The advancements in Track overlay

Fang-Ying Tsai (Stony Brook University)
on behalf of the Fast Chain group

General reconstruction meeting
October 10th 2023

Stony Brook University

BROOKHAVEN NATIONAL LABORATORY

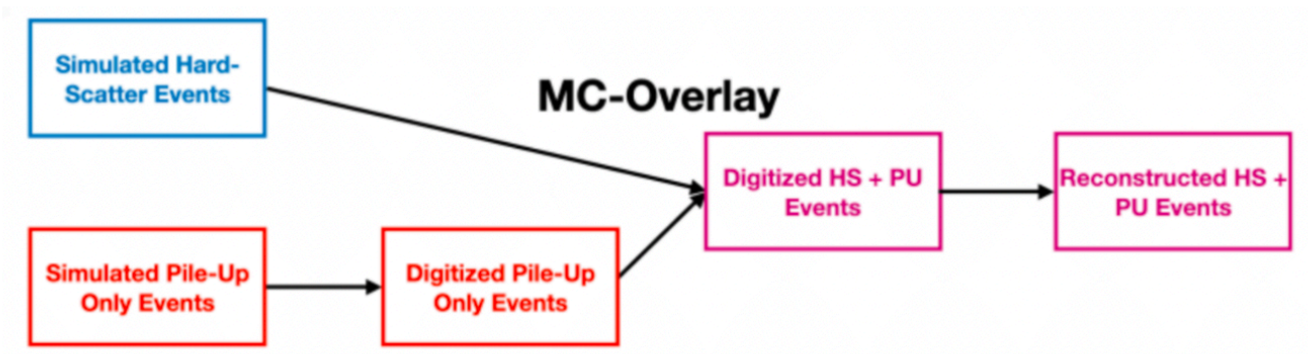
ATLAS EXPERIMENT

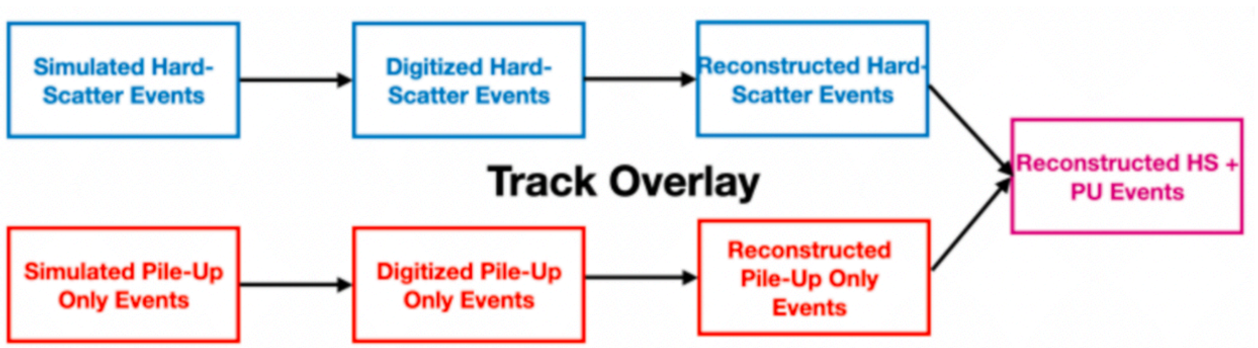
Outline

- The development and integration of the track overlay workflow into the Athena.
- Validation of the track overlay technique.
- Integration with machine learning and the filtering method.

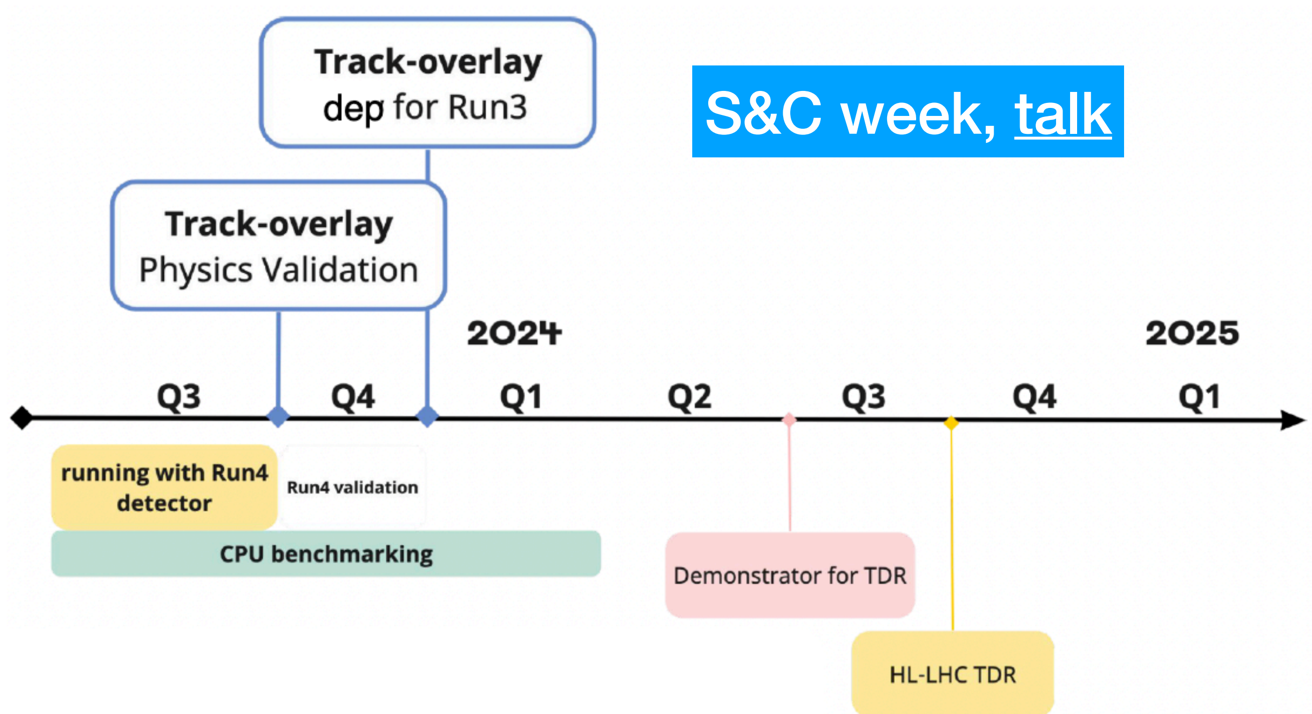
MC overlay v.s. Track overlay workflows

- MC-overlay is a well-established workflow that effectively models real data conditions.
- Track overlay is a complementary technique that is used in addition to the MC-overlay workflow.
- In the track-overlay workflow, we perform **pre-reconstruction of pileup tracks** and utilize them to mitigate the computational overhead associated with pileup vertex reconstruction.





- Track-overlay will be ready to be used in Run3!

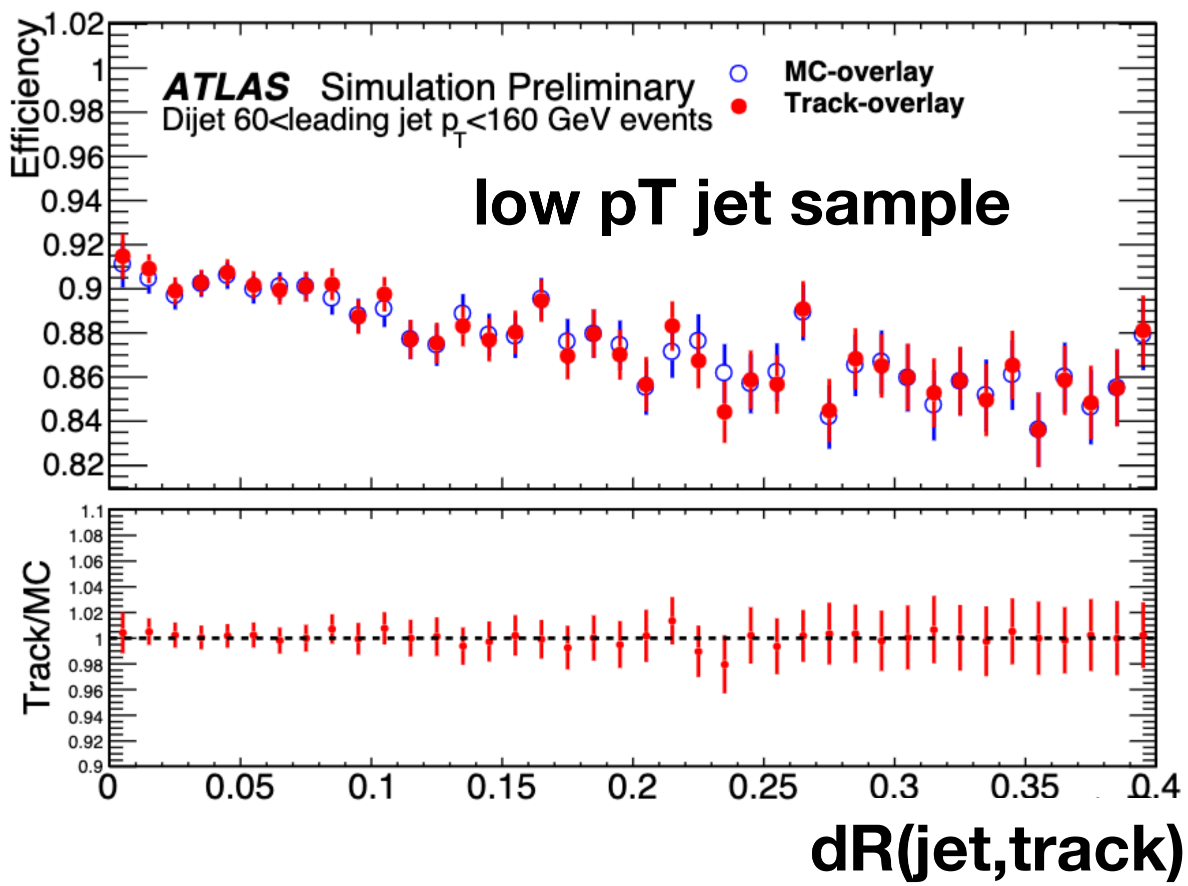


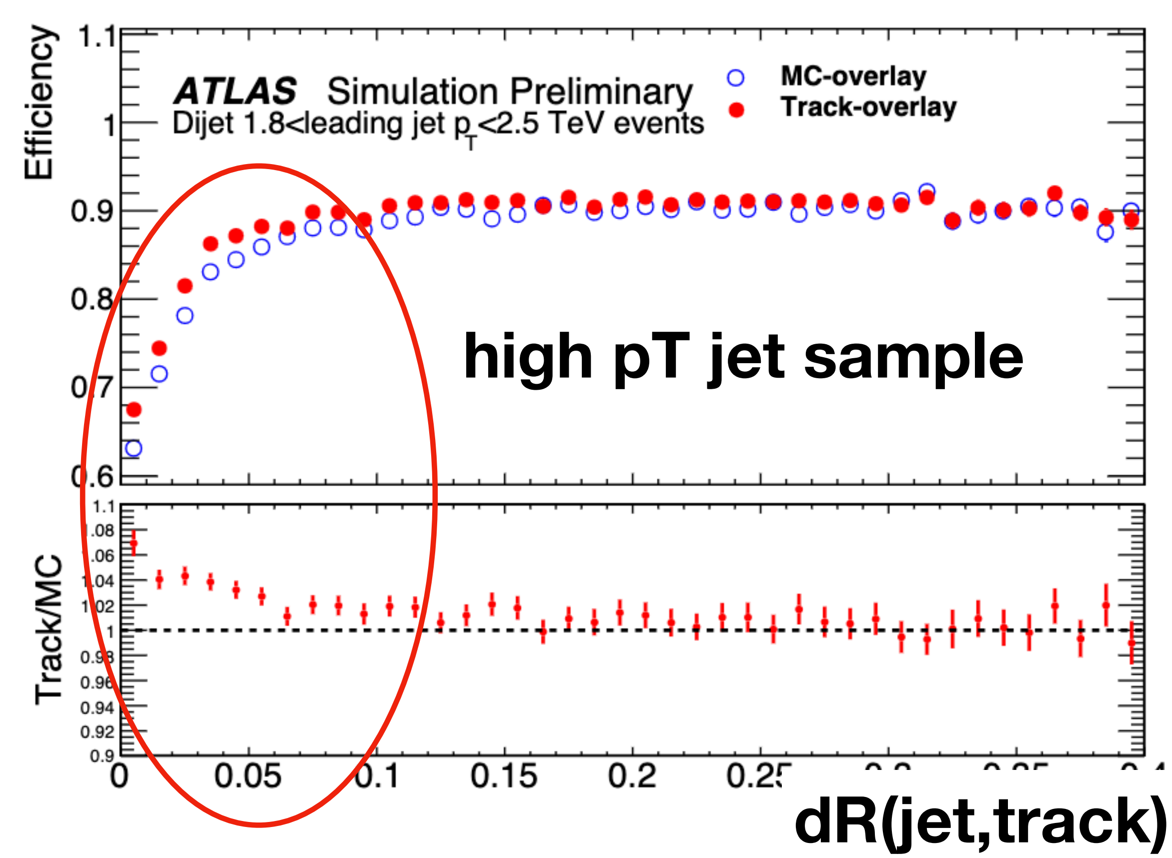
S&C week, talk

Validation

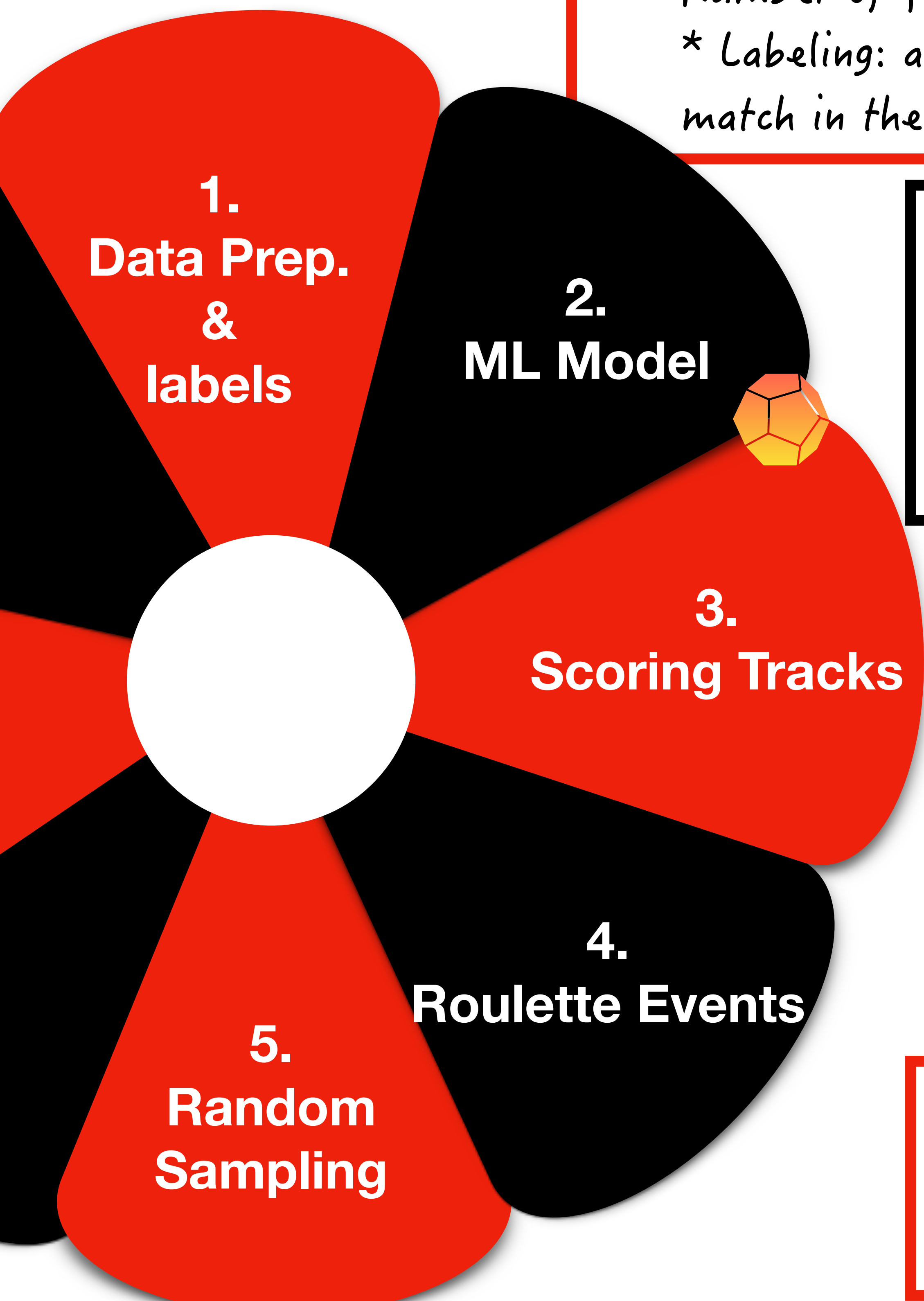
- The reconstruction of tracks inside jets can be challenges.
  - In highly dense environment where hits are often shared or misinterpreted due to overlapping tracks.
  - With the presence of pileup, which can complicate pattern recognition.
- We want to train a model to identify events that are sparser and then flag them for reconstruction using Track-overlay in the high pT jet sample.

efficiency of reconstructing HS tracks





Roulette ML model



1. * Truth particle features (Px, Py, Pz, E, Pt) from the track and density (R02, sumpTR02…) as well as conditions:
Number of pileup, Multiplicity, event pT.
* Labeling: assign a Diff label to each track based on whether it has a truth match in the Track-overlay but not in the MC overlay.

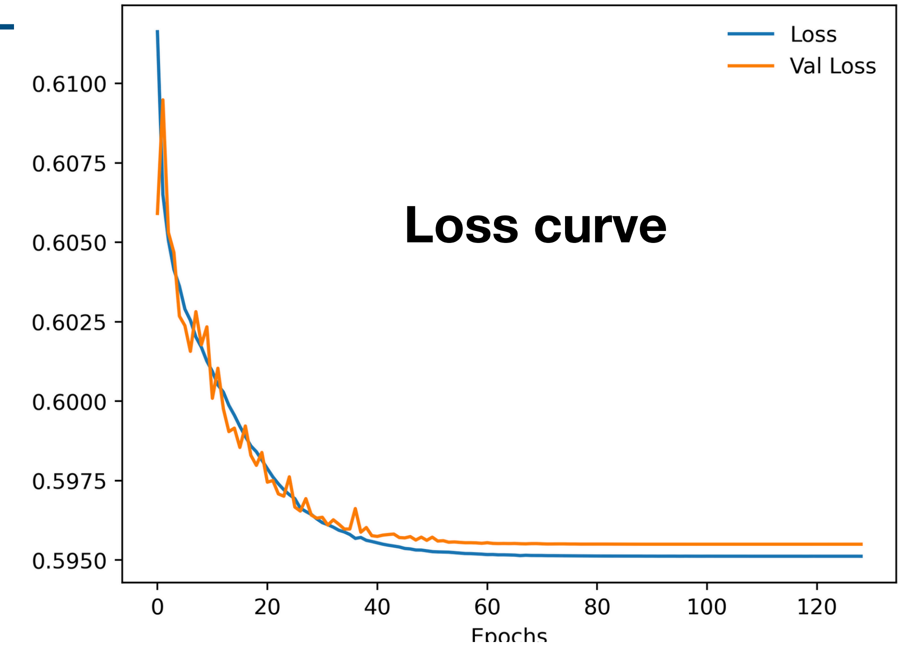
2. *Train a ML model using the collected data with labels so the model can learn patterns and relationships within tracks.
* Use the model to predict the track quality of particle tracks based on features and conditions.

3. If a track is associated with a high-density event, the model assigns it a higher score, indicating a stronger LH of being considered a 'Bad track.'

4. Calculate Roulette score for each event:
N_Bad tracks/N_total tracks.

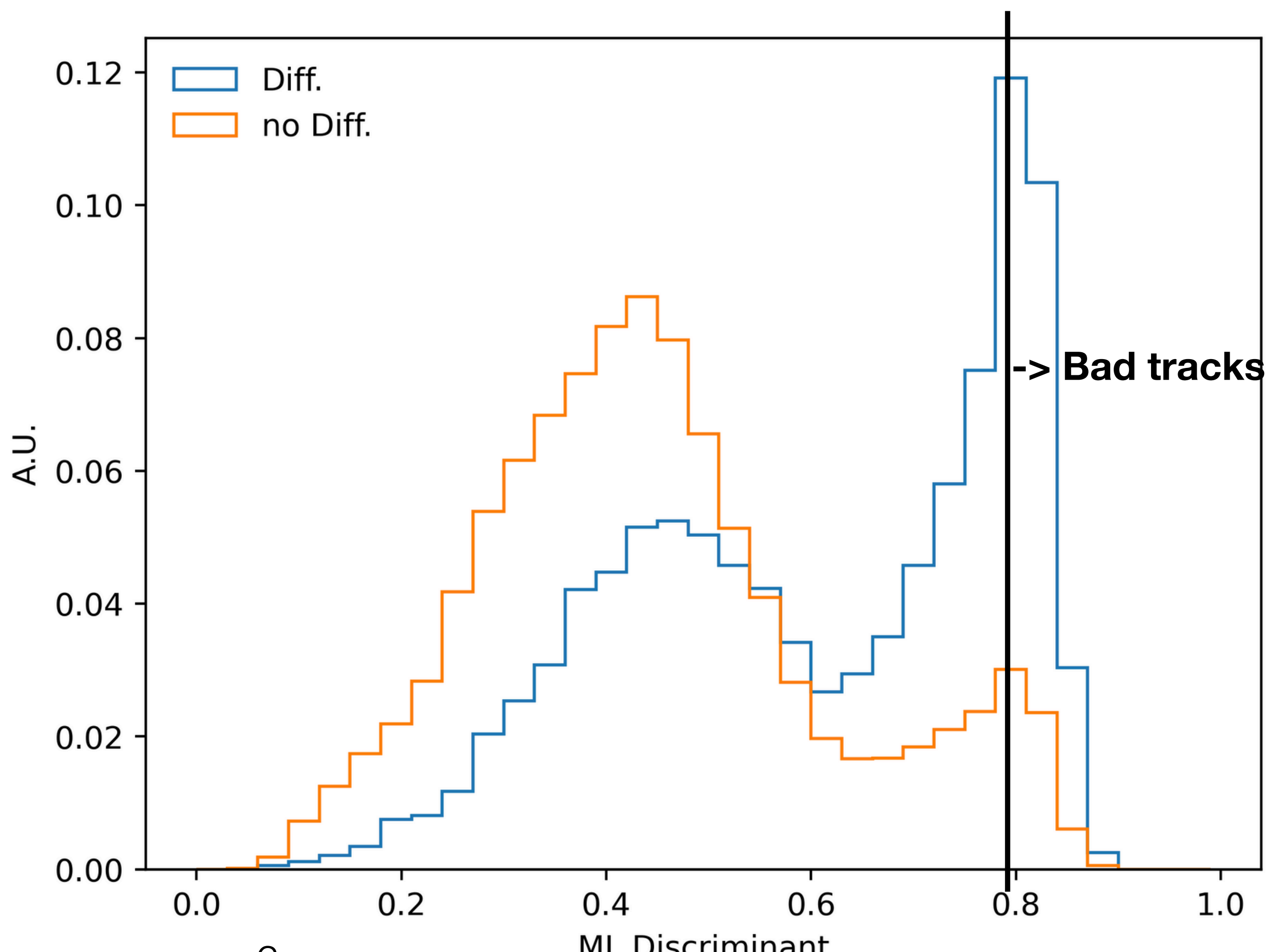
5. Some types of events with Roulette scores just above a threshold are sent to the Track-overlay, while those just below the threshold are sent to the MC-overlay.

Performance Metrics: loss function and ML scores



Model evaluation - J7 model on J7 data:

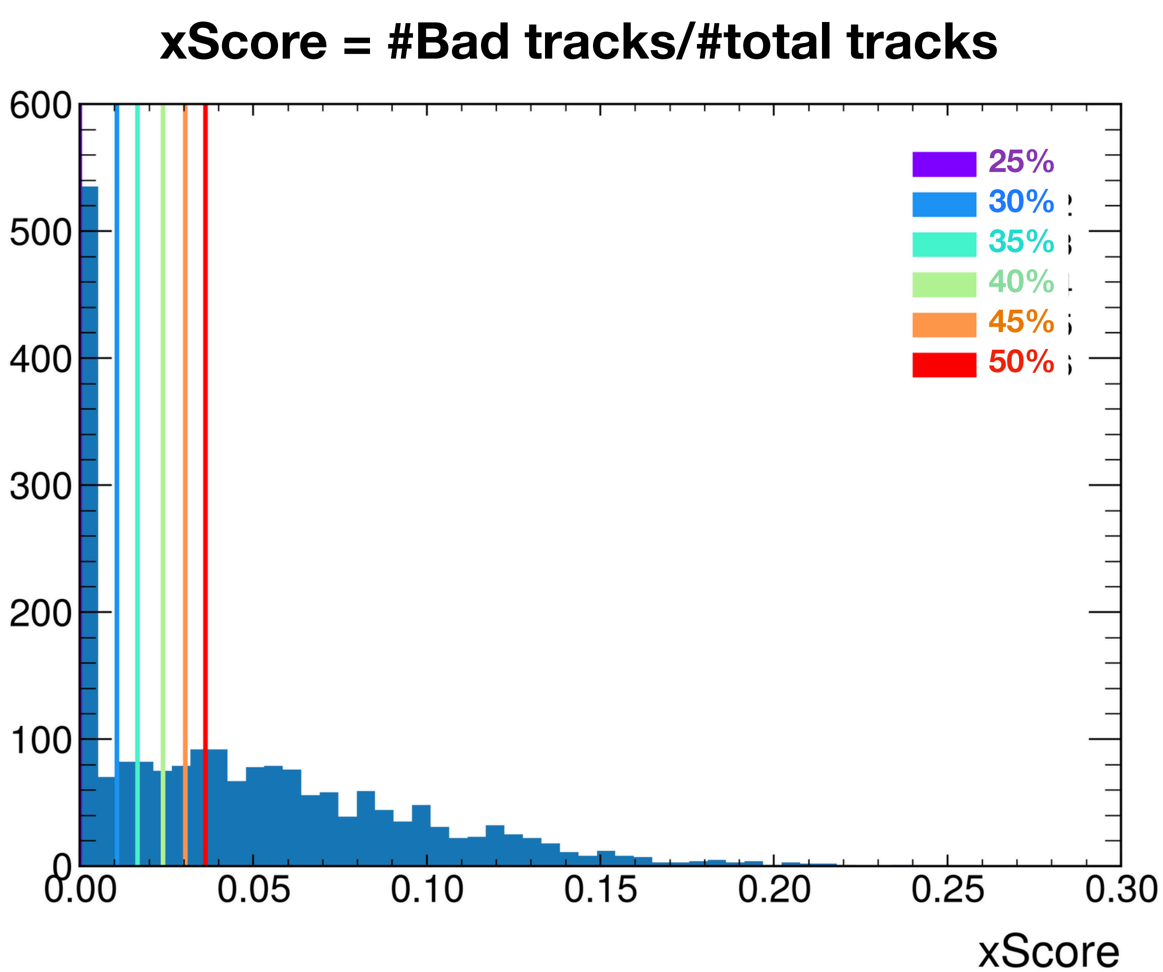
* Input Data:
  - Diff Labels:224871 samples
  - NoDiff Labels: 324871 samples
* Features: :['px', 'py', 'pz', 'E', ''pT', 'R02', 'R05', 'sumR02', 'sumR05', 'sumpT02', 'sumpT05']
* Conditioning Features: ['number of PU', 'TruthMultiplicity', 'event pT']
* Hyper-parameters:
  - Layers: [45,35,30]
  - Batch Size: 80
  - Learning rate: lr*tf.math.exp(-0.1)

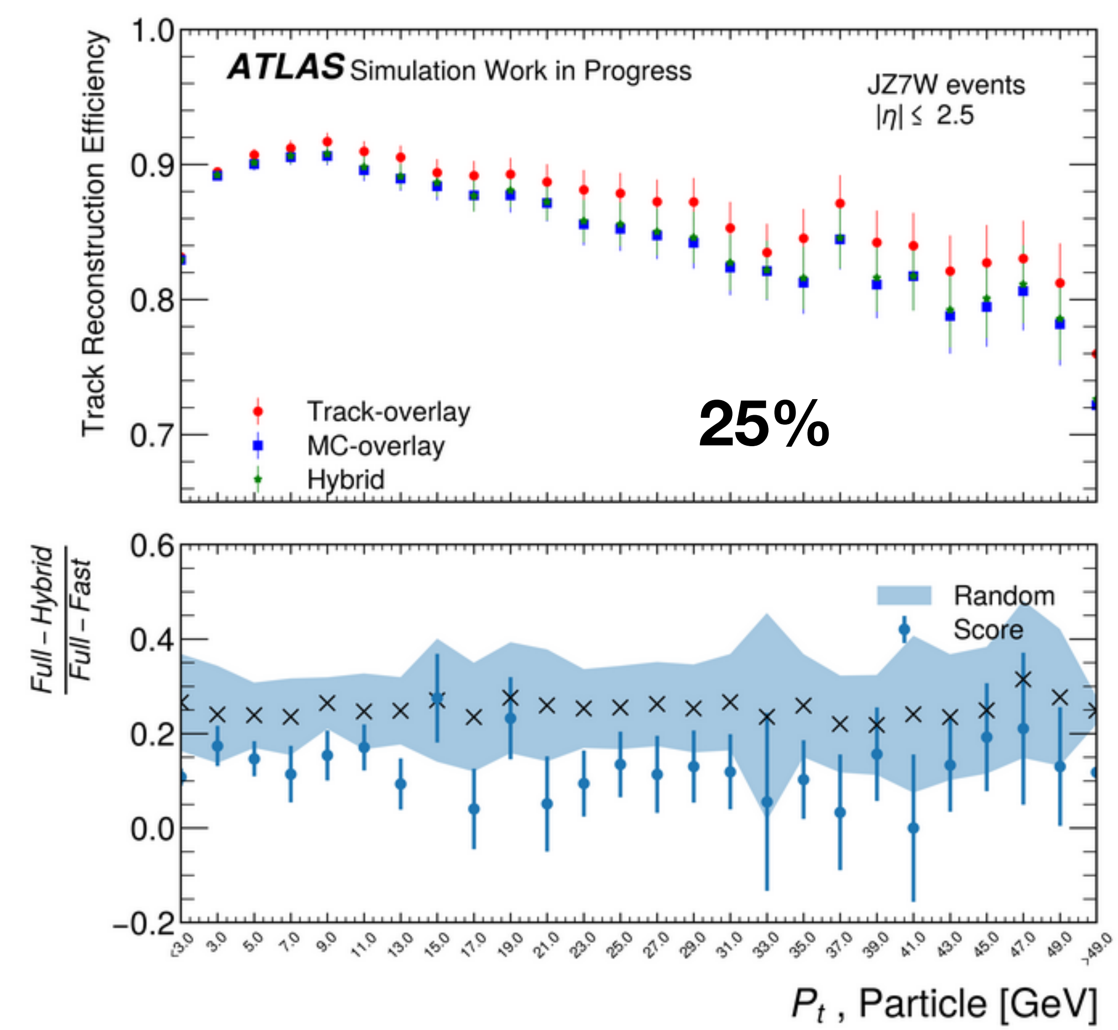


0.7999 (82% of Dif)

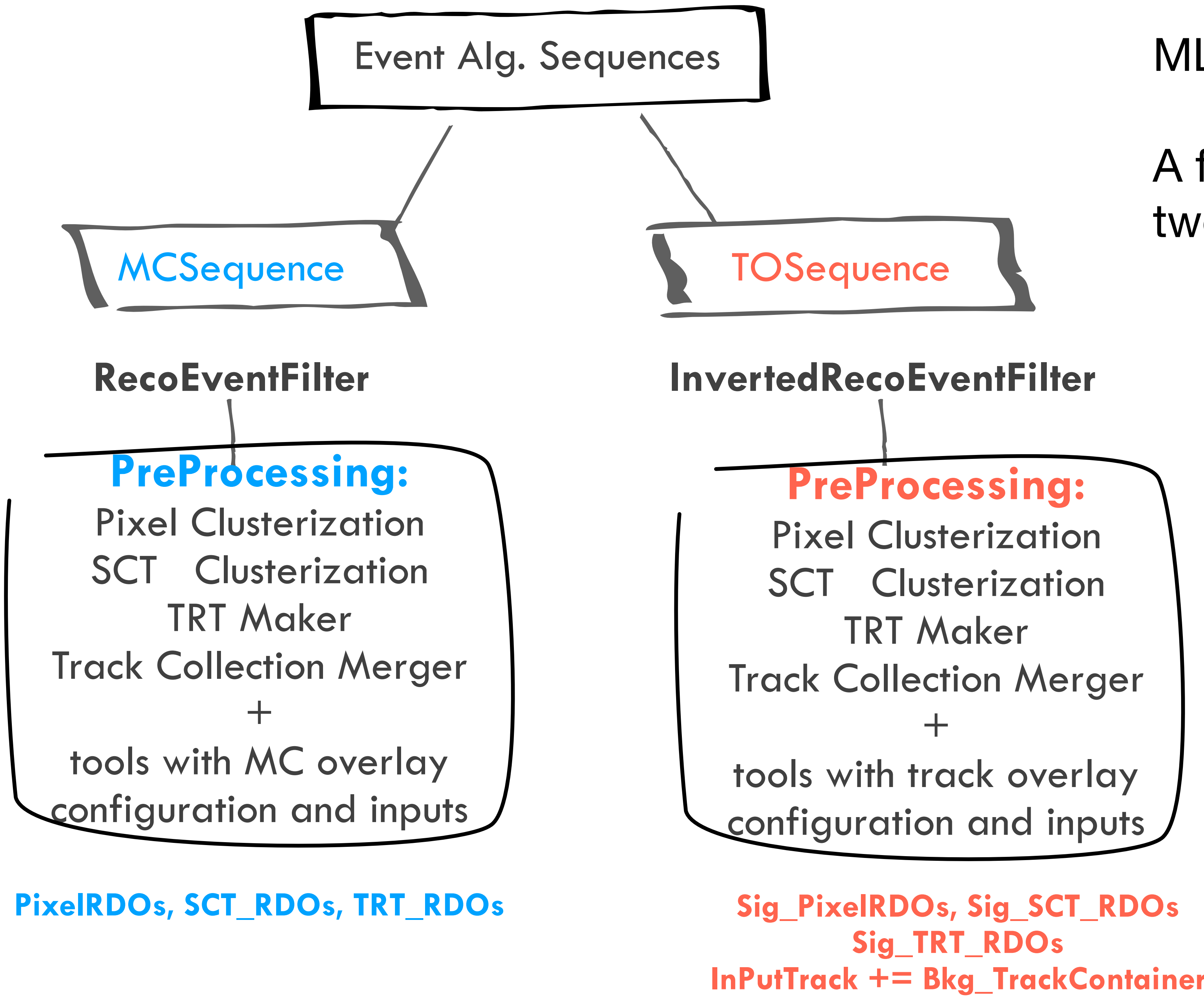
Roulette events selection with random sampling

- Quantiles of xScore: 25%, 30%, 35%, 40%…
- At 25% quantile, xScore is located at 0, indicating that 25% of events have no Bad tracks (above the ML discrimination score threshold.)
- At 25% quantile, Hybrid and MC efficiencies align better, and the ML decision is better than random selection.





Athena Filter: reconstruction



ML decision (event by event)

A filter to switch between two sequences (1configuration.)

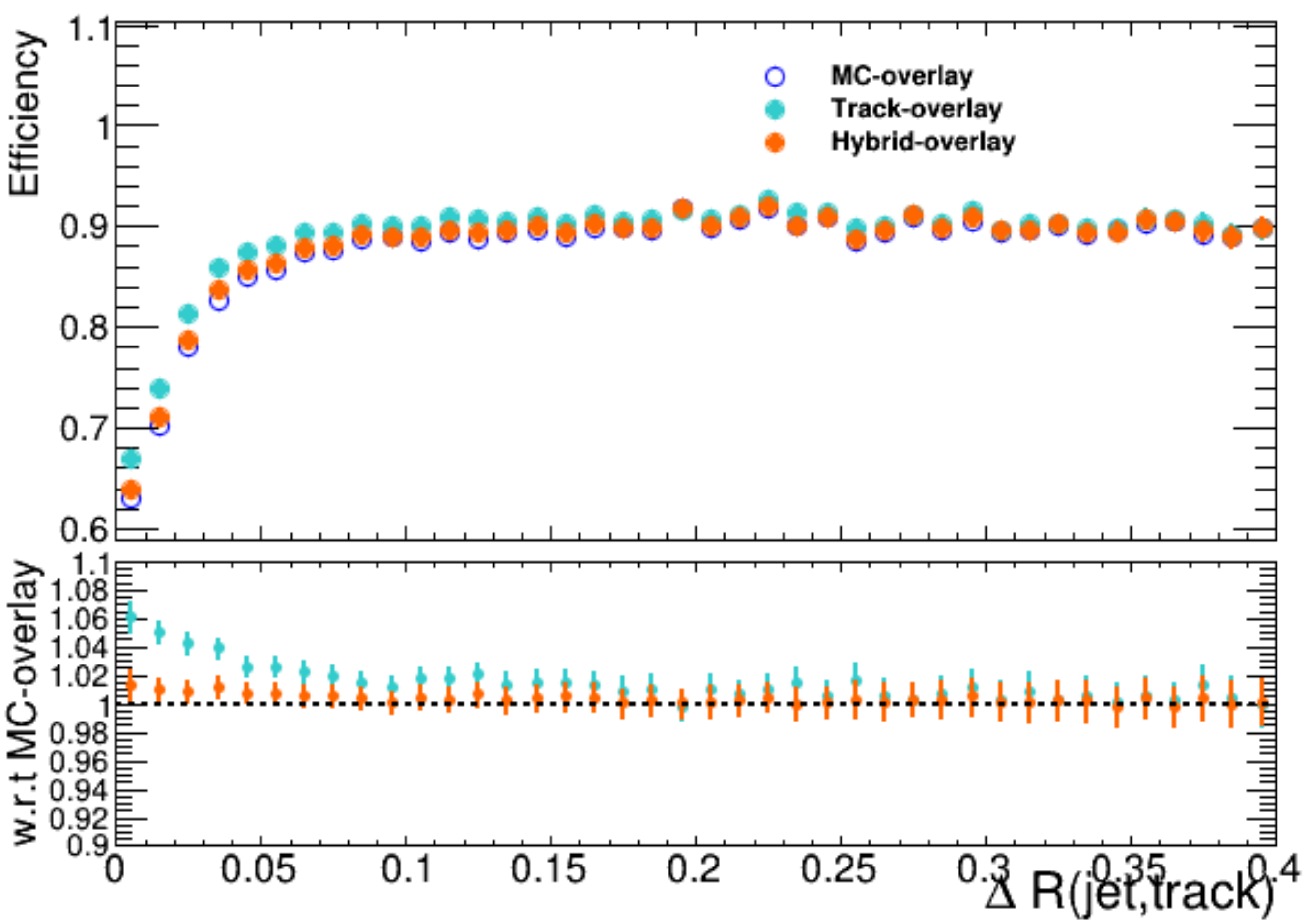
- The tracking algorithm will process the regular PixelRDOs, SCT_RDOs, and TRT_RODs, while when doTrackOverlay is True, the tracking runs on the HS RDOs.

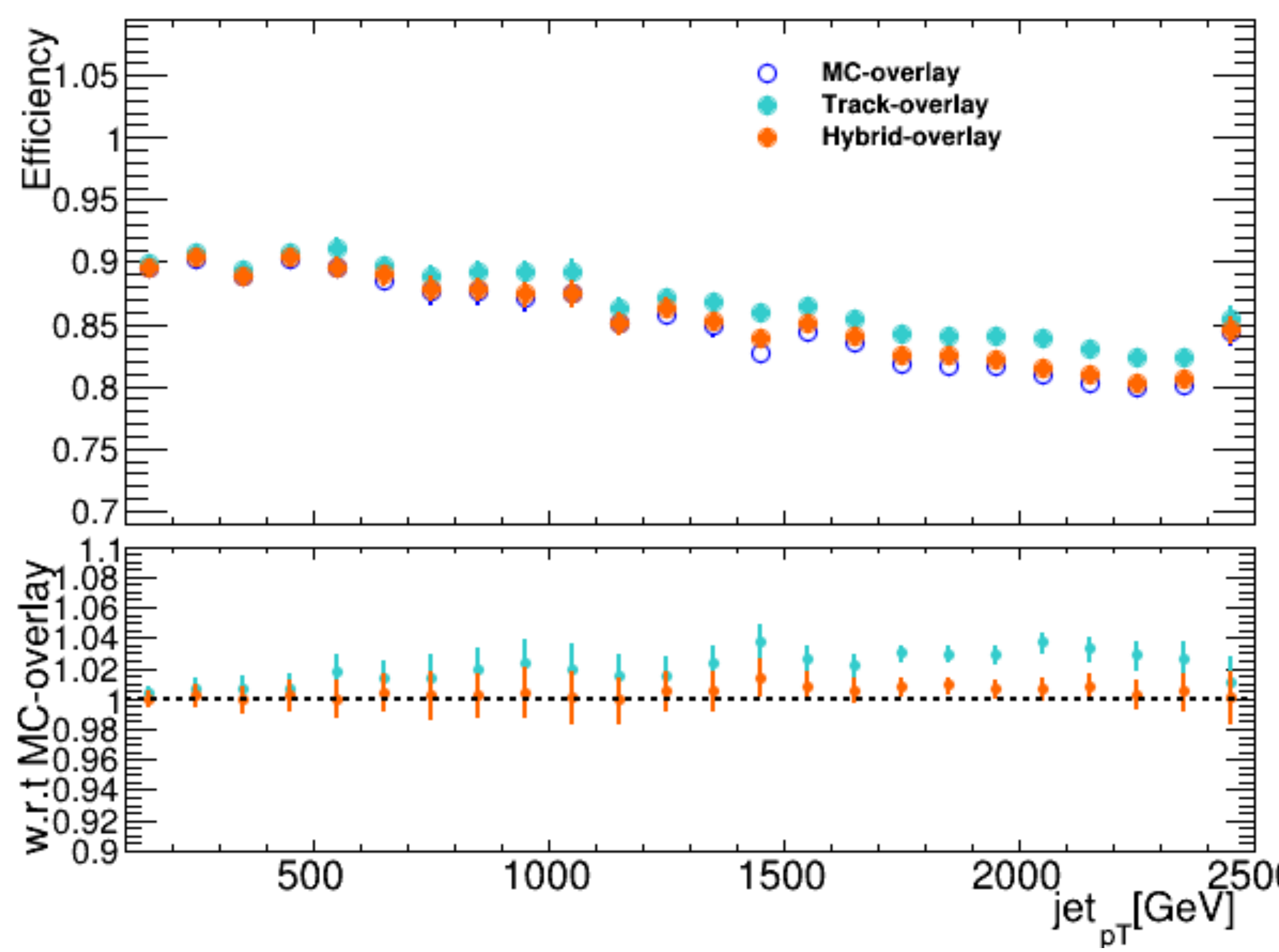
- The background tracks are added to the InputTracks list along with the signal tracks in the track overlay workflow.

Model evaluation - J7 model on J7 data

* Bad Tracks if ML Score > 0.7999
* Select **25% of J7 events** to be sent to Track-overlay.
* Loose tracks

efficiency of reconstructing HS tracks

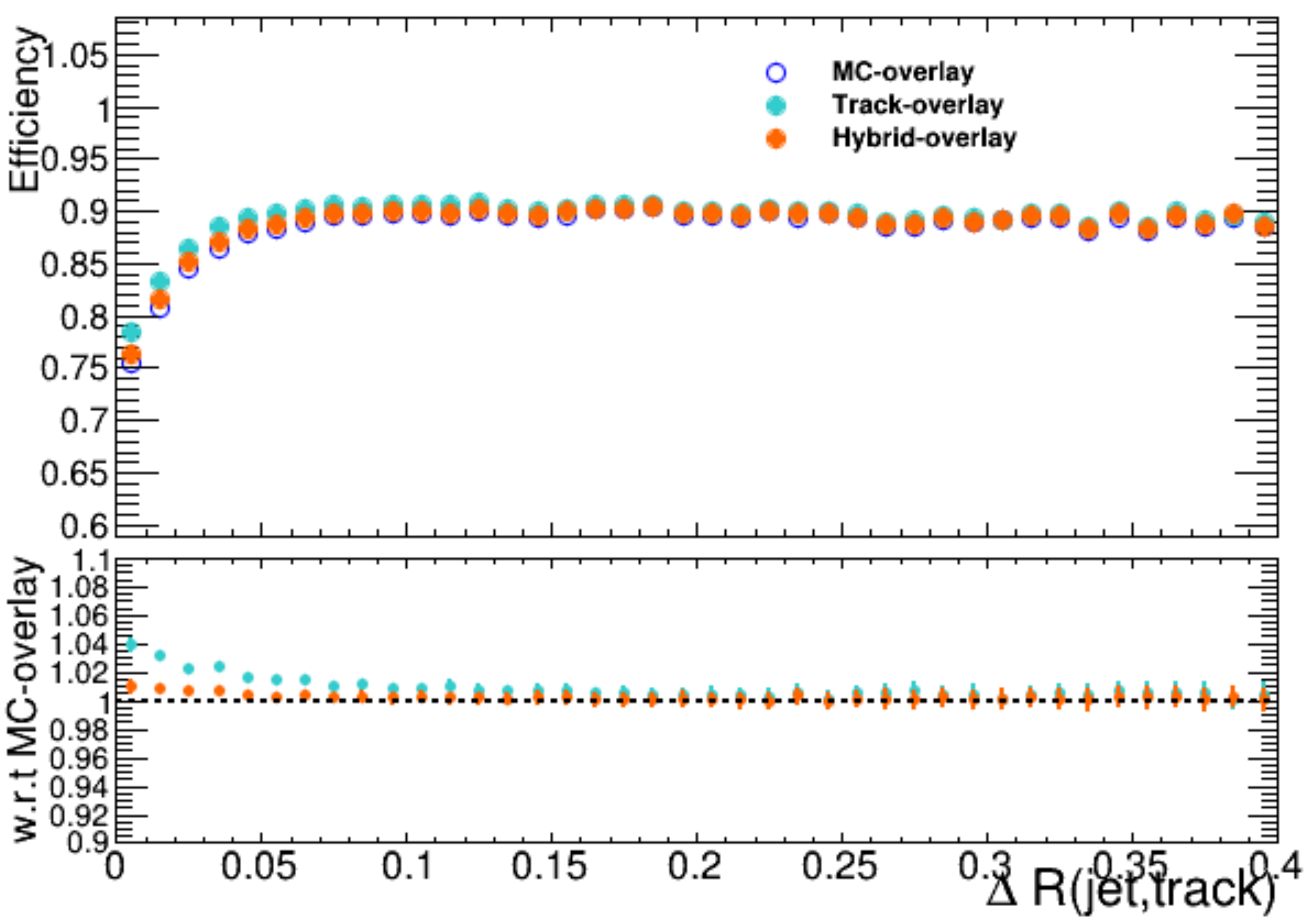


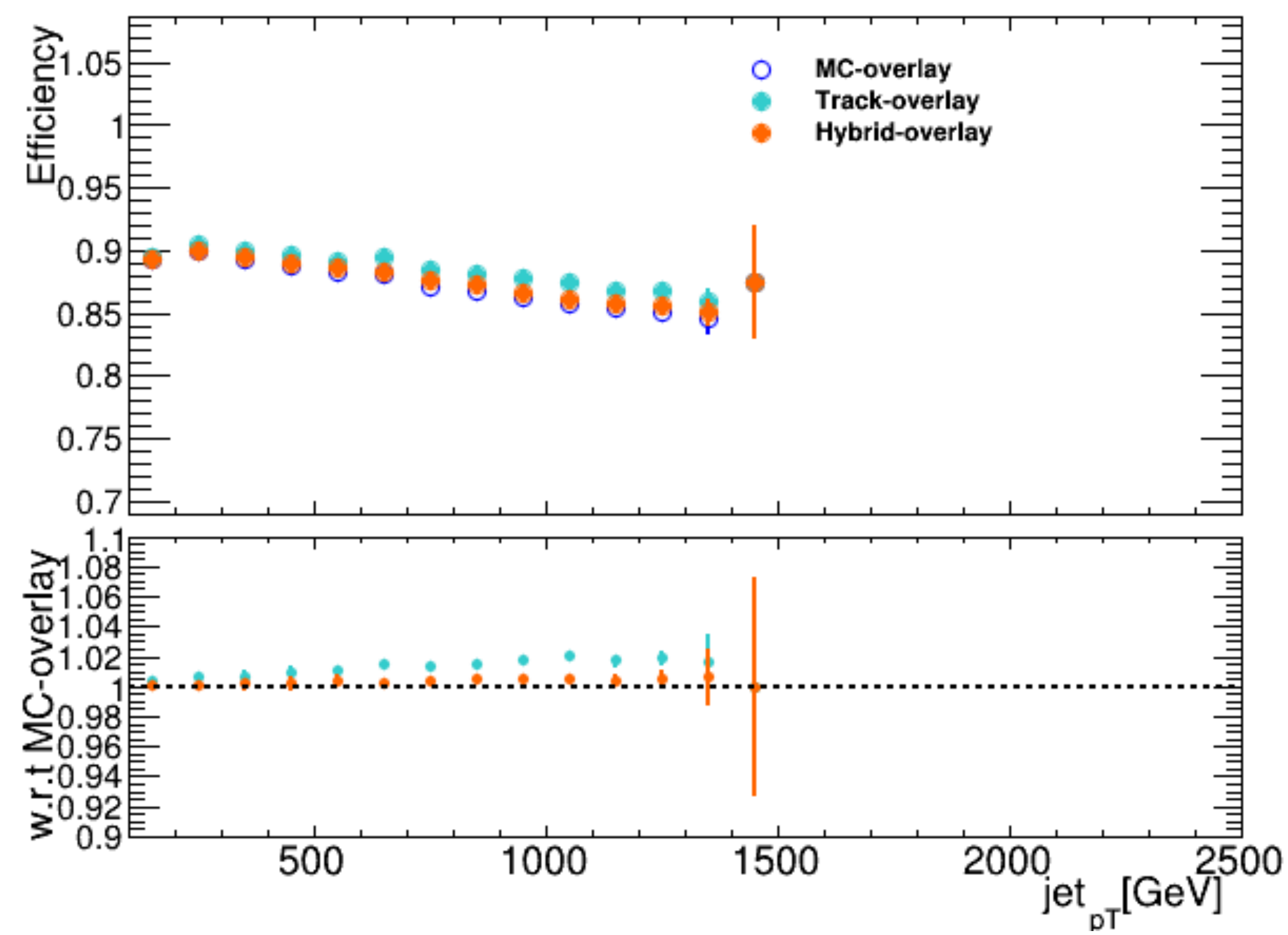


Model evaluation - J7 model on J5 data

* Bad Tracks if ML Score > 0.7999
* Select **40% of J5 events** to be sent to Track-overlay.
* Loose tracks.

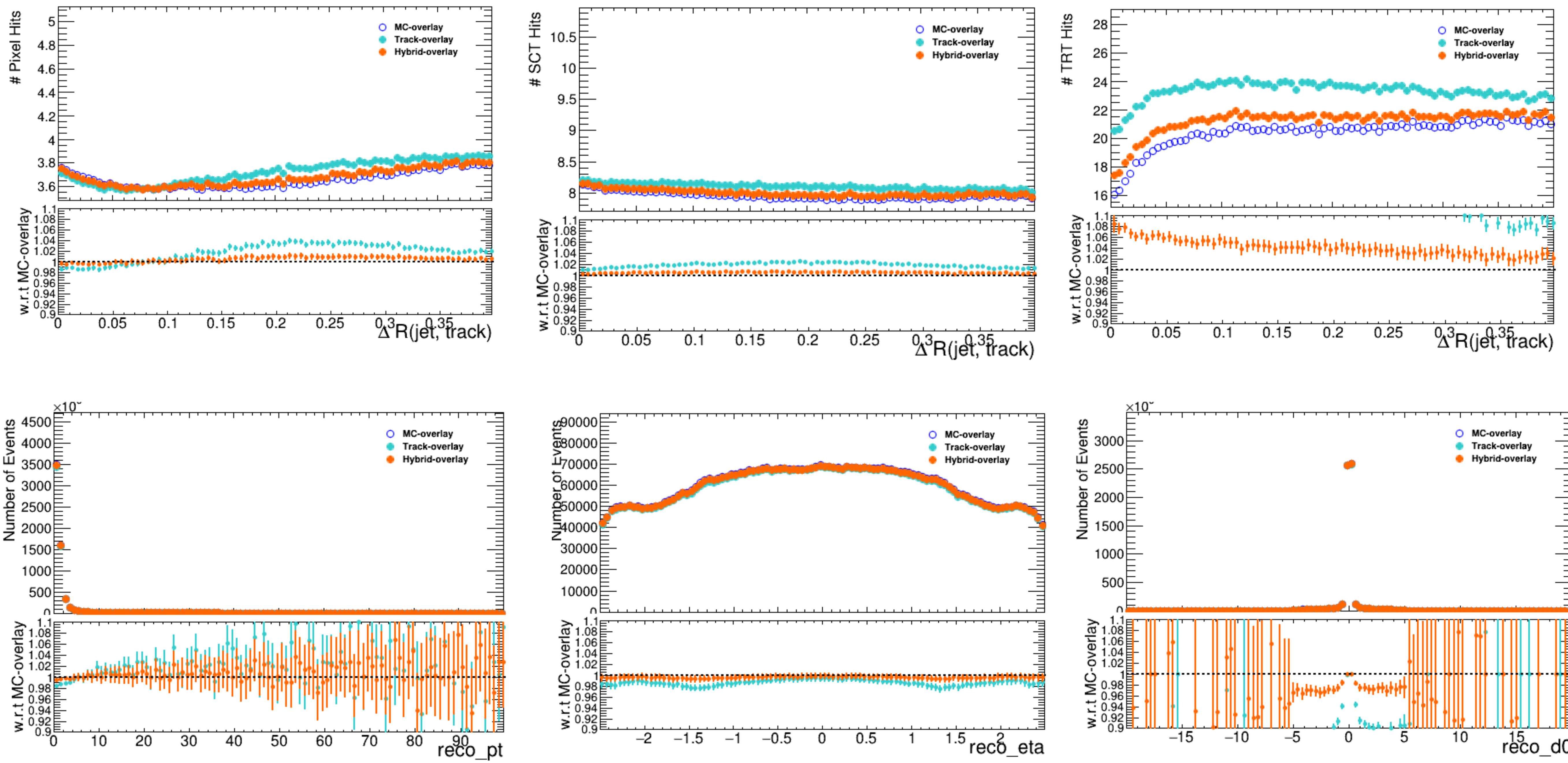
efficiency of reconstructing HS tracks





Model evaluation - J7 model on J5 data

Hit merging?

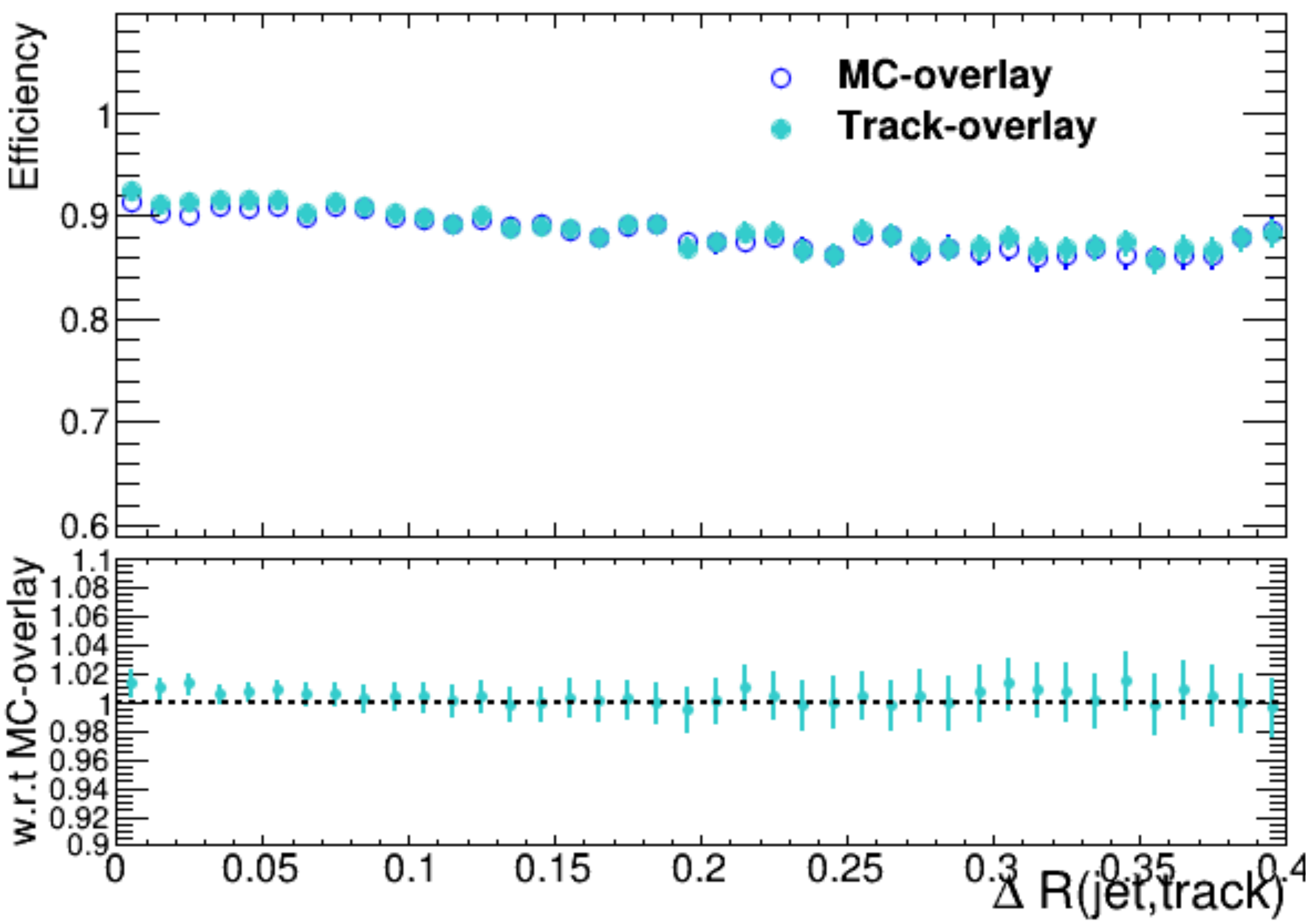


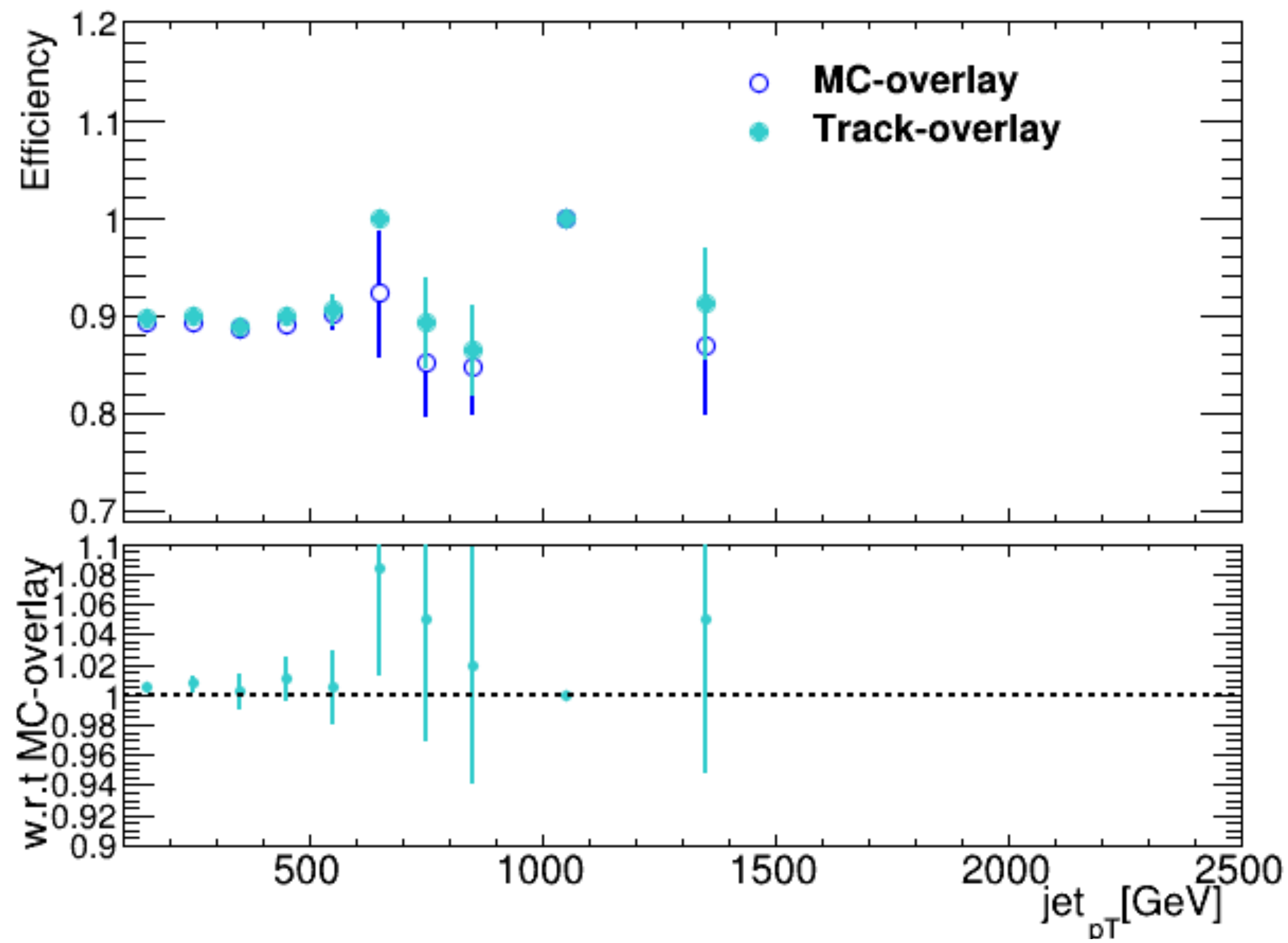
The discrepancies in reco_d0 may arise due to the influence of pileup events, which could be considered 'fake' in the reconstruction process. Further investigations and physics validation are planned to understand the impact.

Model evaluation - J7 model on ttbar data

* Bad Tracks if ML Score > 0.7999
* Select **100% of ttbar events** to be sent to Track-overlay.
* Loose tracks

efficiency of reconstructing HS tracks





Summary & Next

- The Roulette ML technique with the filter has demonstrated its capability to identify sparser events within high pT jet samples and subsequently reconstruct them using the Track overlay workflow.

- The evaluation extends to J7, J5 and ttbar sample.

- In Run3, the agreement between Track-overlay and MC-overlay, as determined by the ML decision, shows less than 2% uncertainty (except TRT Hits).
  - Ongoing QT, Jira, for Run4.

- Mini-analysis: Track-overlay performs well in boosted jet observables used by W' → Wh → lvbb, talk.

- Next:
  - Push the filter code for switching between Track-overlay and MC-overlay to the Athena (~20 changed files.)
  - Run the trigger part of RDOtoRDOTrigger on the pileup file.
  - Convert the TF model format and integrate it into the Athena (under investigation).

Backup

More comparisons on dcube

	Run-3 (μ>50)					
Validation	**ttbar**	**b-tagging**	**JZ2W**	**JZ7W**	**Resolution(J7)**	**doTau**
Performance link	Plots	Plots	Plots	Plots	Plots	Plots
Events	2000					
geometry	ATLAS-R3S-2021-02-00-00					
conditions	OFLCOND-MC21-SDR-RUN3-05					

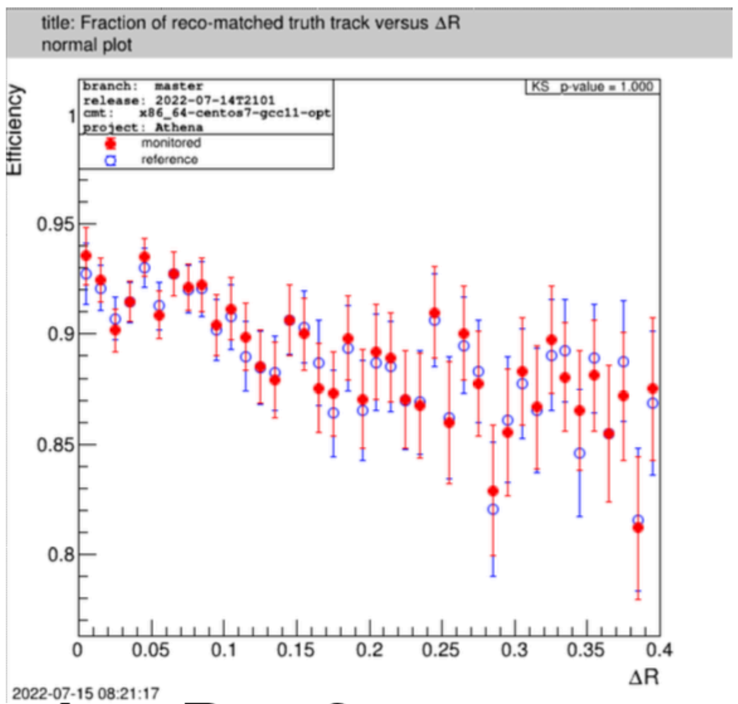
TracksInJets: Efficiency vs dR

MC-overlay
Track-overlay

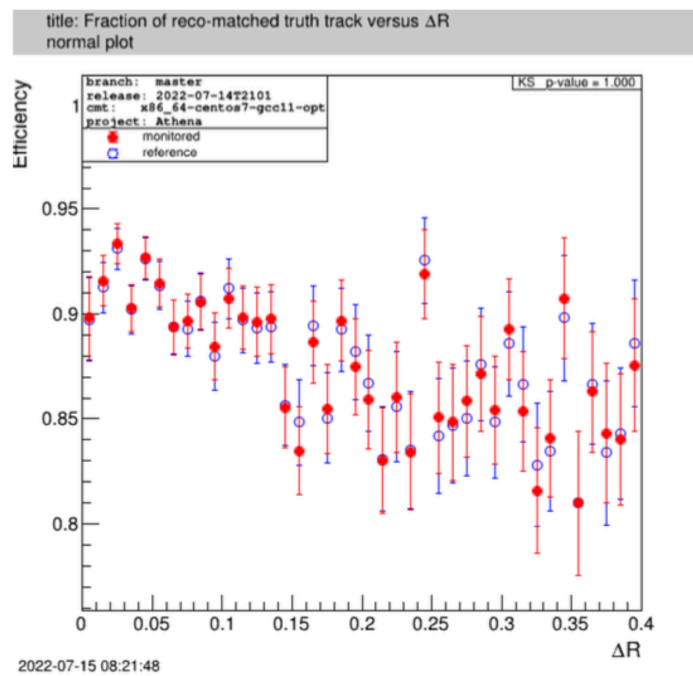
* Performance is expected to be worse here as jets will increase occupancy (worse reconstruction and less efficiency in MC-overlay).

S&C week, talk.

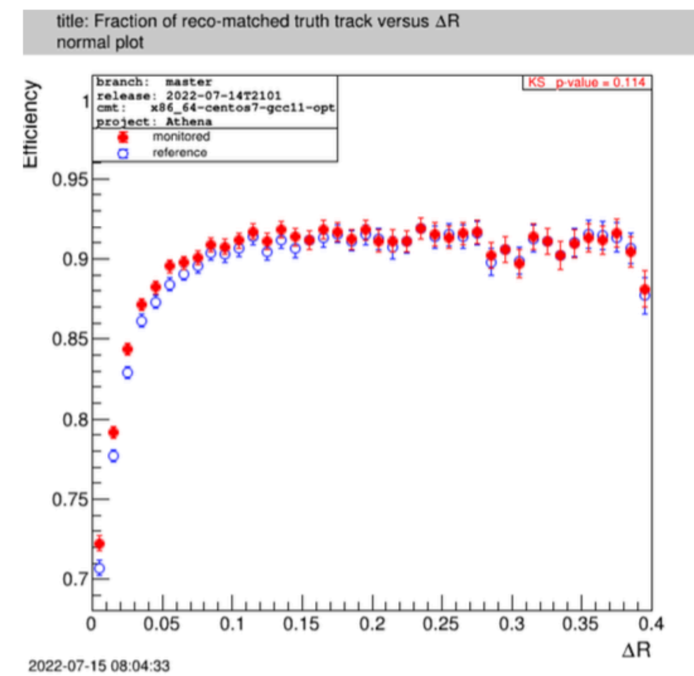
ttbar, Run-2



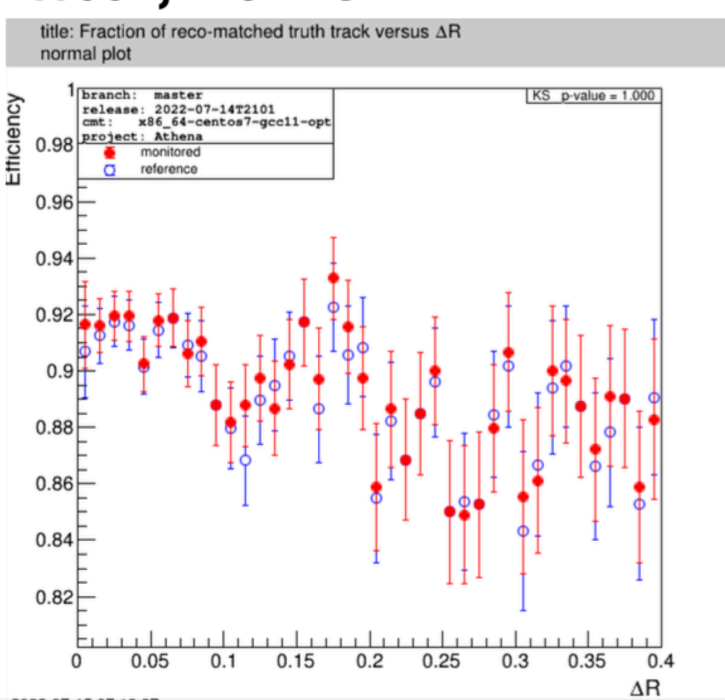
JZ2W, Run-2



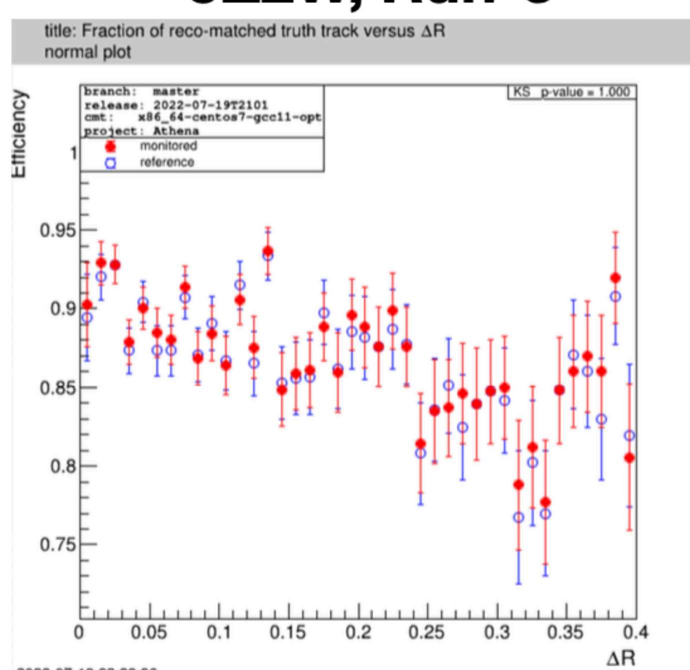
JZ7W, Run-2



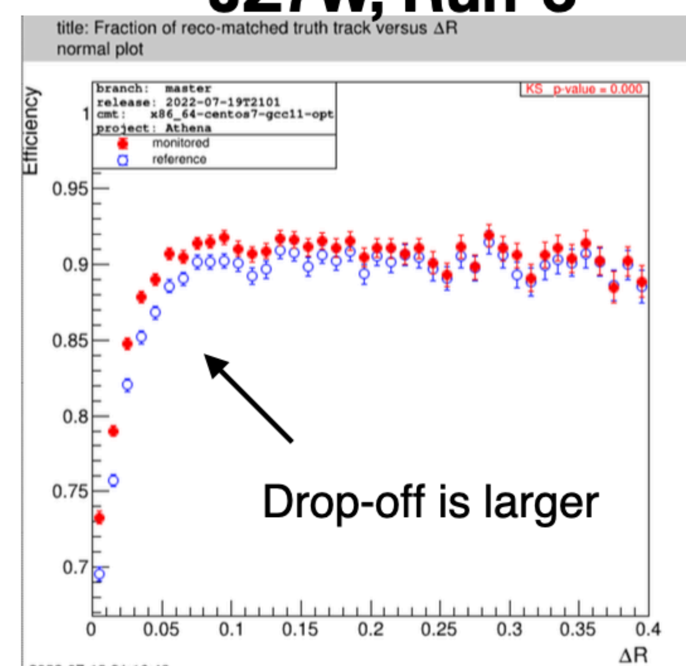
ttbar, Run-3



JZ2W, Run-3



JZ7W, Run-3



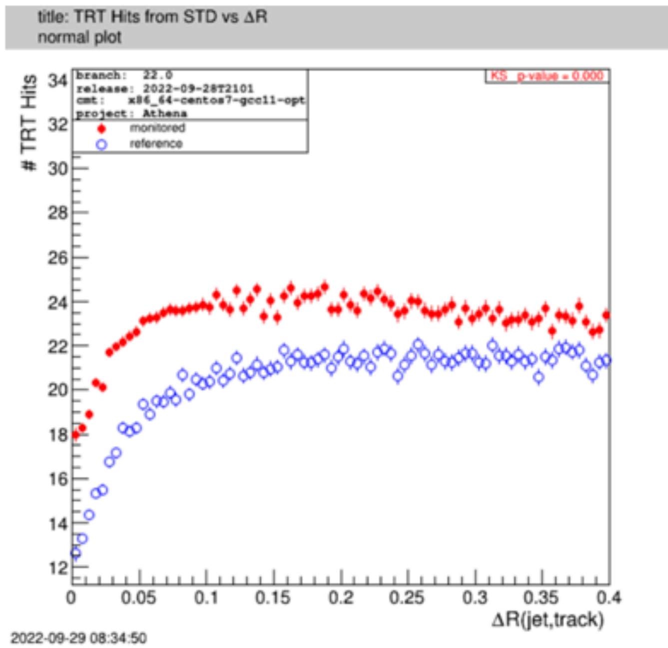
18 Investigate: nTRTHits

MC-overlay
Track-overlay

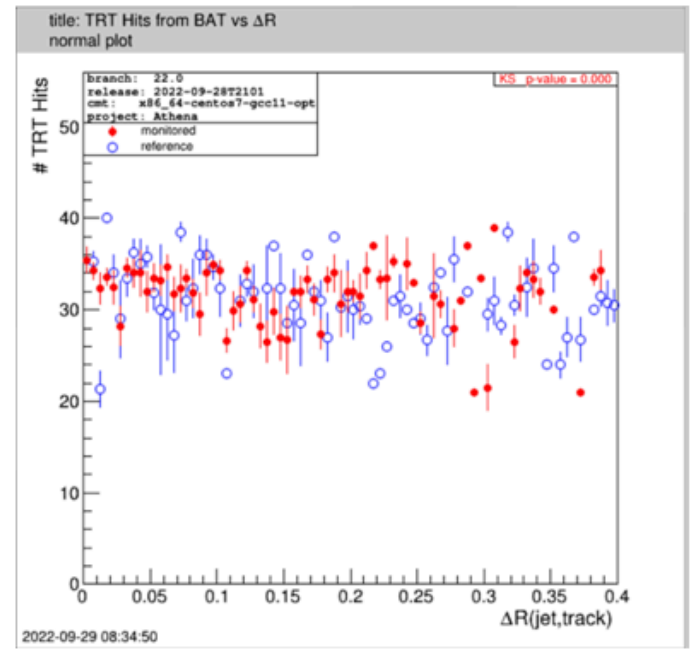
S&C week, talk.

* The discrepancy is not seen in outside-in TRT Hits.

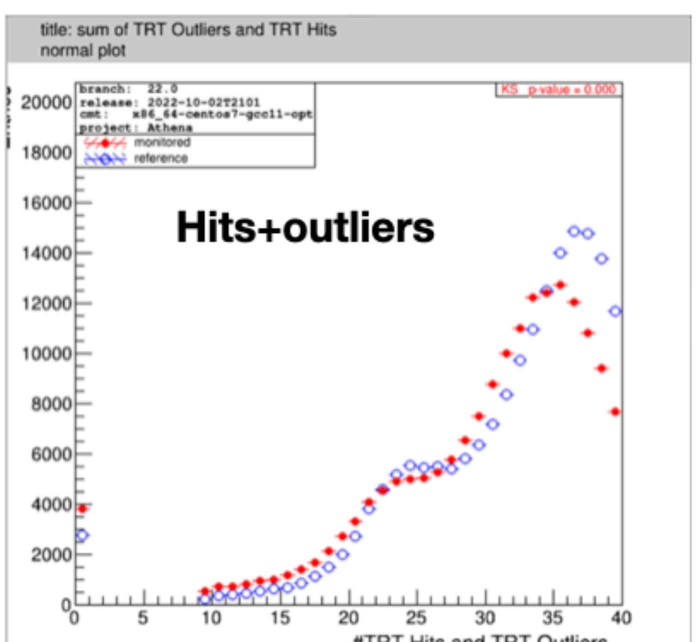
inside-out

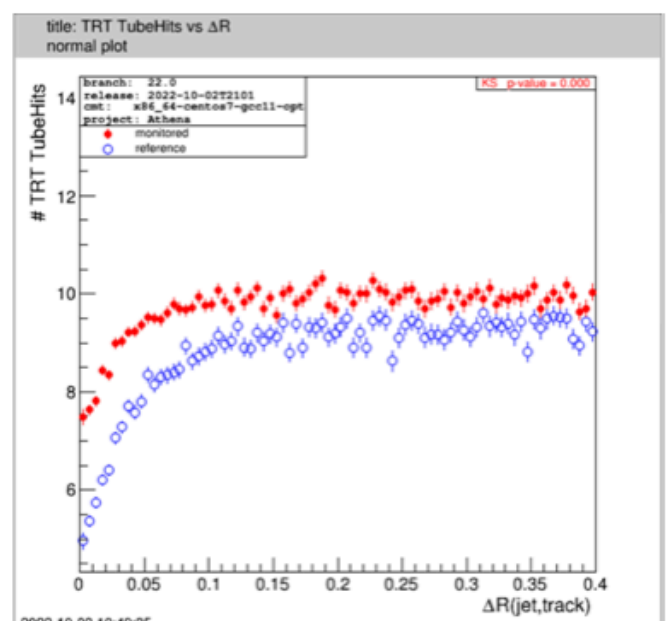


outside-in



* Some of TRT hits are moved to outliers due to reconstructed drift radius but not sure why it gets more in MC-overlay.
- More outliers in MC-overlay, so hard to tell it from the “Hits+outliers” check.





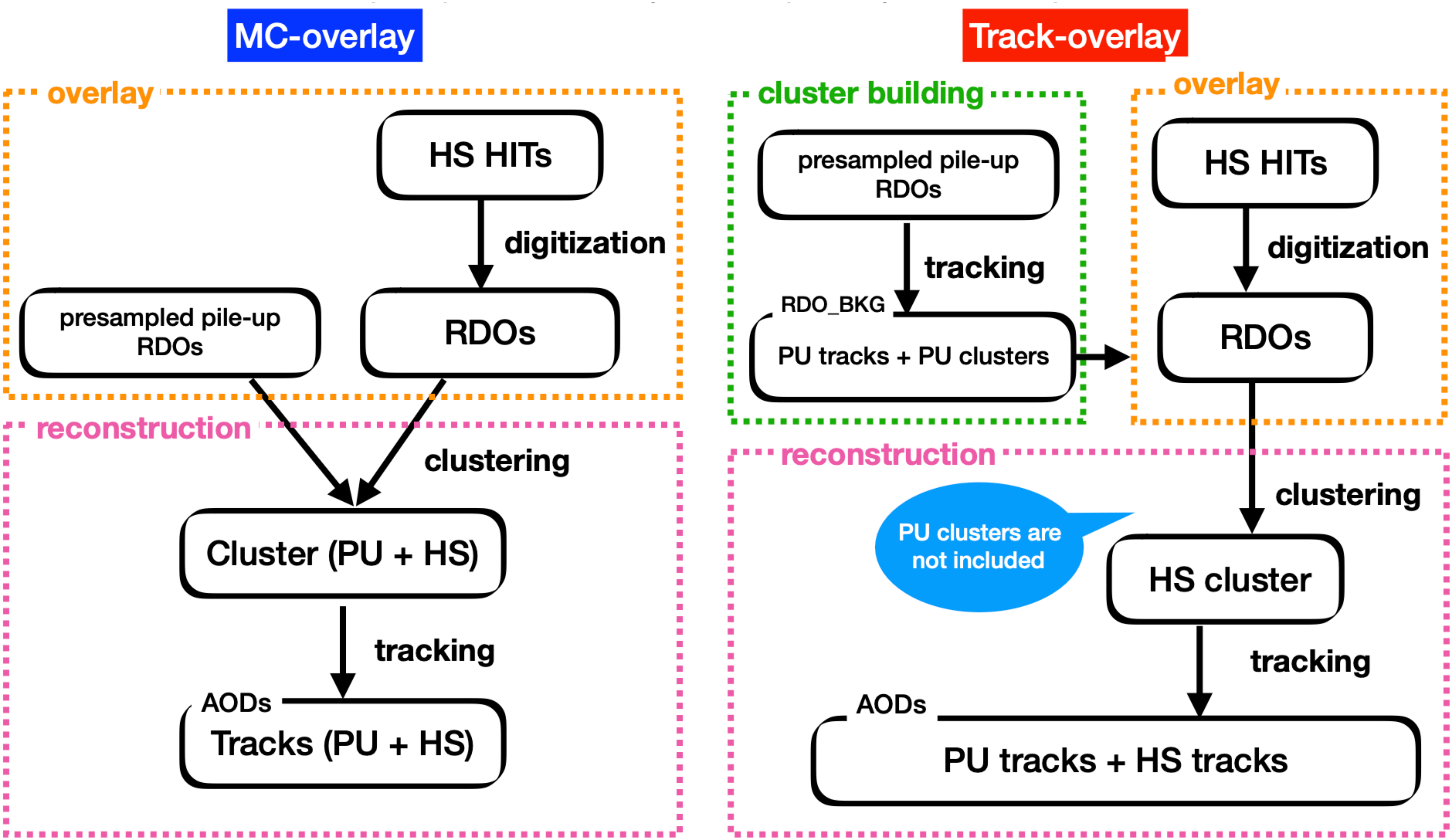
The number of tube hits has the same pattern.

Tommaso's slides

Resource used for fast simulation in 2026

BSM samples	events simulated in 2026	events reconstr. in 2026	CPU simu (kHS06*year)	CPU reco (kHS06*year)	CPU total (kHS06*year)	tape space (PB)	disk space (PB)	formats saved
AF3 + std reco	15B	50B	250	580	830	59	39	HITS, AOD, DAOD_PHYS
AF3 + track overlay	15B	50B	250	300	550	59	39	HITS, AOD, DAOD_PHYS
AF3 + track overlay	15B	2 x 50 B	250	600	850	32	11	HITS, DAOD_PHYS
AF3Fatras + std reco	15B	50B	45	580	630	59	39	HITS, AOD, DAOD_PHYS
AF3Fatras + std reco	50B	50B	150	580	730	28	39	AOD, DAOD_PHYS
AF3Fatras + track overlay	15B	50B	45	300	345	59	39	HITS, AOD, DAOD_PHYS
AF3Fatras + track overlay	2 x 50 B	2 x 50 B	300	600	900	0	11	DAOD_PHYS

* Track overlay alone offers a substantial benefit, saving 33% of CPU usage for datasets where it is applied. As an alternative, it allows for skipping the writing of AOD files, resulting in a 40% reduction in the combined disk and tape storage footprint.



For the MS and Calo, MC- and Track-overlay work the same.

Model evaluation - J7 model on ttbar :

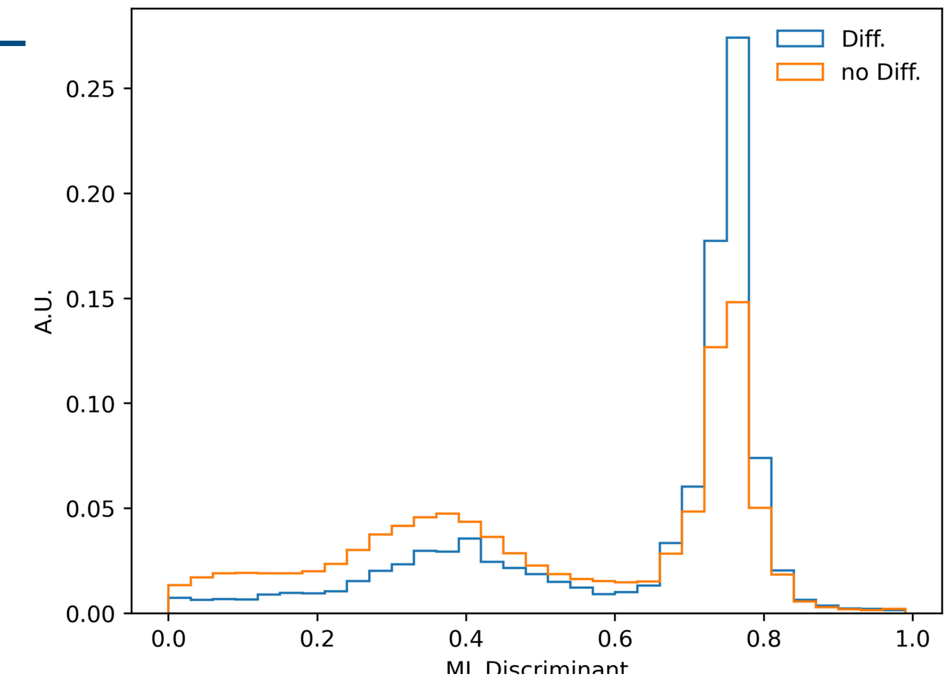
* Input Data:

  - Diff Labels:15119567 samples

  - NoDiff Labels: 15119567 samples

* Features: :['px', 'py', 'pz', 'E', '’pT', 'R02', 'R05’]

* Conditioning Features: ["wo_numvtx", "w_numvtx", "TruthMultiplicity", "eventPt", "wo_leadJetpT", "wo_nleadJetpT", "w_leadJetpT", “w_nleadJetpT"]



ttbar events are not the primary concerned dense environments we focus on. High scores for ttbar events would be problematic for the J7.

Model evaluation - J7 model on ttbar :

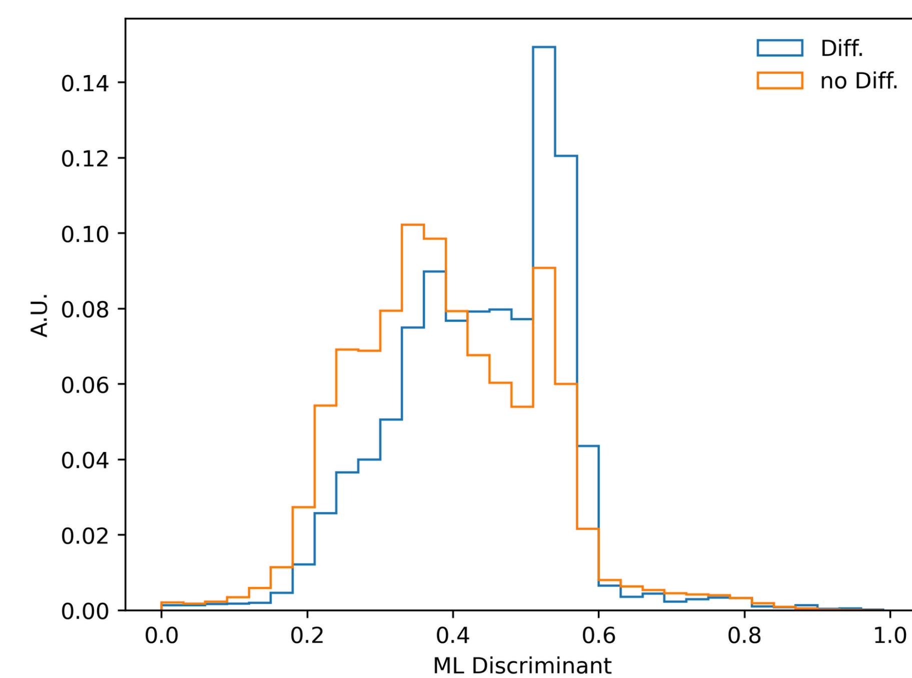
* Input Data:

  - Diff Labels: 93276 samples

  - NoDiff Labels: 4319097 samples

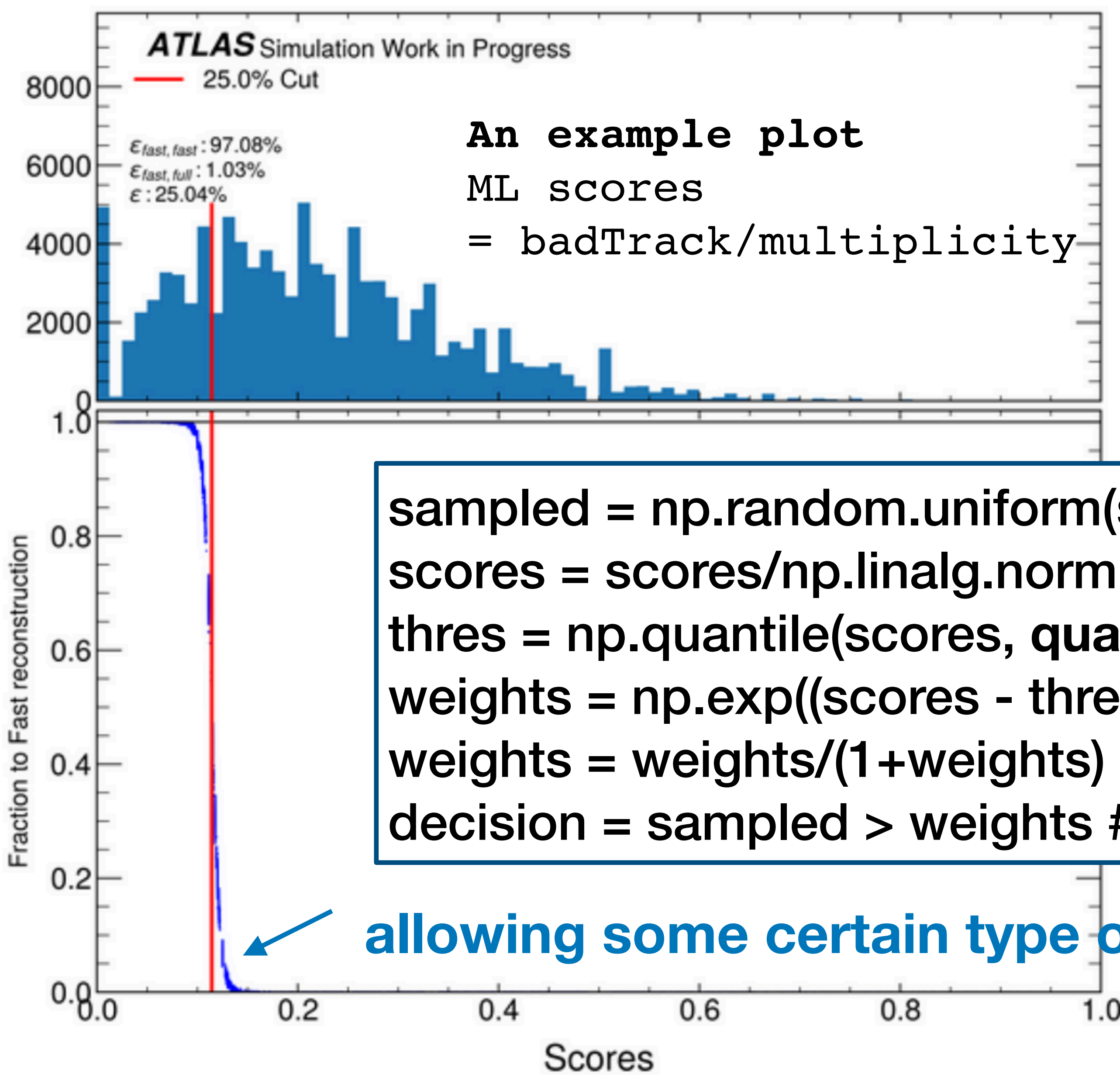
* Features: :['px', 'py', 'pz', 'E', '’pT', 'R02', 'R05', 'sumR02', 'sumR05', 'sumpT02', ‘sumpT05’]

* Conditioning Features: ["wo_numvtx", "w_numvtx", "TruthMultiplicity", "eventPt", "wo_leadJetpT", "wo_nleadJetpT", "w_leadJetpT", “w_nleadJetpT"]



Roulette events selection with random sampling

- Event Classification: the number of BadTracks/ the total number of tracks.
- The final Roulette decision involves generating random samples, normalizing scores, calculating thresholds, assigning event weights to determine whether an event should be classified as track-overlay.




```
sampled = np.random.uniform(size=num_samples)
scores = scores/np.linalg.norm(scores)
thres = np.quantile(scores, quantile) # 25% events going to track-overlay.
weights = np.exp((scores - thres)/temperature) #temperature=0.00005
weights = weights/(1+weights)
decision = sampled > weights #if decision = True -> going to Track overlay.
```

allowing some certain type of events going to track overlay.