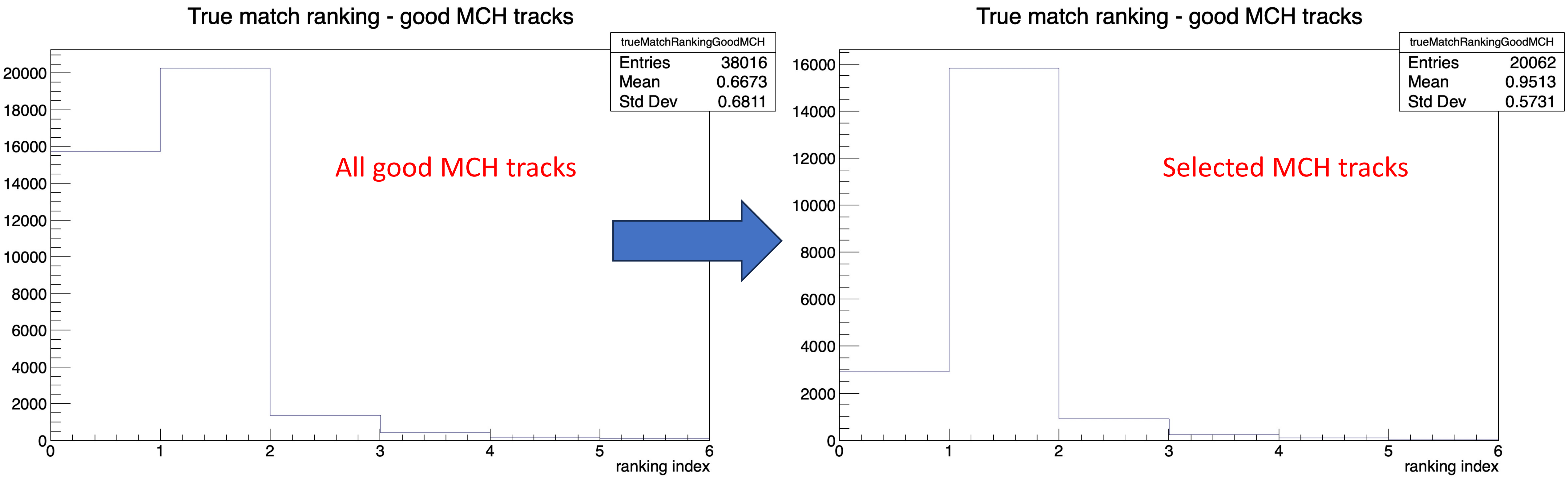
- What if we include the vertex constraint in the track extrapolation?
- The track at the vertex can then be extrapolated back to each MFT, to see if it crosses the active area of the corresponding discs



Branson plane

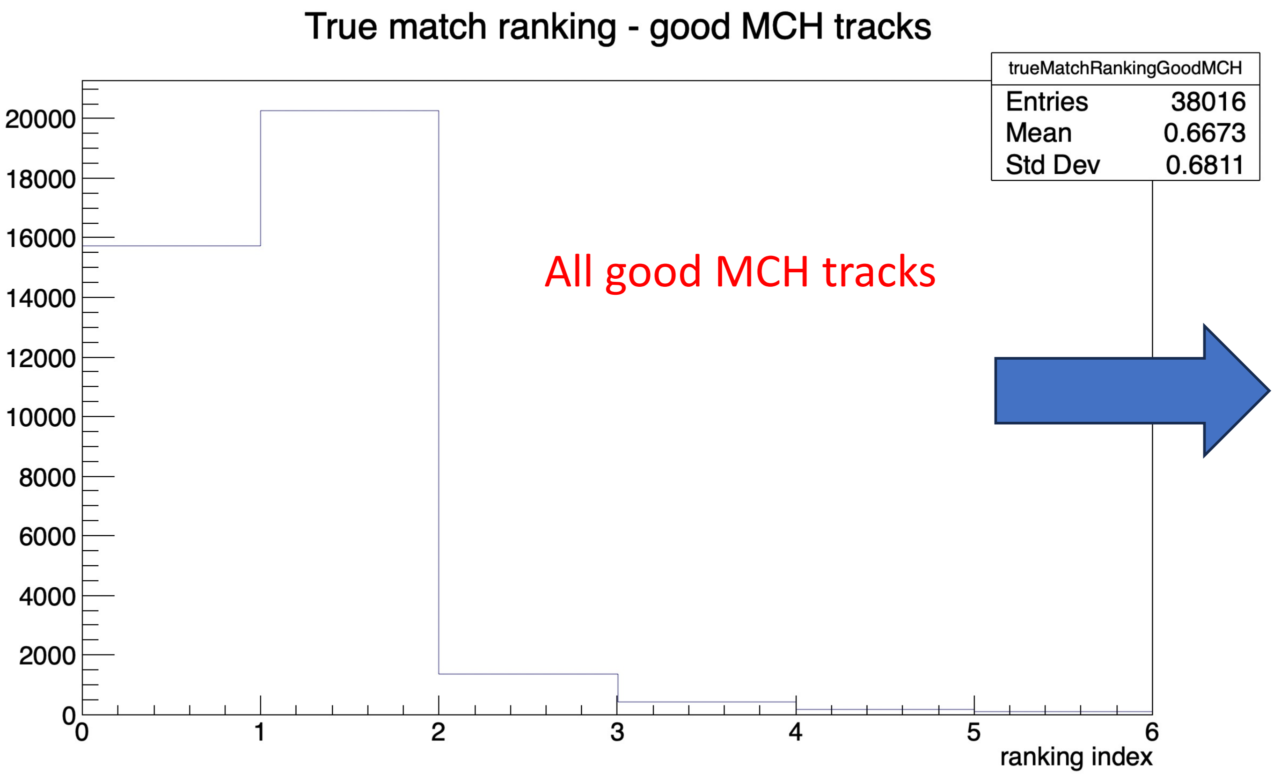
Tracks selection: does it work?

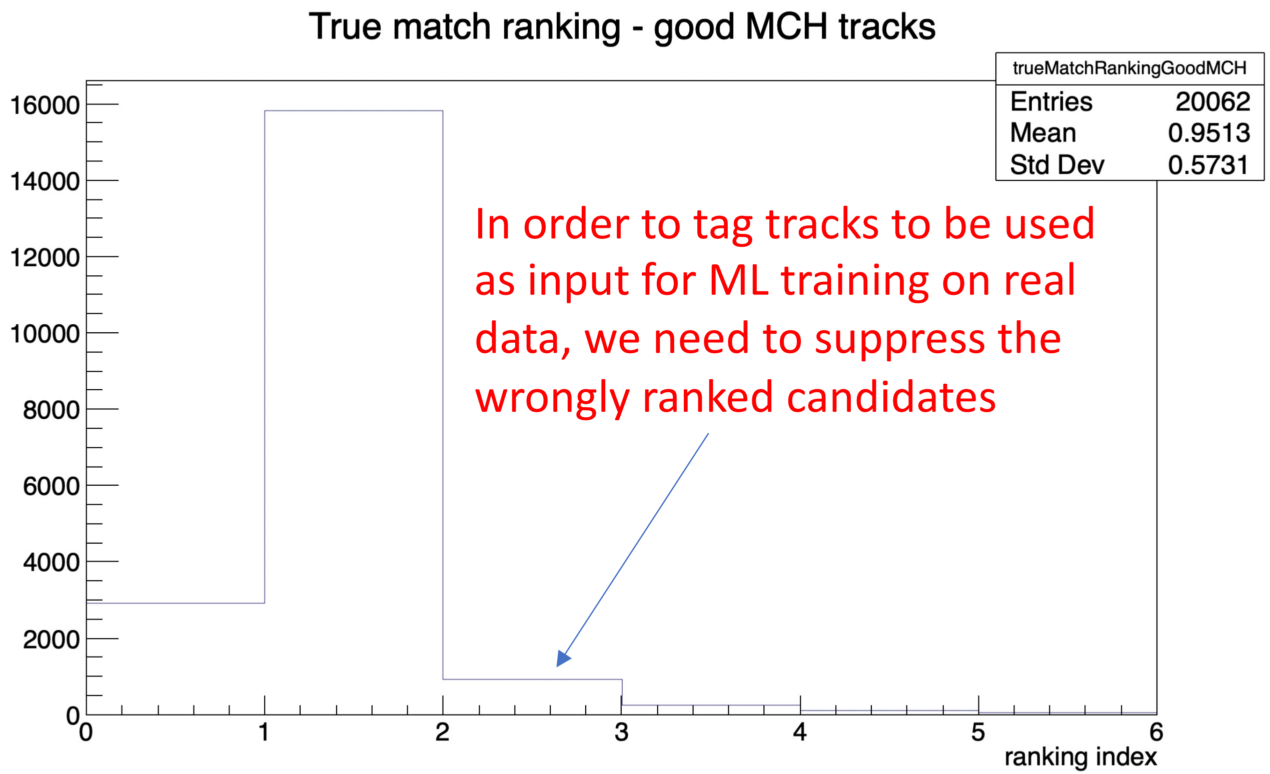
- This is how the proposed tracks selection allows to suppress the fraction of non-paired candidates
- Quite good already, but there is probably room for improvement…



Tracks selection: does it work?

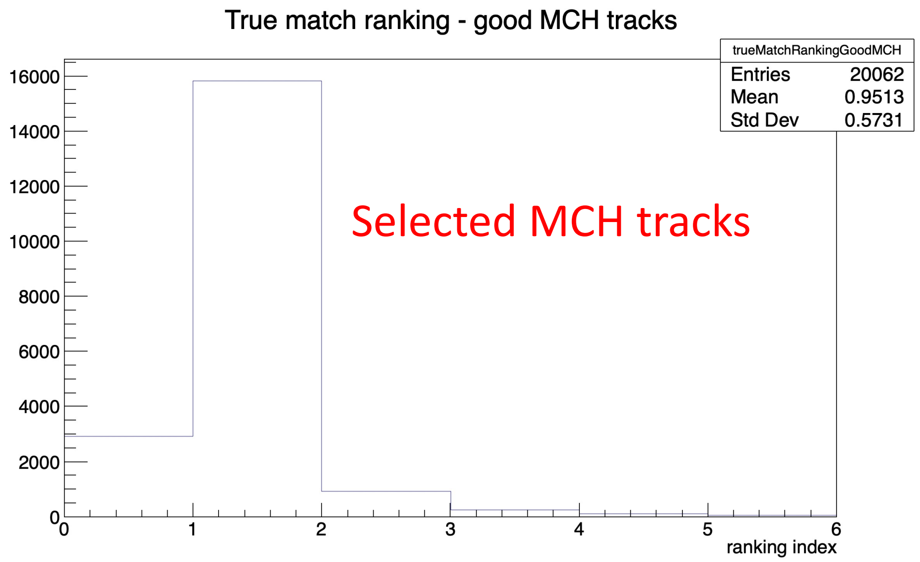
- This is how the proposed tracks selection allows to suppress the fraction of non-paired candidates
- Quite good already, but there is probably room for improvement…

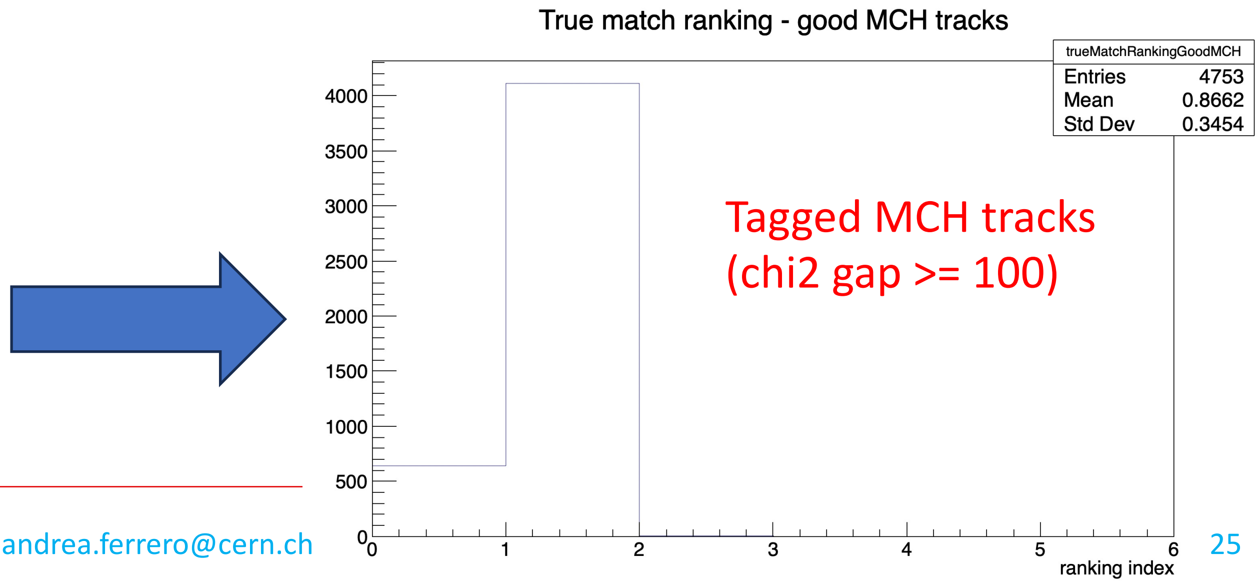




From selection to tagging

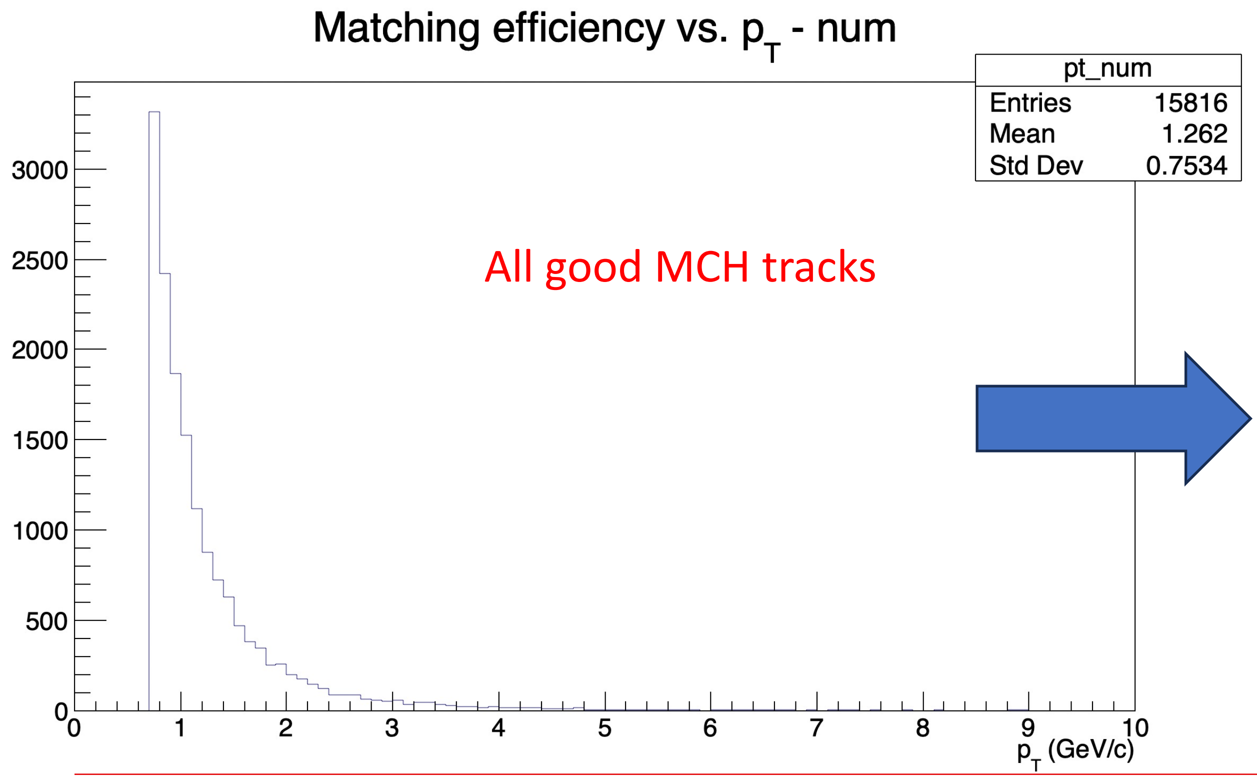
- The main reason for incorrect ranking of candidates is the large extrapolation errors at low momentum, combined with multiple MFT tracks close to each other
- What if we request a large chi2 gap between the leading candidate and the second one?
- This should be equivalent to selecting events where there is a large separation between MFT tracks. Since we are only interested in tagging matching candidates that are “very likely true”, this should be still ok for the purpose of training ML models on true matches in real data

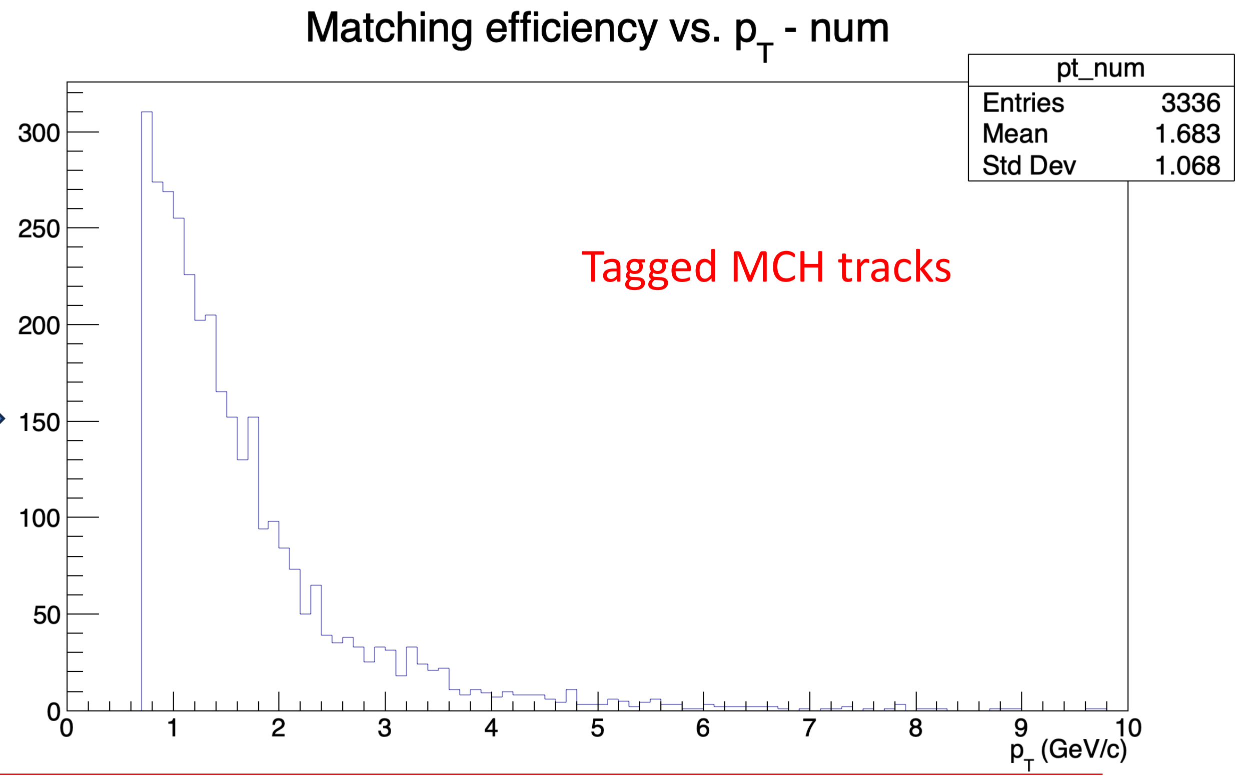




Kinematics of tagged muons

- The tagging with the chi2 gap method introduces some distortion of the kinematics, mainly suppressing low-pT tracks.
- Is this a serious problem for the ML training?

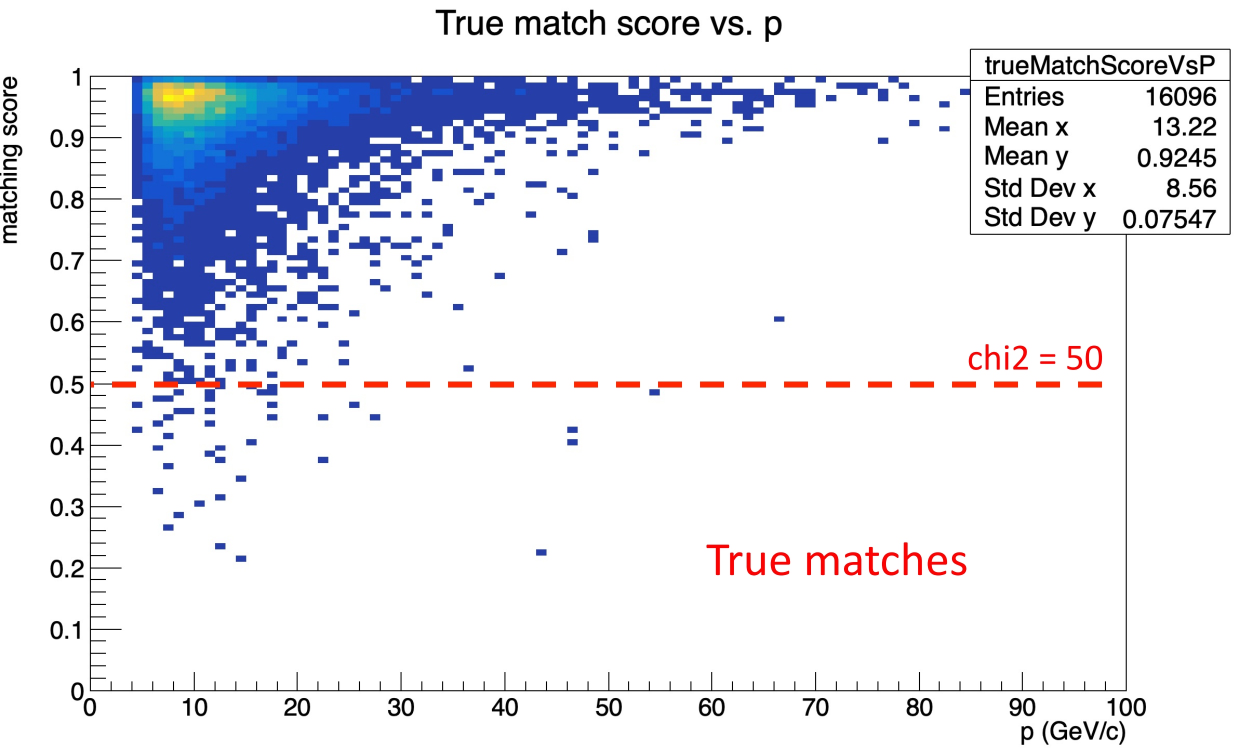


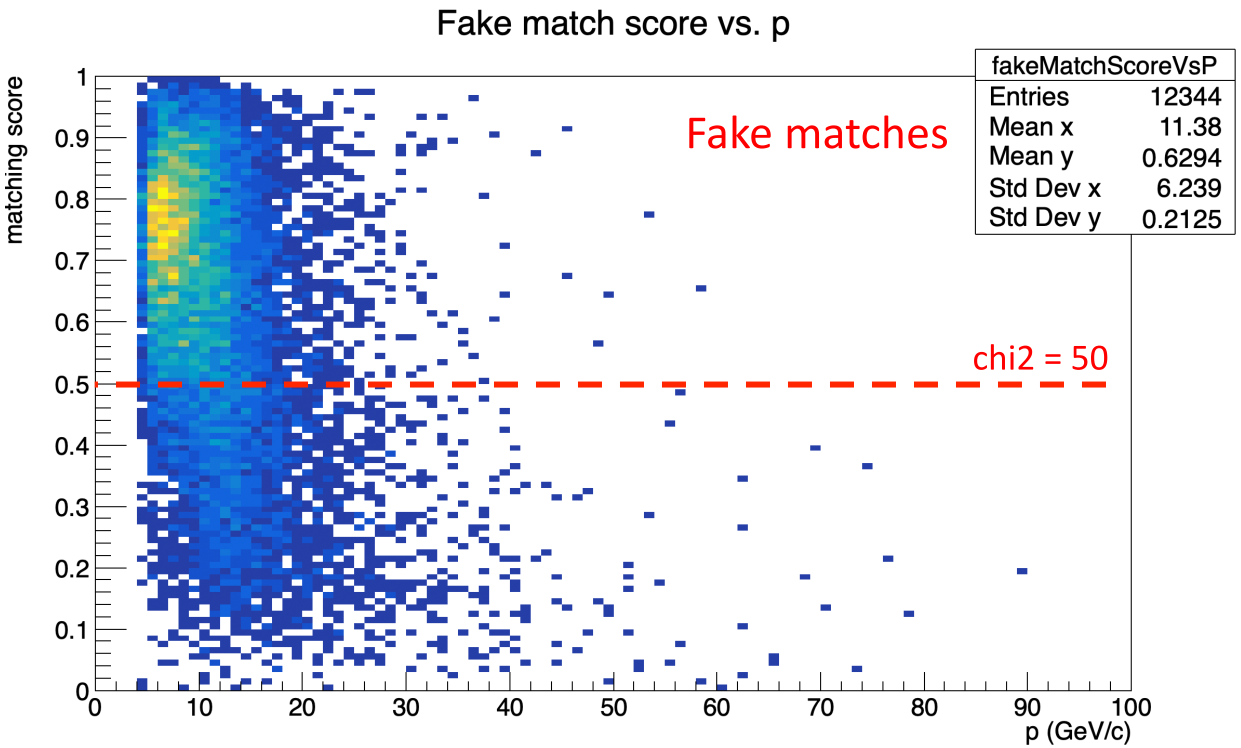


Track extrapolation to the matching plane

Match score distribution vs. momentum

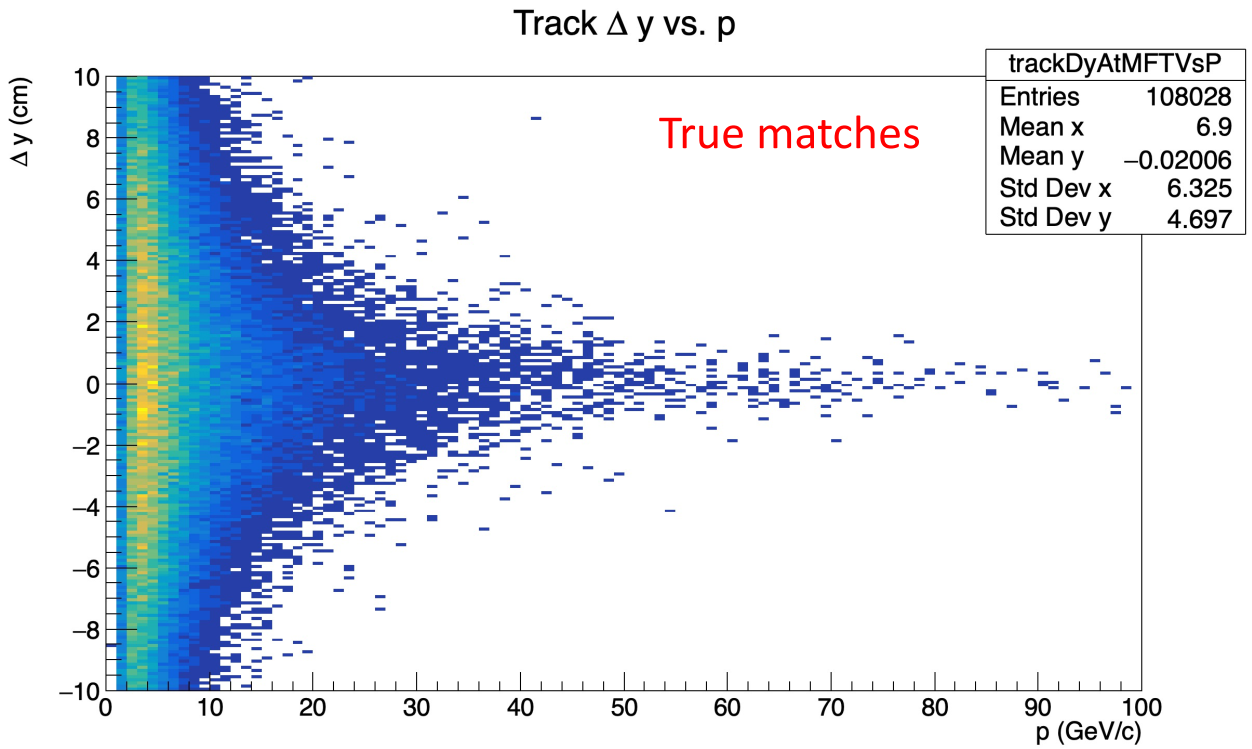
- The match score separation between true and fake matches is rather weak at low momentum, mostly due to the large extrapolation errors for the MCH tracks
- The plots below show the score of the leading candidate, for true and fake matches separately

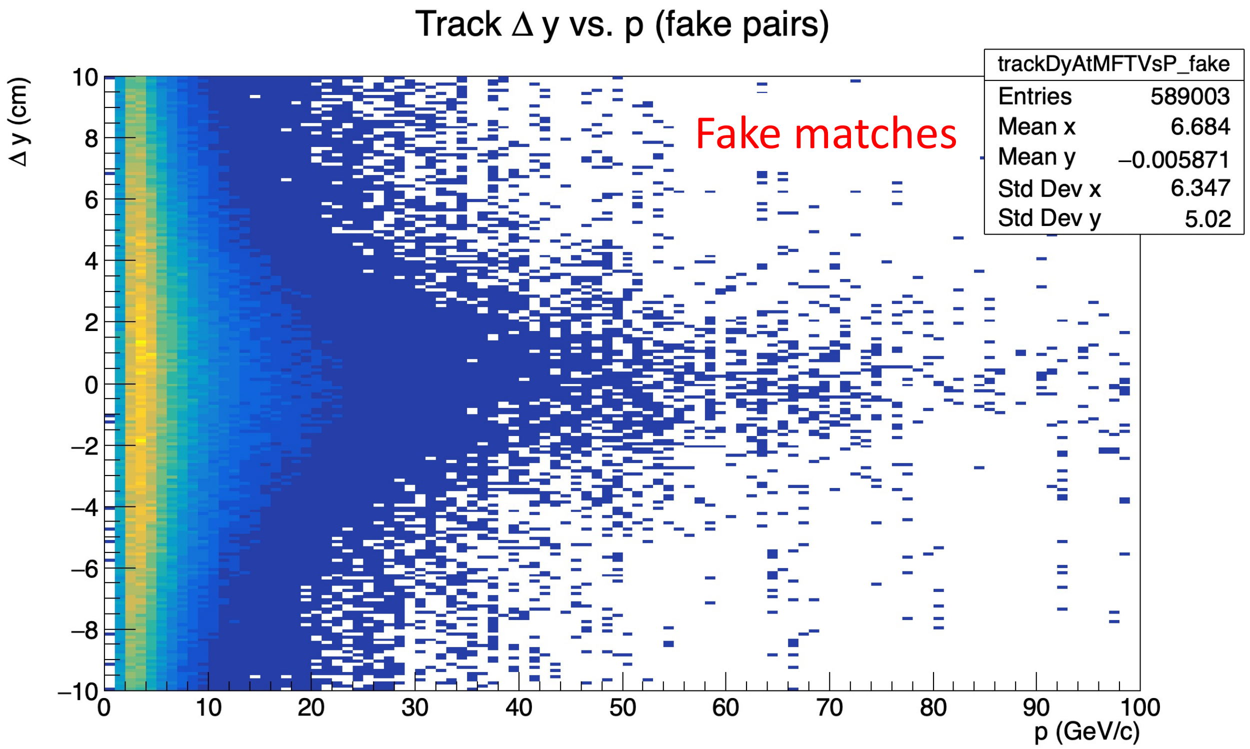




Track extrapolation to matching plane

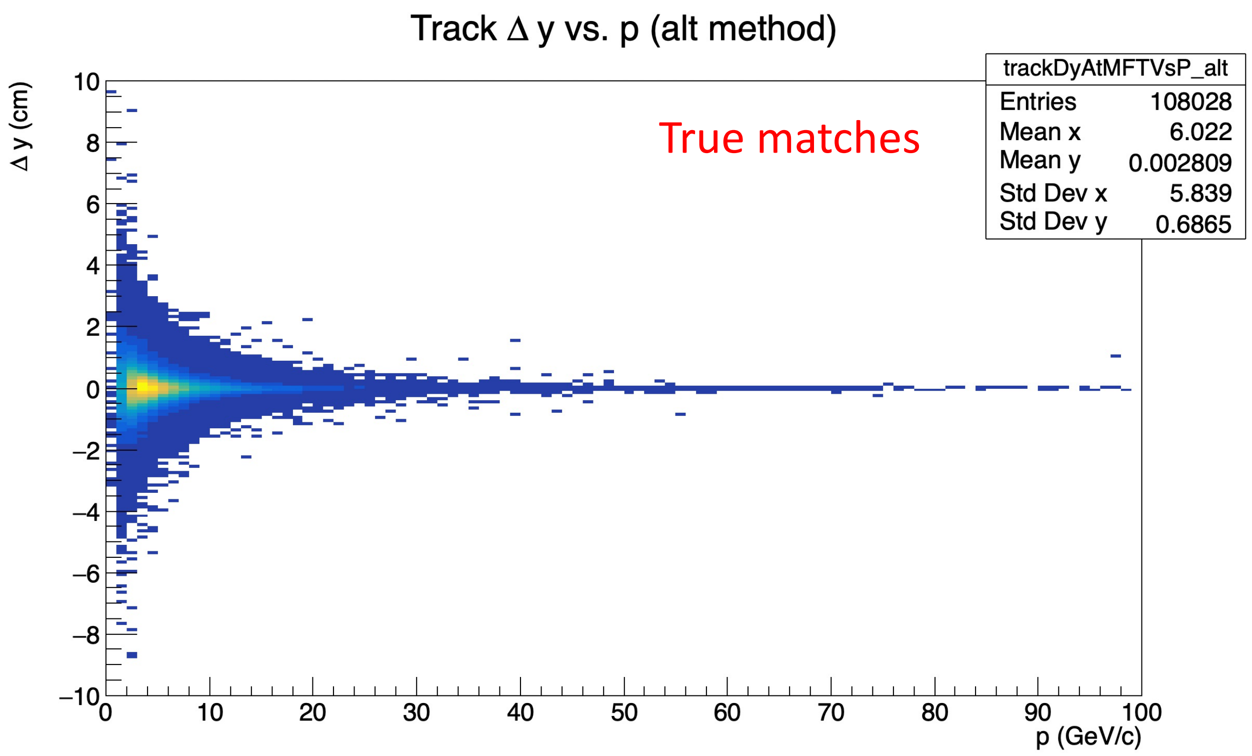
- The plots below show the difference in vertical track position at the matching plane between MCH and MFT tracks (left: true matches, right: other MFT tracks in the same collision)
- There is very little difference between the two at low momentum (not a surprise!)

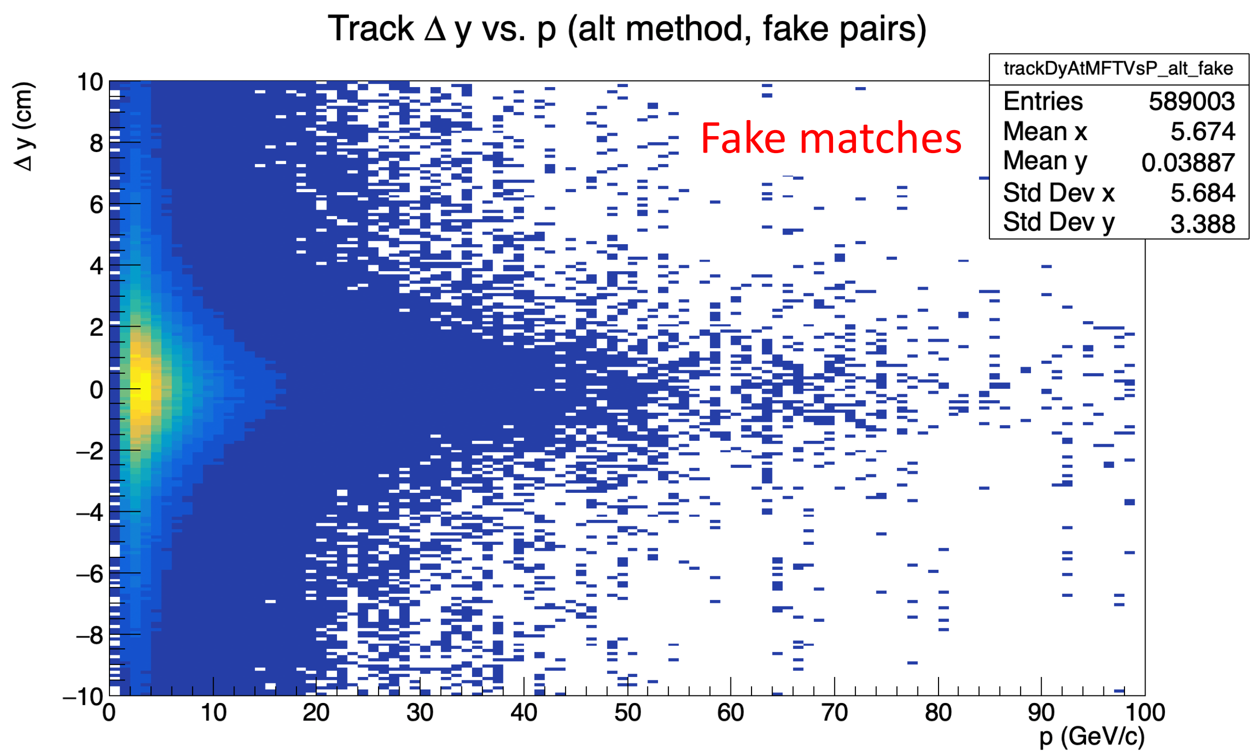




Track extrapolation to matching plane

- The situation gets better when adding the vertex constraint, i.e. first extrapolating MCH to the vertex and then back to the matching plane
- Could this be a promising alternative method for the MFT-MCH matching? Any thoughts?





Conclusions and Outlook

- A first draft of the matching QA task is available on GitHub:
  - Already provides a rather complete characterization of matching performances in MC
  - Can load and run several ML models via the `MlResponseMFTMuonMatch` class

- First look at a simple tagging method for “likely true” matches in real data
  - Work still needed to optimize the selection and understand the remaining fake matches

- First look at an alternative method for propagating the MCH tracks to the matching plane
  - Might provide a more precise estimate of MCH track parameters at low momentum
  - Not clear if the vertex constraint can introduce some bias in the matching output…