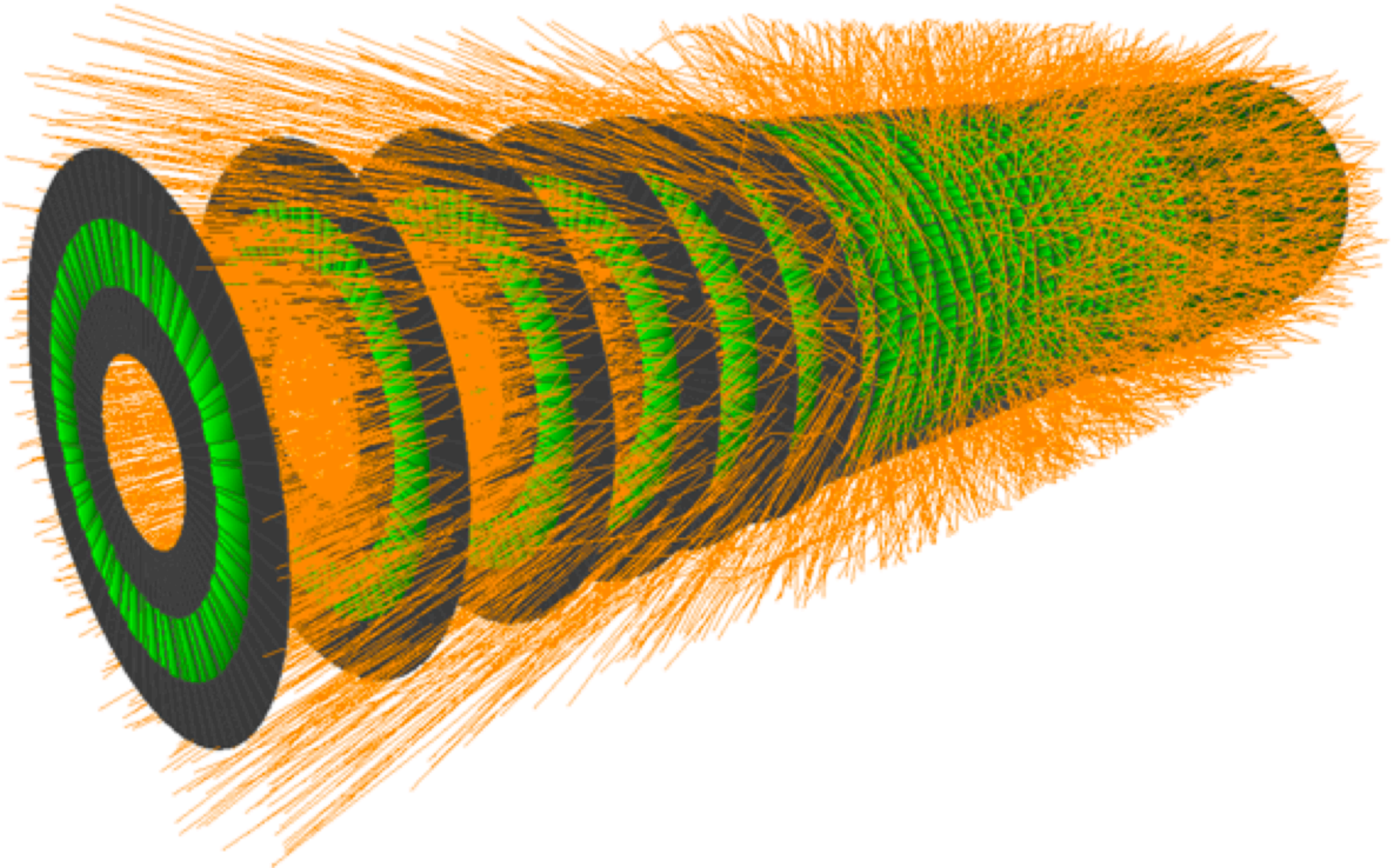
Triggering and online calibration with machine learning techniques



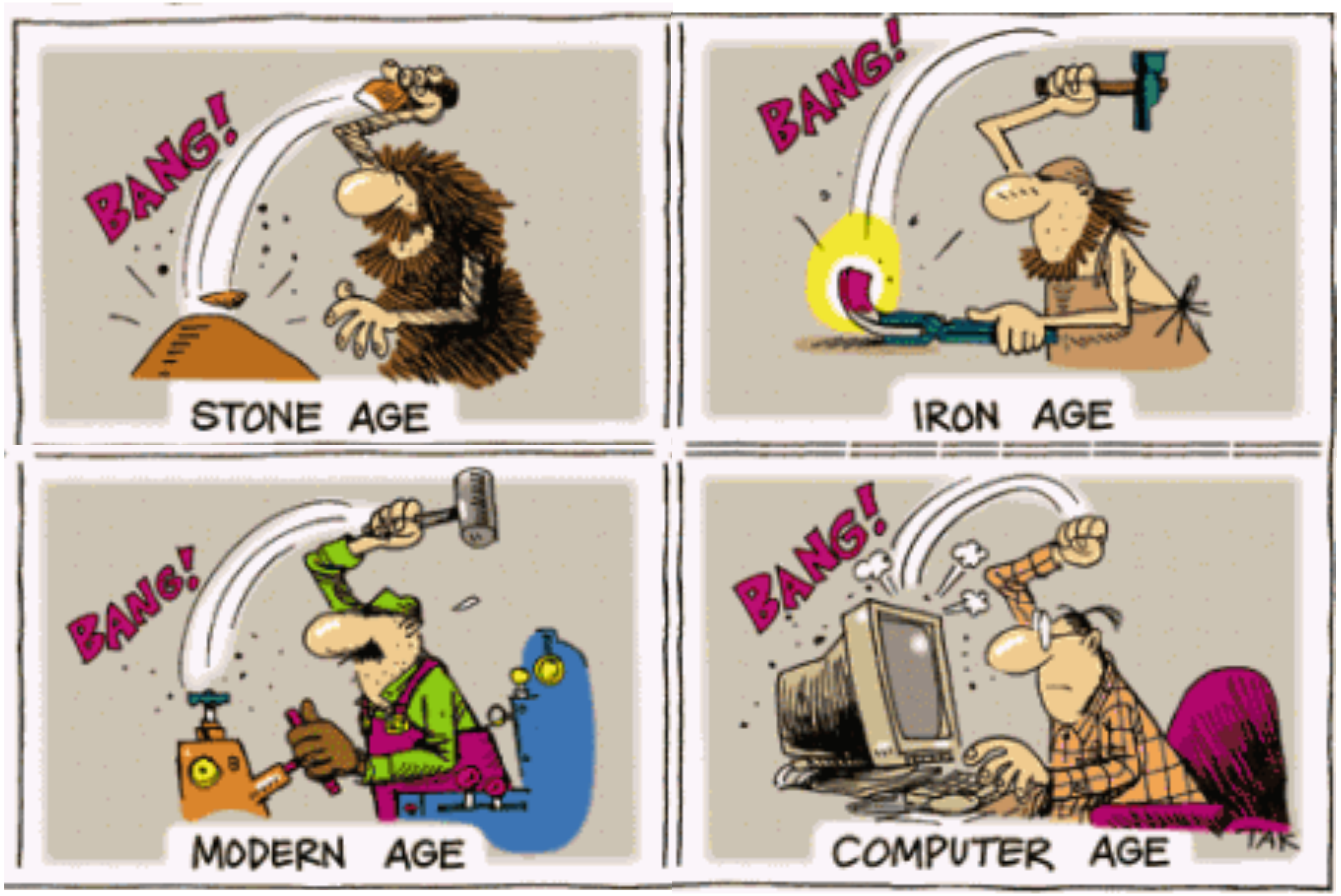
Vladimir V. Gligorov, CNRS/LPNHE

With material from the LHCb, CMS, ATLAS, and ALICE collaborations

LHCP 2020, a virtual space in Paris, 25.05.2020

The ages of ML at the LHC

Courtesy of funnyjunk



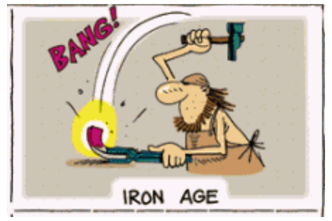
Disclaimer: this is a 15 minute talk and does not attempt to be comprehensive

At first we just didn't want to mess it up

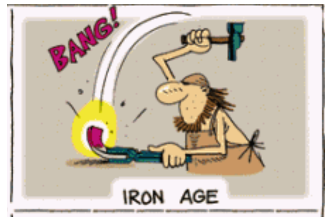
Pre Run-I: can we trust our detector?

...but then we got more ambitious

Pre Run-I: can we trust our detector?

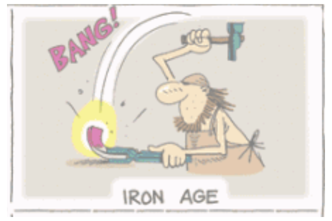


Run-I: yes! Adoption of BDTs across offline analyses and for real-time classification.

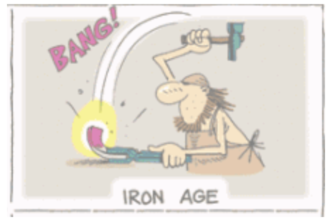


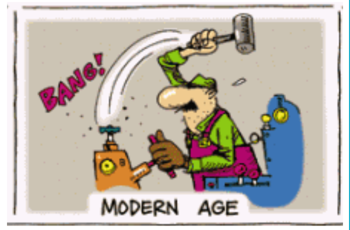
In the twilight of the classification epoch

Pre Run-I: can we trust our detector?

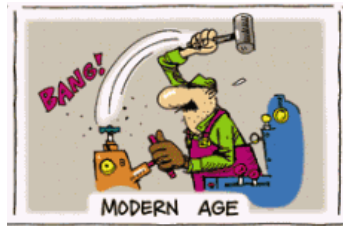


Run-I: yes! Adoption of BDTs across offline analyses and for real-time classification.



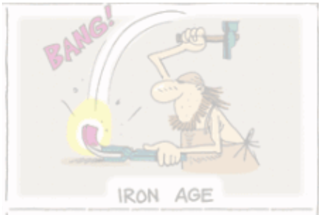


Run-II: increasing use of real-time alignment, calibration, & analysis. Deploy ML to assist classical feature building, in generators & for calibration.

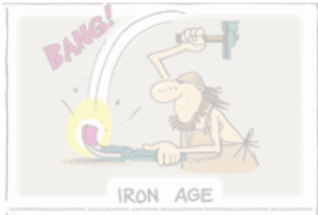


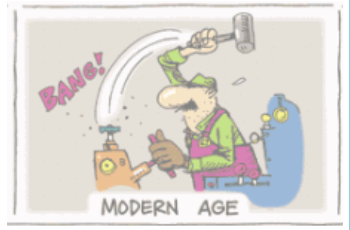
...looking to the pattern recognition age

Pre Run-I: can we trust our detector?

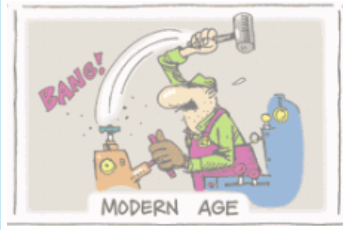


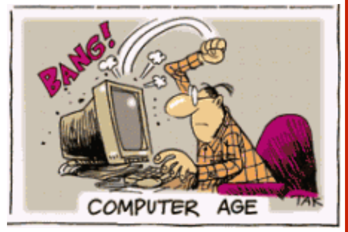
Run-I: yes! Adoption of BDTs across offline analyses and for real-time classification.



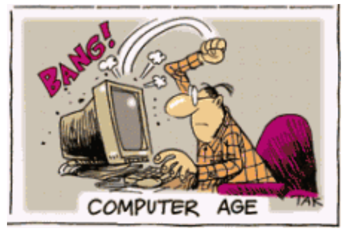


Run-II: increasing use of real-time alignment, calibration, & analysis. Deploy ML to assist classical feature building, in generators & for calibration.



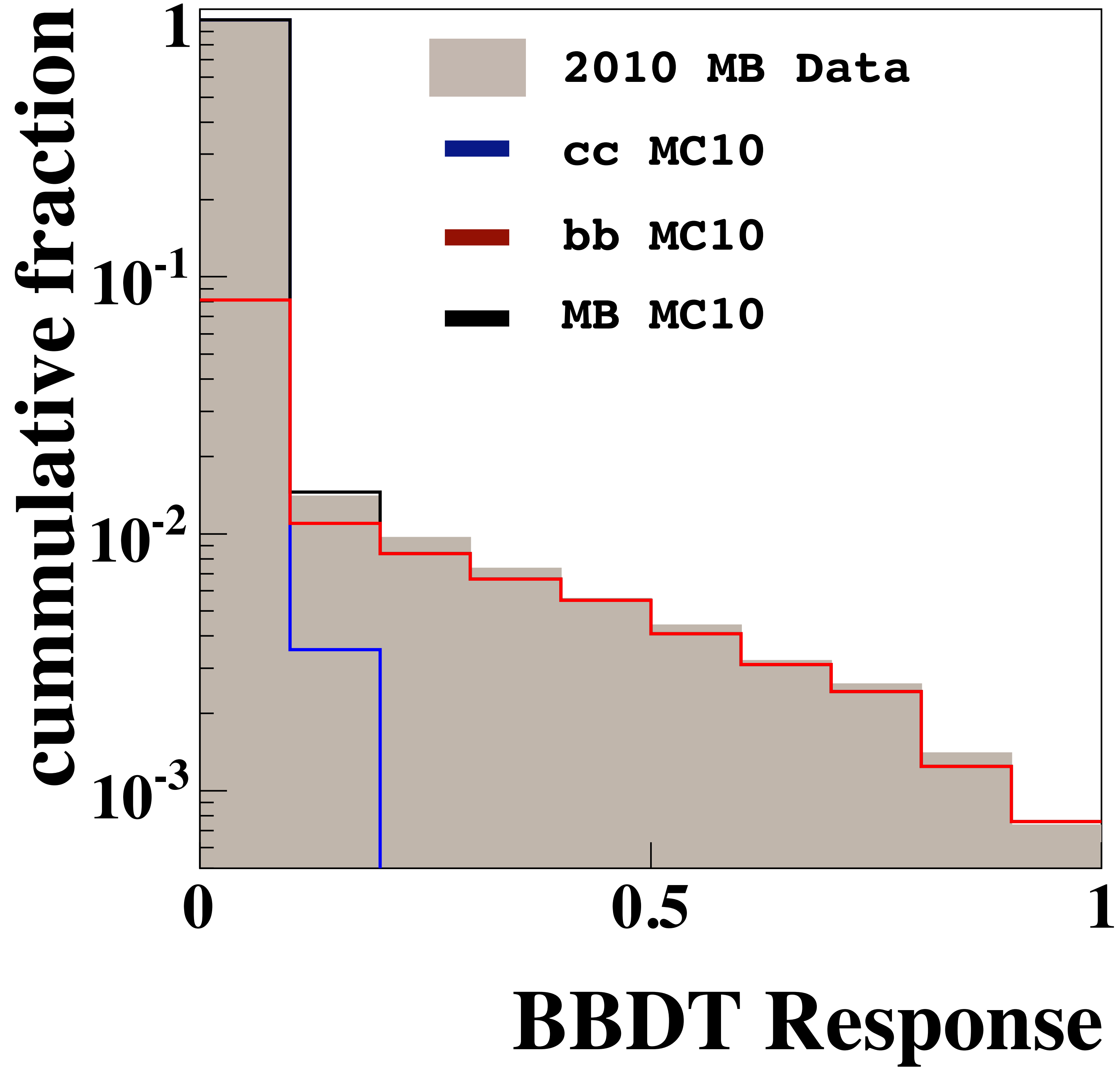


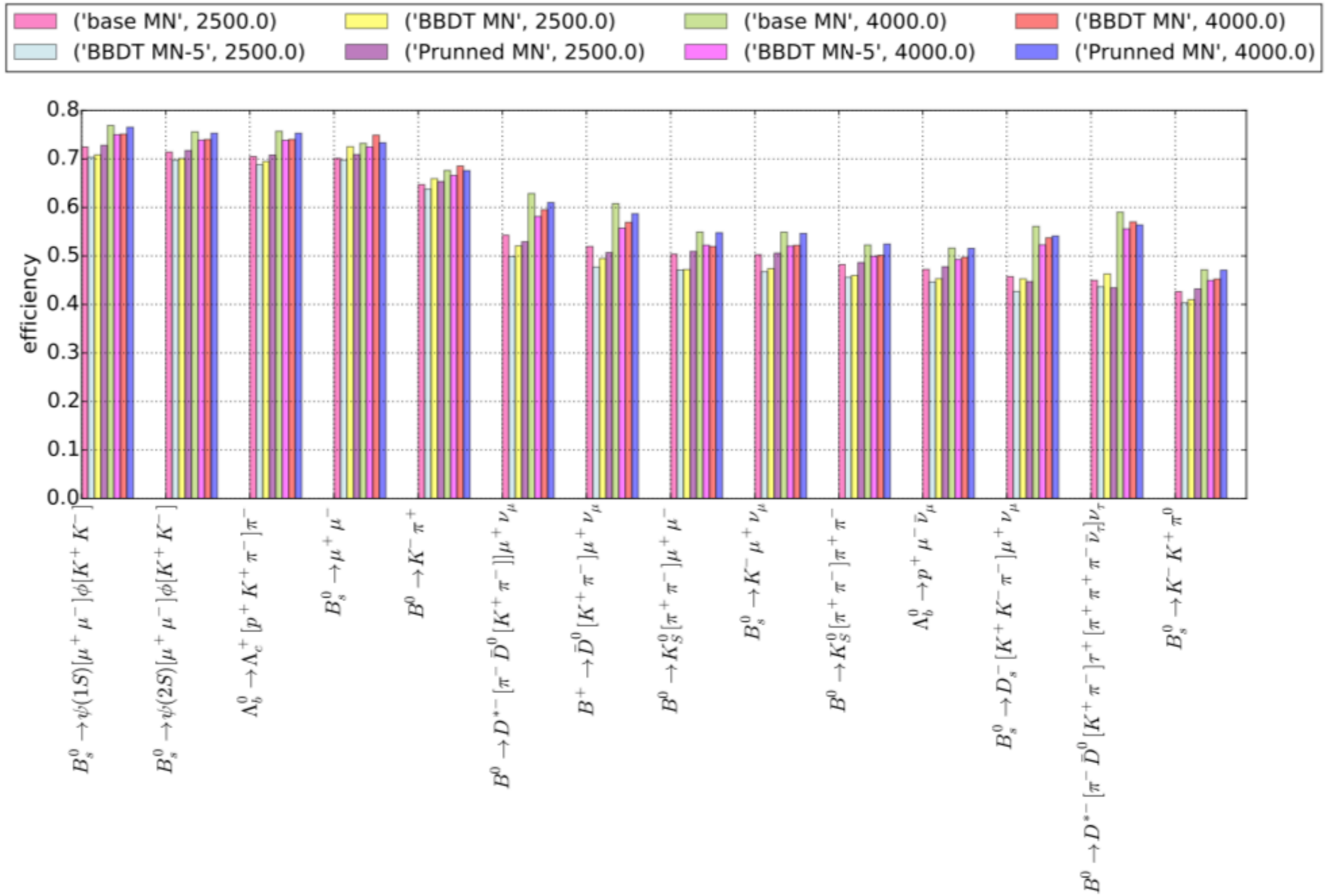
Run-III: Full interleaving of classical and ML methods in reconstruction and analysis, real-time & offline?



Classification

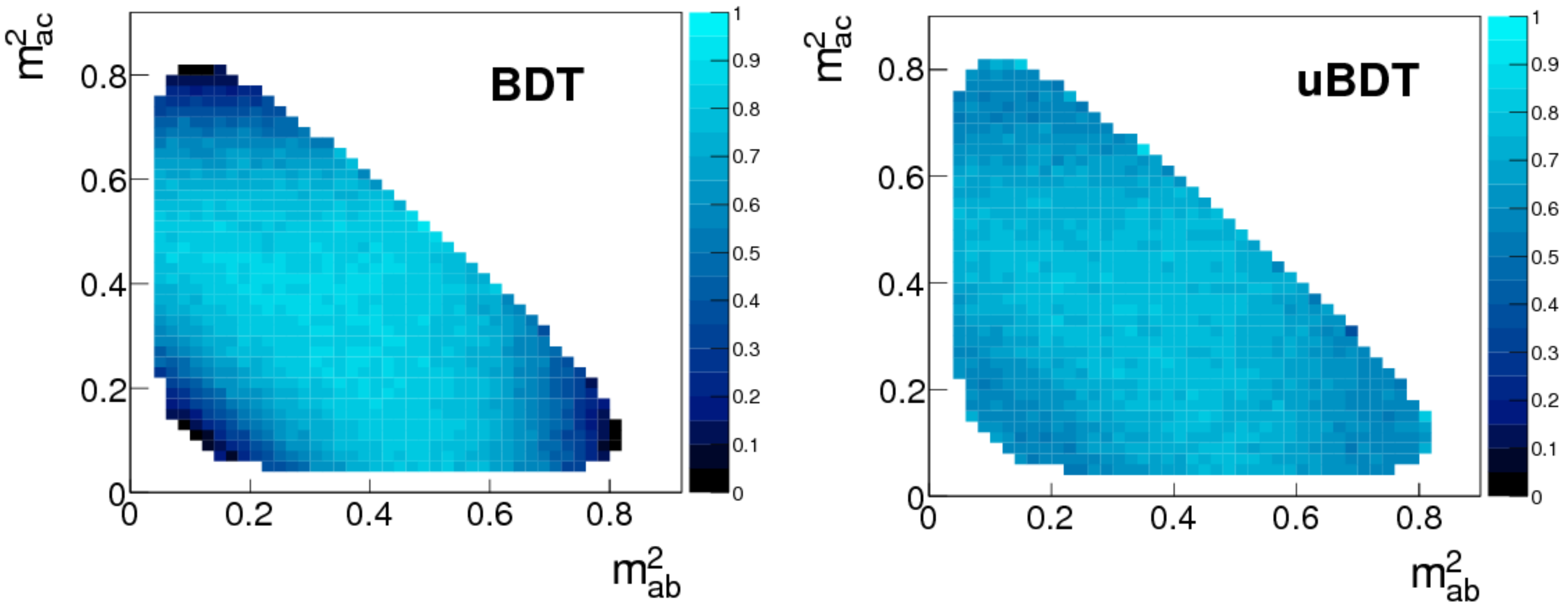
ML for real-time classification at LHCb





Main inclusive beauty trigger based on a BDT since 2011, widespread use of different classifiers since. Classifier implementation optimized for execution speed from the start. See the BBDT paper, Run 2 topological trigger proceedings, LHCb PUB notes 2011-002,003,016, the 2011, 2012, and Run 2 trigger performance papers.

Flattening classifier efficiency curves

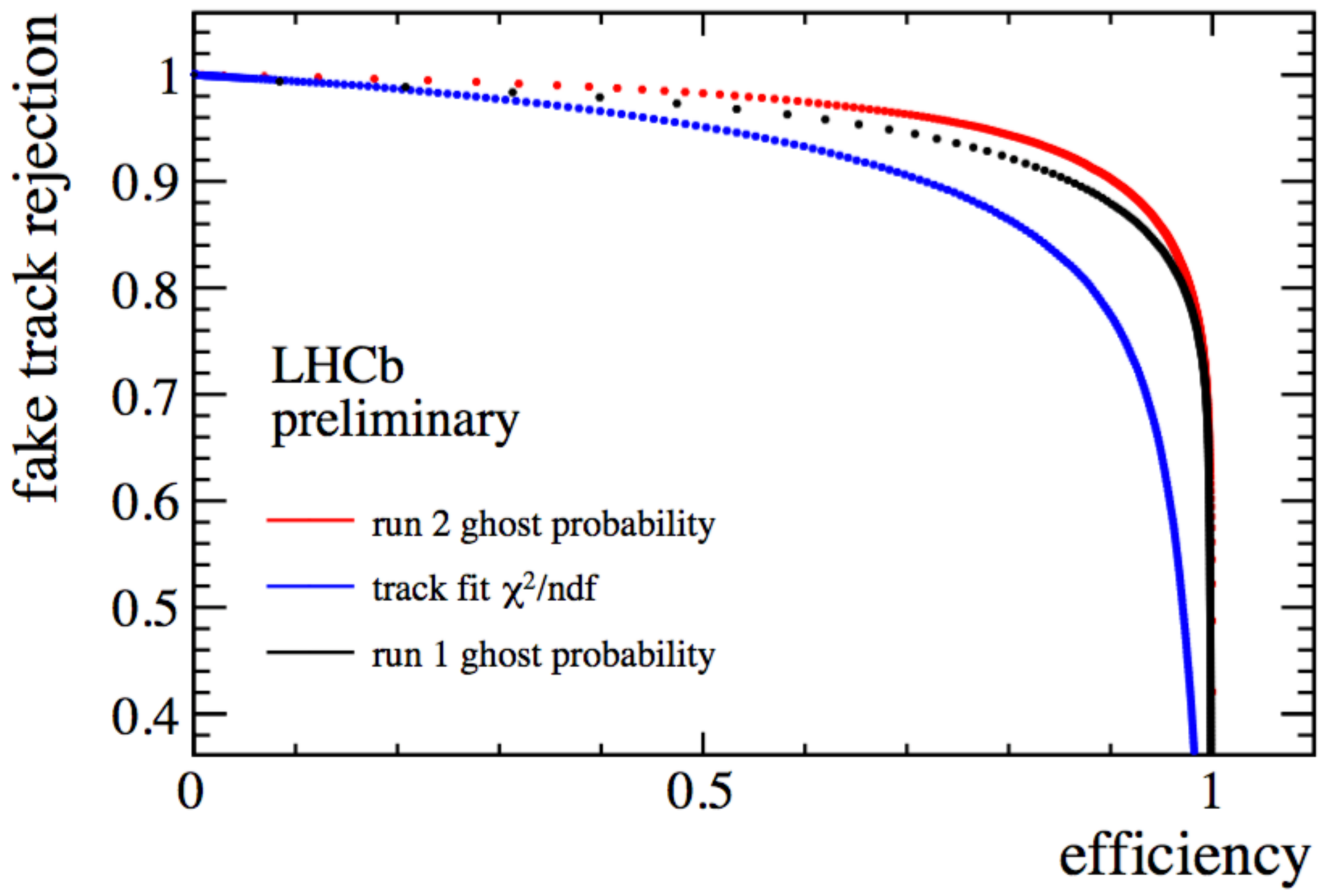


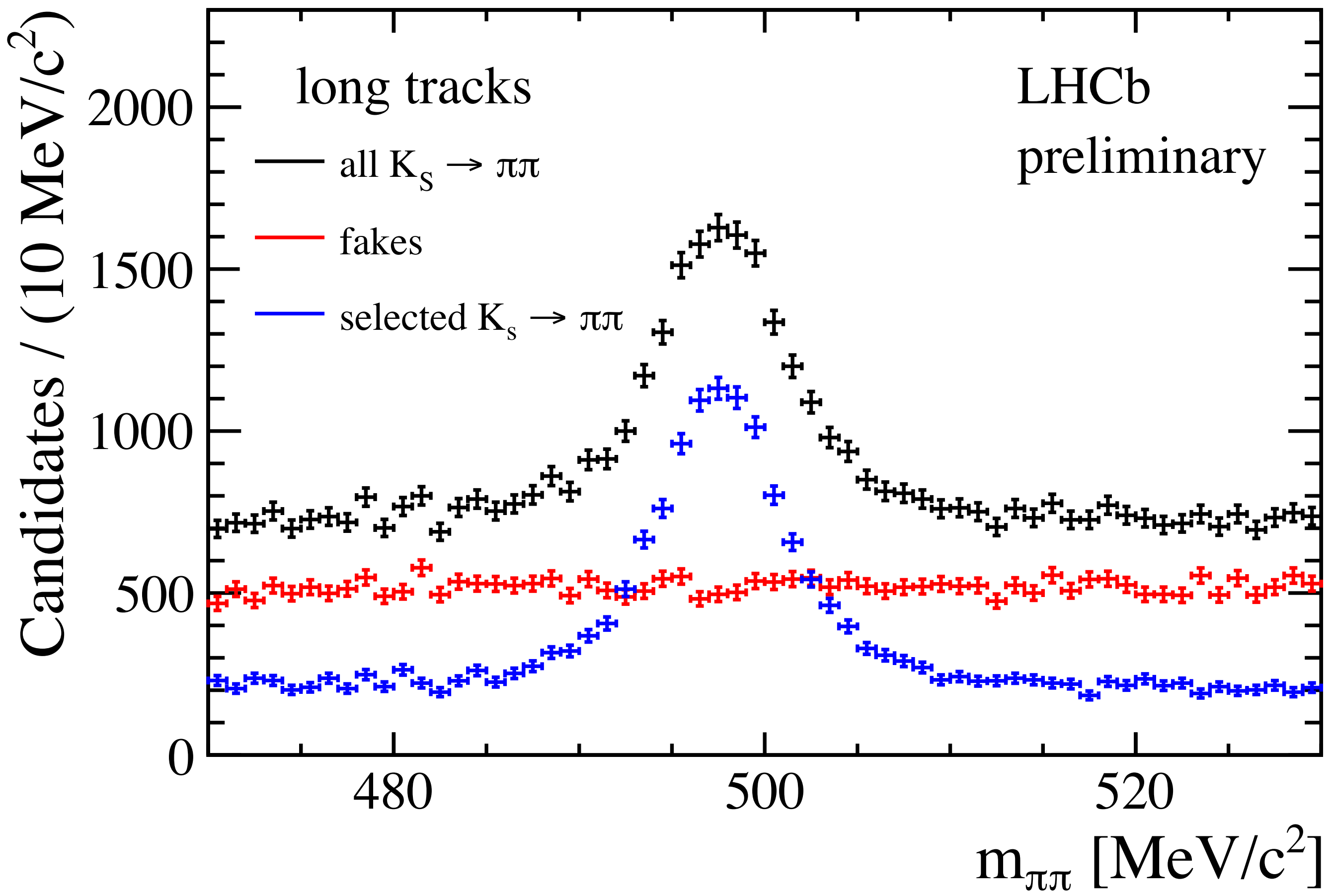
Efficiency of a BDT classifier as a function of position in the Dalitz plane for a three body meson decay. Left: without training for flatness. Right: with training for flatness.

For analyses which may be systematics limited, we might well be happy to trade the last few percent of classifier performance to have a flatter efficiency dependence on kinematic variables. Again part of the toolkit since the earliest days, see Stevens&Williams and Rogozhnikov et al. One lesson of the last decade is that well designed ML classifiers are actually less biasing than "simple orthogonal cuts". With a stress on "well designed"!

Fake track classification at LHCb

De Cian et al. Note

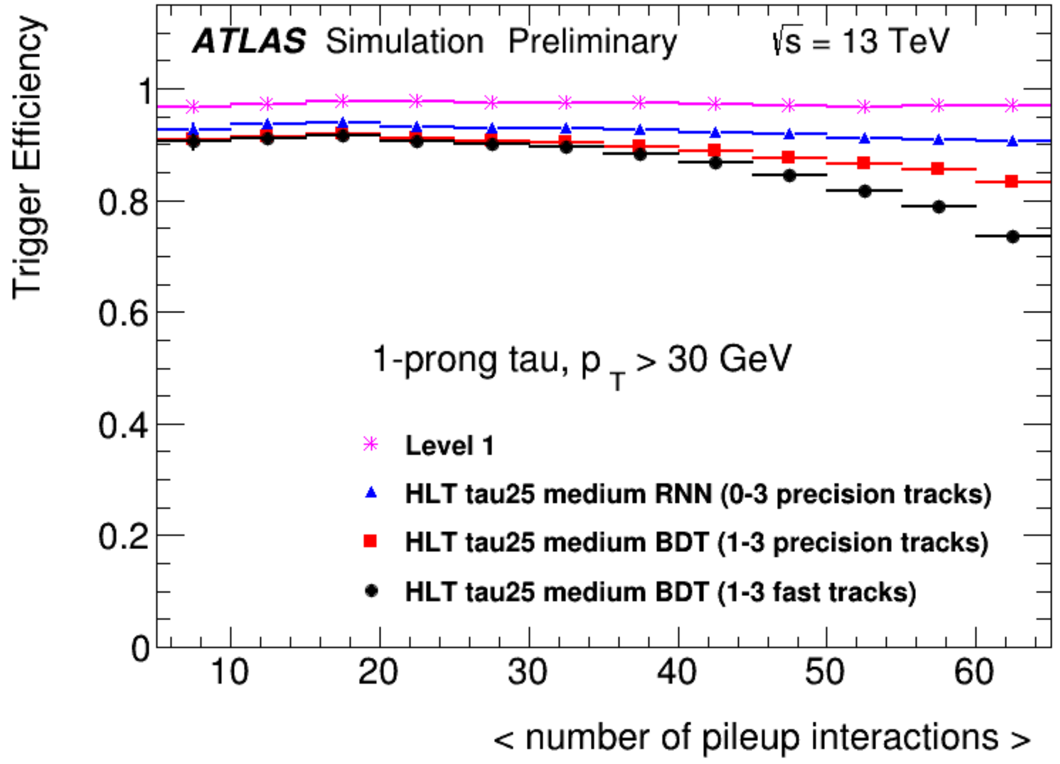




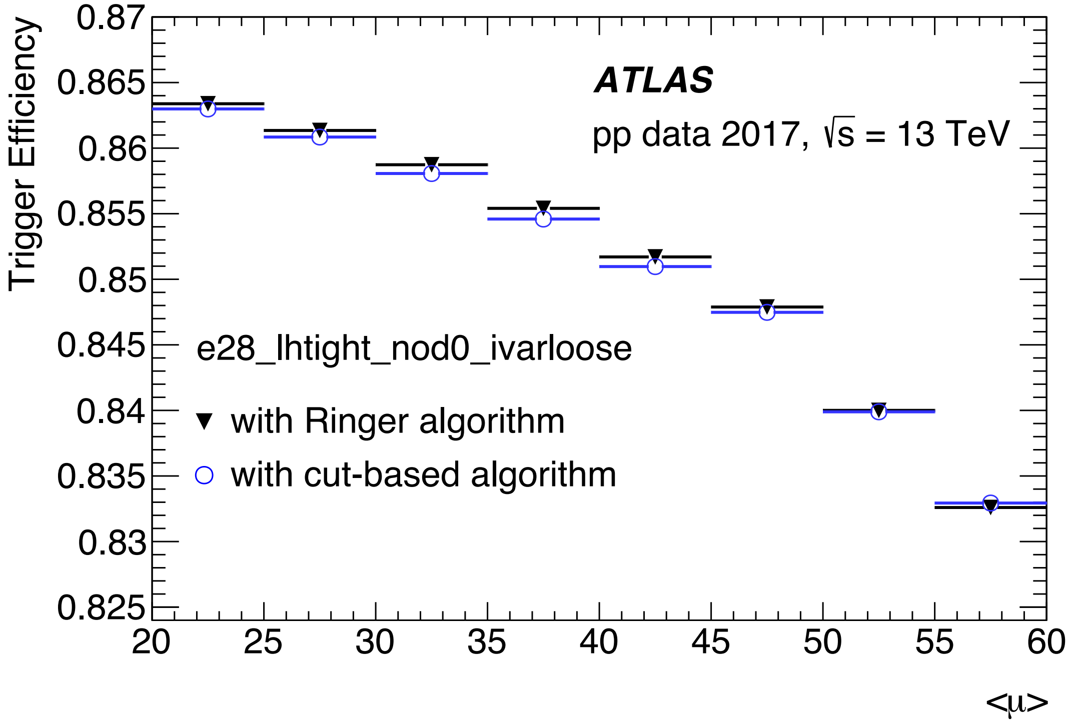
Different particles leave hits in different parts of LHCb. Gain by training an NN to put the different detectors in a global selection. Classifier implementation hand-tuned for execution speed. No room to discuss here but several different classifiers are also used for particle identification at LHCb, including in the trigger.

Trigger classification at ATLAS

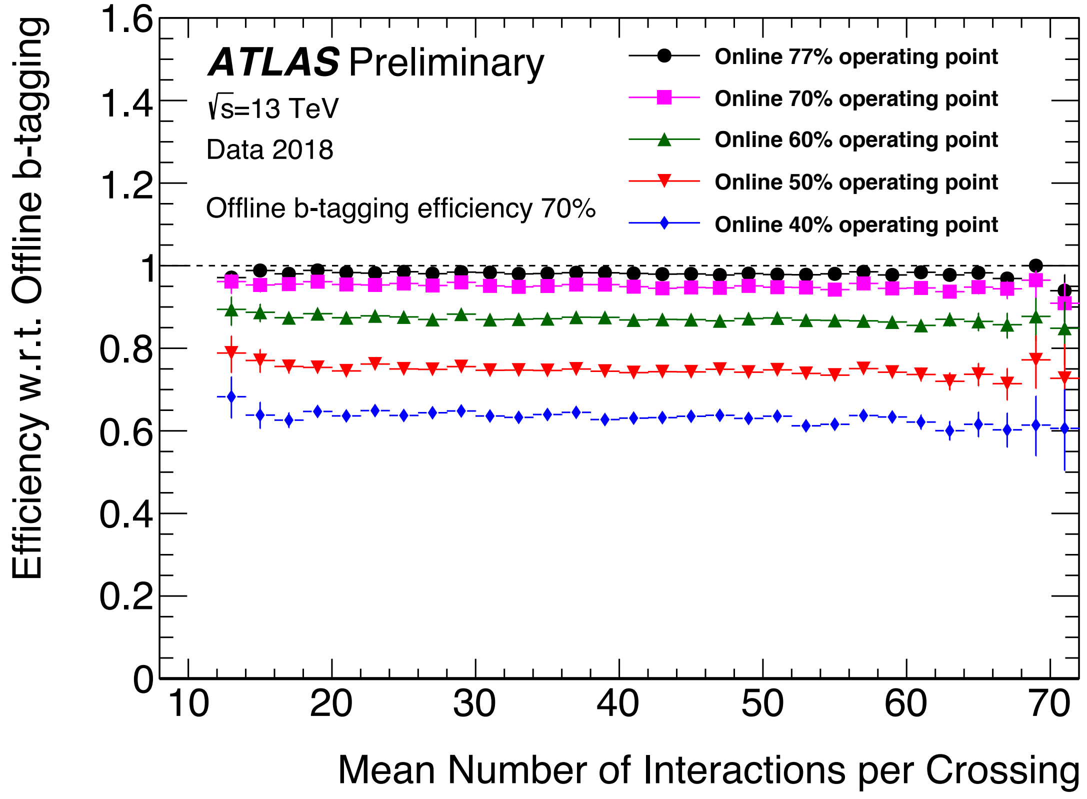
Performance of Tau HLT algorithms



Performance of NN based electron ID



Performance of online b-jet tagging



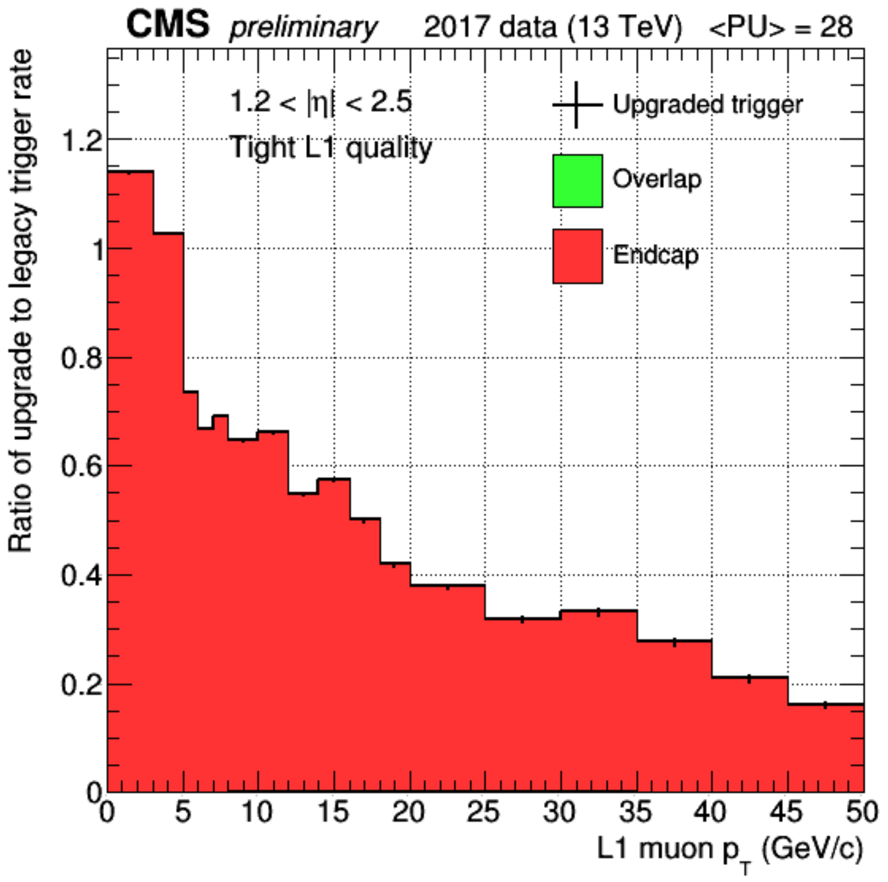
Evolving from BDTs to NNs, with large rate reductions at the same efficiency working point and latency reduction through reducing combinatorics earlier in processing chain.

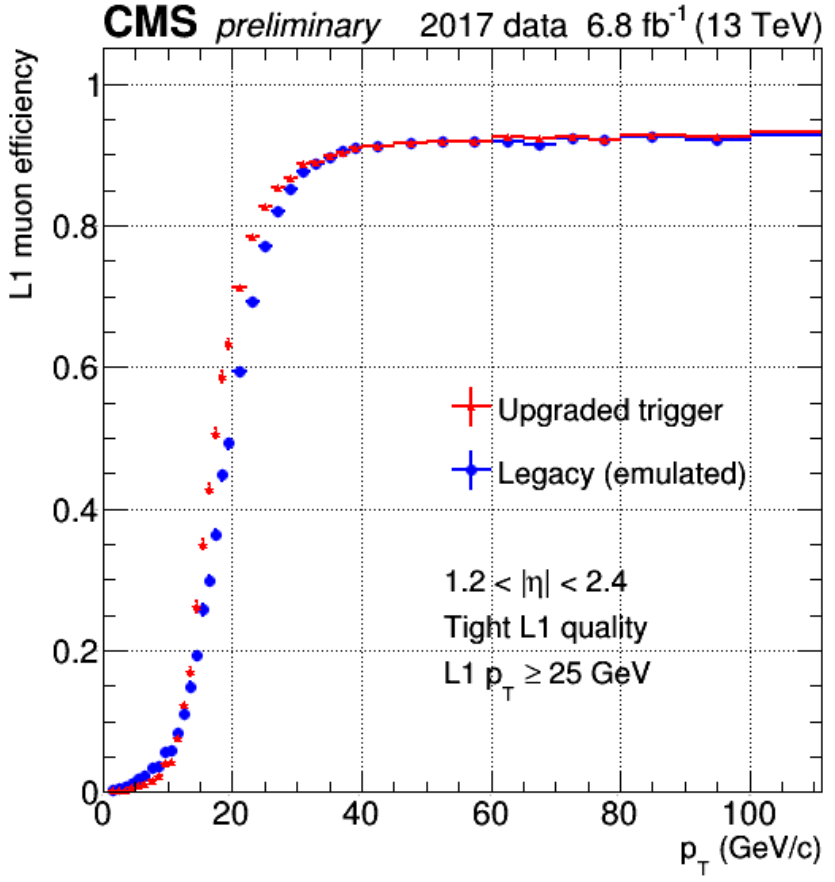
- Electron triggers make use of a neural-network-based ringer algorithm for an early background rejection and overall CPU reduction.
- Tau triggers make use of boosted decision tree and recurrent neural network algorithms for hadronically decaying tau-lepton identification.
- b-jet triggers make use of a multivariate b-tagging algorithm (MV2) which consists of a boosted decision tree algorithm that combines the outputs of low-level taggers: impact parameter-based algorithms (IP2D/IP3D), a secondary vertex finding algorithm (SV1) and a topological multi-vertex finding algorithm (JETFITTER)

See Tau classification public results, e/gamma performance paper, and b-jet ID paper for more details.

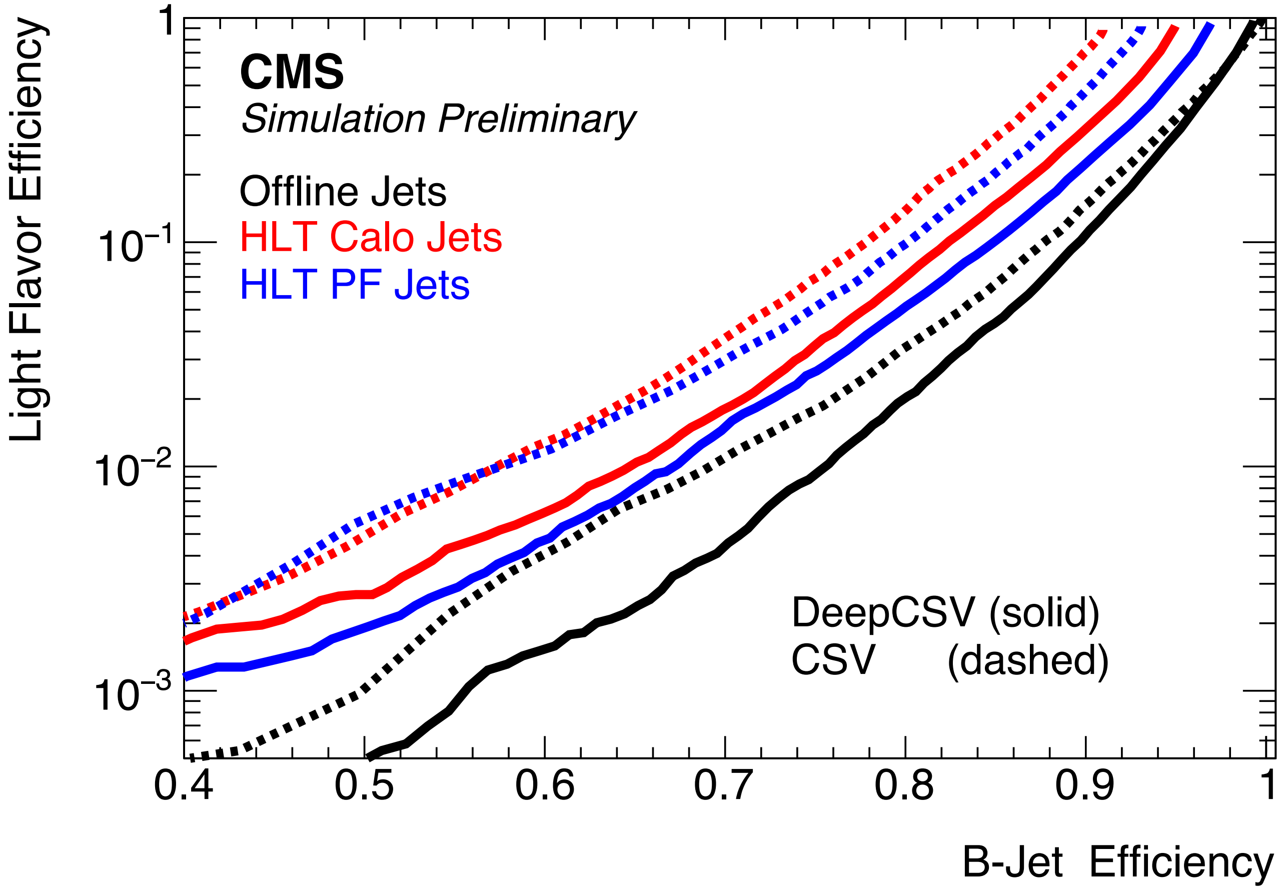
Trigger classification at CMS

Performance of L1 muon endcap trigger





Performance of deep b-jet tagging



In general selection in the CMS trigger is largely cut-based, but BDTs are used for some calibration tasks and track classification. Some specific use cases are

- **The L1 muon endcap trigger is based on a BDT, factor 3 rate reduction at same efficiency**
- **The b-tag triggers are based on a NN deep csv**

Personal remark: interesting that ATLAS, CMS, and LHCb all use ML of one kind or another to tag beauty hadrons or beauty jets in their triggers.

See the L1 muon endcap public note and b-jet trigger performance plots page for more details.

Deploying classifiers in FPGAs

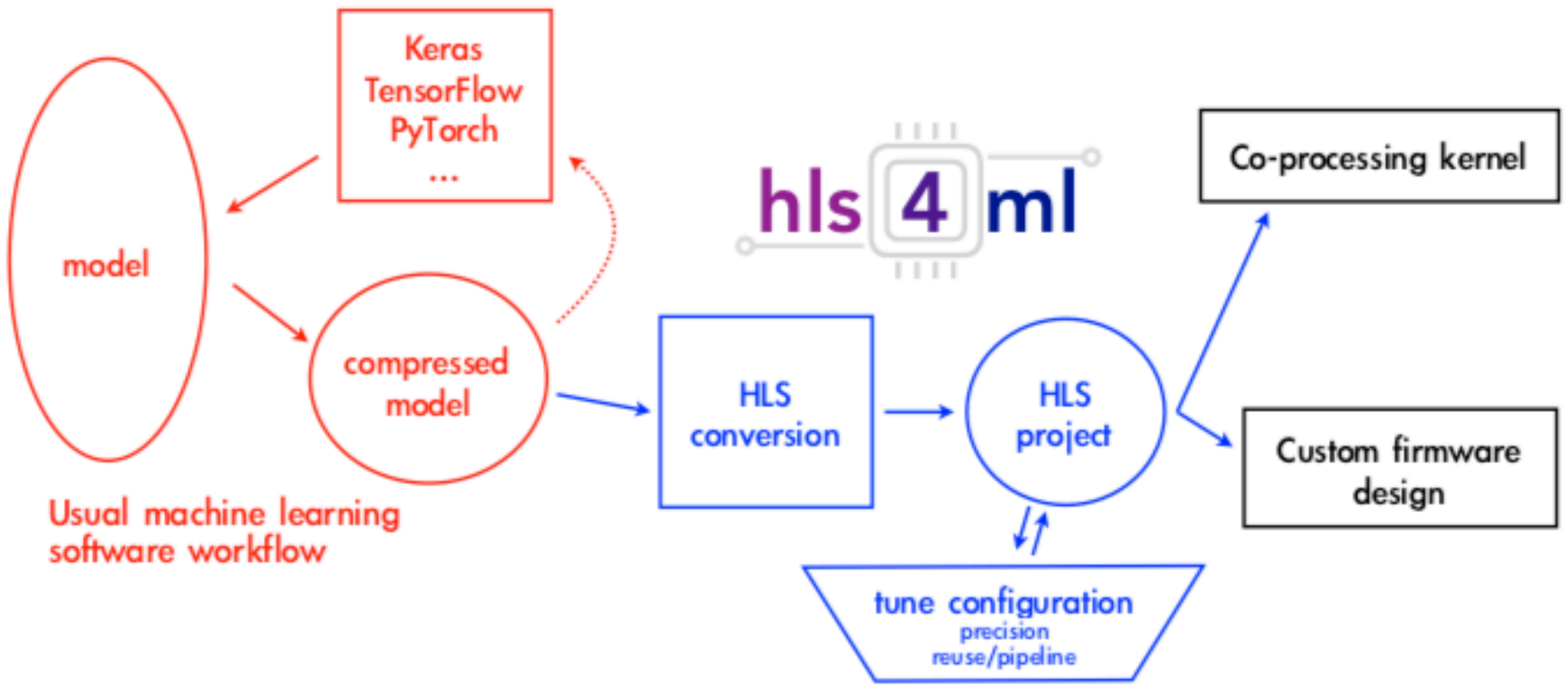
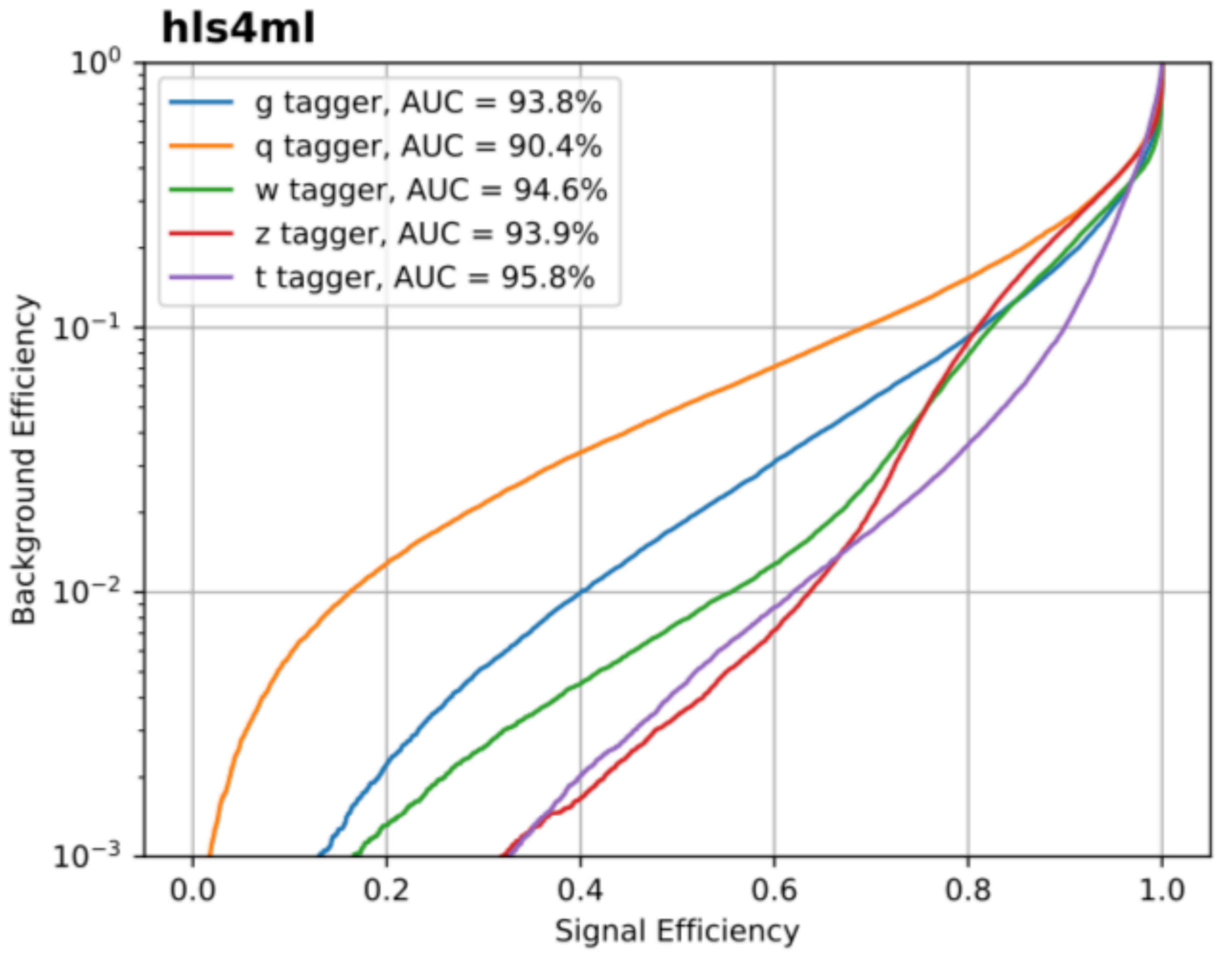


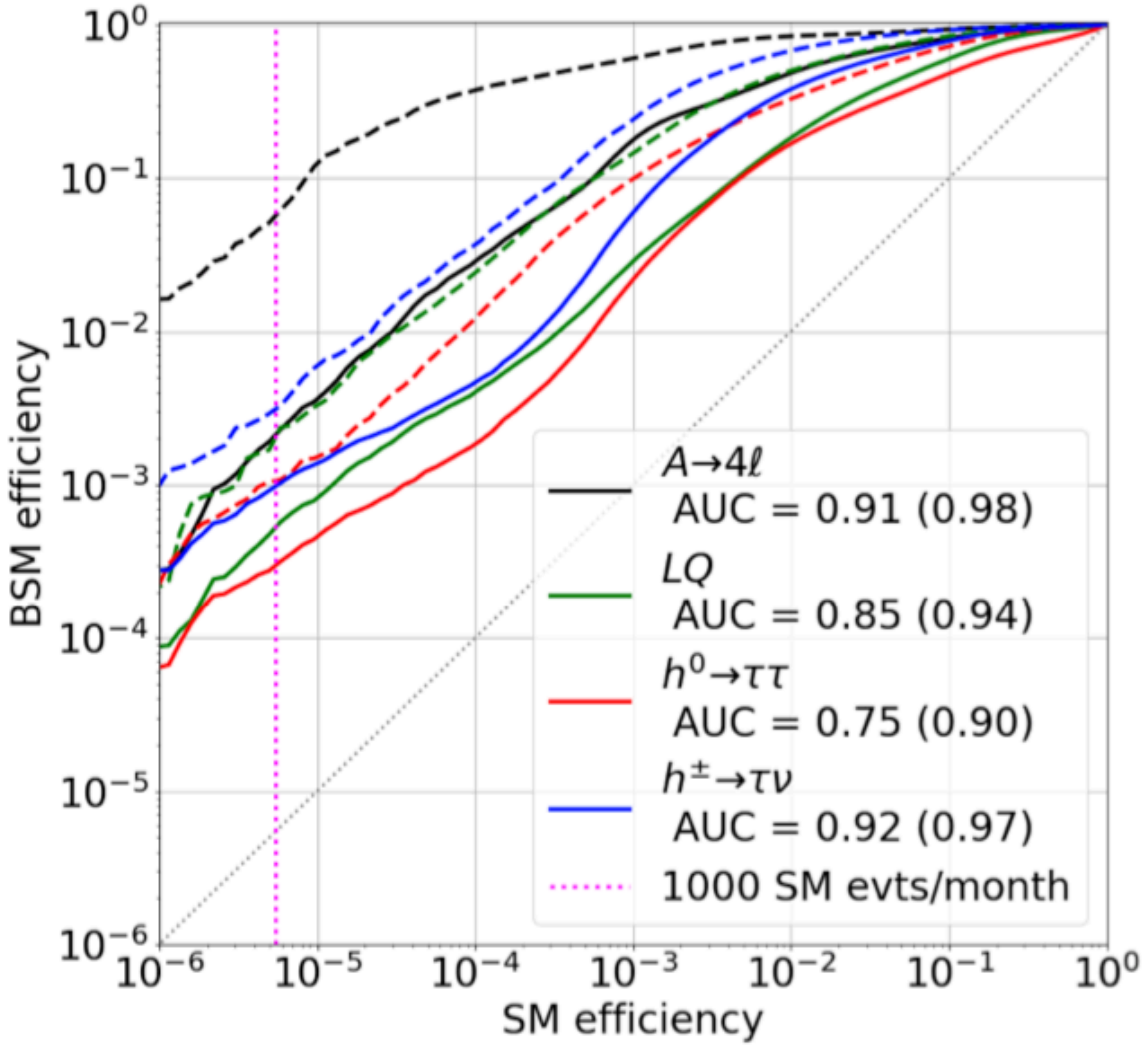
Figure 1: A typical workflow to translate a model into a FPGA implementation using `hls4ml`.



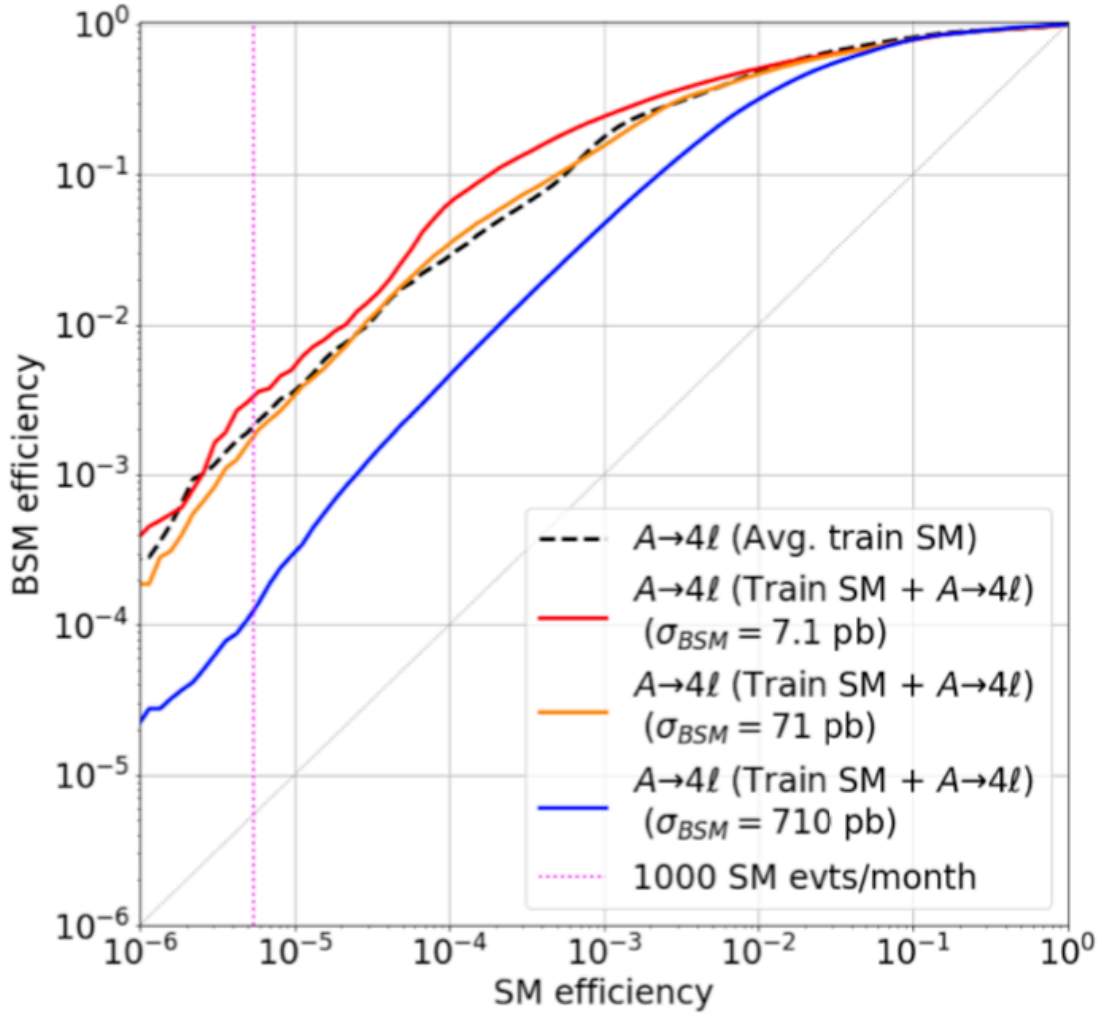
An increasing interest in deploying NN classifiers in FPGAs, which would allow their widespread use in first level triggers or in CPU-FPGA coprocessor based architectures. Compression of NNs is particularly important in the first level trigger case, to fit into available resources. See Duarte et al., Gugliemo et al., Summers et al. for more.

Unsupervised trigger classification

Unsupervised (solid) and supervised (dashed) classifier performance for BSM signals.



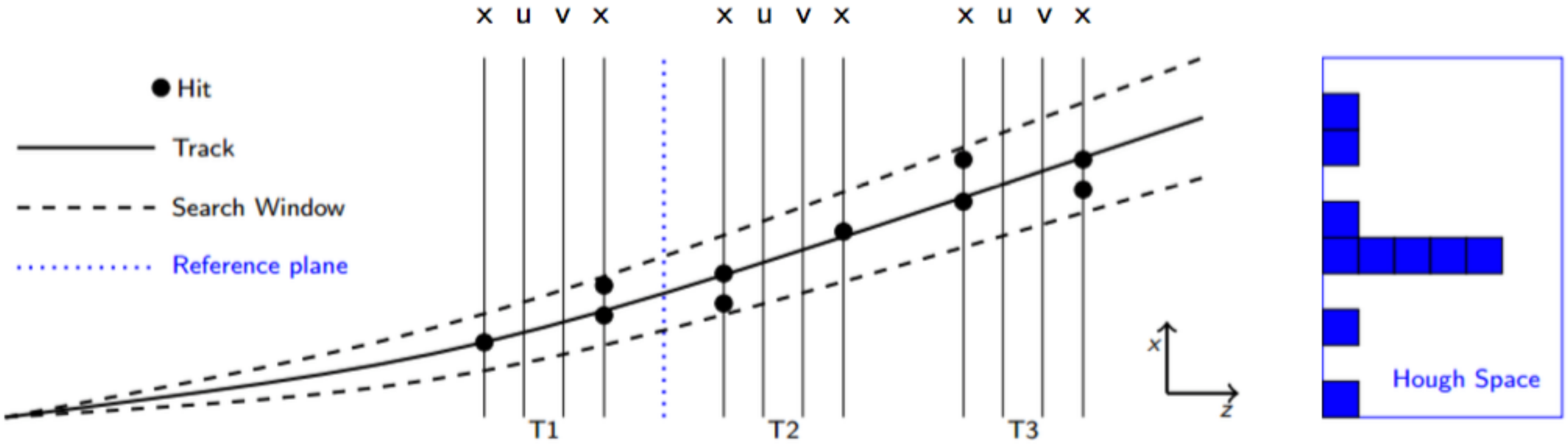
Evolution of unsupervised learning performance when injecting SM background into training sample.



Approaches based on unsupervised learning and targeting a kind of generalized anomaly detection are also being explored in the community. See Cerri et al. for more details.

Reconstruction

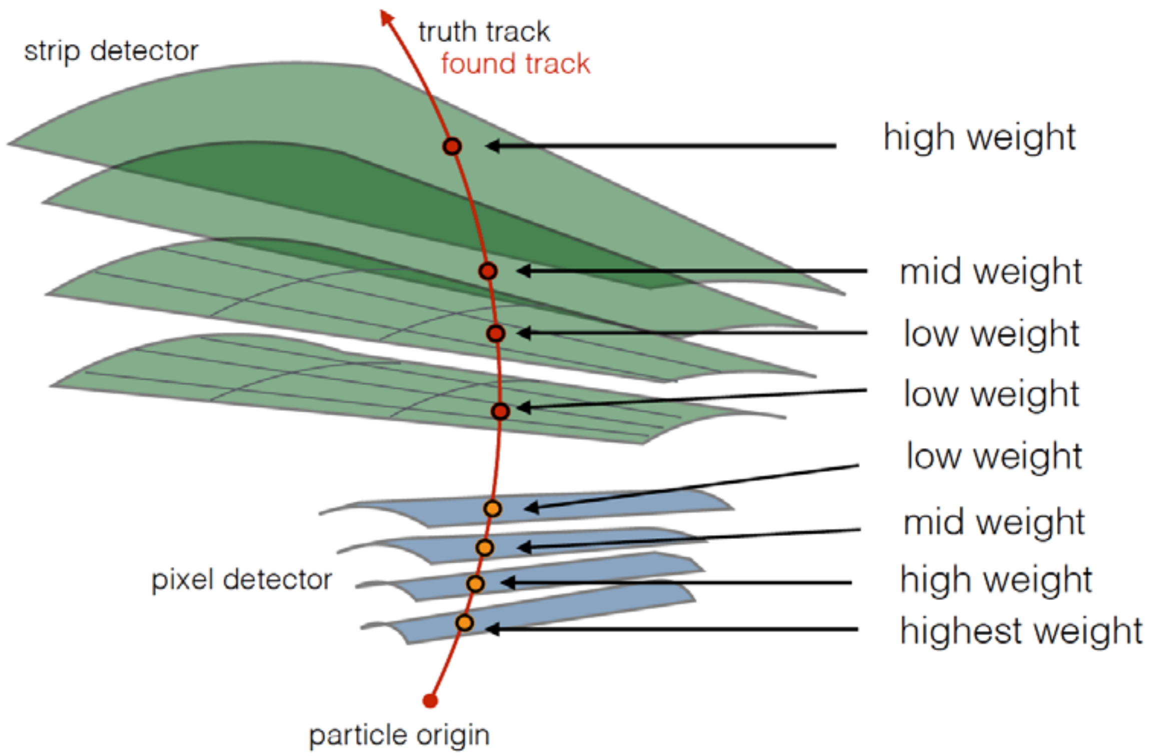
ML classifiers to speed up reconstruction



In LHCb, the first steps towards an ML based reconstruction were to add NN classifiers inside the classical Hough-transform based pattern recognition for an early rejection of bad hit combinations => significant speedup of the pattern recognition code. But the NN is still not making trajectories from hits, but rather classifying trajectories.

TrackML challenge & Exa.TrkX projects

Scoring model of TrackML challenge



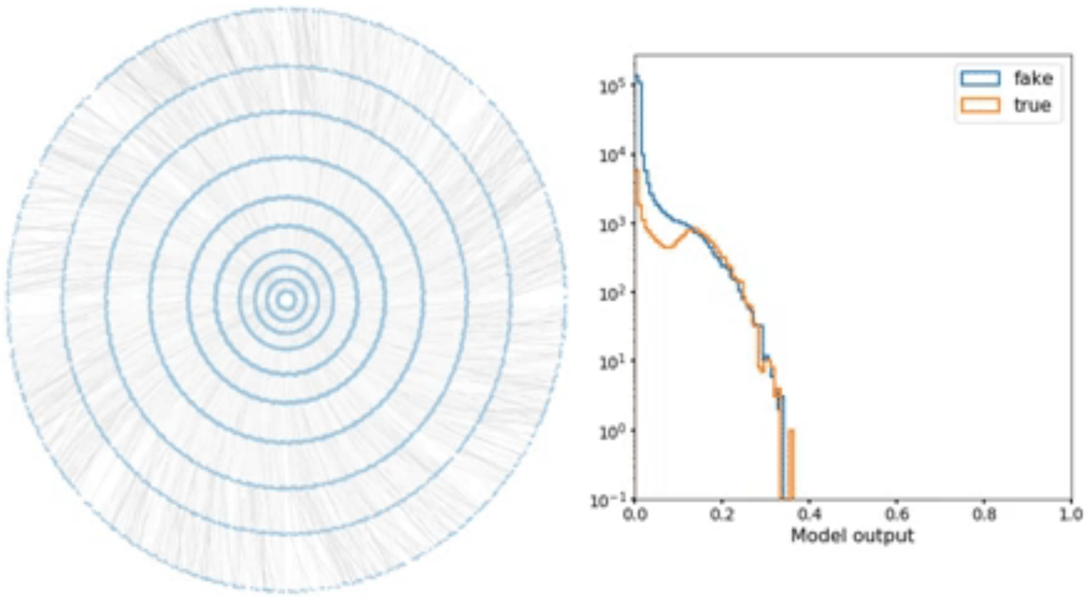
Challenge to see what physics performance (first phase) and throughput (second phase) machine learning approaches could get in track reconstruction.

Input dataset based on a "realistic" HEP detector with the physics score taking into account relevant criteria like "which hit did I miss on the track" to also fold in things like resolution without making the scoring too complex.

Best approaches mixed classical tracking (e.g. Kalman filter style track following) and physically motivated models of the track path through the detector with machine learning aspects in rejecting fake combinatorics and training the algorithm to find the optimal search parameters.

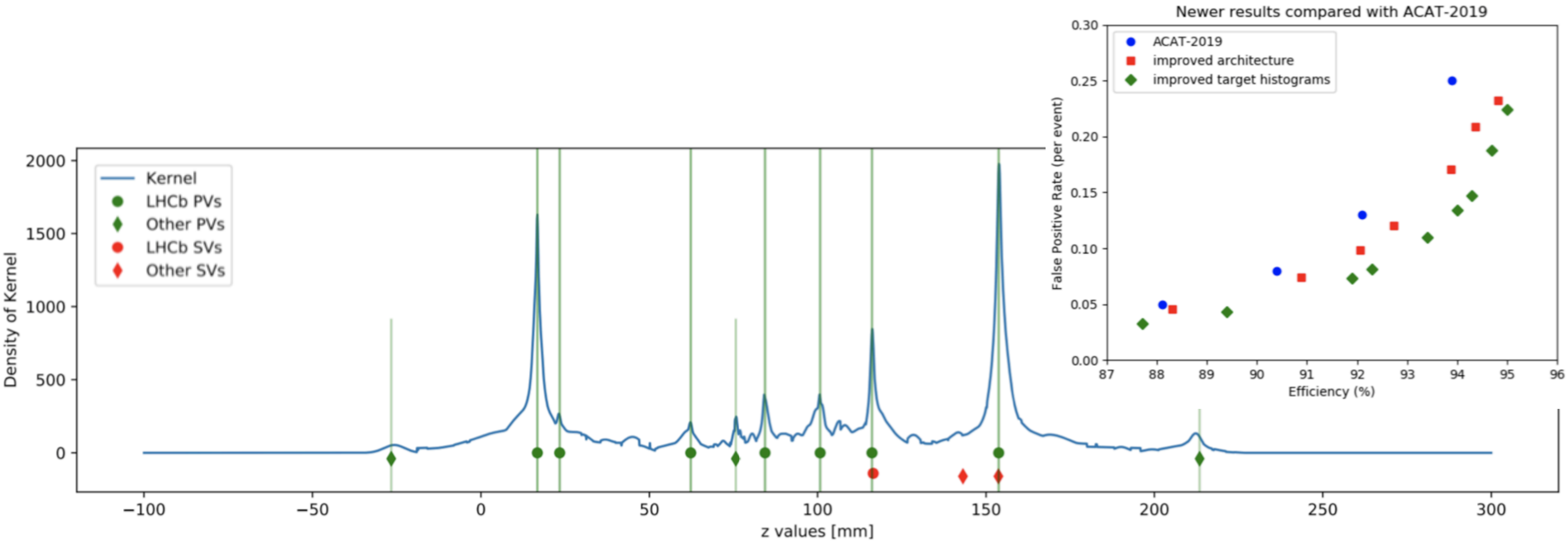
The Exa.TrkX project, which is a continuation of the HEP.TrkX pilot, is looking to use Graph Neural Networks for a more completely "ML based" tracking across different experiments (LHC and beyond, e.g. DUNE).

On the right (click to play) is shown the performance of a track finding Graph Neural Network as a function of the message-passing iteration. Best results are found after eight iterations.



See the [TrackML challenge page](), [Kiehn et al.](), [the final workshop](), and the [Exa.TrkX github repository]()for more information.

NN based primary vertex reconstruction

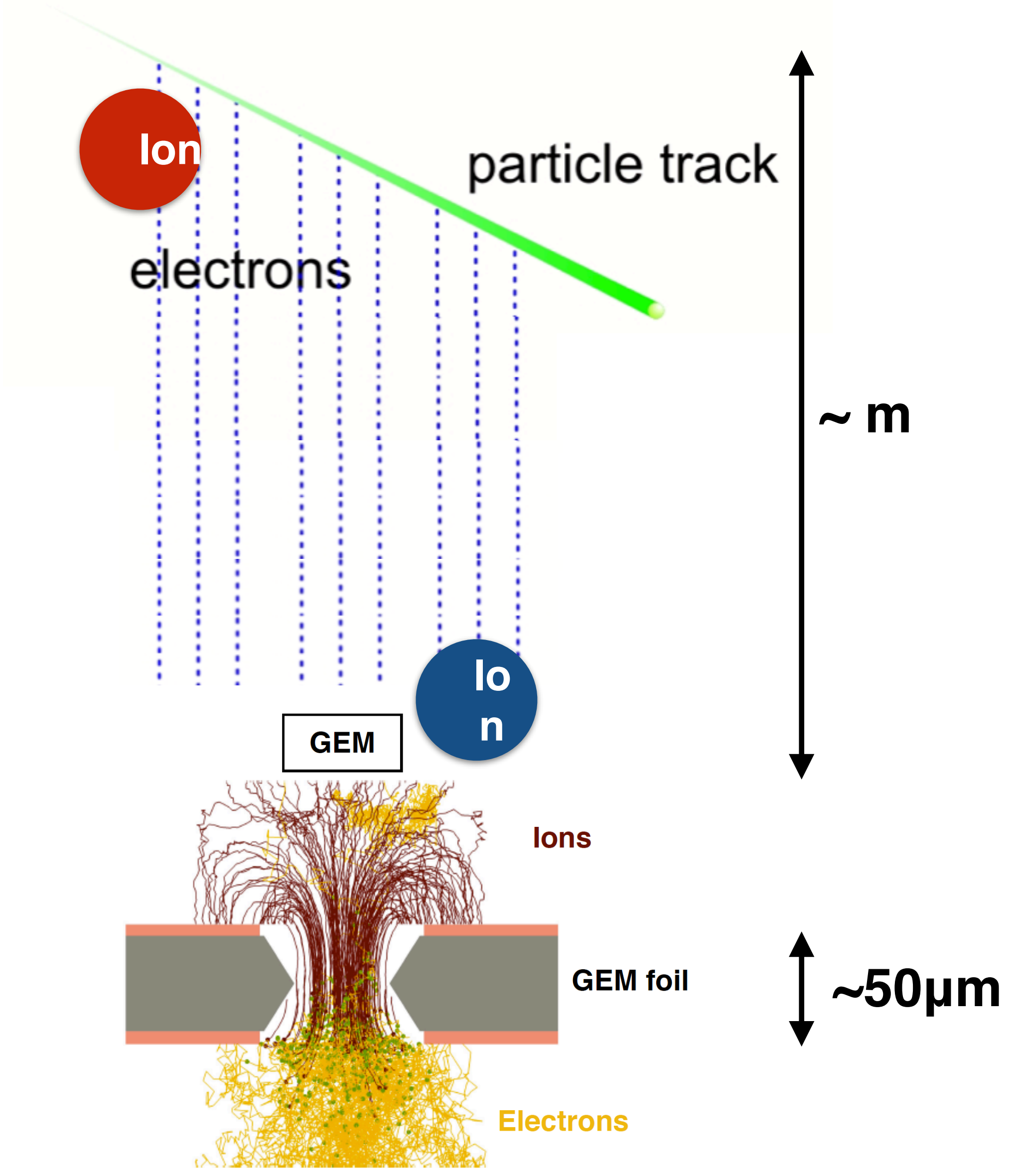


Another hybrid approach: classical algorithm finds the tracks, a NN then finds the PVs. The algorithm's physics performance is very promising, optimization for computational efficiency is ongoing. See Fang et al. and a recent CTD talk for more details.

Calibration

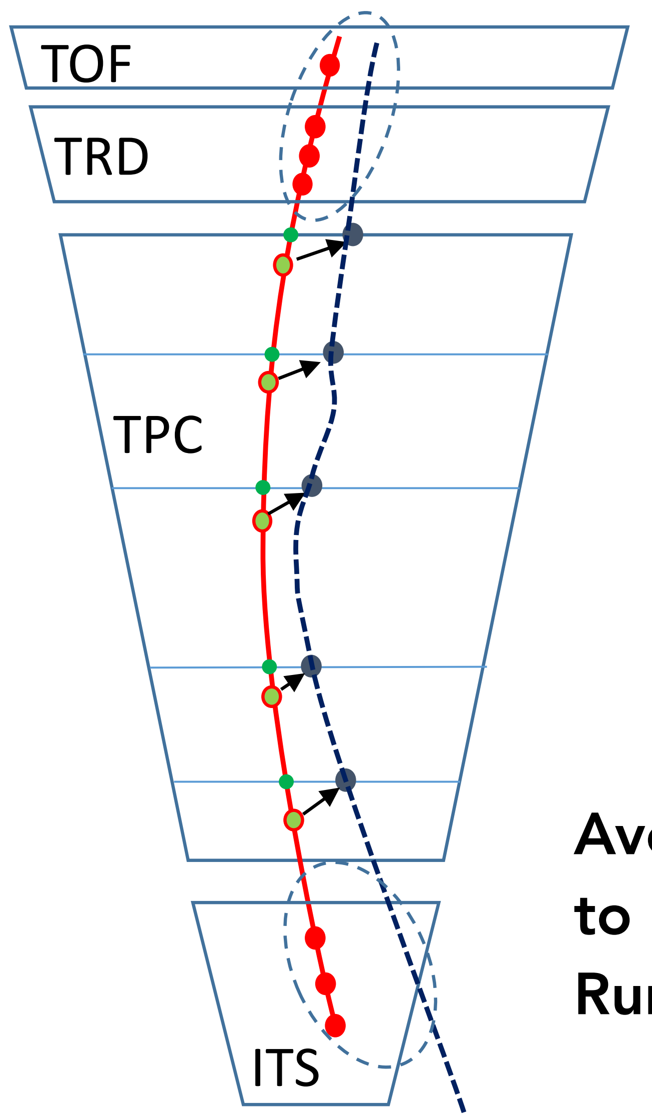
Charge distortion corrections in the ALICE TPC

In Run3, new readout system of the ALICE Time Projection Chamber will allow to collect data in continuous mode



- electron drift time ~ 100μs
- ion drift time ~ 200ms
- Ions belonging to 8000 different PbPb collisions!

GEM detectors will release in the TPC active area slow ions that can distort the electric field that guides the electrons



↓

- average "shift" (distortion) and fluctuations in the positions of the reconstructed TPC clusters

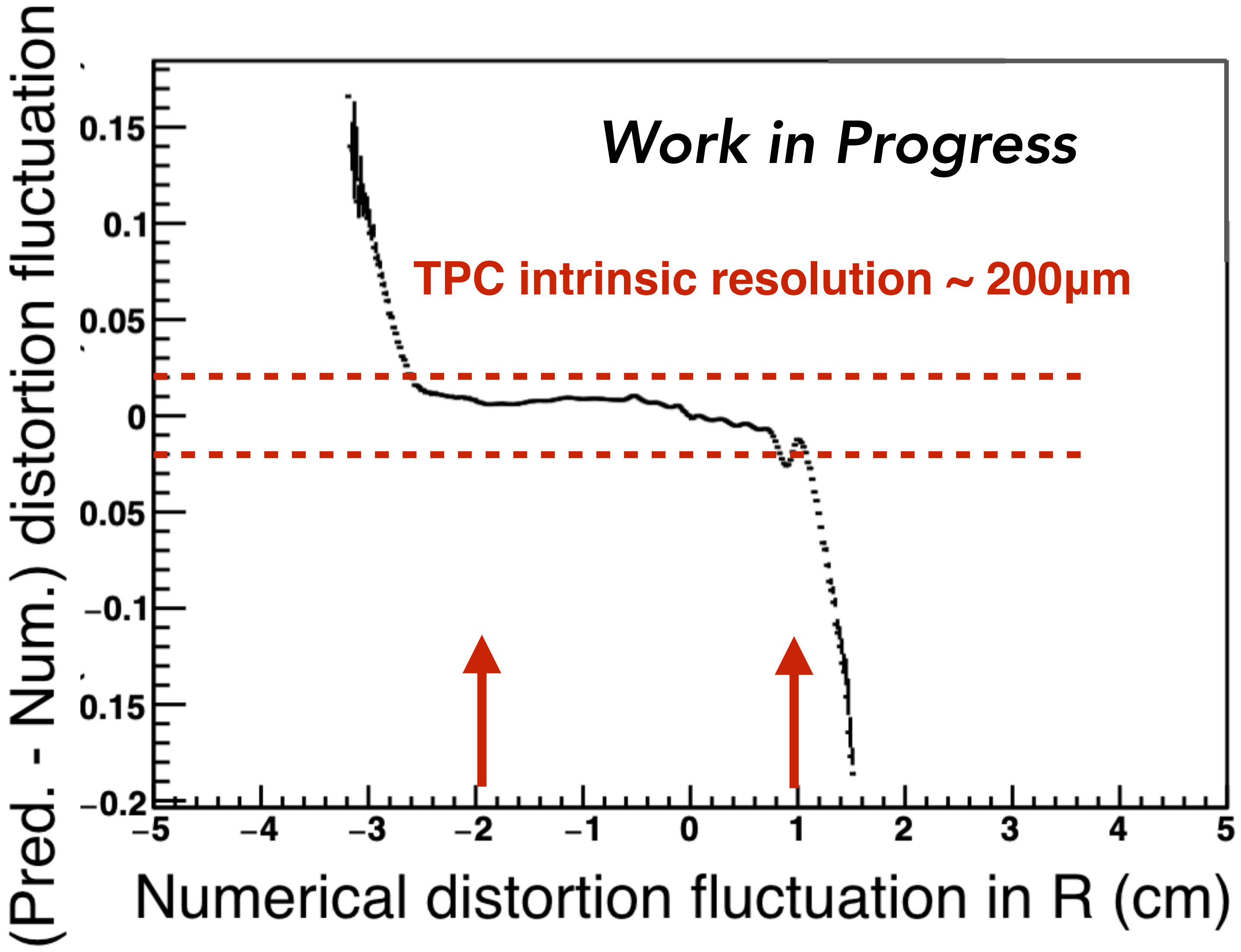
↓

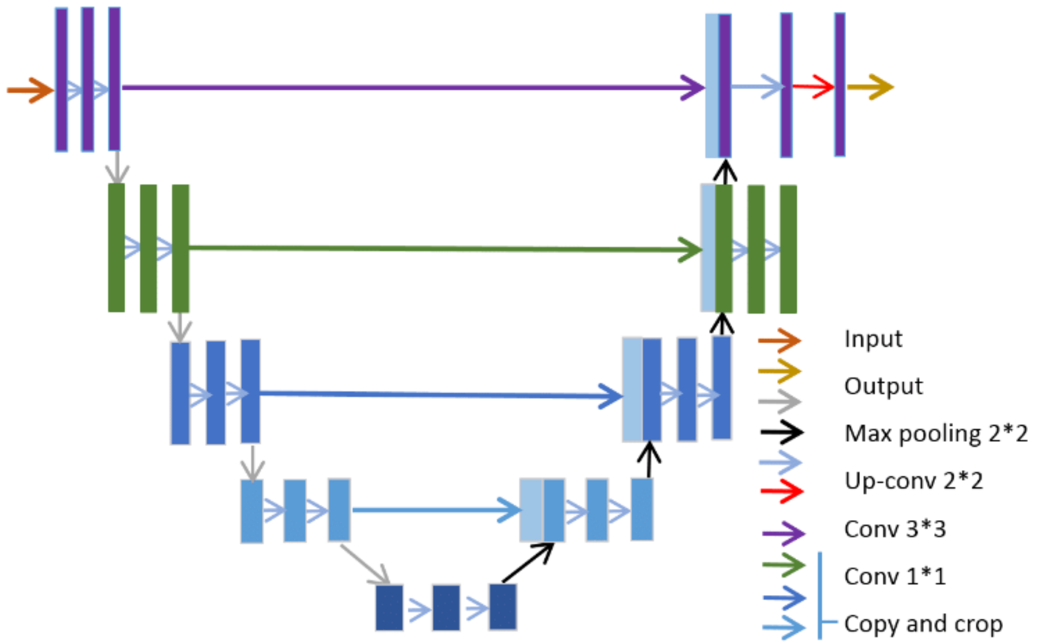
Average distortions and distortion fluctuations need to be corrected before the tracking is performed (in Run3 mostly performed online):

→ Analytical correction procedure is way too slow!

UNets for correcting TPC distortion fluctuations

→UNets (type of convolutional NN) being explored to correct for the TPC distortion fluctuations using as an input the TPC space charge densities or currents





<u>Very promising results:</u>

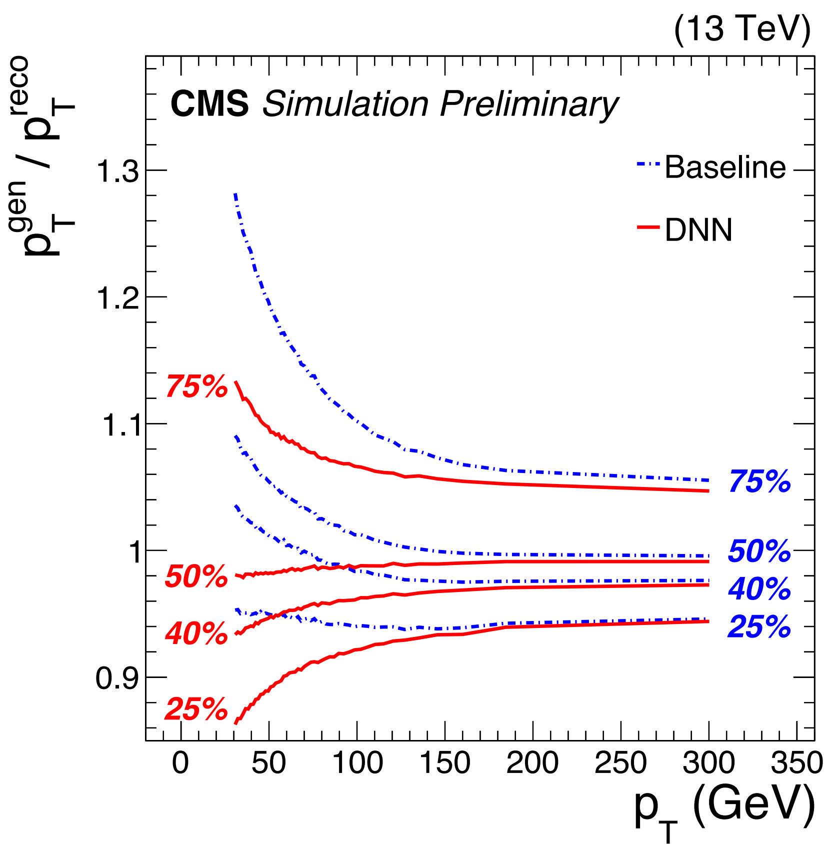
- Resolution of the predicted distortion fluctuations are ~ TPC intrinsic resolution
- orders of magnitude faster than analytical methods

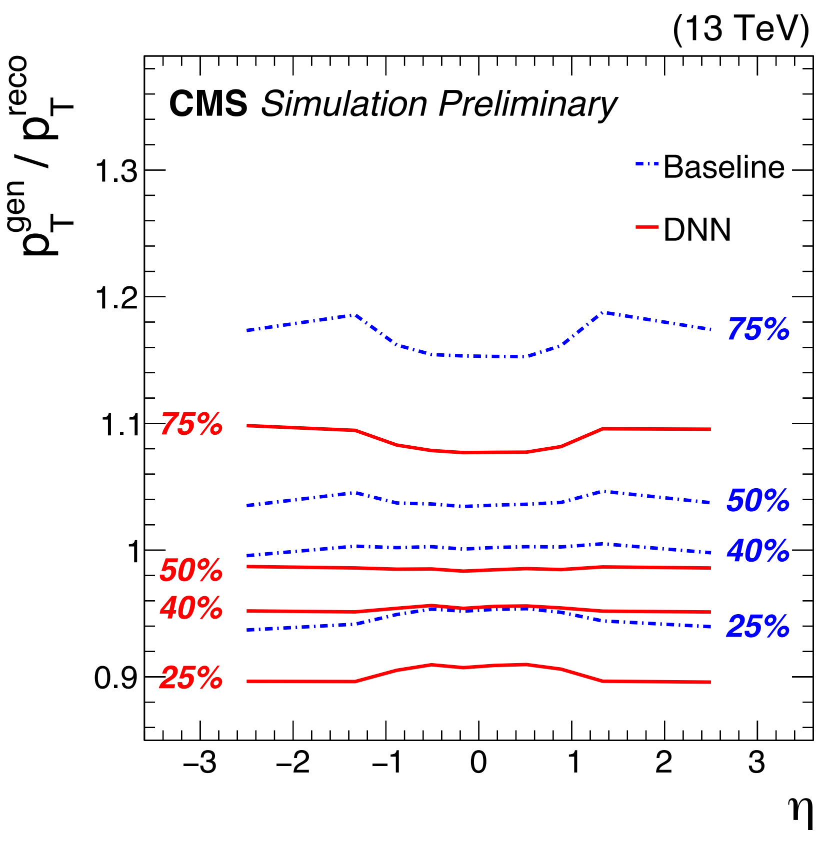
<u>Development on going to:</u>

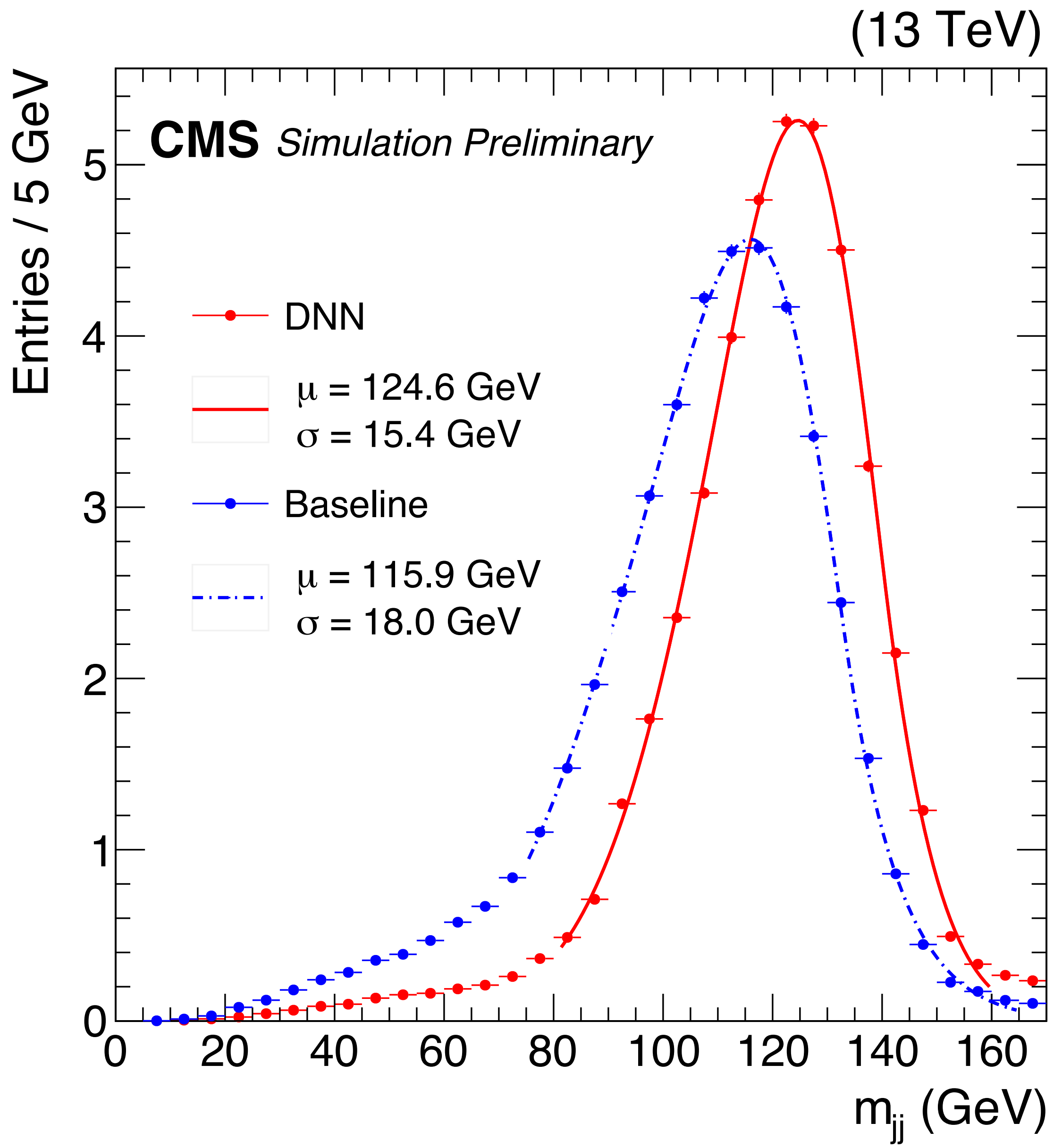
- optimize the network architecture and the input training data format
- define a solid training/testing routine that fits the tight time budget of sync/async reconstruction strategy

See backup slides for more details and references

Estimating jet energy with DNNs at CMS







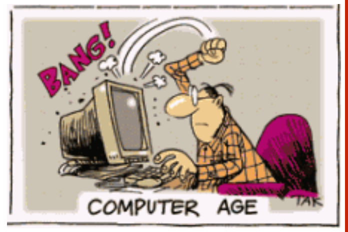
Use a deep neural network to calibrate the reconstructed energy of b-jets.

Both information about the jet constituents and information about the rest of the event (pileup) is used in the training.

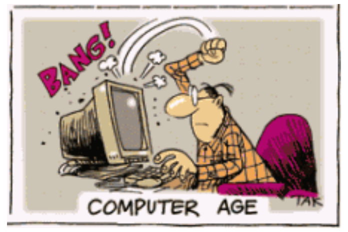
See the public note for more details. Many other interesting use cases, see also Adversarial NNs for data-simulation corrections, and deep learning for per-object systematic uncertainties.

Conclusions

A diverse and heterogeneous future



Run-III: Full interleaving of classical and ML methods in reconstruction and analysis, real-time & offline?



Some personal remarks

1. We know by now that ML methods can outperform classical methods in physics terms in classification. There are increasing hints that the same may hold for reconstruction.
2. We are accumulating operational experience deploying these methods in real-time and using them to not only get more physics but also better control and understanding of our detectors.
3. We are also developing experience deploying ML across a range of computing architectures, which may be important if large-scale computing continues to become more heterogeneous.
4. A key challenge is to systematically understand the computational efficiency of different ML approaches and where best to mix classical or hand-optimized elements to speed things up. Both algorithms and the data structures used to pass information between them are crucial to this.

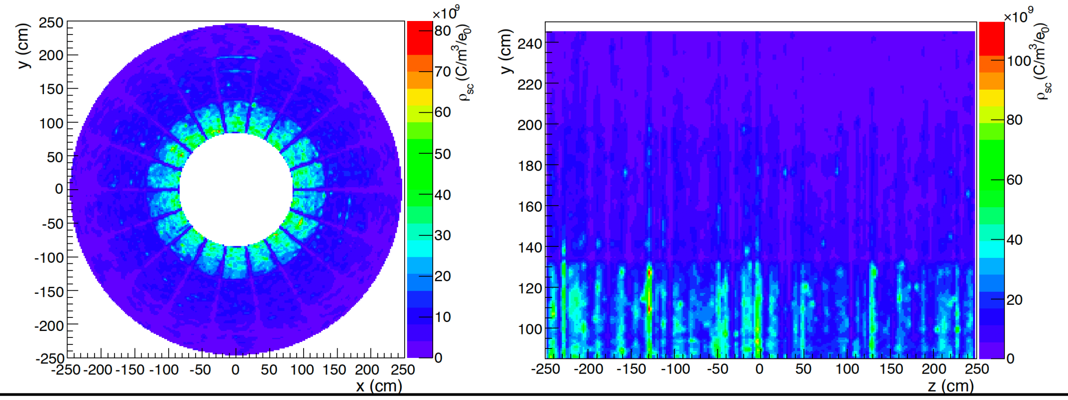
Backup

UNet for correcting for distortion fluctuations

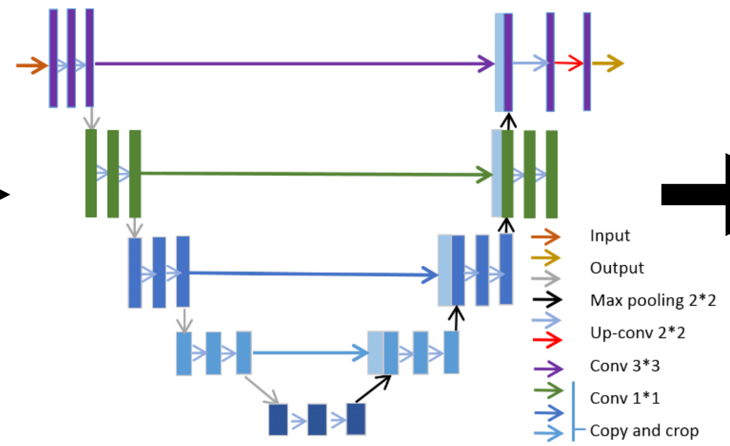
UNets being explored to correct for the **TPC distortion fluctuations** using as an input the TPC space charge densities or currents

inputs:
- space charge Ion densities and fluctuations

average map: $\langle\boldsymbol{\rho}(r,\varphi,z)\rangle$
fluctuation maps: $\boldsymbol{\rho}(r,\varphi,z) - \langle\boldsymbol{\rho}(r,\varphi,z)\rangle$

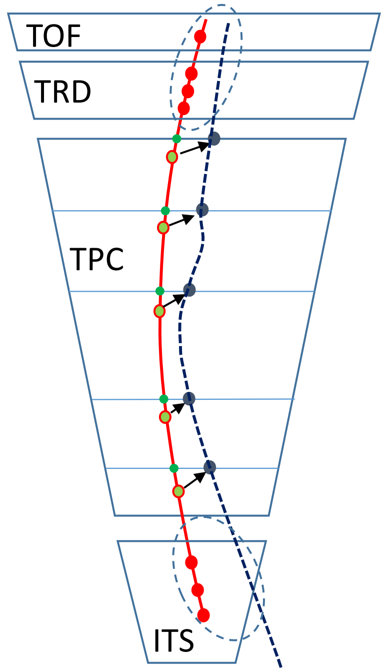


UNets



Predicted quantities:
- Distortion fluctuations along the 3 direction (R,Rφ, z)

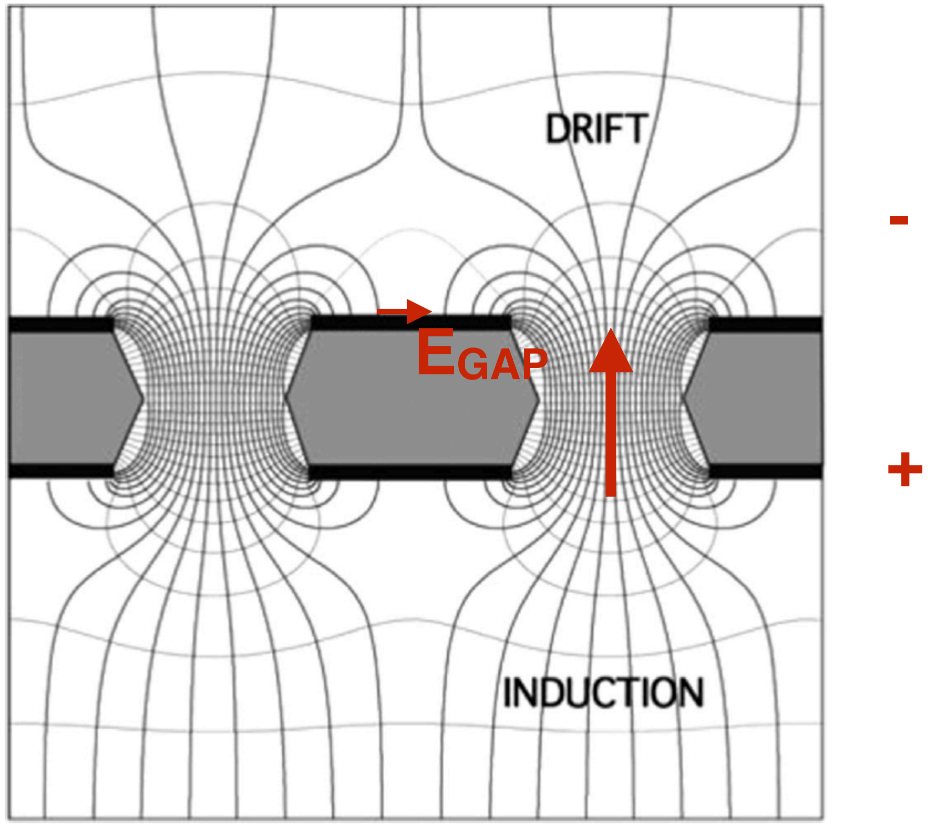
fluctuation maps:
- $\langle\delta_r\rangle - \delta_r$,
- $r\langle\delta_\varphi\rangle - r\delta_\varphi$,
- $\langle\delta_z\rangle - \delta_z$



In this strategy a **new training could be performed periodically to build the "map"** using simulated data or real data and **used to produced corrections based on realistic space charge densities**

TPC readout in Run3: GEM chambers

Goal of Run3 is to collect all the MB PbPb events delivered:
- 50kHz of collisions for total of ~50 billion events over 2/3 years of data taking
- Current TPC can run up to few KHz (limitation imposed by the process of IBF gating)



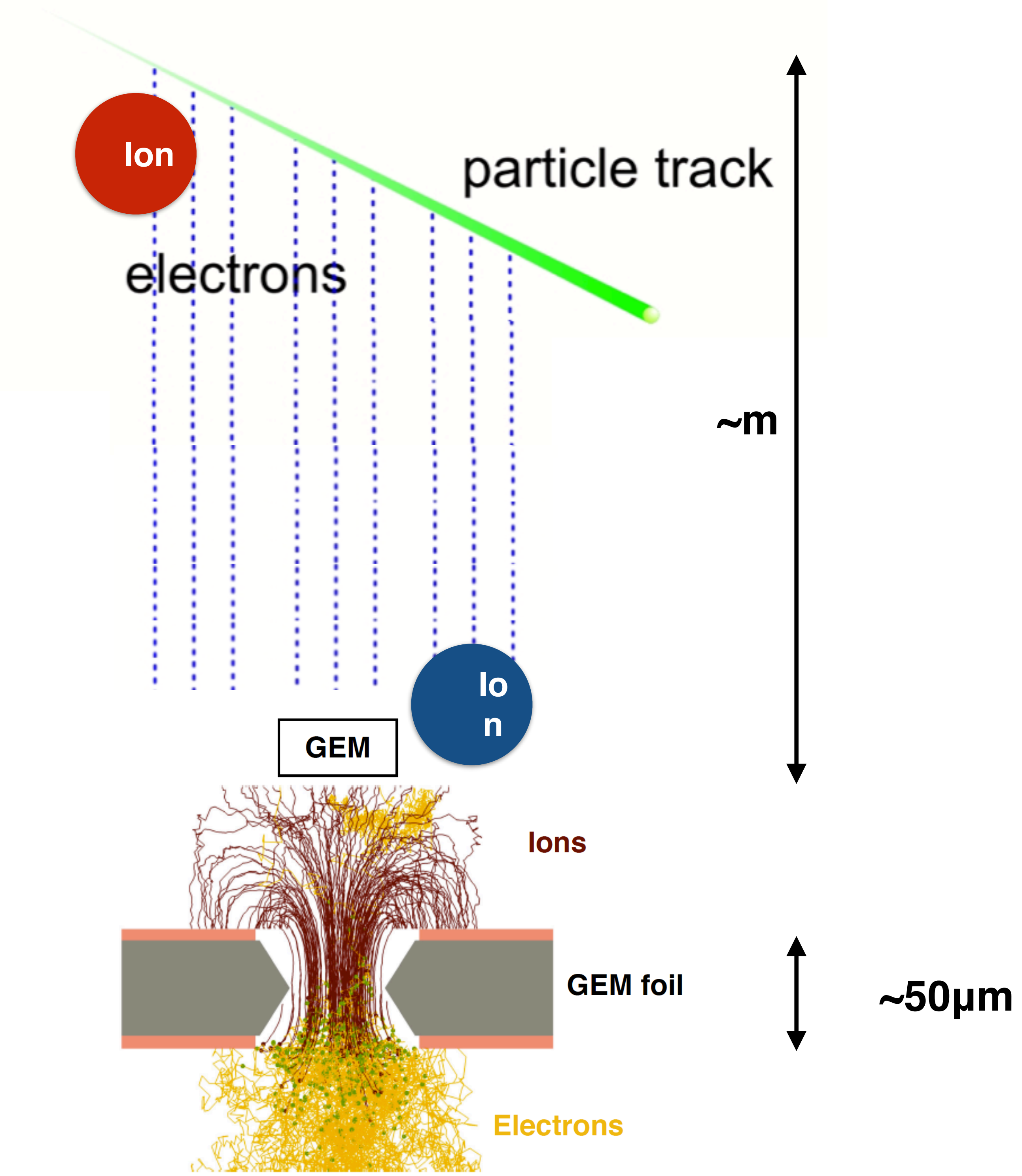
GEM (Gas Electron Multipliers)

Thin polymer foil, metal-coated on both sides and pierced with a high density of holes where strong electric fields to generate multiplication:

→ **faster:** signal collected only using electrons, not ions
→ **more stable:** readout region and multiplication region are separate: propagation of discharges is much less likely
→ **reduced IBF:** ions produced in multiplication are partially collected by the GEM

x, y "segmented" electrode for signal collection

Distortion scenario with GEM in Run3



BUT there is no ion gating! In the drift region we will have:

- **Primary ions**: produced by the passage of the tracks
- **Secondary ions (or ion backflow)**: produce in the GEM multiplication region ~ uniformly distributed in the drift region

Some numbers:

- time between two collisions → **t=1/(50kHz) = 20μs**
- **electron drift time ~ 100μs**
- **ion drift time ~ 200ms**

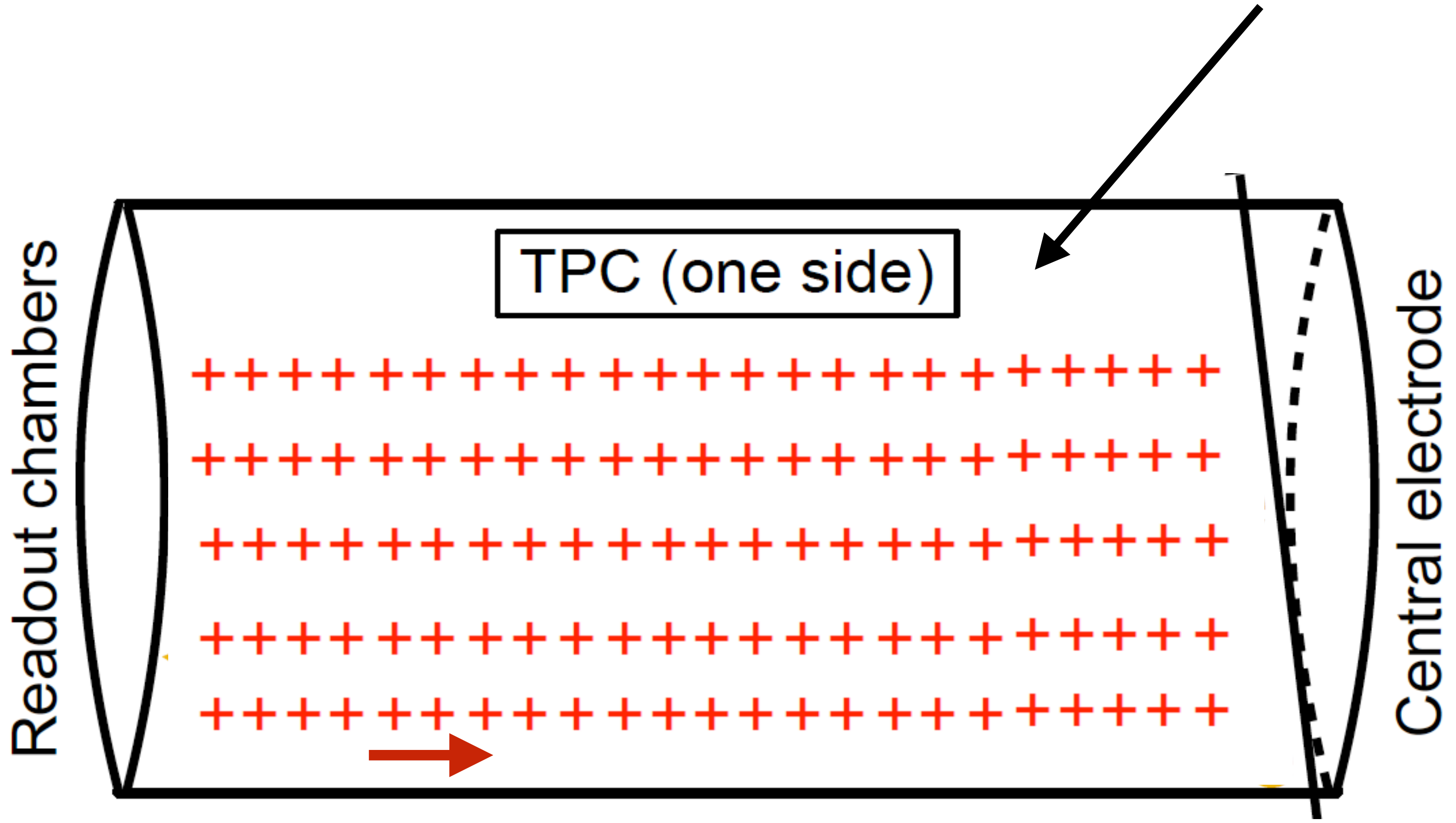
↓

- **Screenshot of the TPC at any given time:**
  - electrons belonging to 5 different PbPb collisions
  - BF Ions belonging to **8000 different PbPb collisions!**
- BF ions will be >> primary produced ions

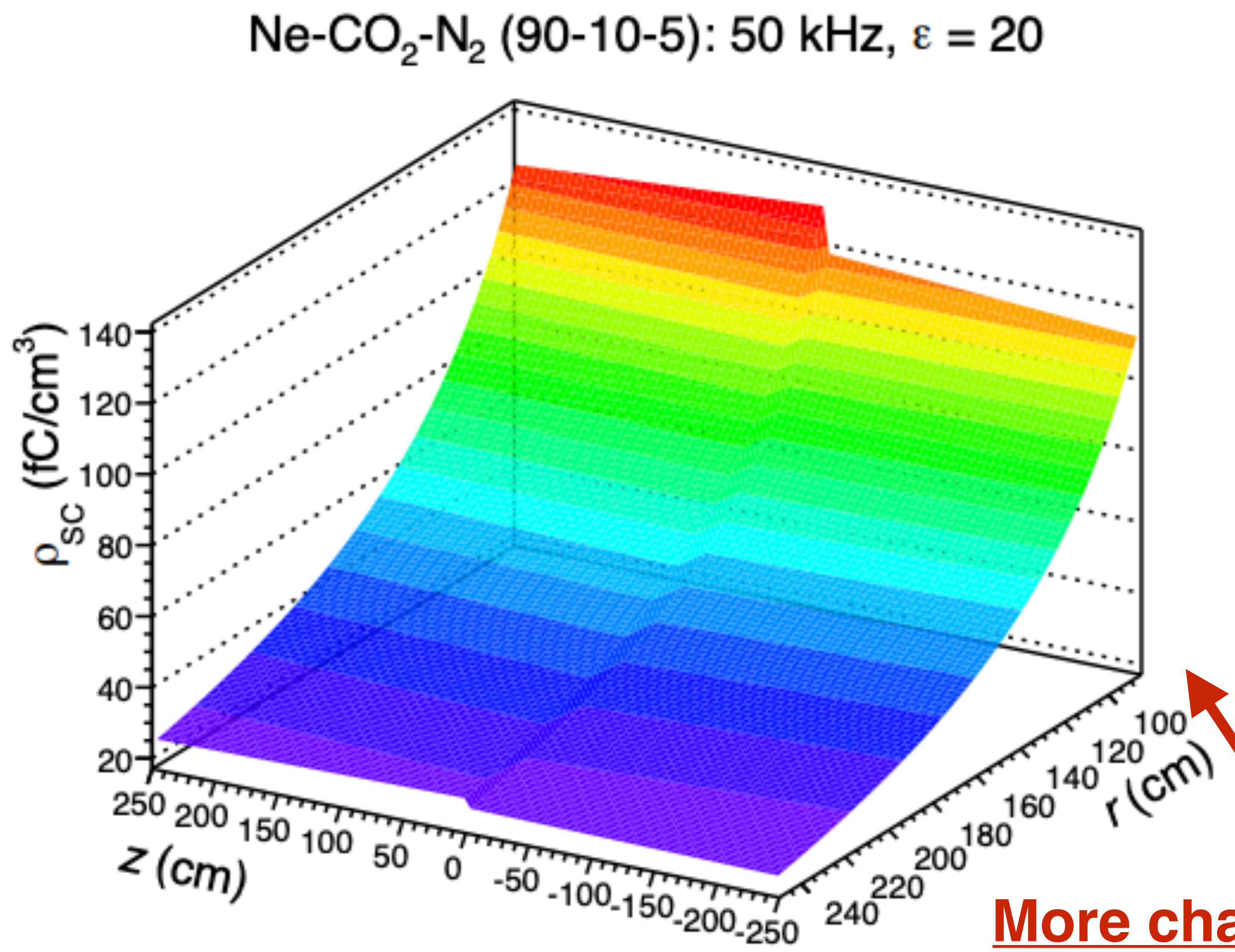
How will the space charge distribution look like in the TPC?

Average space charge (SC) densities <ρ_{SC}>

A ~ uniformly distributed flow of BF flow ions that go from the readout pads to the center of the TPC :



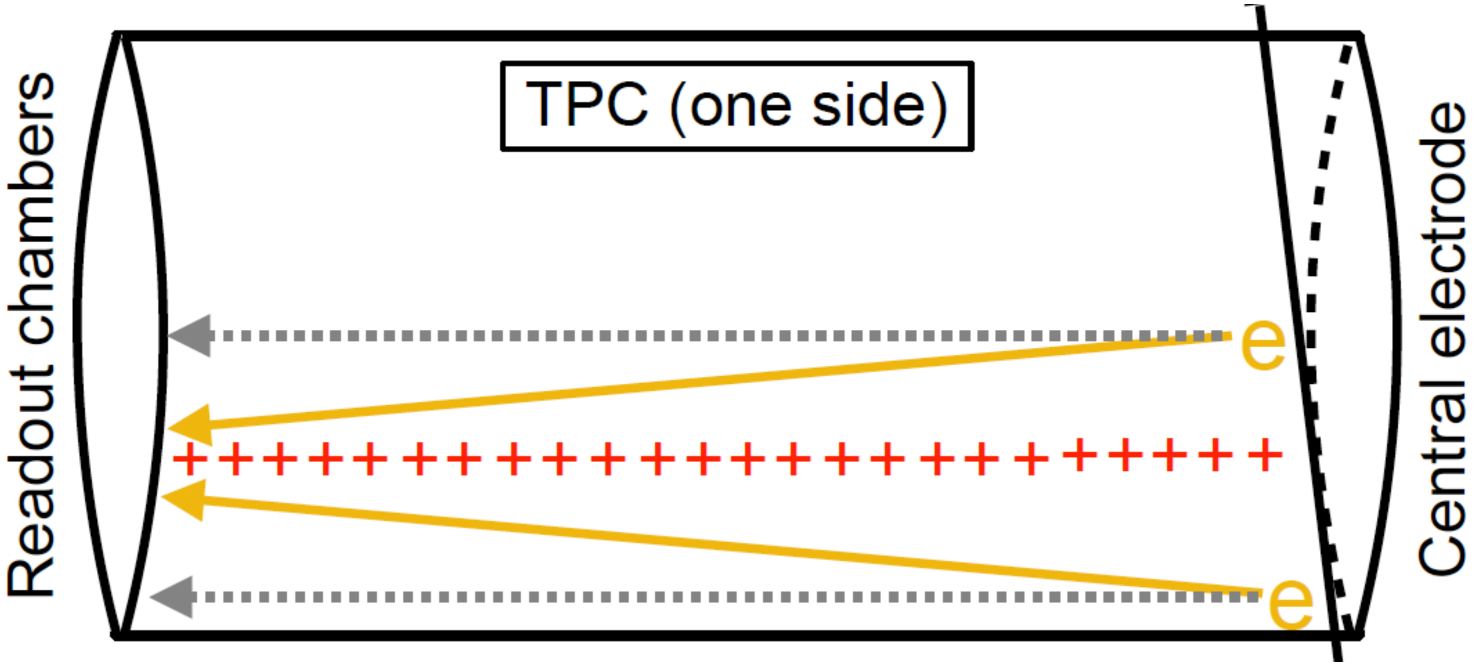
ρ_{sc} = charge /volume



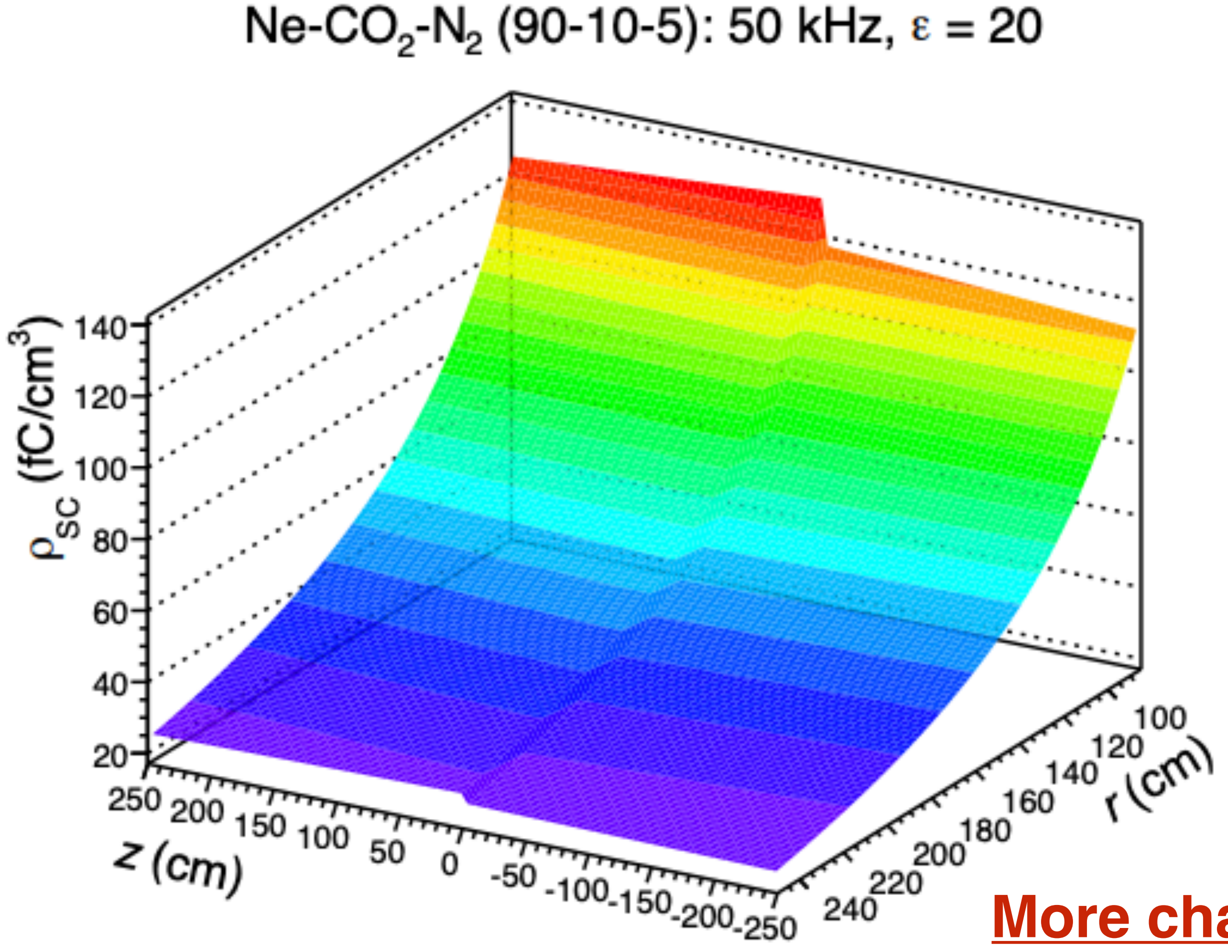
More charge density at small R

Average distortions <δ>

A ~ uniformly distributed flow of BF flow ions that go from the readout pads to the center of the TPC :



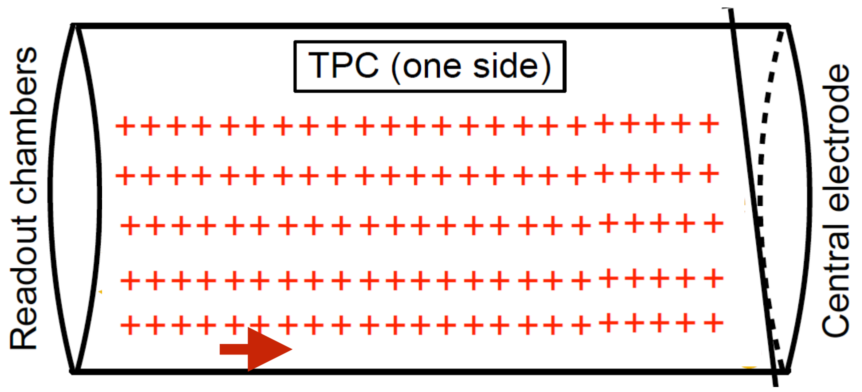
ρsc = charge /volume

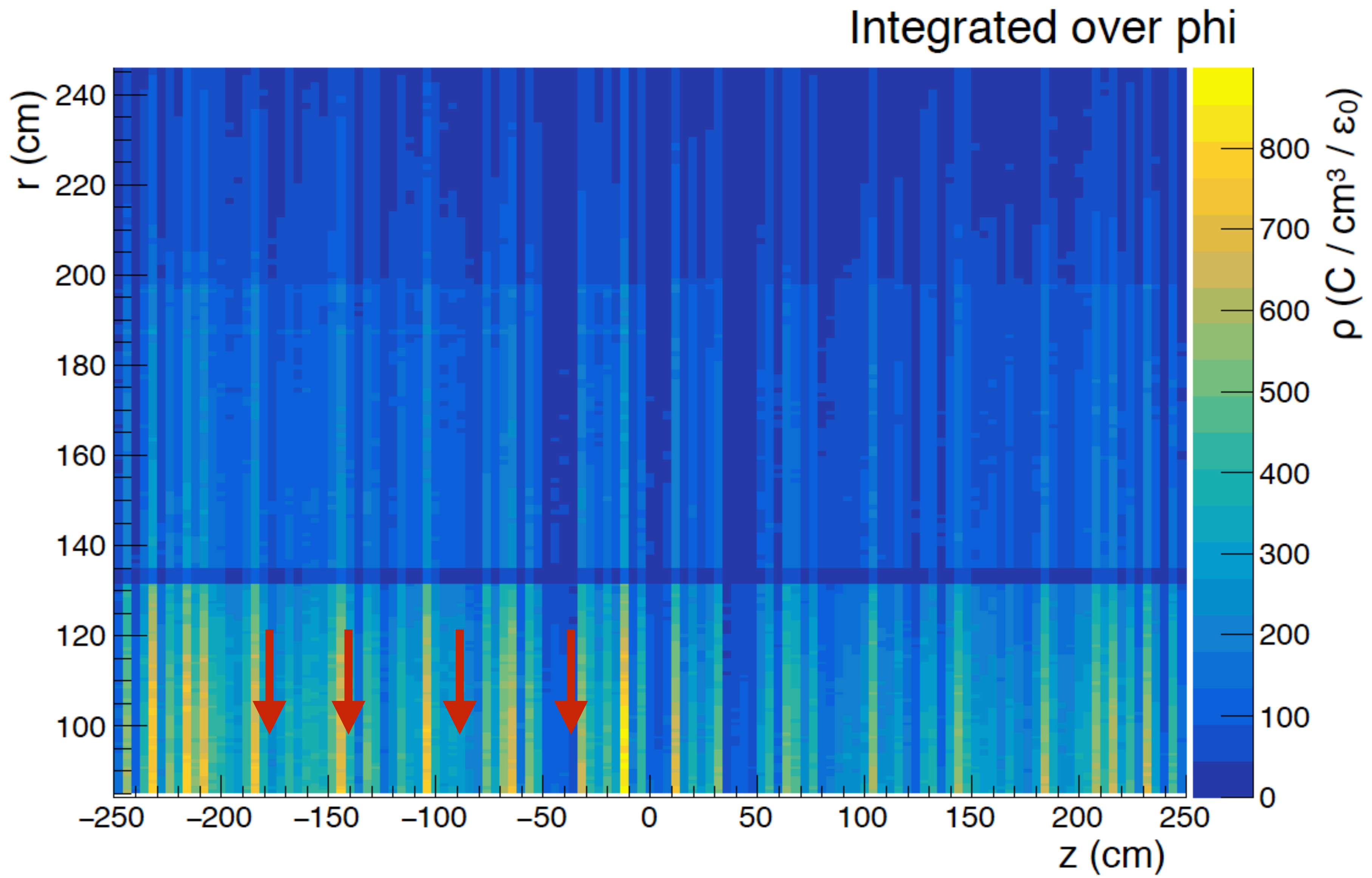


More charge density at small R

Is that all? Are we fine if we correct only the average effect of space charge densities as in Run2?

Space charge fluctuations

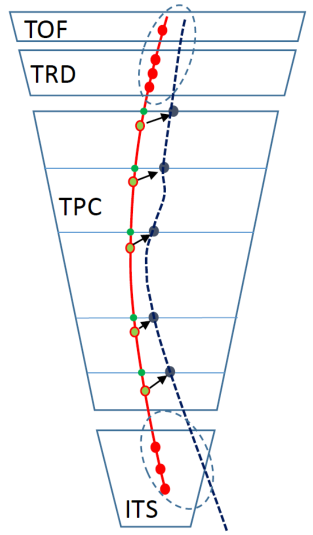




- **Screenshot of the TPC at any given time:**
  - BF Ions belonging to **8000 different PbPb collisions!**

Space charge fluctuations generated by:

- fluctuations in the n. of pile up events in TPC
- fluctuations in the n. of BF ions in each collisions (e.g. multiplicity fluctuations)
- fluctuations of ionization charge/ track
- …



Fluctuations in the distortion of the TPC clusters: substantial effects (~mm)

Run3 requirements and strategy

Synchronous stage:

- **distortions corrected with precision ~mm** for cluster-track association, tracking and track matching

Correction for average distortions:

- pre-calculated correction maps computed based on MC events or data from previous data (with ITS-TRD interpolation techniques)

Asynchronous stage:

- **distortions corrected with precision ~ 200µm** (TPC resolution)

Correction for average distortion and fluctuations:

- correction maps for average distortions computed with real data (within ~min)
- **corrections for distortion fluctuations (?)**

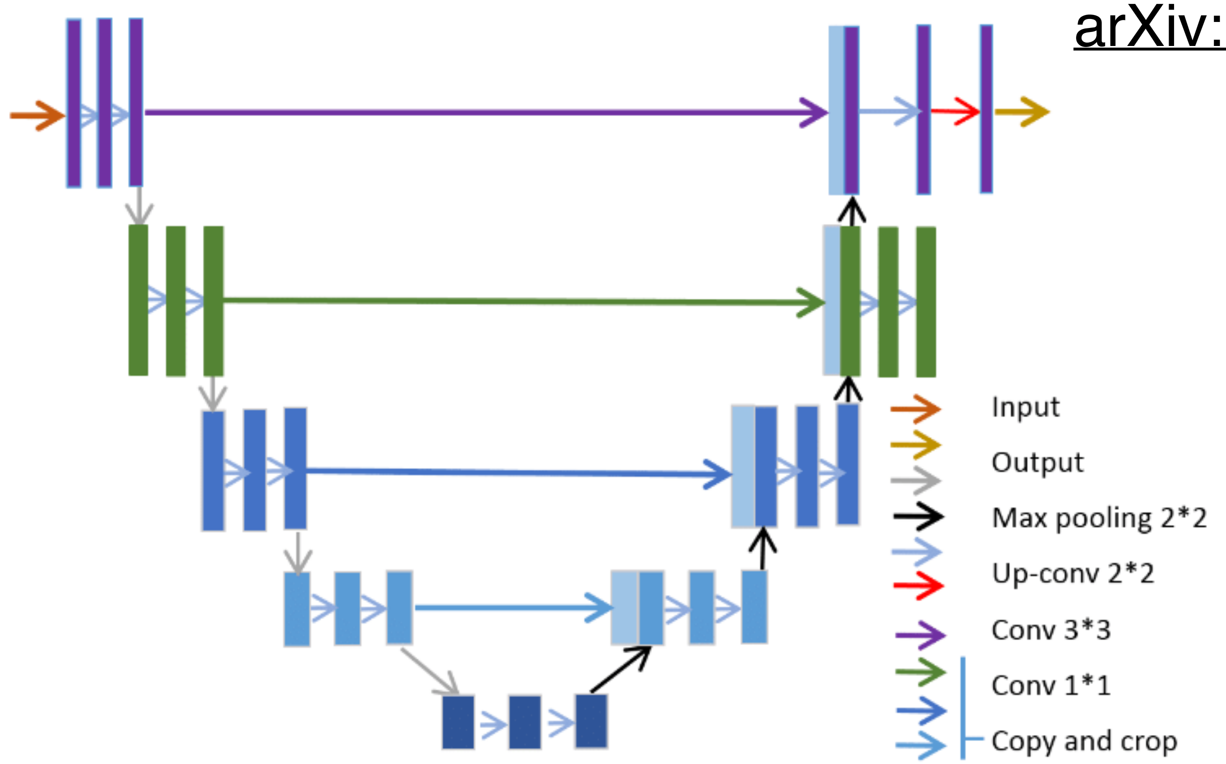
Correcting for distortion fluctuations is currently the biggest challenge!

- to efficiently correct fluctuations we need new distortion correction **~5ms** to account for changes in the SC densities
- analytical corrections take too much time

→ **3D convolutional neural network can help to map the effect of SC densities into distortion corrections**

Model and training setup

UNet: convolutional neural network developed for biomedical image segmentation. Includes:
- Series of 3D CNNs + Pooling/Upsampling layers



arXiv:1505.04597

For network structure and configuration (link)

filters: 4
pooling: 0
batch_size: 27
shuffle: false
depth: 4
batch_normalization: 0
dropout: 0.0
ephocs: 20
lossfun: mse
metrics: mse
adamlr: 0.001000

Some timing information:

→training on ~800 events with 90x17x17 grid takes ~ 2min with 1 NVIDIA V100
→training on ~800 events with 180x33x33 grid takes ~ 15 min with 1 NVIDIA V100