# Progress on ML-based Particle Flow with CLD


Gregor Krzmanc, ETHZ

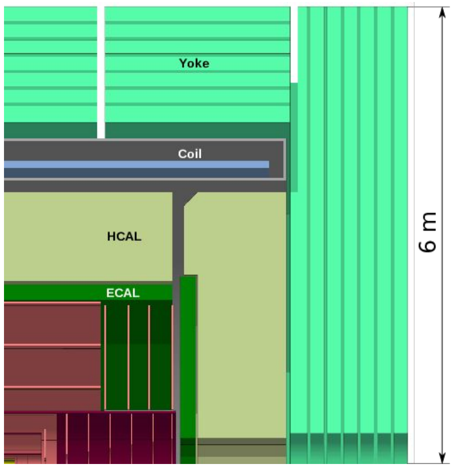
Dolores Garcia, Michele Selvaggi

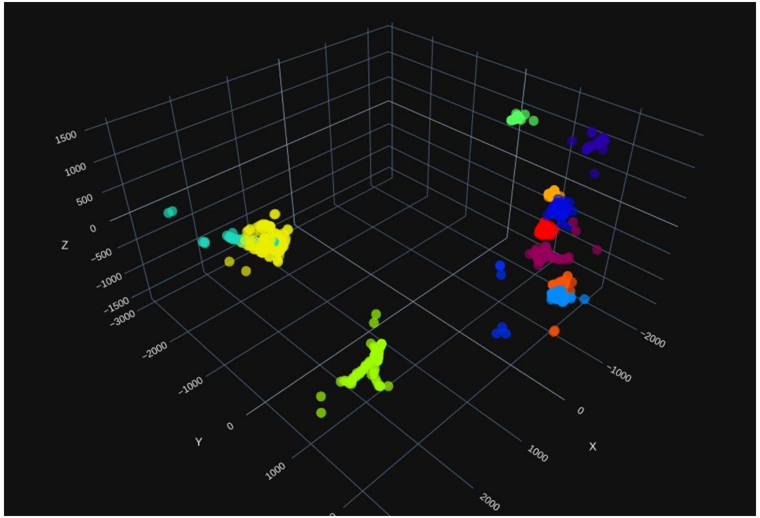

10 July 2024

# Introduction - **Particle-Flow reconstruction**

**Task: Reconstruct particles and their properties** from the hits in the calorimeters and tracking information

CLD detector

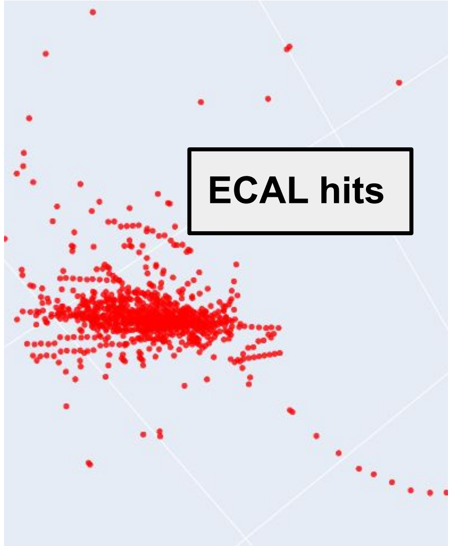


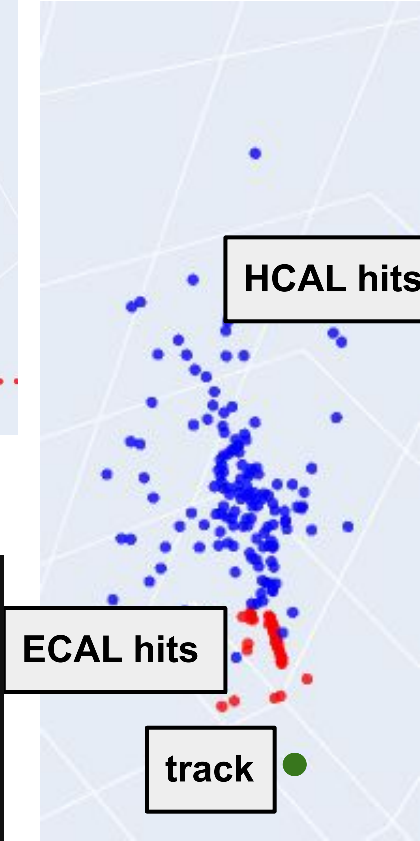



# Problem

**Input:** Point cloud:

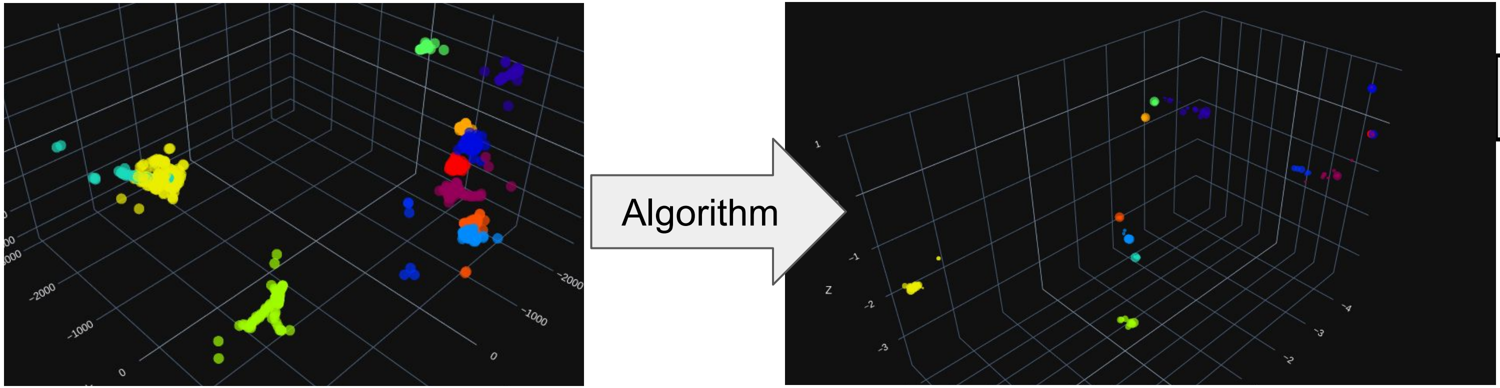
- Hits (**ECAL**/**HCAL**): x, y, z, hit energy
- **Tracks:** x, y, z, p, $\chi^2$ of the track fit
- 1 hit/track = 1 node



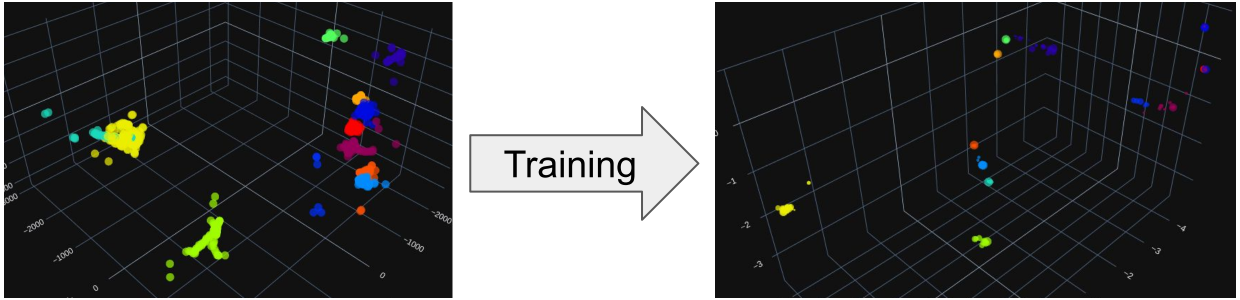




**Output:**

Set of reconstructed particles



# Method: Object Condensation



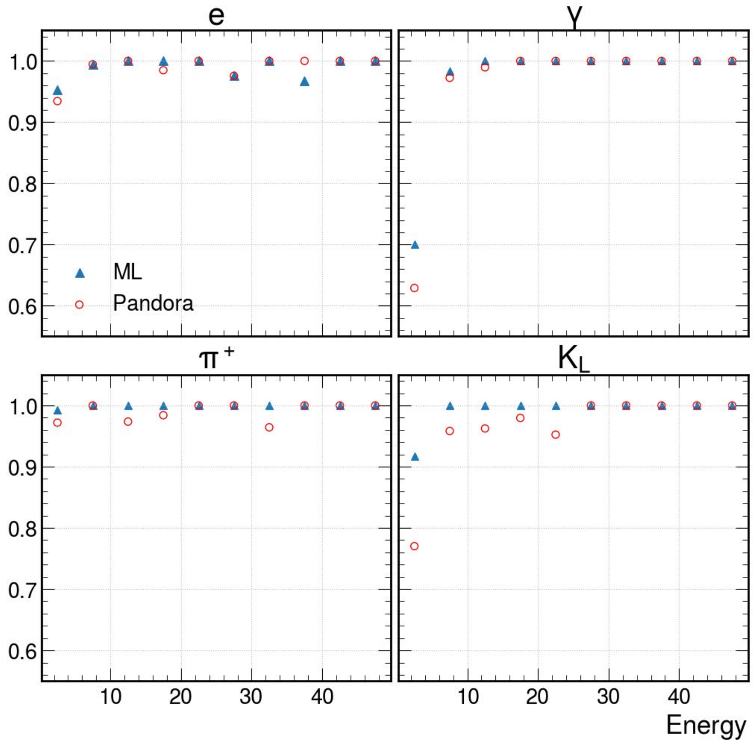

- Works well for clustering!
- GATr - Geometric Algebra Transformer works reasonably well, outperforming PandoraPFA (clustering baseline)

**Object Condensation** (Jan Kieseler, 2020, https://arxiv.org/pdf/2002.03605)
https://openreview.net/pdf?id=PYcp183GBL

# Clustering evaluation

- Particle gun dataset: $K_L$, n, $e^{+/-}$, $\pi^{+/-}$ ,
- 10-15 collimated particles, $\Delta r$=0.5

Efficiency

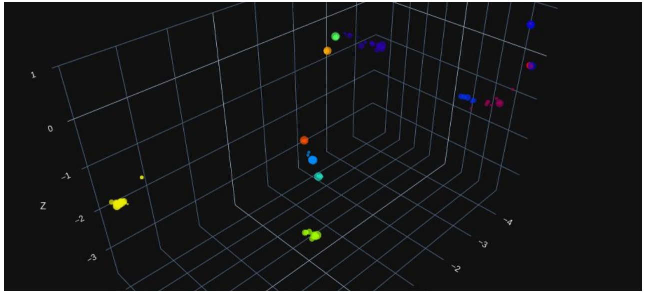


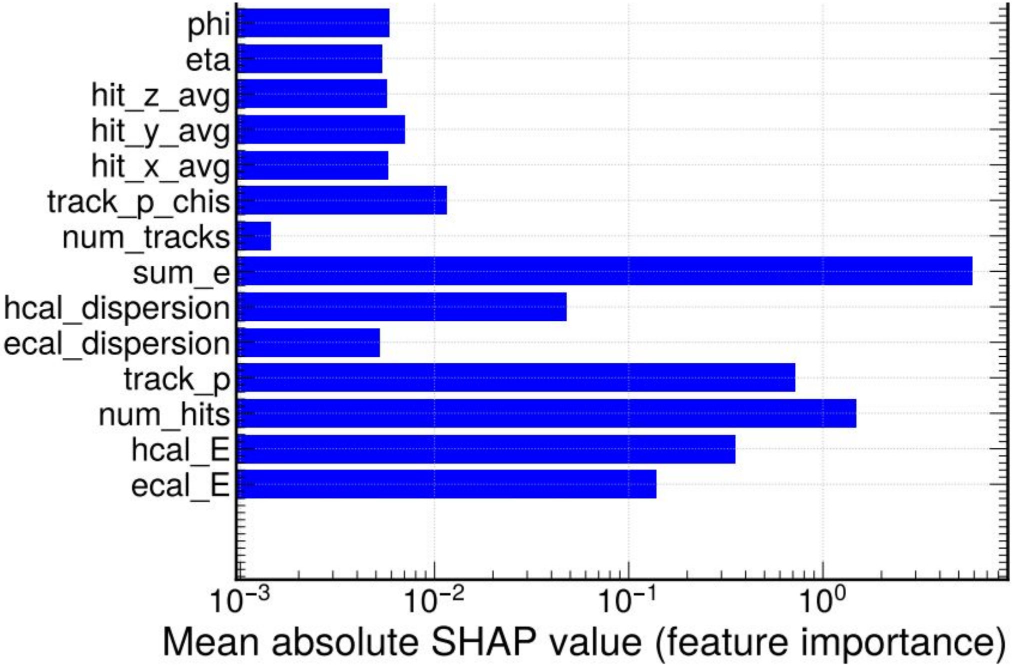
# Task: Energy regression

- HGCal approach (Qasim et al., https://cds.cern.ch/record/2775923/files/document.pdf) - regress *energy correction factor* for each hit

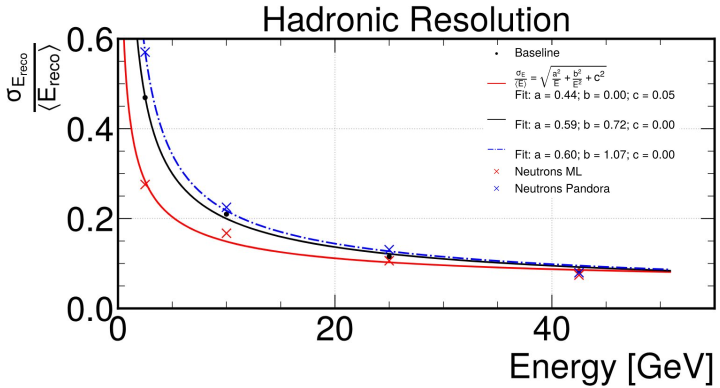
# Simple neural network model

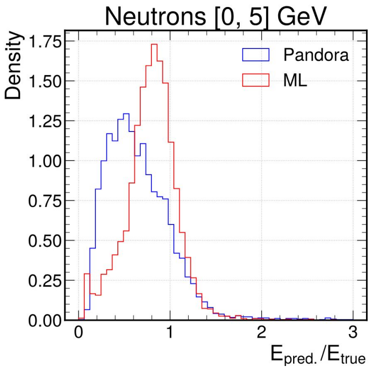
- Model: Simple deep neural network: 2 hidden layers of 64 dim. with ReLU
- Input: Clusters made with the model → Compute **graph-level features** and train on that
- Separate model for charged and neutral particles

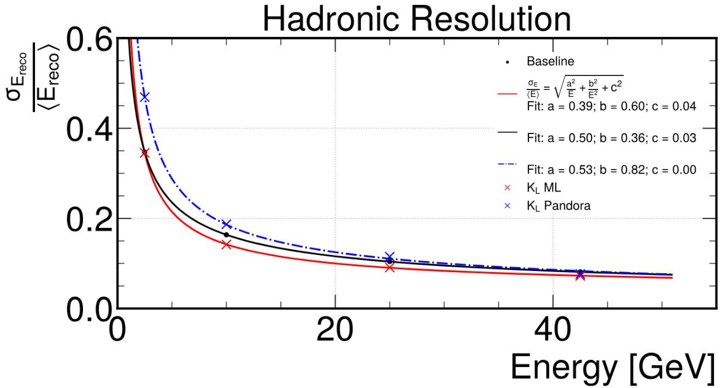


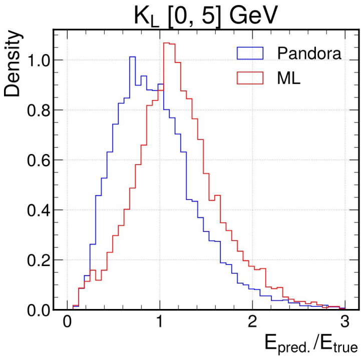



# Results - neutrons, $K_L$

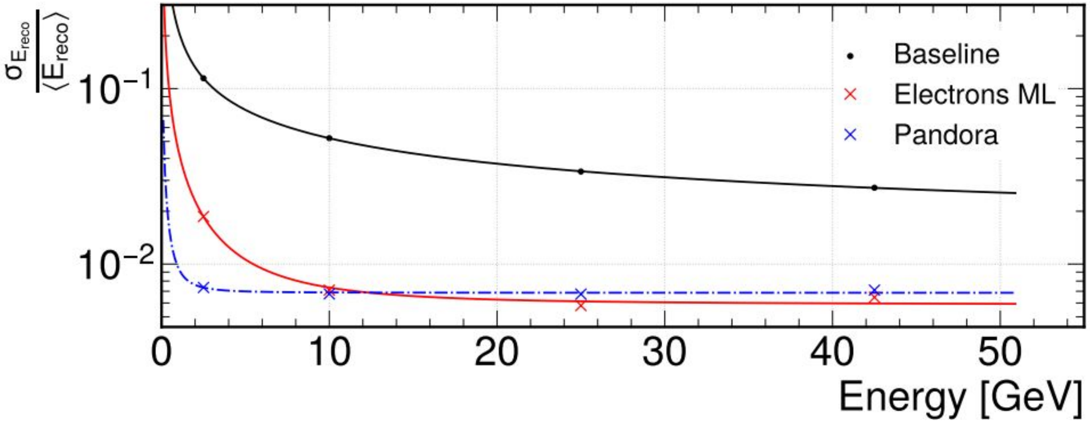


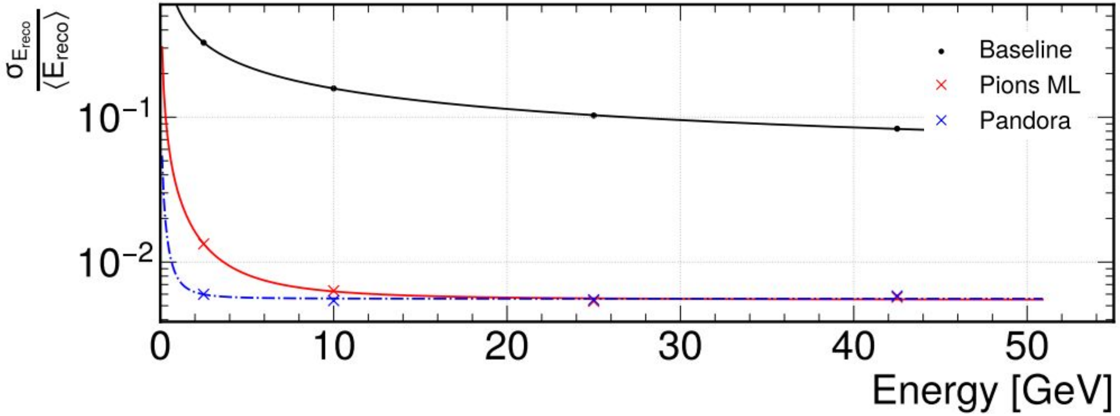



# Charged particles

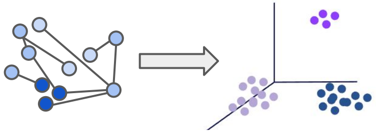


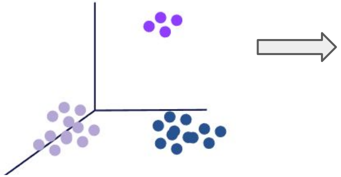

- Needs improvement at lower energy

# Model

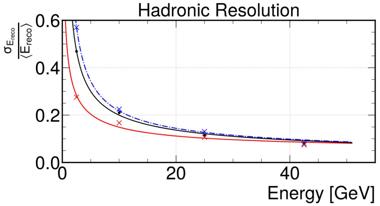
Clustering GNN



- Energy correction
- Momentum: unit vector regression for neutral, pick track at DCA for charged
- PID: separate model for particles with track and particles without track



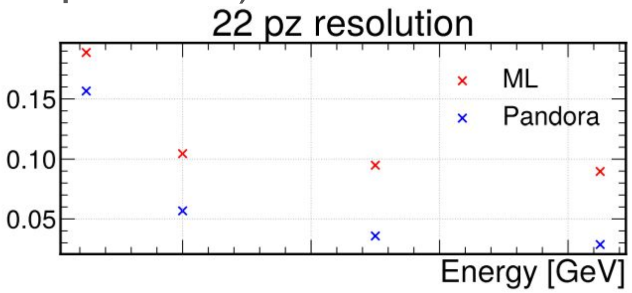
Energy

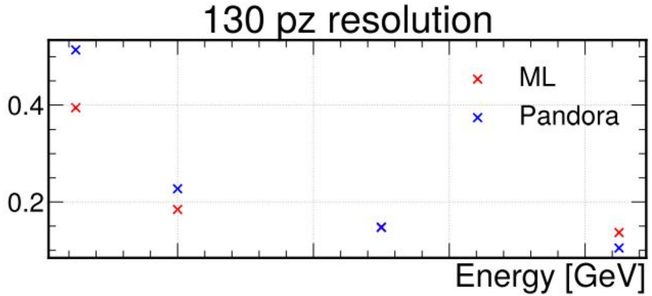


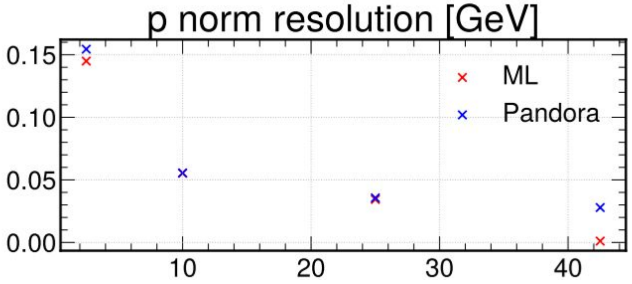

PID, Unit $p_x$, $p_y$, $p_z$

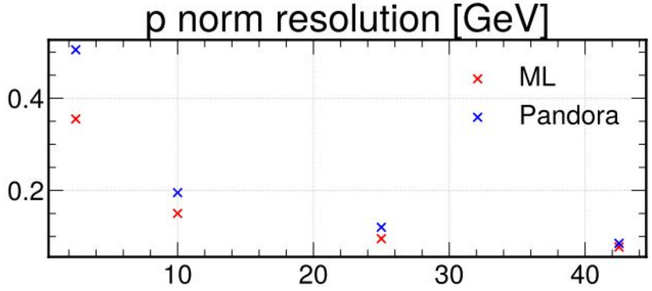
# Momentum regression

- Similar to energy, better momentum resolution for neutrals. Worse for photons (no gain from energy regression, worse direction reconstruction than pandora).



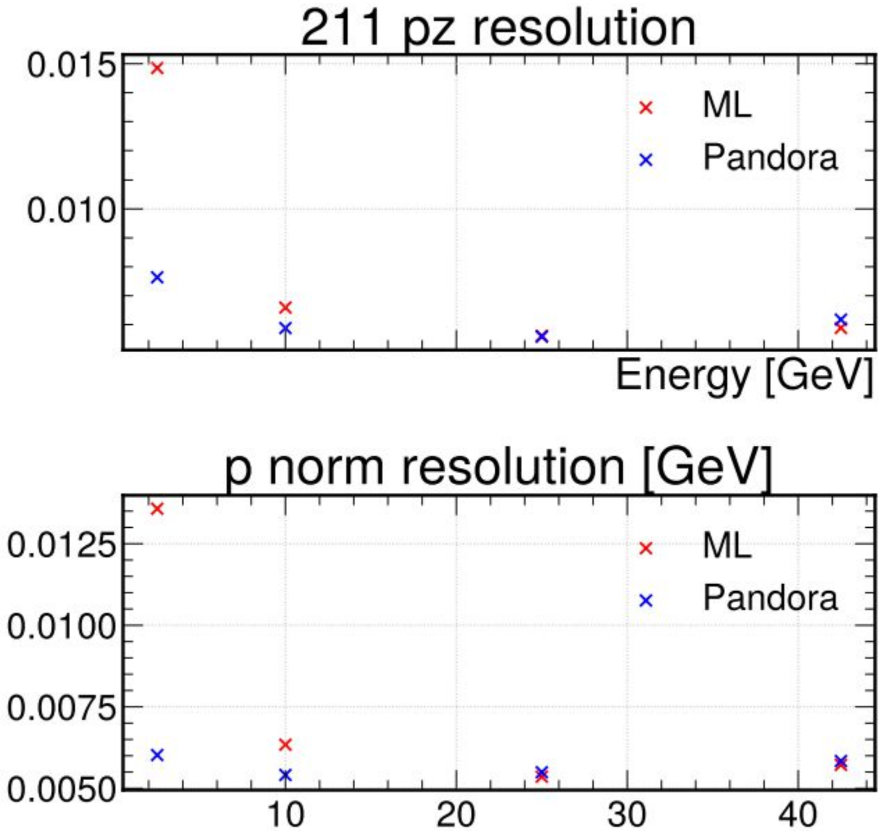



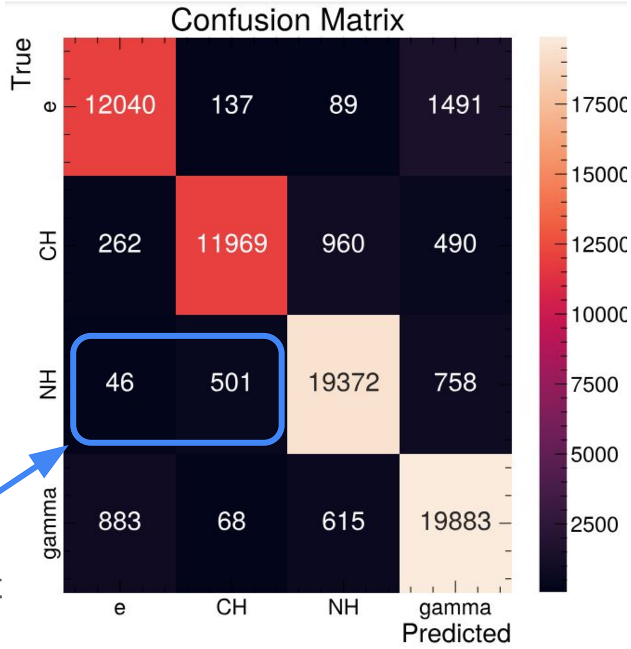
# Momentum regression - $\pi^{+/-}$

- Needs improvement, especially at lower energies



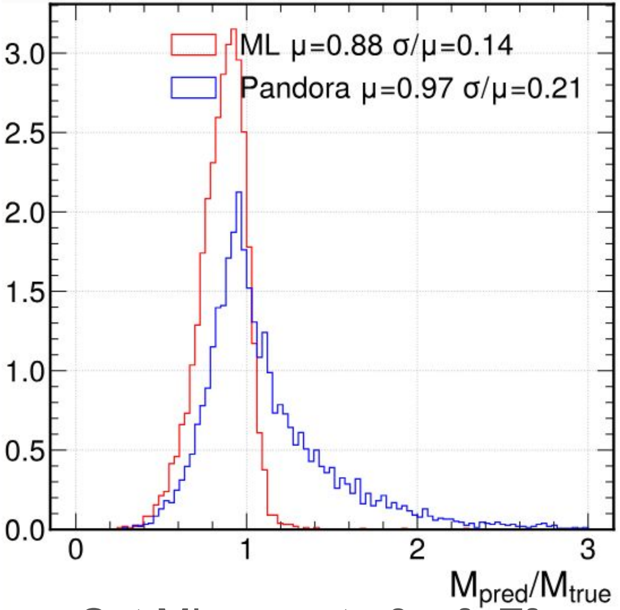
# PID

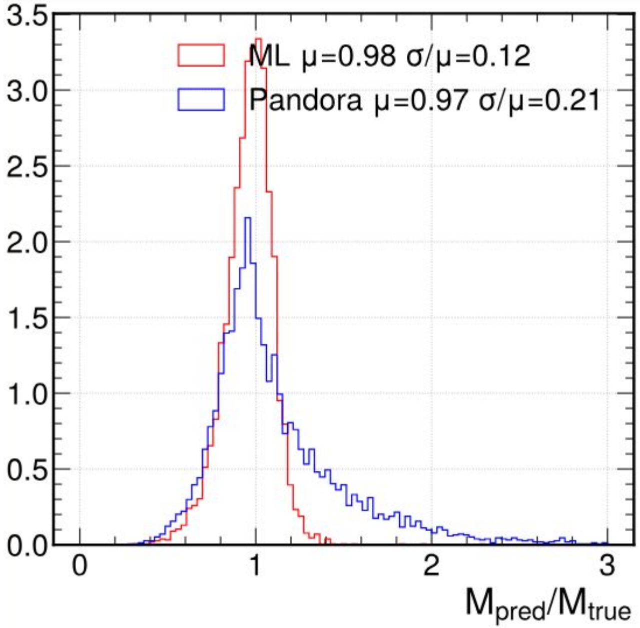
- Separate model for clusters with a track and clusters without a track - same classes for both, to account e.g. for missing tracks
- Charged hadrons: assign pi+ mass
- Neutral hadrons: assign neutron mass

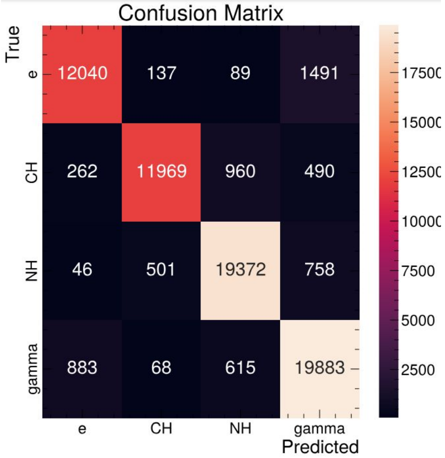



Model confused as some charged don't have a track

# Overall, we improve mass resolution (10-15 particles collimated dataset)



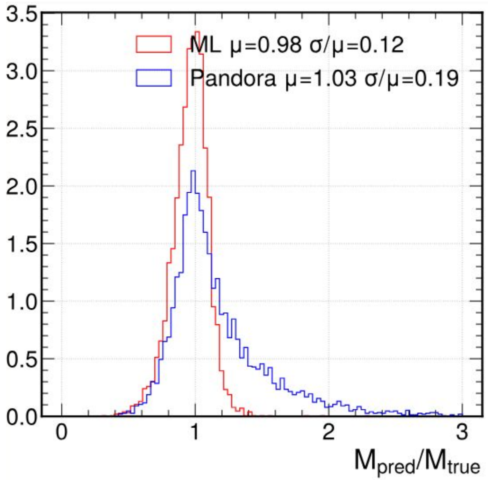




Set ML mass to 0, $p^2=E^2$

ML PID mass

- Pandora PID is around 0 for neutrals, peak around the correct mass for charged
- Our model: assign pi+ mass to CH, neutron mass to NH




- Assume "perfect" Pandora PID (take MC truth) to ensure it is a fair comparison →

# Key findings

- Improved clustering performance and energy resolution in neutral hadrons at low energies
- Improved efficiency of clustering
- Improved energy resolution mostly for neutral hadrons at low energies and mass resolution for a toy dataset

# To-dos

- Charged hadrons, photons need more work - momentum direction is not good for neutrals
- Physics events

# Outlook

- Approach adaptive to detector geometry
- Next steps: full momentum regression, physics events

# Backup slides

The baseline model is a 4-layer **GravNet** [2] a message passing GNN which dynamically updates the KNN graph after each layer.

The model predicts:

- Hit coordinates in abstract clustering space.
- A measure per hit of how likely the given hit is a condensation point β.

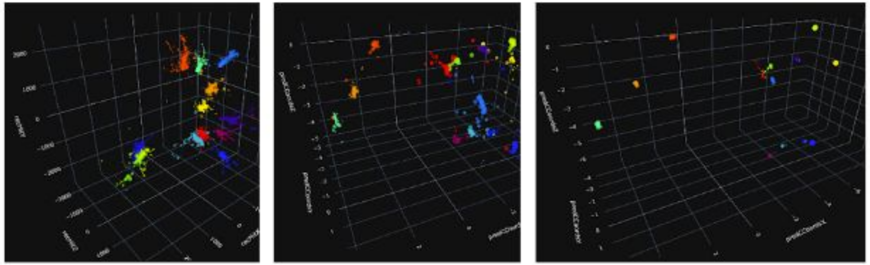
The loss then pulls together hits belonging to the same particle (1) and pulls apart the hits belonging to different particles (2). Additionally, the beta loss (3) is trained to pull the beta values of the points other than the condensation point to zero [3].

$$\mathcal{L}_{V,\text{repulsive}} = \frac{1}{N_{part.}} \sum_{\alpha_j \in \alpha} \frac{1}{N_{hits,j}} \sum_{i \notin \alpha_j} \left( q_i q_{\alpha_j} \exp(-4\,|x_i - x_{\alpha_j}|) \right) \qquad (1)$$

$$\mathcal{L}_{V,\text{attractive}} = \frac{1}{N_{part.}} \sum_{\alpha_j \in \alpha} \frac{1}{N_{hits,j}} \sum_{i \in \alpha_j} \left( q_i q_{\alpha_j} |x_i - x_{\alpha_j}|^2 \right) \qquad (2)$$

$$\mathcal{L}_{\beta} = \frac{1}{N_{part.}} \left( \sum_{\alpha_j \in \alpha} \frac{1}{N_{hits,j}} (\beta_{\alpha_j} + \sum_{i \in \alpha_j} (1 - \beta_i)) \right) \qquad (3)$$

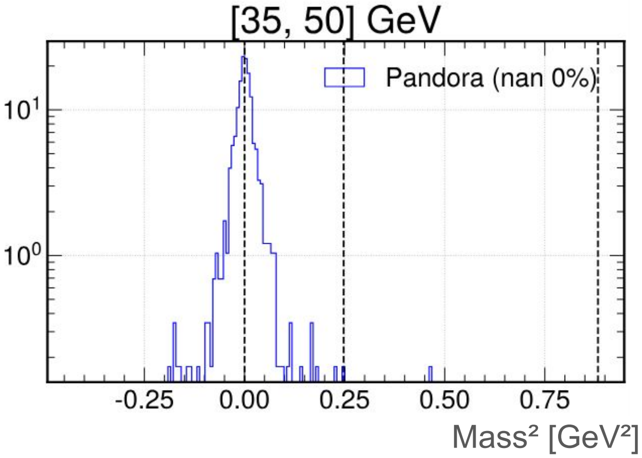
Baseline (Pandora [4]): Multi-step algorithm which makes use of domain knowledge to form clusters. It starts from the inner layers of the detector and aggregates groups of hits following a set of defined rules.



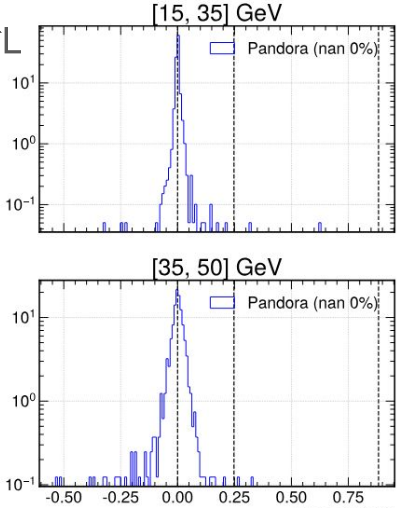
Detector coordinates (left), epoch 3 (center), epoch 12 (right)

# Pandora $E^2$-$p^2$

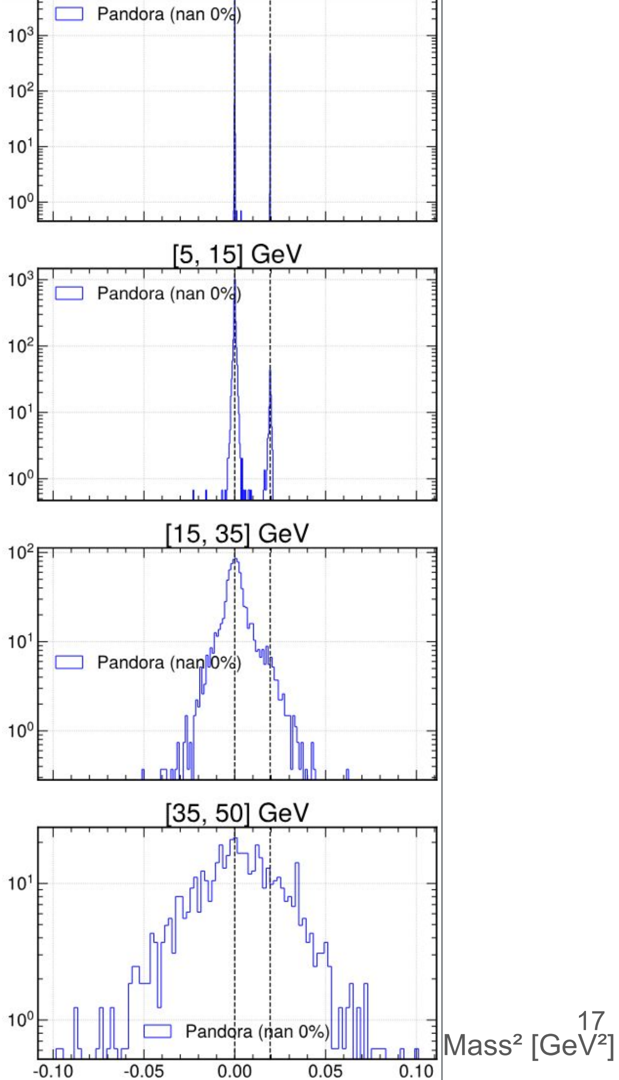
Neutrons




$K_L$




Electrons



# Evaluation of clustering - energy resolution of the sum of the hits (GATr outperforms PandoraPFA)