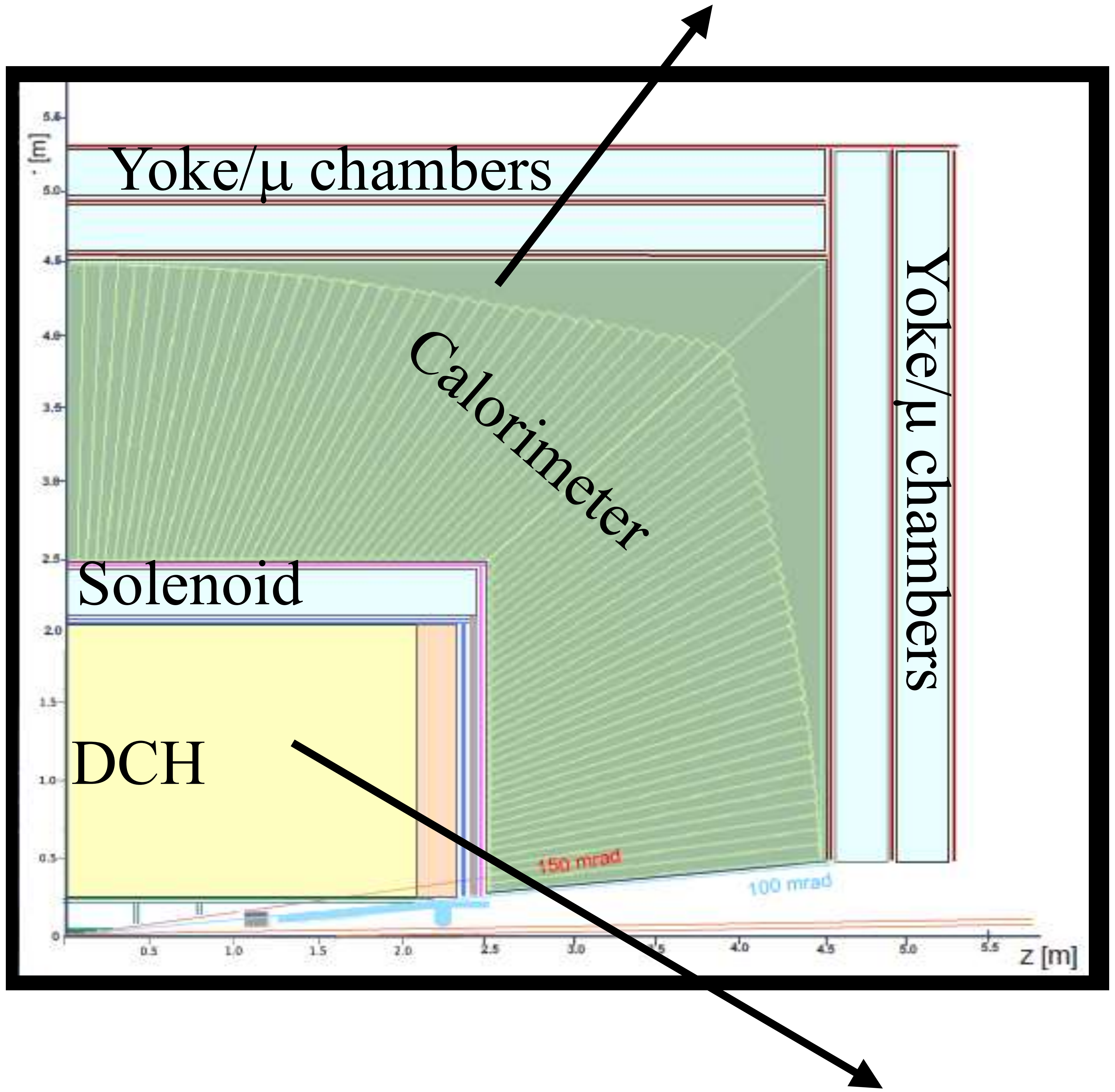
Particle Flow reconstruction with the DR calorimeter in Pandora Framework

work in collaboration with:
M. Biglietti, A. Farilla, R, Di Nardo, A. D'Onofrio (INFN Roma Tre), S. Vallecorsa (CERN), I. Vivarelli (Sussex)

B. Di Micco 11-07-2023

The IDEA detector for FCC-ee

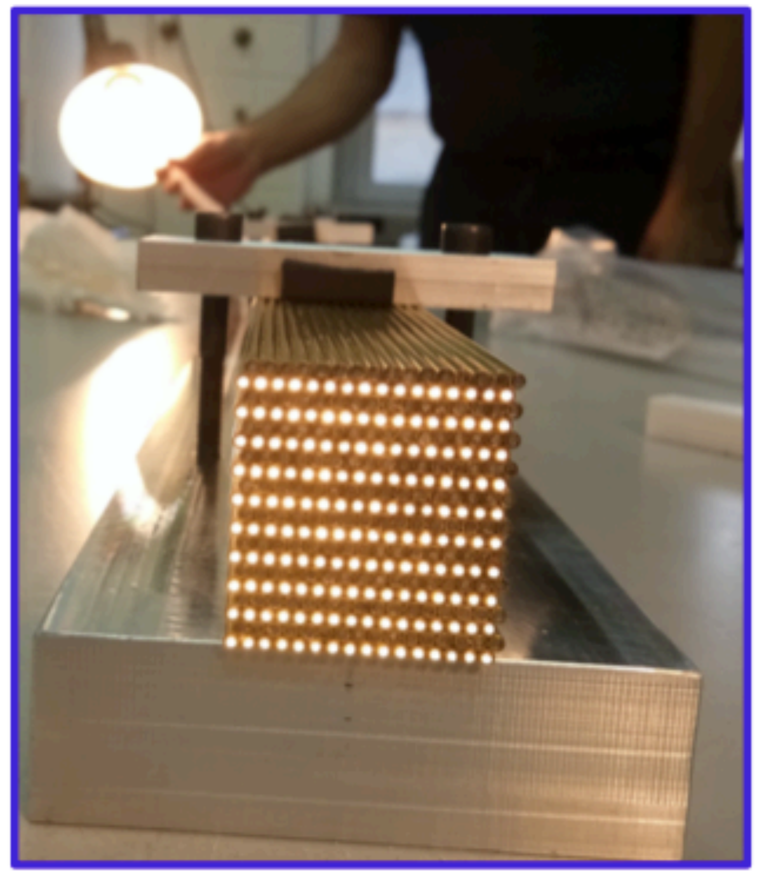
spaghetti dual read-out calorimeter



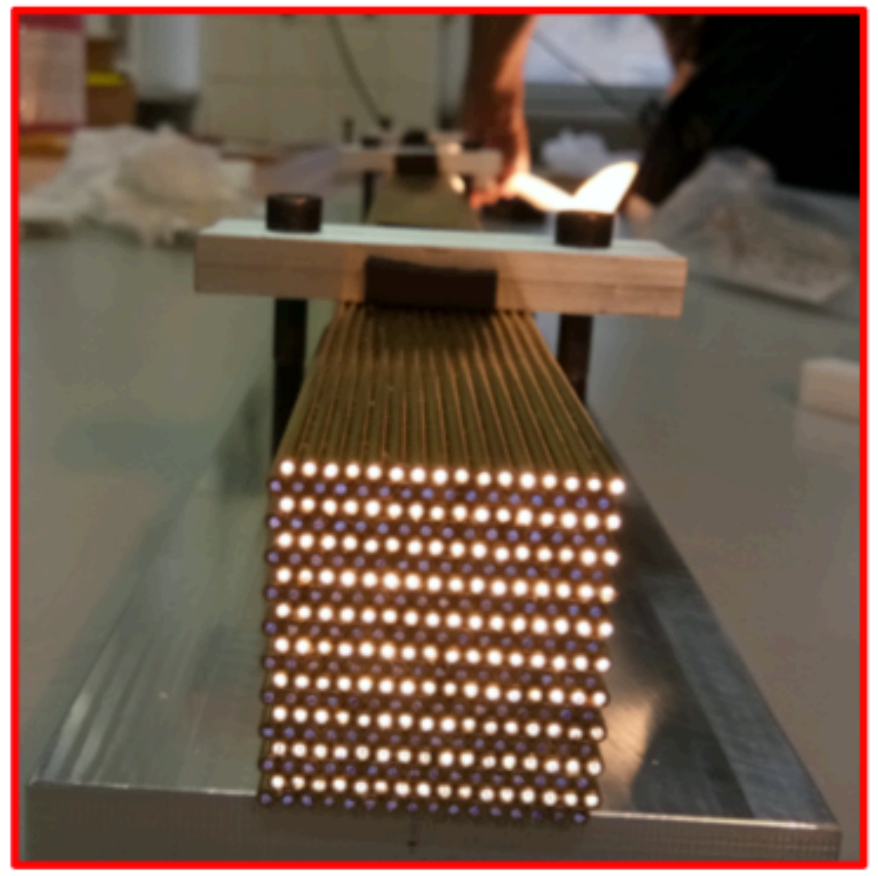
cilindrical drift chamber

- Dual readout calorimeters aim at improving the energy resolution of hadronic calorimeters
  - Generally driven by the fluctuations between the electromagnetic and the hadronic component of showers
- Measure the hadronic component and the electromagnetic component (dual readout) of the showers separately, to derive proper correction factors to be applied to each component to reconstruct the energy of the impinging hadrons
- Exploit a passive/material - fibre layout where two type of fibres, one sensitive to the usual scintillation process, a second type of fibre producing Cherenkov light when ultra-relativistic particles cross with a speed higher than the speed of light in that fibre (S or C fibres)

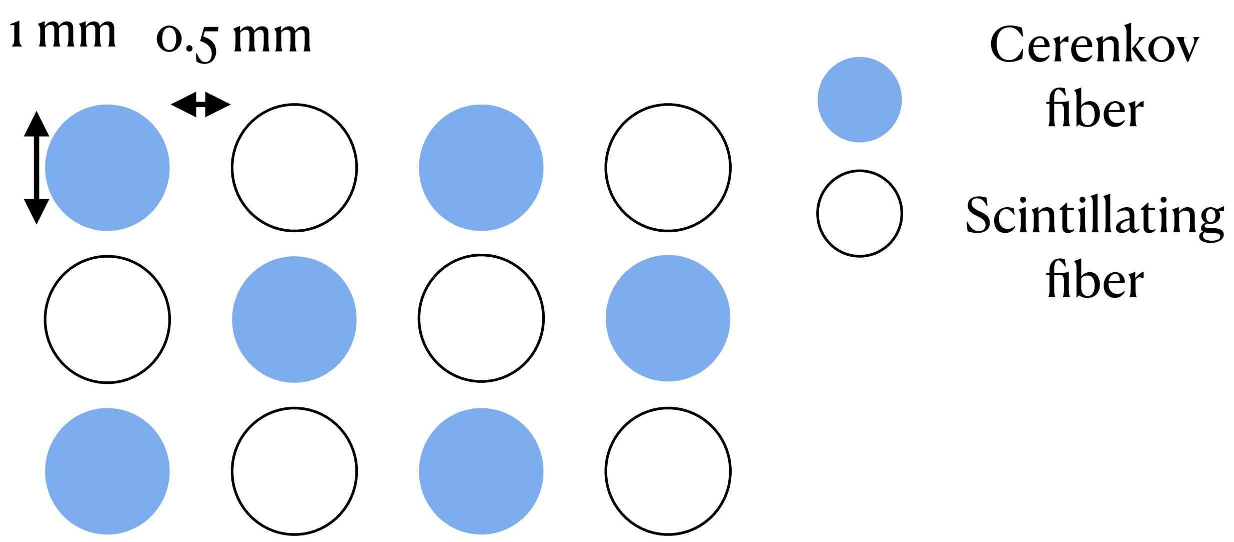
Test beam module



Scintillation fibers



Cherenkov fibers



The particle flow project (ambitious)

★ The basic idea is to exploit the high granularity of the DR calorimeter (up to 50 fibers per cm²), to build up a NN based particle reconstruction and identification algorithm, use truth level information for charge particle tracks from DC and check a NN based jet reconstruction algorithm.

The aim of the project is to build a Neural Network based algorithm that, from a given collection of energy deposits in the calorimeter, is able to completely reconstruct a jet in the detector and maximise the energy resolution of the dual read-out calorimeter

Workflow

1. Geant4 simulation ✓
2. Extract particle info at the input layer: position, momentum, particle type
   Extract calorimeter info: fibre position, fibre type, collected light by the fibre ✓
3. Clustering
4. NN based particle identification: charged: $e^{\pm}$, $\mu^{\pm}$, $\pi^{\pm}$, $K^{\pm}$, p, neutral: γ, π^{0}, $K_{L,S}$, n, Λ.
   Output: momenta of the particles and particle identification (PID) weights

NN based jet reconstruction:
starting from particle lists, their momenta, and PID weight build a jet using NN regression algorithm

Work in progress

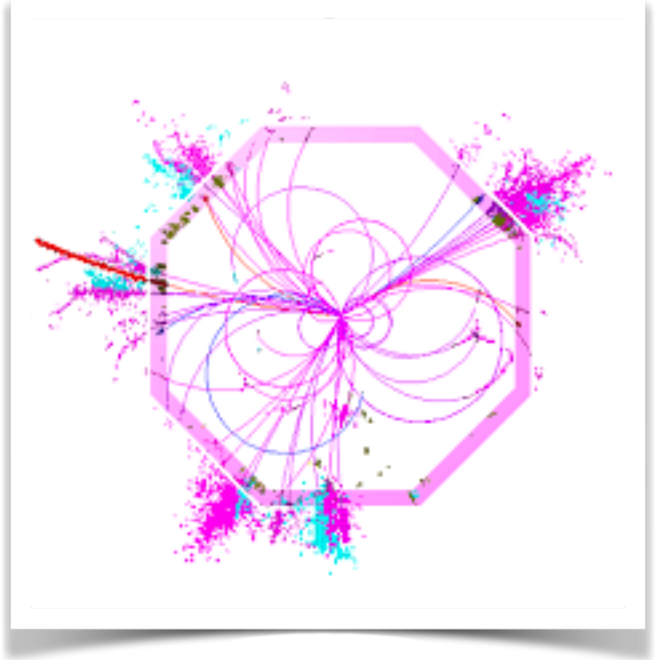
Software Implementation

- Input from detector simulation (EDM4HEP) format
- Reading using key4HEP *code*
- Dumping algorithm, input variables for NN training
- NN training using *Tensorflow* on CPU/GPU

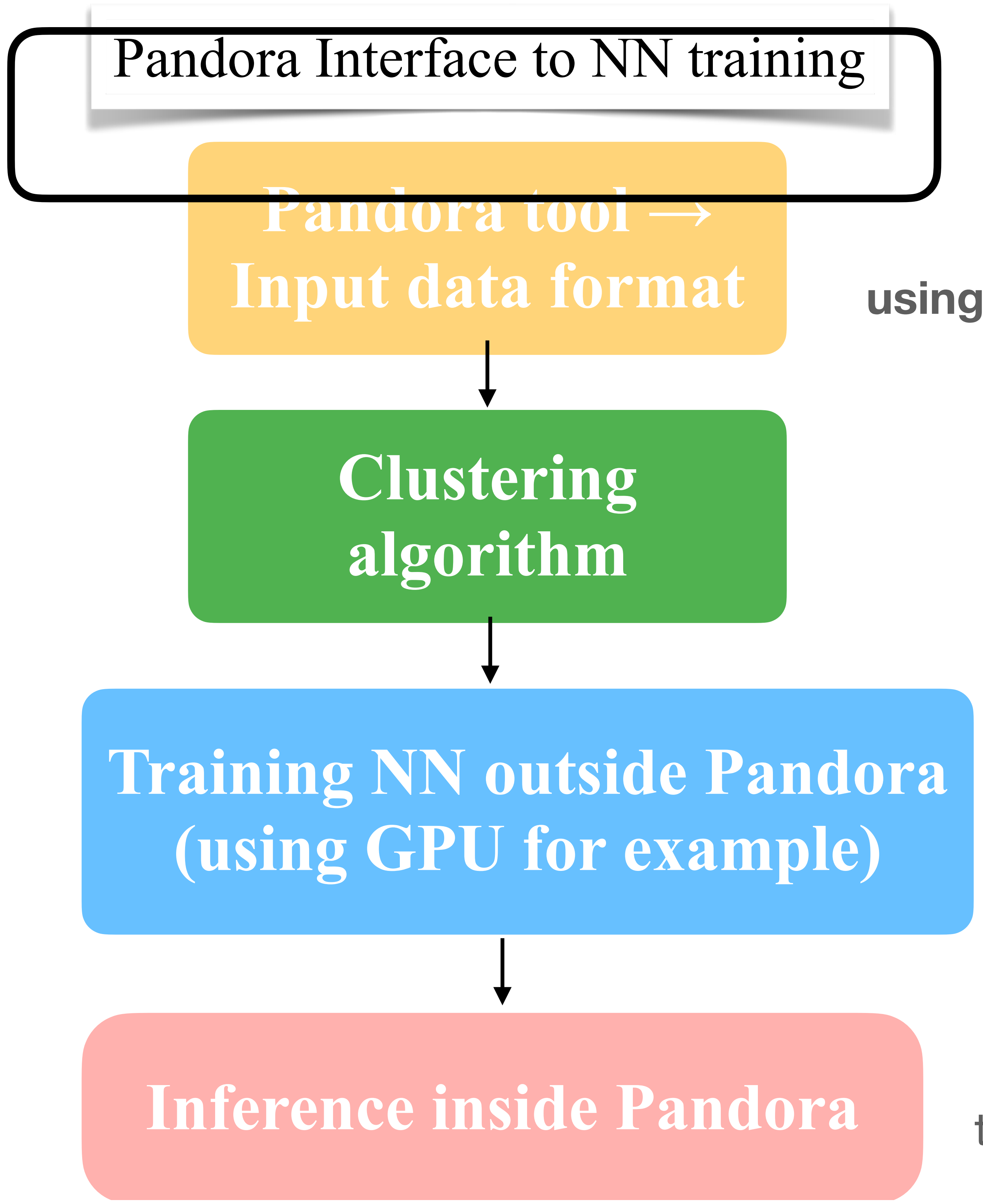
Software stack

The project will be developed in key4HEP and Pandora → interface Pandora with key4HEP

- key4HEP: general software framework developed for many experiments *https://github.com/key4hep*
  - GEANT4 implementation in KEY4HEP already started



- Pandora Particle Flow Algorithm *https://github.com/PandoraPFA*
  - Collection of pattern recognition algorithm, the idea is to insert our algorithm inside Pandora and compare its performance with already existing algorithms
  - Algorithm already present: several clustering algorithms, non NN based Particle Flow algorithms, NN based reconstruction algorithm for liquid Argon TPC for the DUNE experiment
  - Training outside Pandora, use interfaces to produce input to the training data format and inference



not feasible, using dumper directly from key4hep

using EDM4HEP and key4hep

the only planned interface at the moment

present plan: using Pandora as an algorithm collector, main interface through KEY4HEP and EDM4HEP
present status: NN development for electron identification and reconstruction

Infrastructure and artificial intelligence approach

- CPU & GPU installation performed on INFN Roma Tre cluster
  - The site is equipped with about 50 server (mainly based on Blade technology) with a total amount of cores available (or VCPU) of about 1500 interconnected with Infiniband (DDR 20Gbps e QDR 40Gbps)
  - The site has also 2 Graphical Processor Unit (GPU) K 80 (4 in total: 2 x K40), where jobs can be parallelised if needed
  - There is a storage system present in the cluster for a total amount of about 700TB
  - **Extensive innovation next year**, in order to double the CPU and storage system

Two neural network (NN) approaches tested

Deep NN (DNN) approach
1. **10 hidden layers architecture**
2. **20 hidden layers architecture**

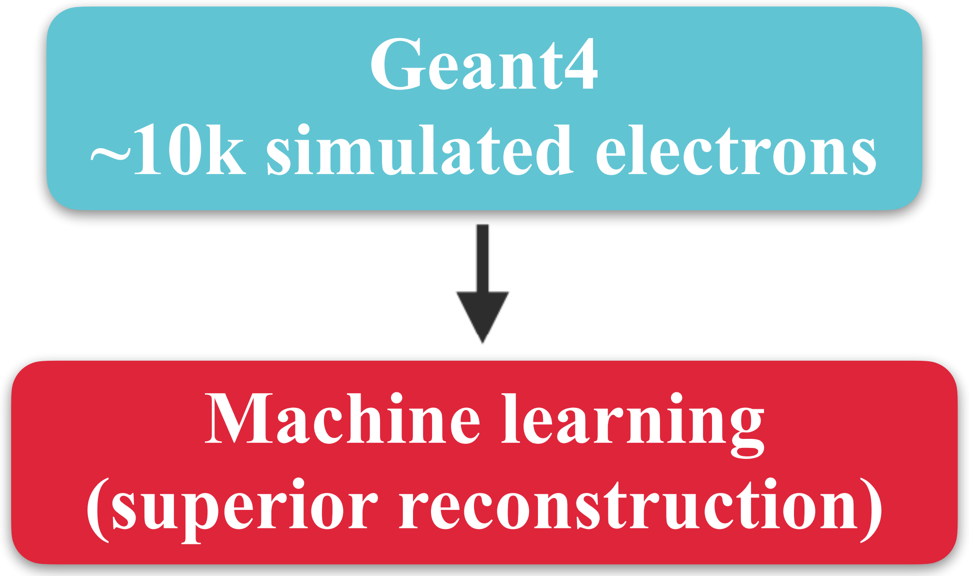
- Pro:
  - Fully exploits the fibres granularity in the calorimeter
- Cons:
  - Memory issues to process events in the full energy spectrum (0-125 GeV) for input electrons
  - Angular resolution not available

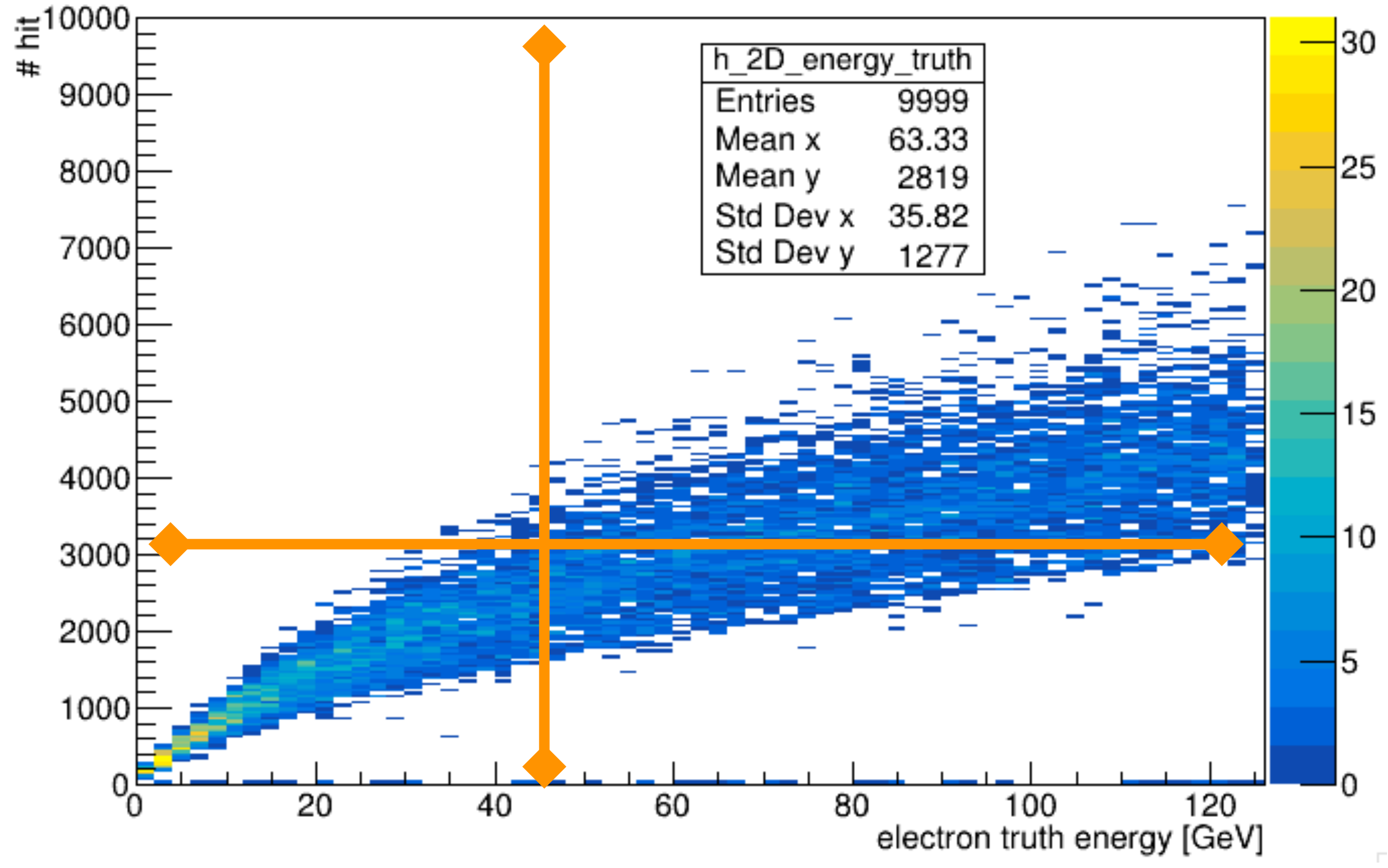
Convolutional NN (CNN) approach
1. **VGG-like architecture**
2. **VGG-like architecture & proto-clustering**

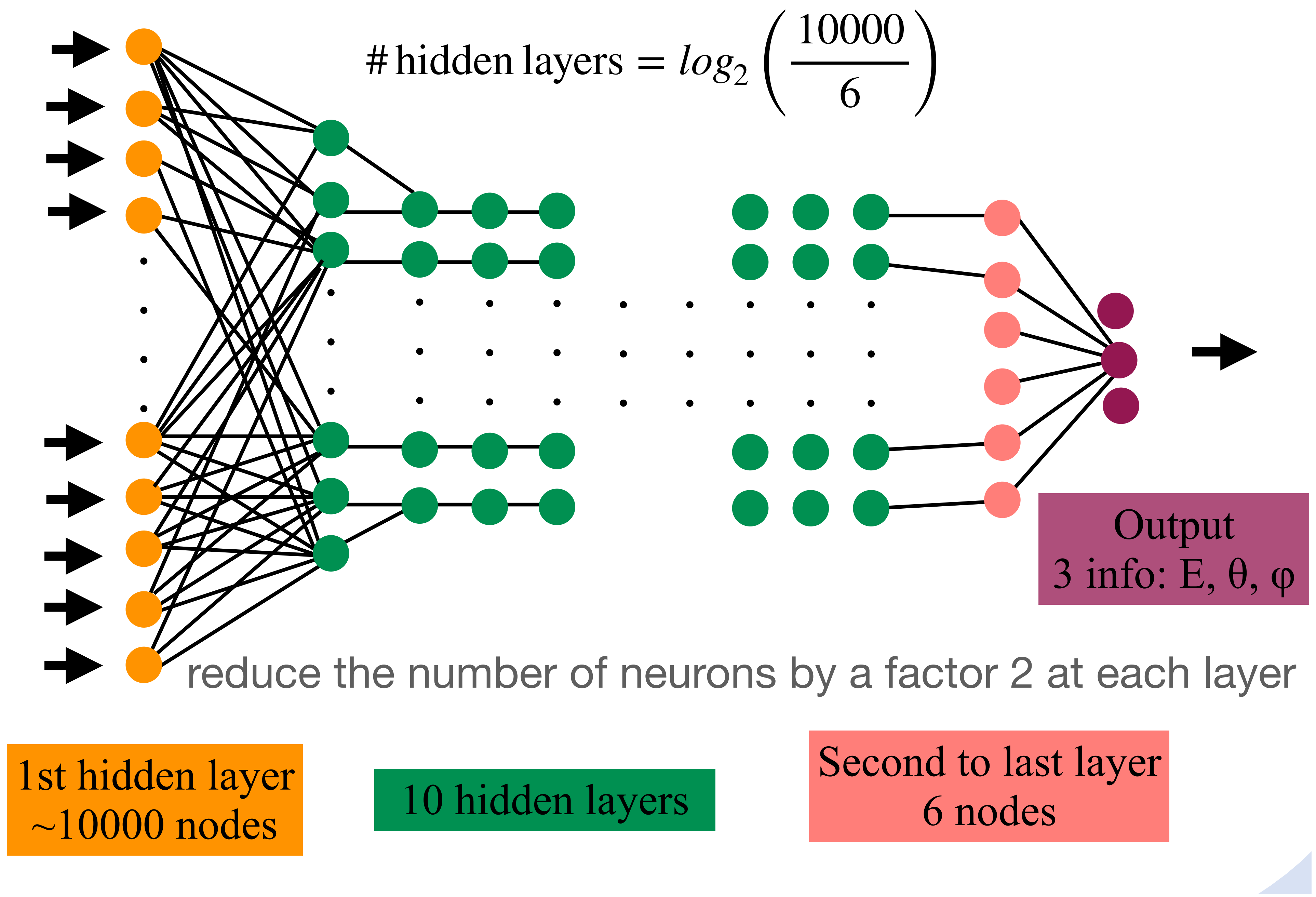
- Pro:
  - Solves the memory issues → able to exploit electrons info in the full energy range (0-125 GeV) for input electrons
  - Able to obtain also the angular resolution
- Cons:
  - Further studies needed to improve the energy and angular resolution results

DNN approach

- **Tensorflow**, interfaced with Keras, is used to build and train a NN on CPU and GPUs
- Inputs: energy and position of each hit in the shower generated by the impinging electron and recorded in both S&C fibres → NN input: 5 kinematic variables (E, x, y, z, fibtre-type) multiplied by hit multiplicity
  - Average hit multiplicity: ~ 10’000 hits per impinging electron per event→ ~ 50’000 info per event
  - Maximum hit multiplicity: ~ 22’500 hits per impinging electron per event→ ~ 112’500 info per event
  - Zero padding approach: if the number of hits in the event is less than the max hit multiplicity, set to zero the remaining positions in the array
- # initial nodes = # input info
  - Exploit the average hit multiplicity × kinematic variables as #initial nodes to reduce the complexity of the problem
- Need to reduce the number of inputs due to GPU memory issues (too many trainable parameters)
- Speed of the algorithm:
  ~10 minutes on GPU
  ~2 hours on CPU

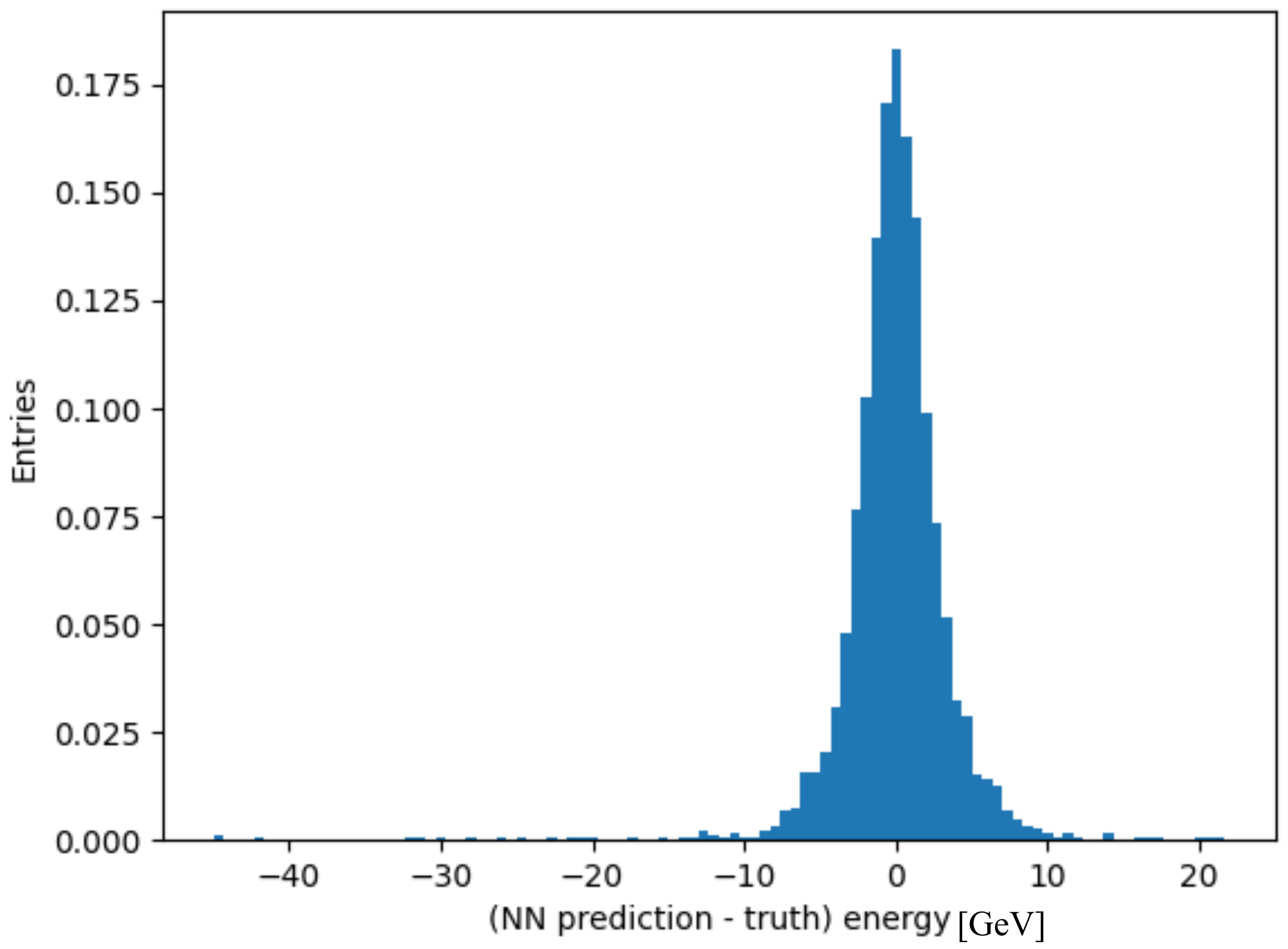


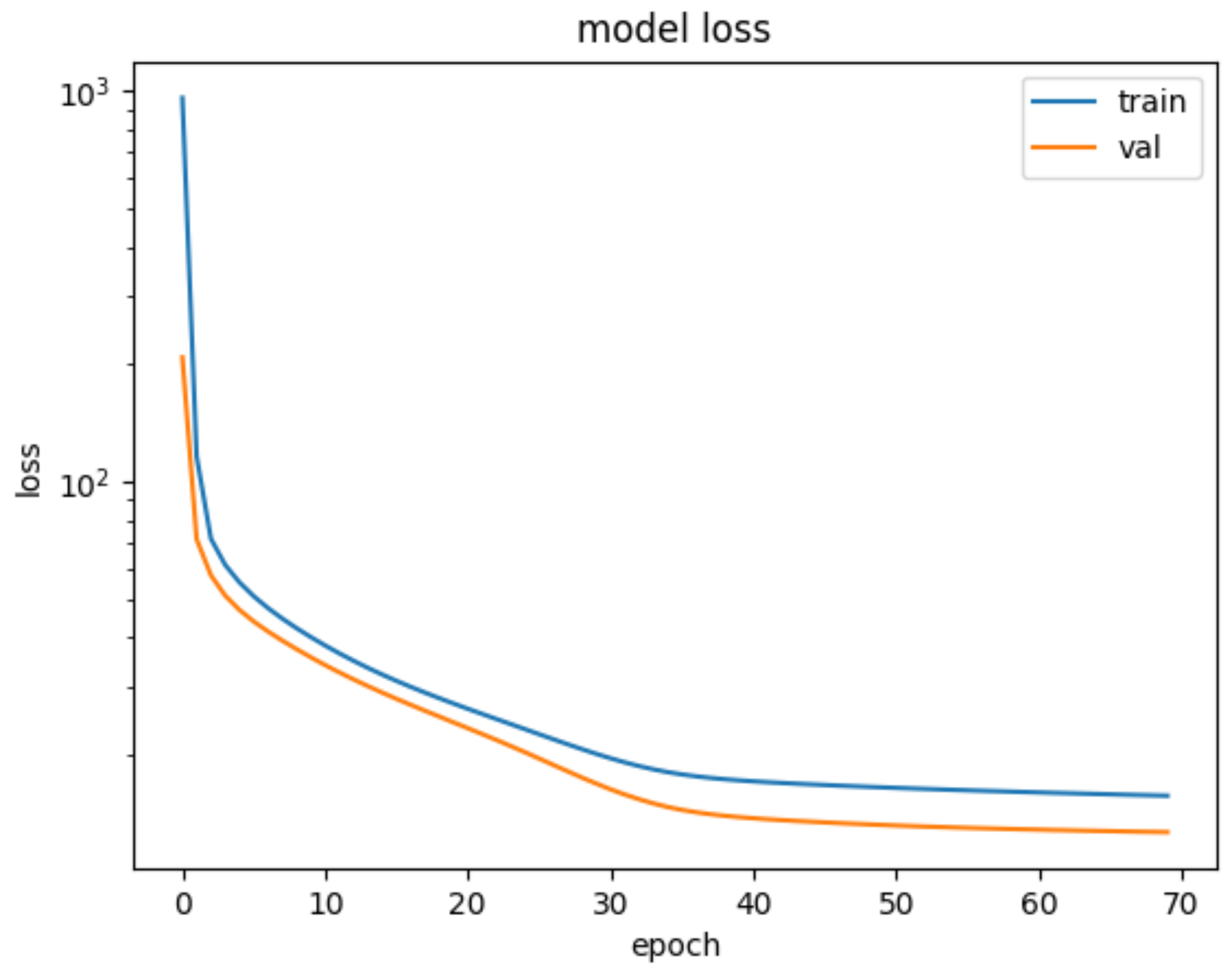


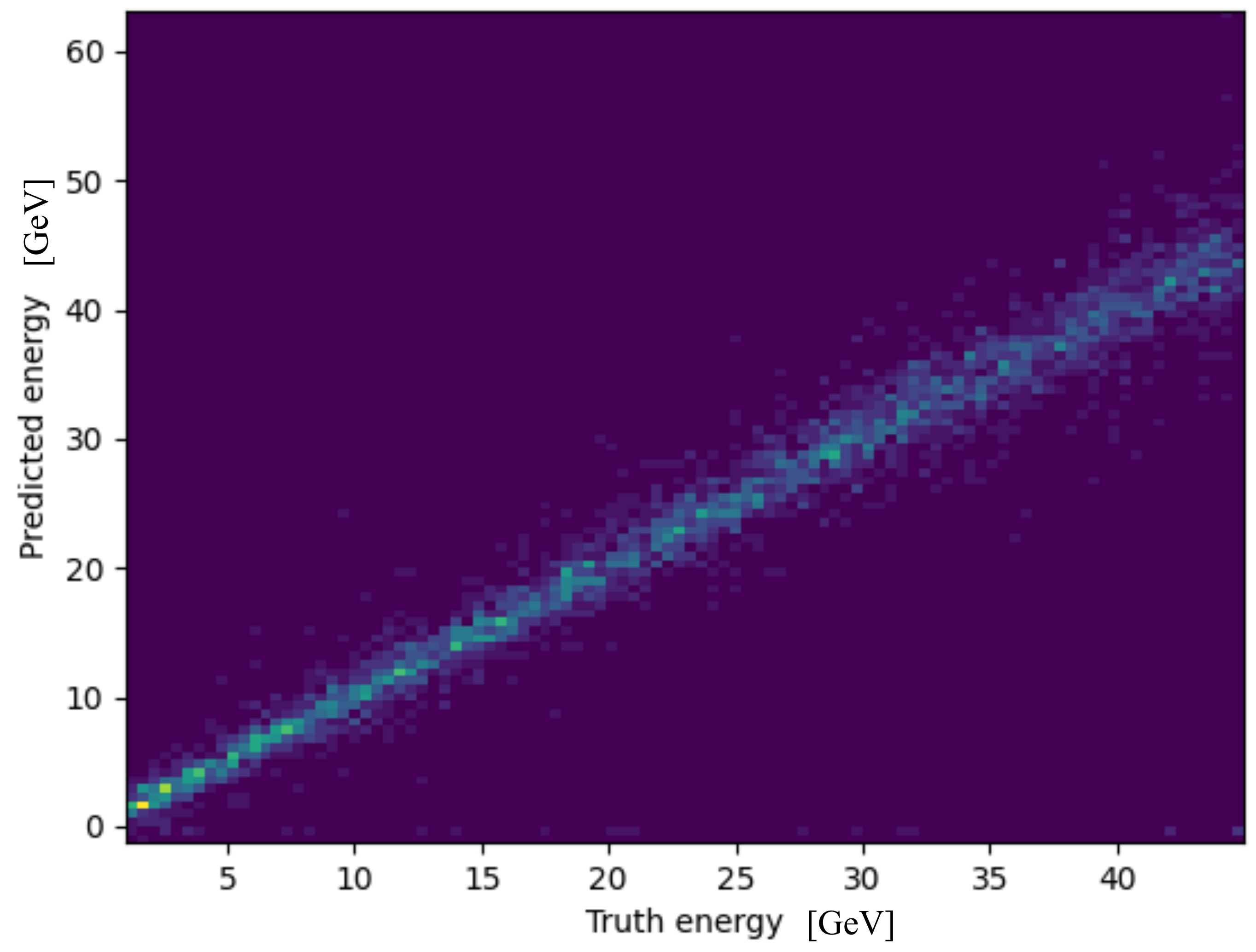


DNN approach: electron energy resolution

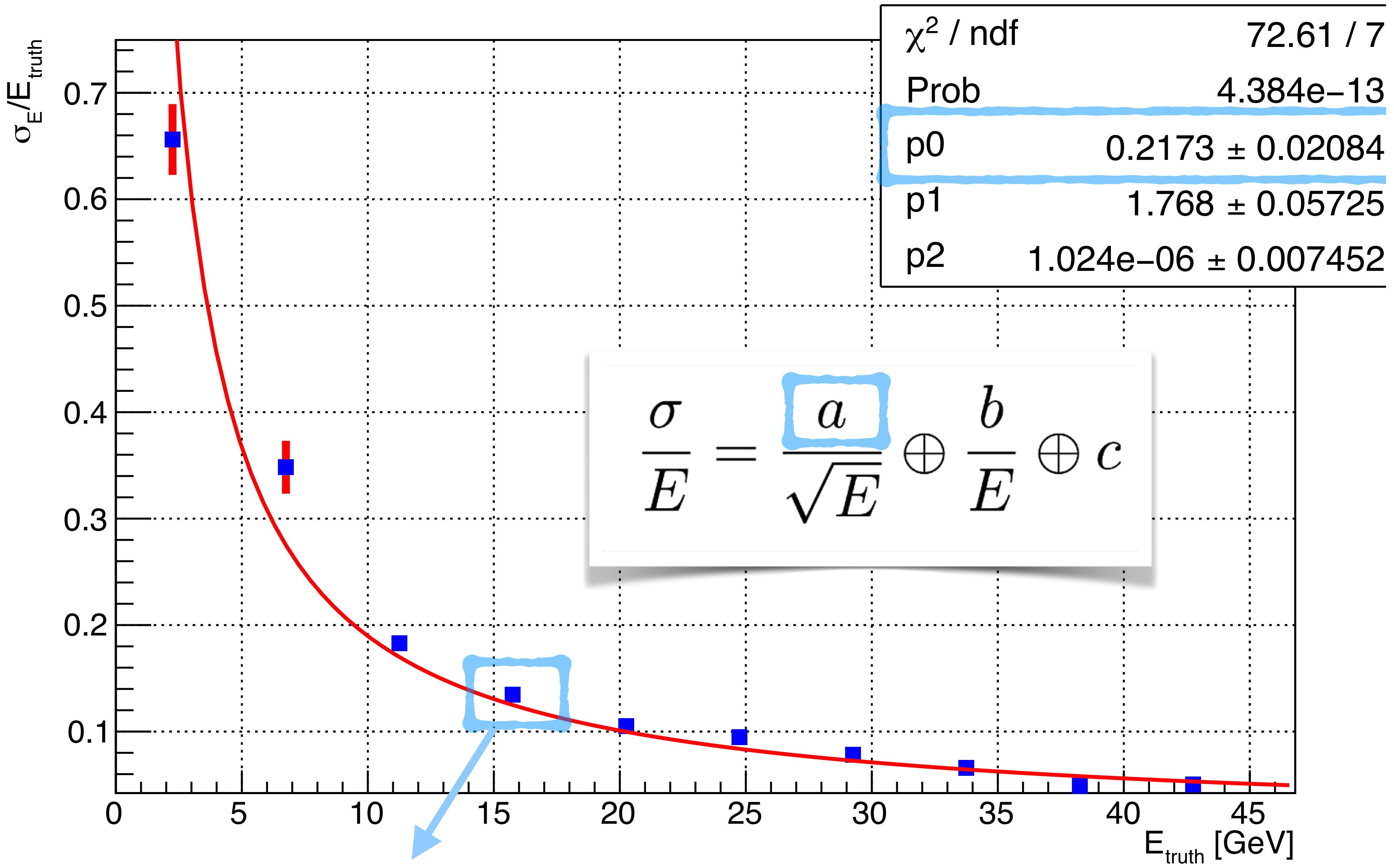
- First attempt: 10 hidden layers
- Halving number of nodes at each layer
- Model loss: MeanSquaredError()
  $$\frac{1}{n}\sum_{i=1}^{n}\left(y_{\text{true}} - y_{\text{pred}}\right)^2,$$ optimised with respect to the simulated energy of the incoming electrons
- Adam, a stochastic optimiser, is used as optimiser to minimise the loss *Reference*

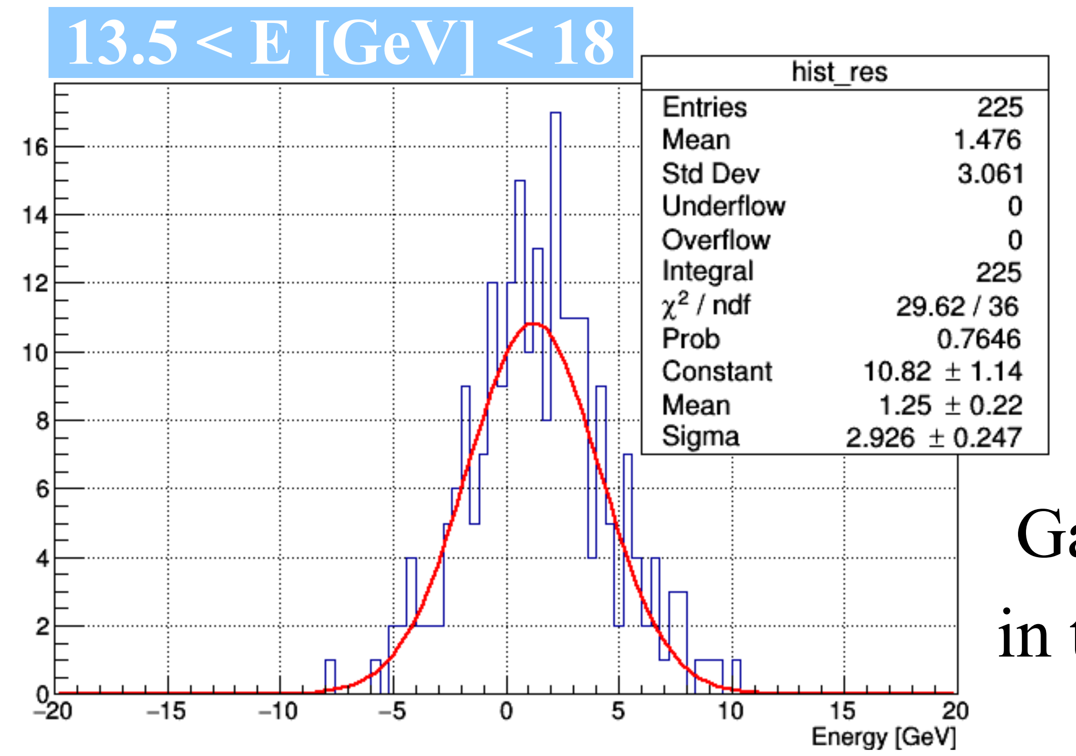






- NN configurations might be under-performing
  - Too easy architecture? **Work in progress**

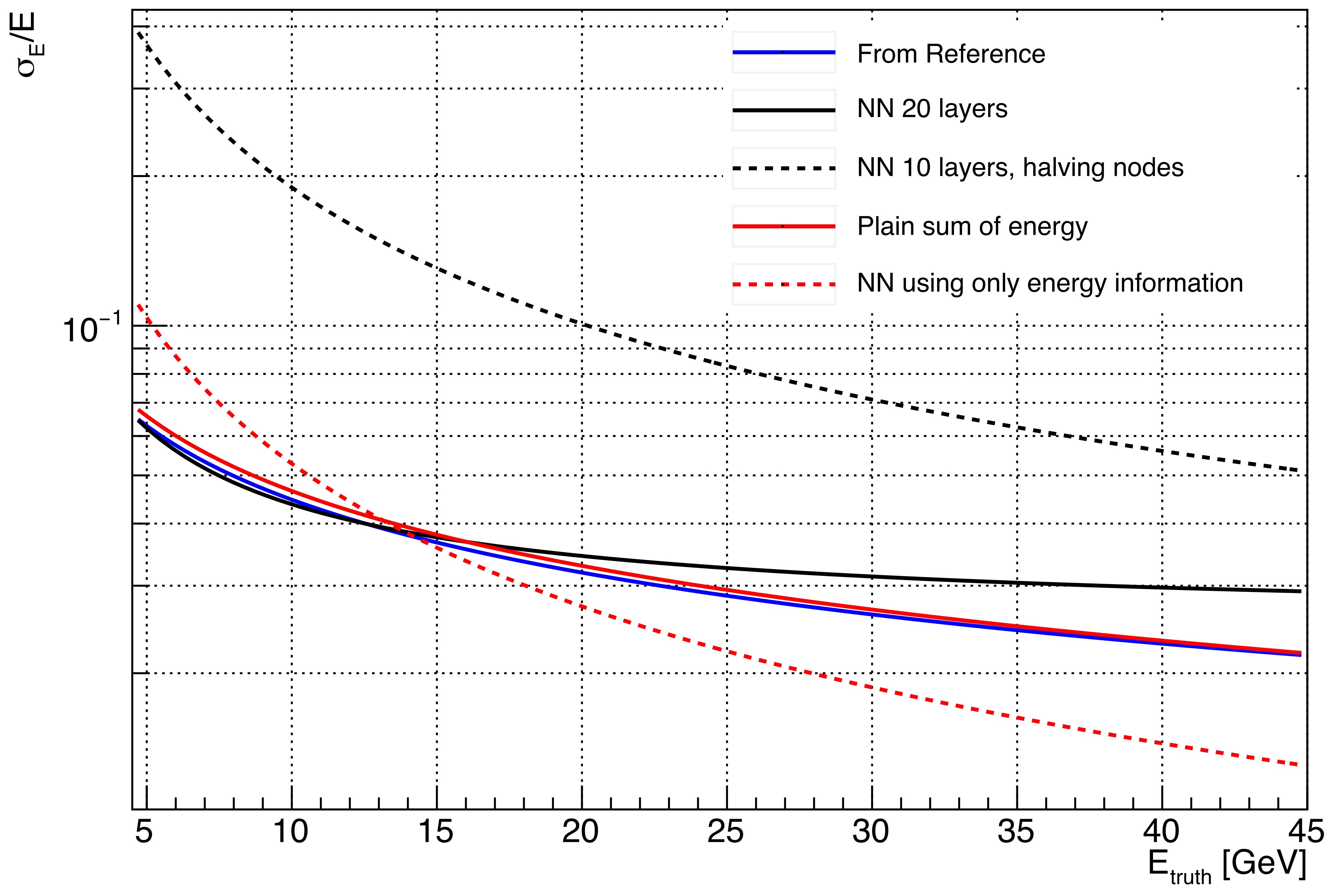




Gaussian fit performed in truth energy slices

DNN approach: comparing algorithms

- As a sanity check, we compared our energy resolution results with:
    - A reference https://inspirehep.net/literature/1861660
    - The energy resolution obtained simply summing up the energy deposits in the fibres (S&C)
- The energy resolution improves if we double the NN layers and we keep constant the number of nodes
- **Issue**: the NN performance is still worse than the standard reconstruction → work in progress



1. using more deep NN (20 versus 10 layers) substantially improves performances, going close to the reference and to the simple reconstruction (sum of light yield in all fibers)
2. simple reconstruction outperforms NN approach when both energy and fiber position are used;
3. reducing the information, using only energy, we get better performance at high energy

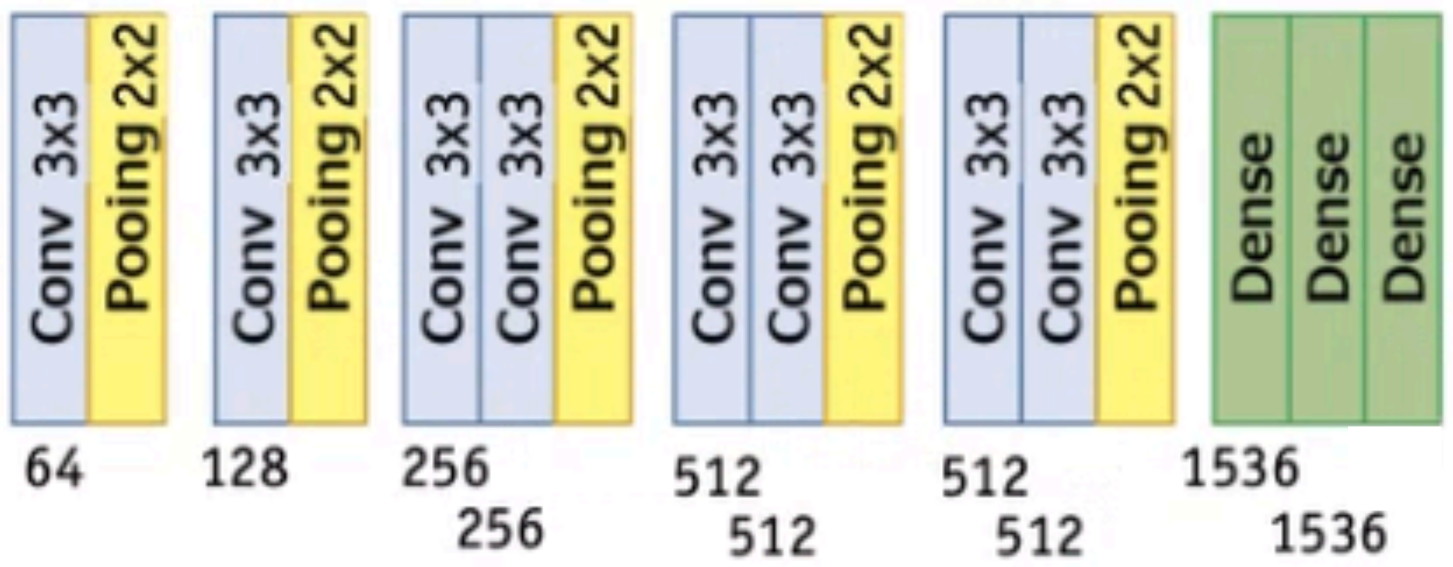
Physics seems too complex for NN to learn it, hit position seems to add confusion more than information. Puzzled by this we tried different NN architectures.

What NN needs to learn: electron bending in a magnetic field, electron radiation, fiber geometry:
the x, y, z coordinates are the 3D coordinate of the fiber endpoint pointing to the Interaction Point.

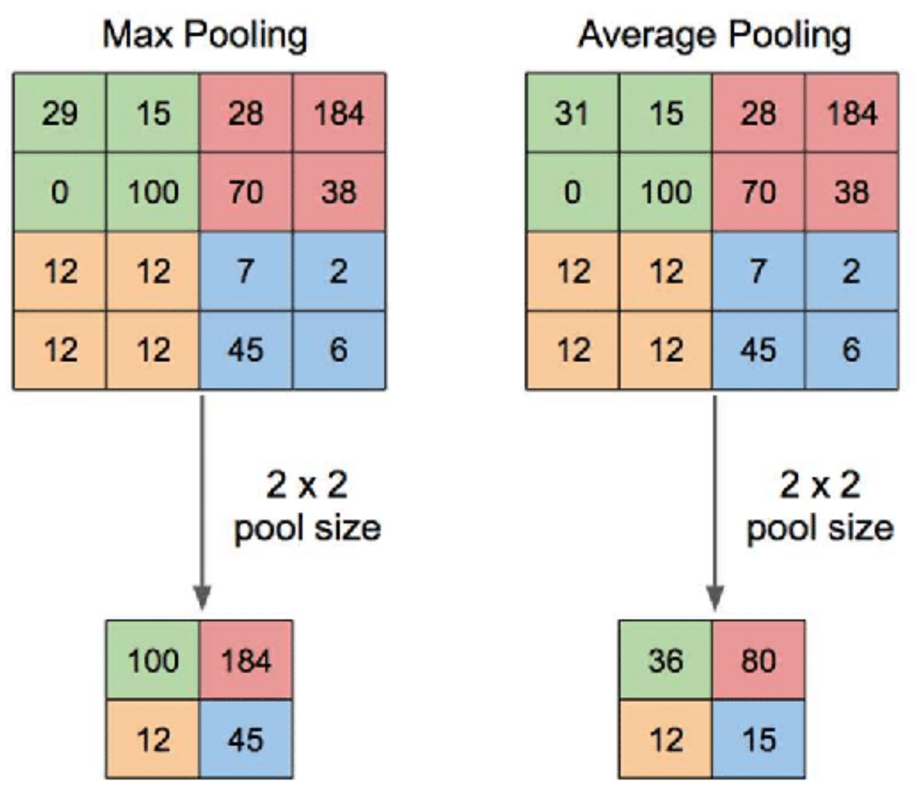
CNN approach

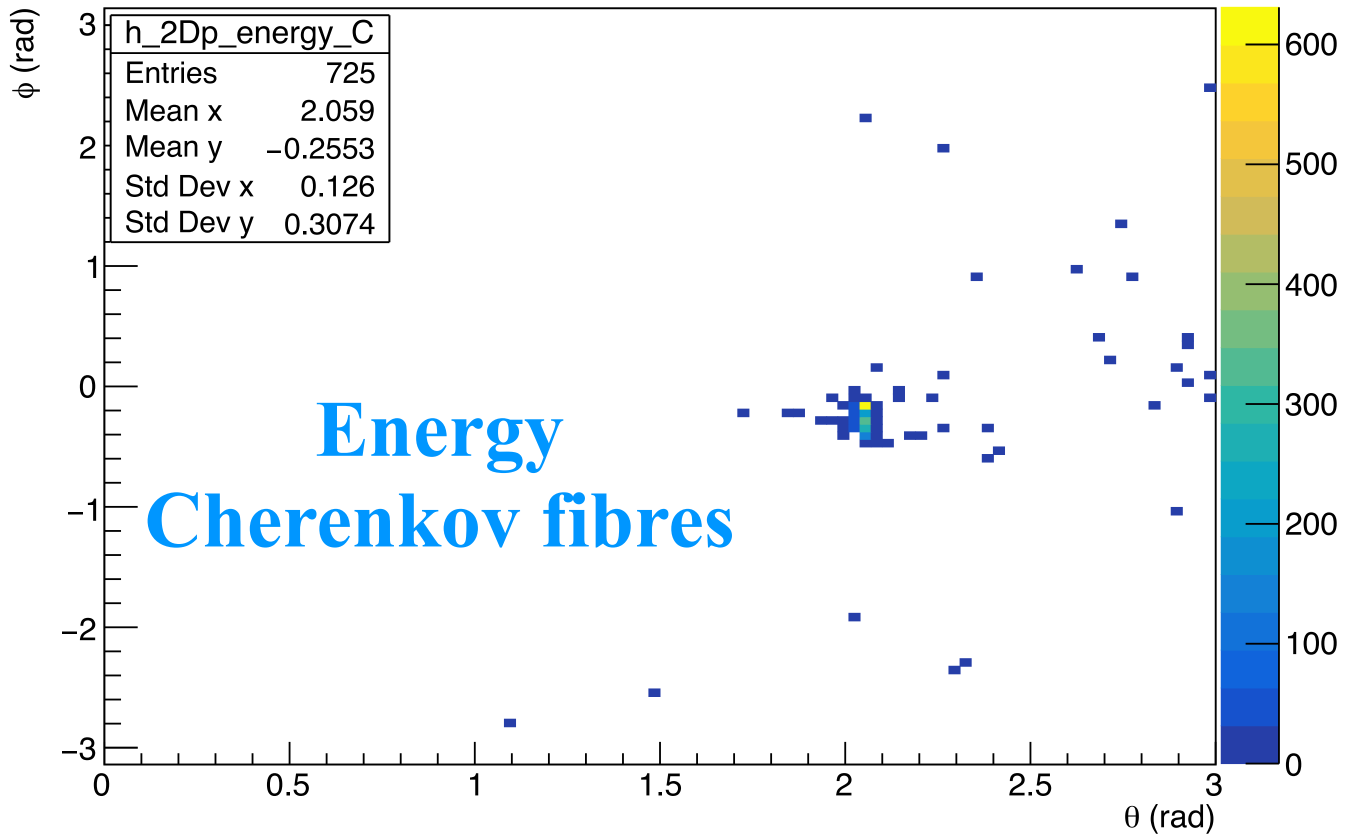
- CNN tests motivated by memory issues with DNN (many fibres for input info)
- VGG-like architecture (Visual Geometry Group - Very Deep Neural Networks)
  - No batch normalisation
  - 5 convolutional 2D layers
  - Flatten and 3 dense layers
  - 3 outputs
  - Overcome memory issues, possibility to use the full energy range
- Tested both MaxPooling and AveragePooling methods

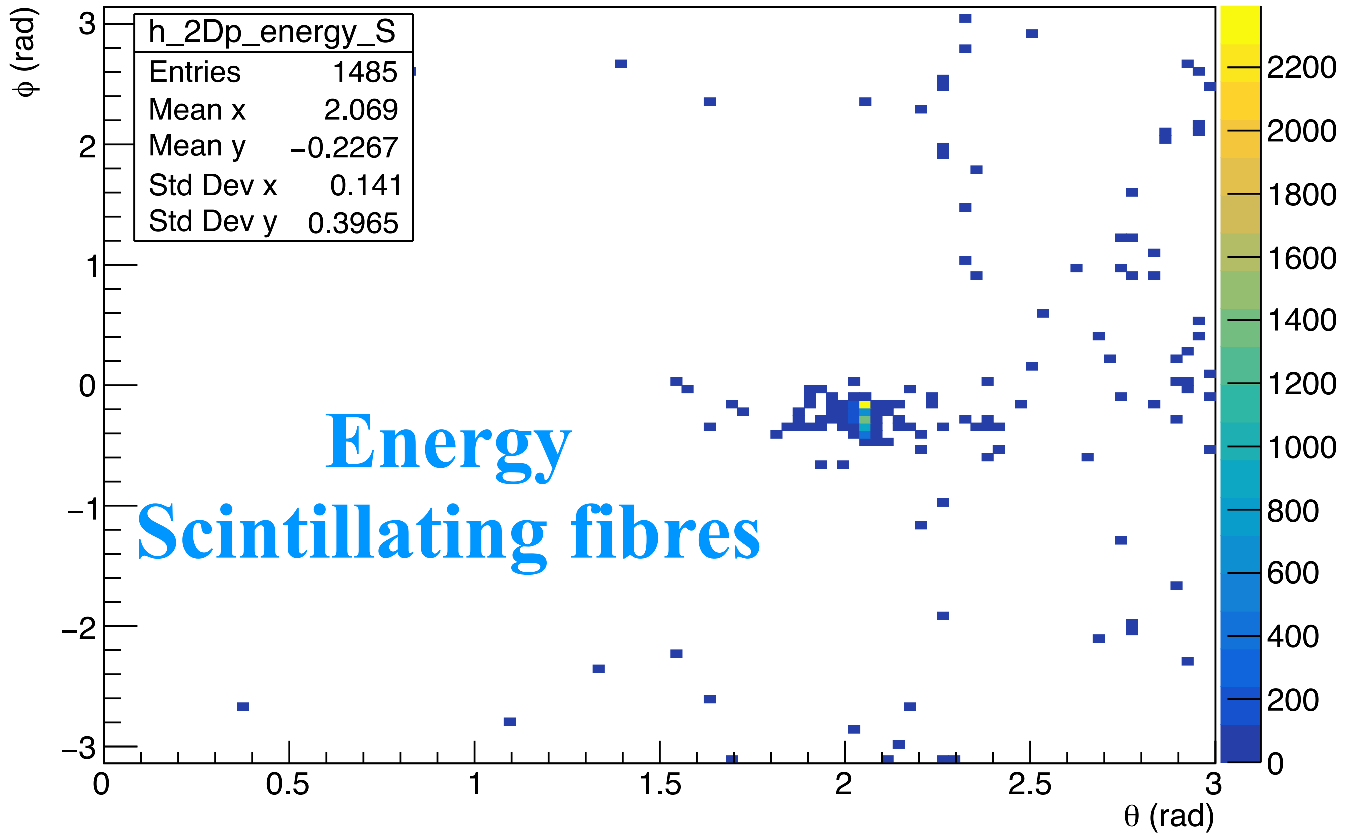
100x100xN Input tensor



MaxPooling or AveragePooling





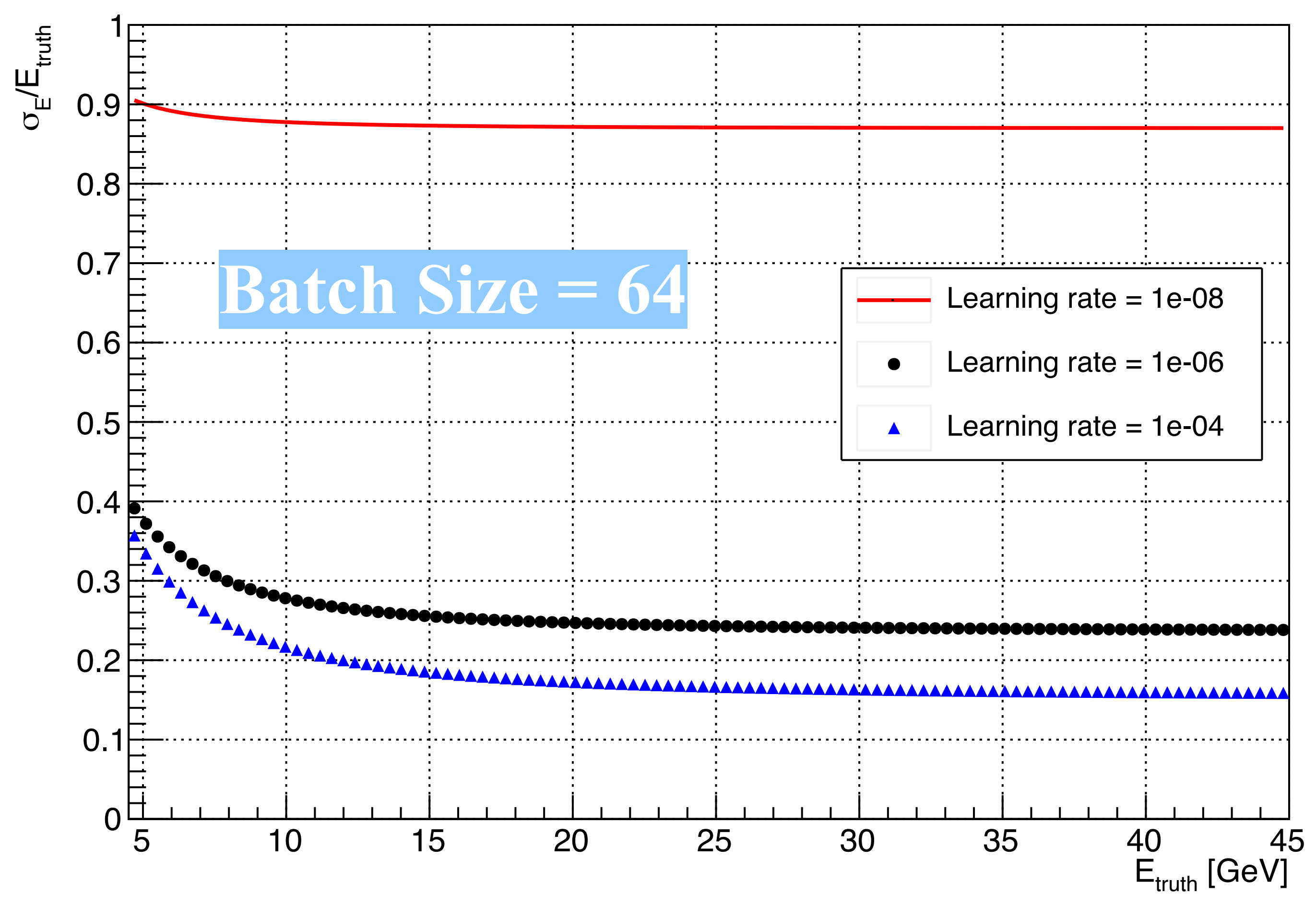


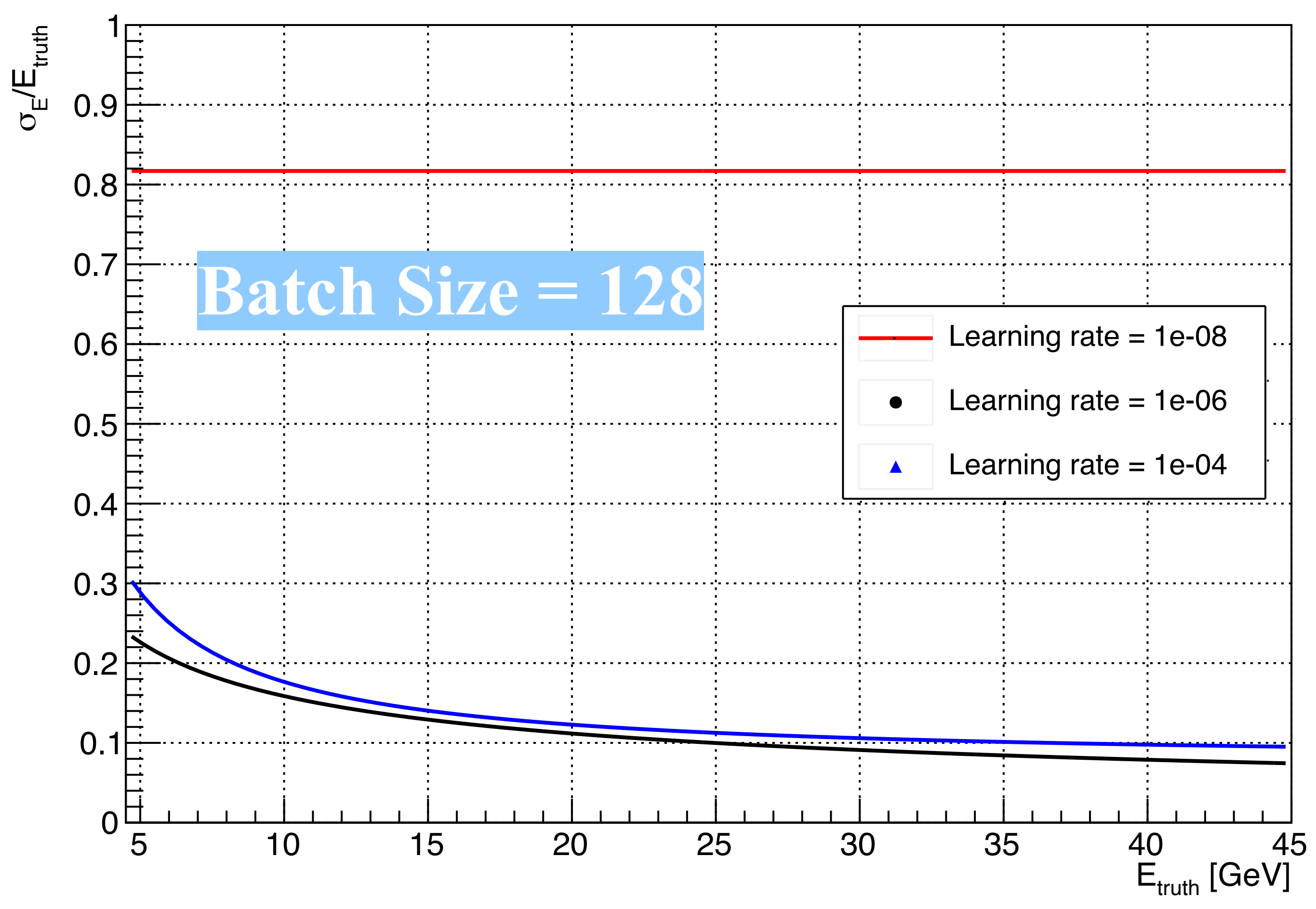
- Testing the CNN approach with zero padding
- Create numpy arrays with shape (N,N,d) where NxN is a matrix for the φ-θ granularity (100x100 bins) and d represents the features associated to each pixel: energy, x, y, z
- Discrimination between scintillating or Cherenkov fibres

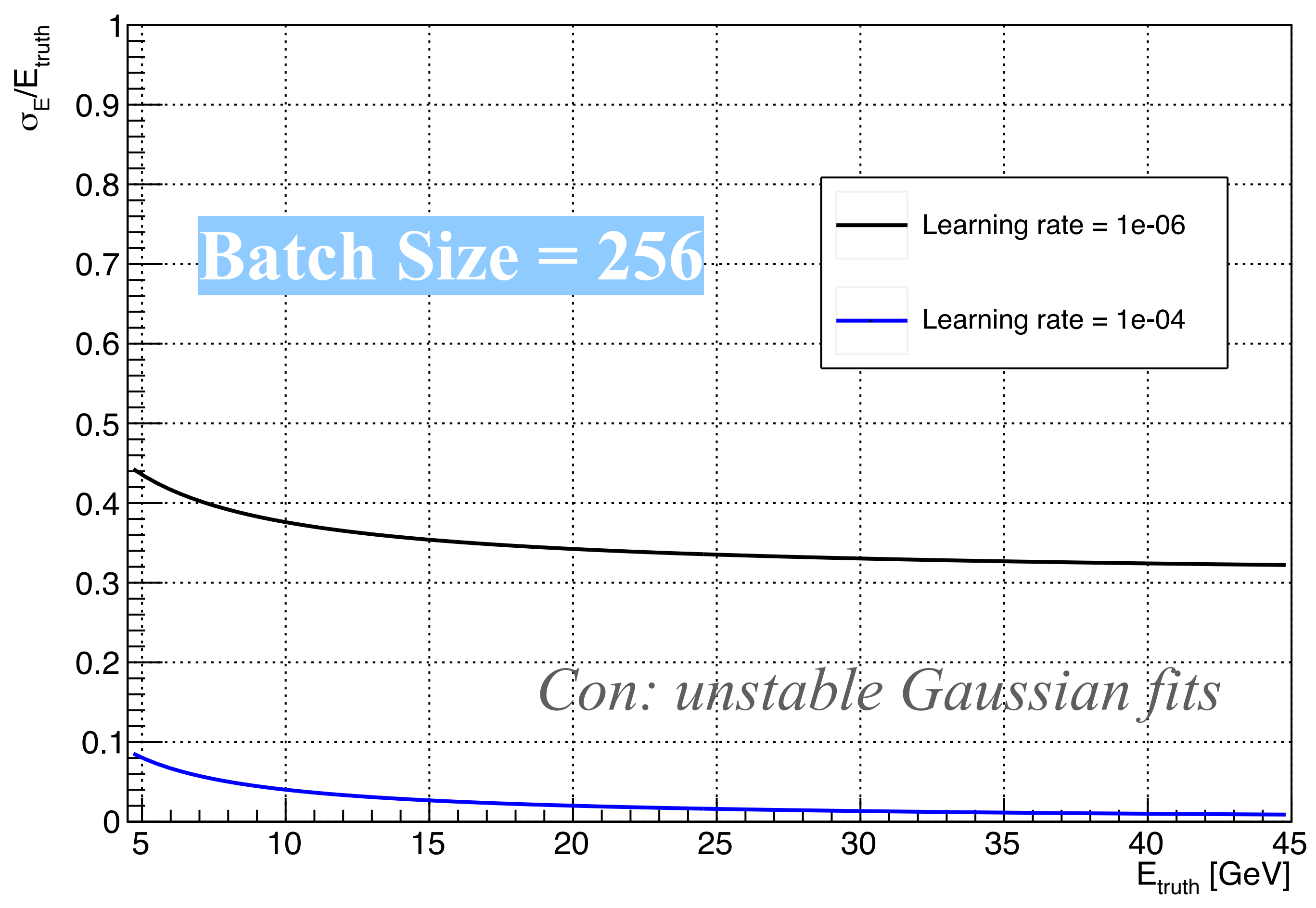
CNN approach: hyperparameter optimisation

CNN approach
1. VGG-like architecture w/o proto-clustering

- Energy resolution used as figure of merit for batch size and learning rate optimisation







**Batch size: it is a number of samples processed before the model is updated*

**Learning rate: it is a hyper-parameter used to govern the pace at which an algorithm updates or learns the values of a parameter estimate*

CNN approach with proto clustering

- as for DNN, CNN is also not able to reconstruct the electron position in the calo, tried a proto-clustering approach: try to simplify the CNN work by defining the average position of the energy deposits, all fired fiber coordinates are defined respsect to the proto-cluster position.

Clustering ?

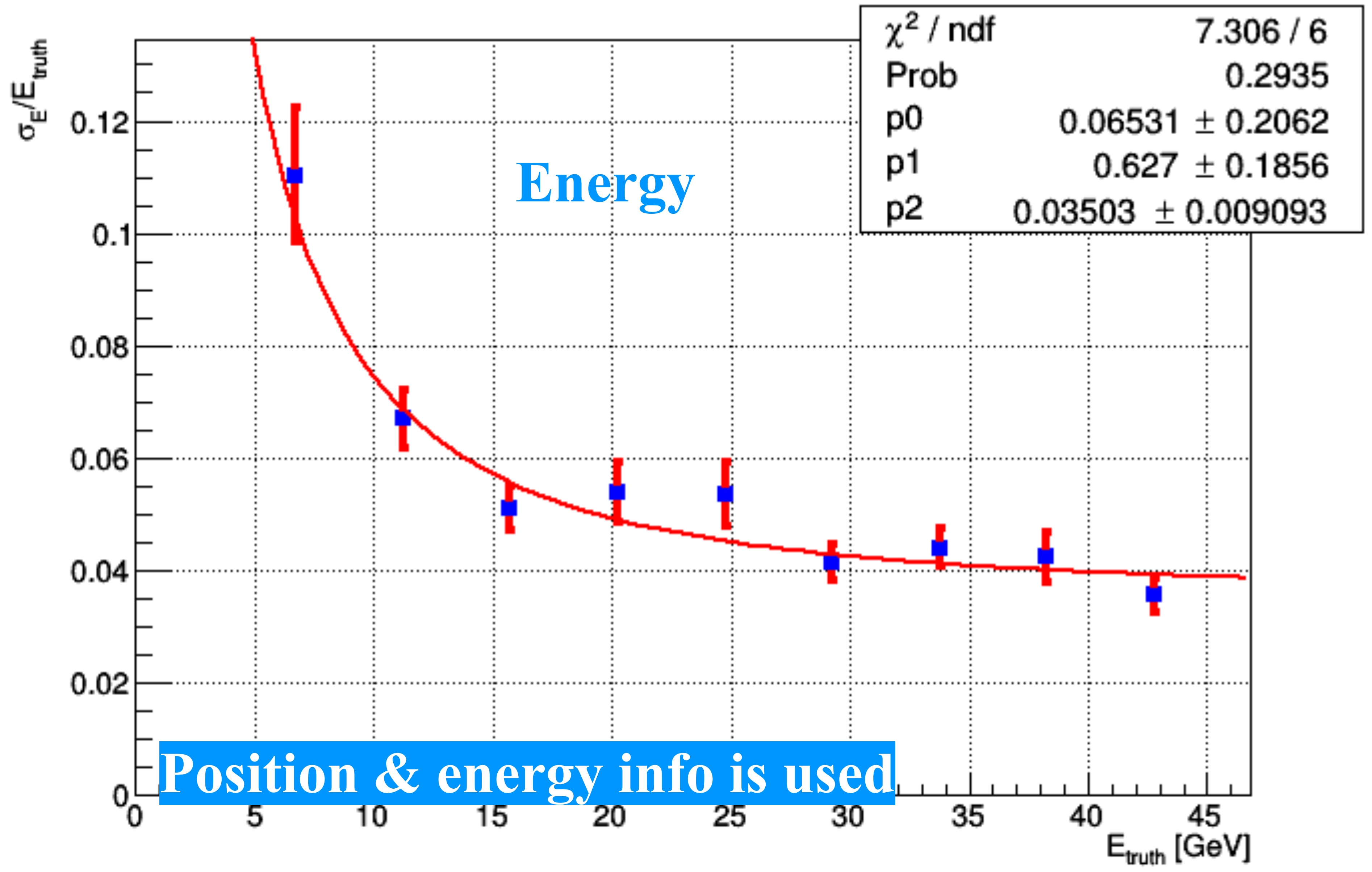
Clustering seems the obvious way to simplify conceptually the algorithm

1. Identify energy deposits released by a single particle, collect them, and apply energy regression at cluster level;

2. Preliminary test: hit energy and distance wrt the centroid used as NN input

$$d_i = \text{position(x, y, z)}_\text{i} - \frac{\sum_{i=0}^{i=N_{hits}} \left(\text{position(x, y, z)}_\text{i} \cdot \text{energy}_\text{i}\right)}{\sum_{i=0}^{i=N_{hits}} \text{energy}_\text{i}}$$

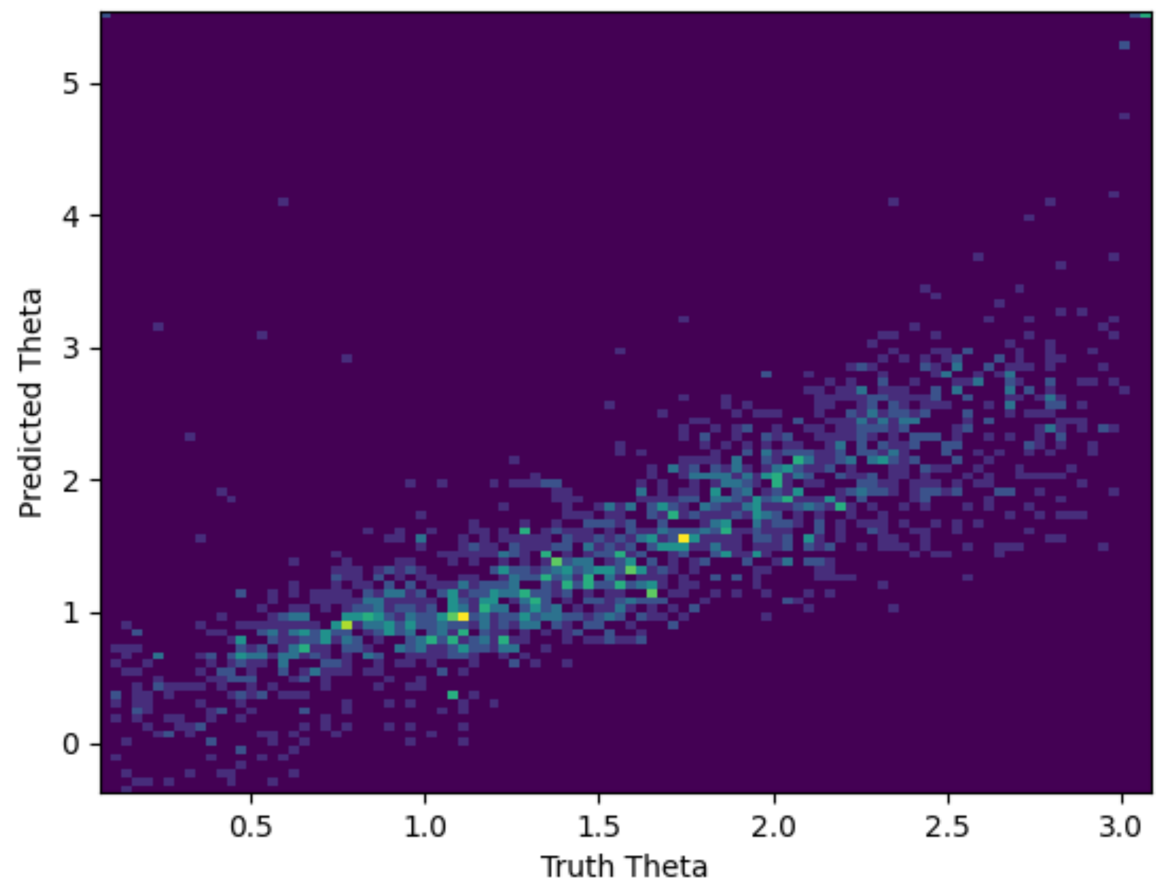
3. Next step: exploit clustering algorithm in the Pandora framework

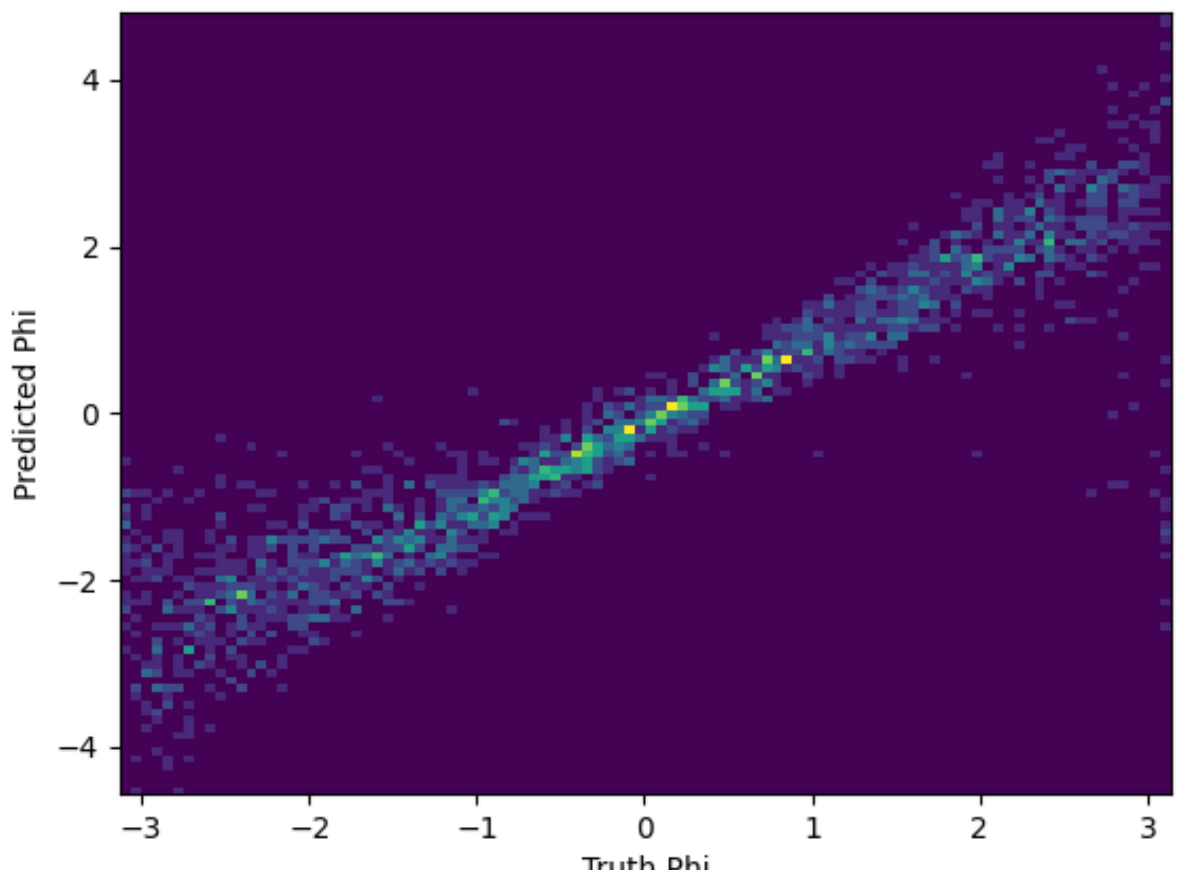


CNN approach angular resolution

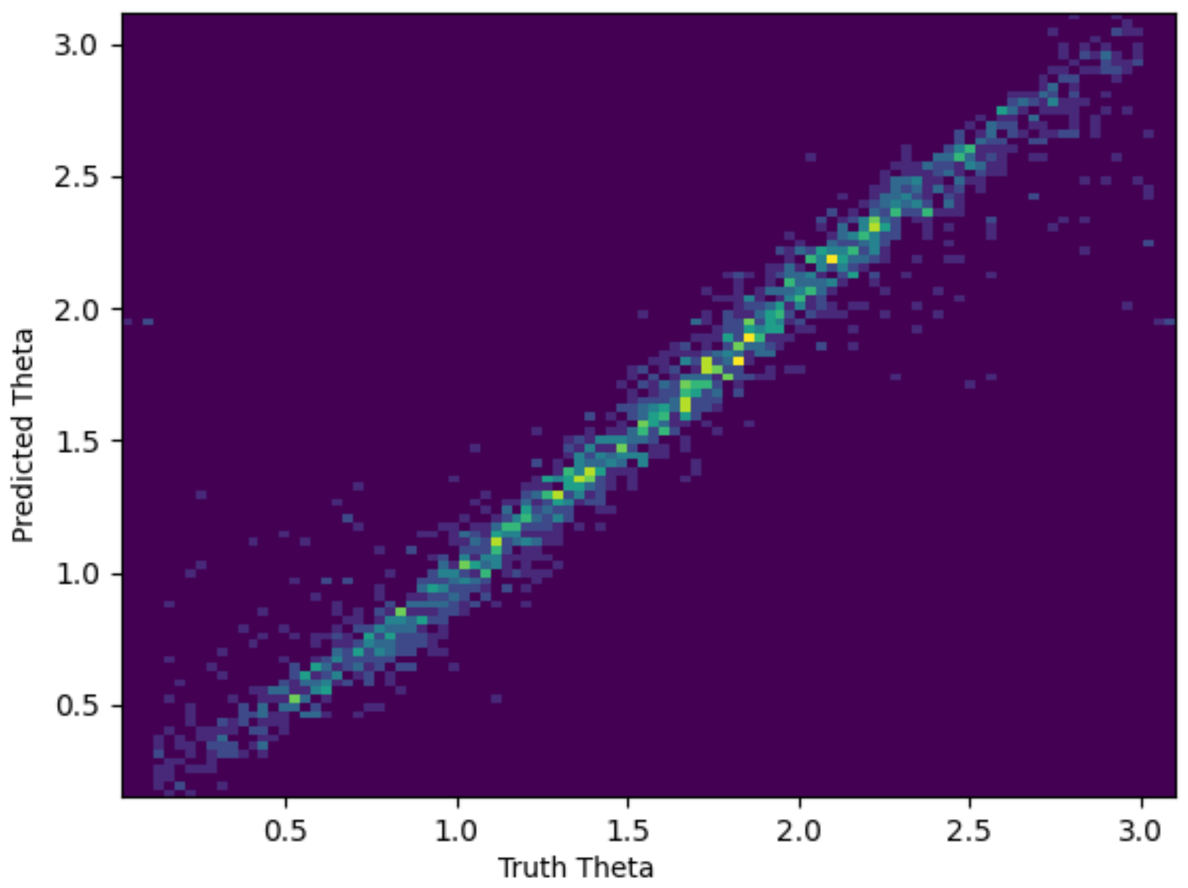
- Improvements observed if a proto-clustering is applied

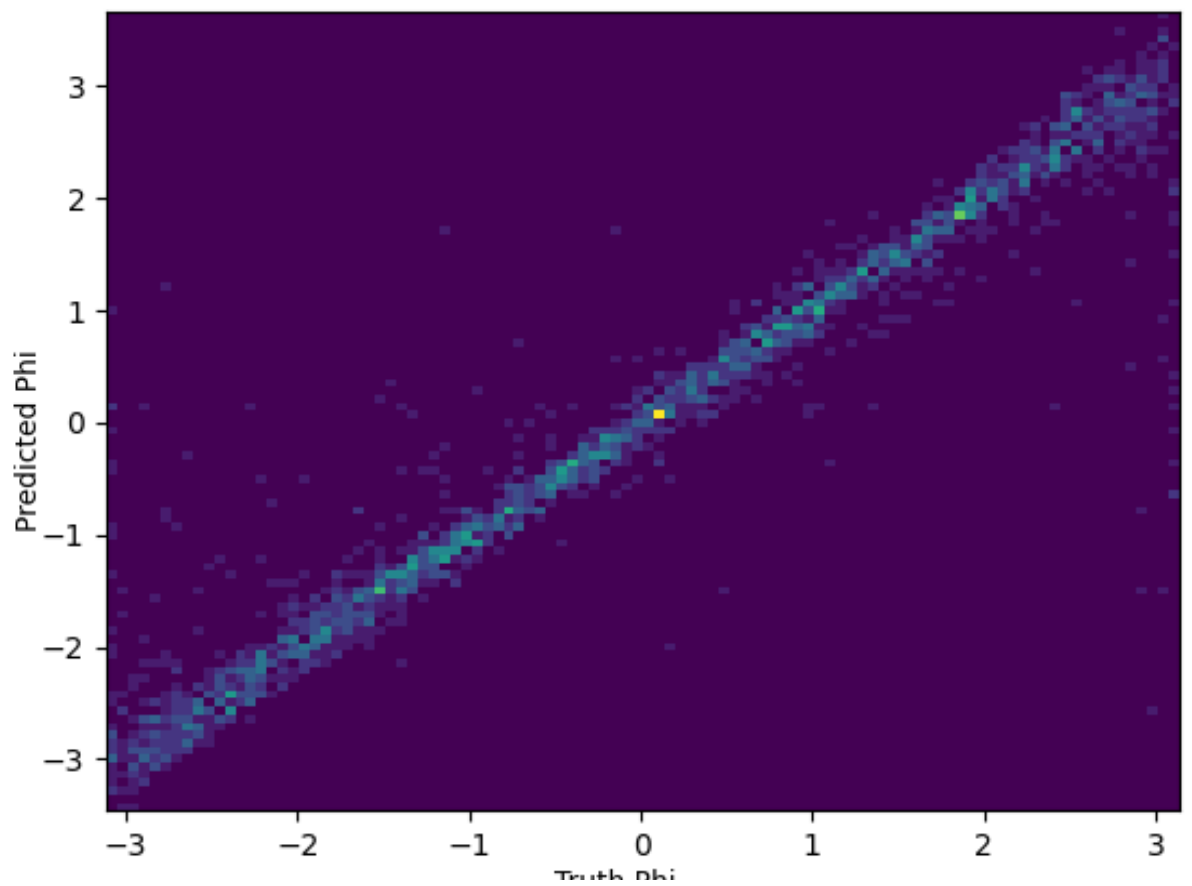
CNN approach
1. VGG-like architecture w/o proto-clustering



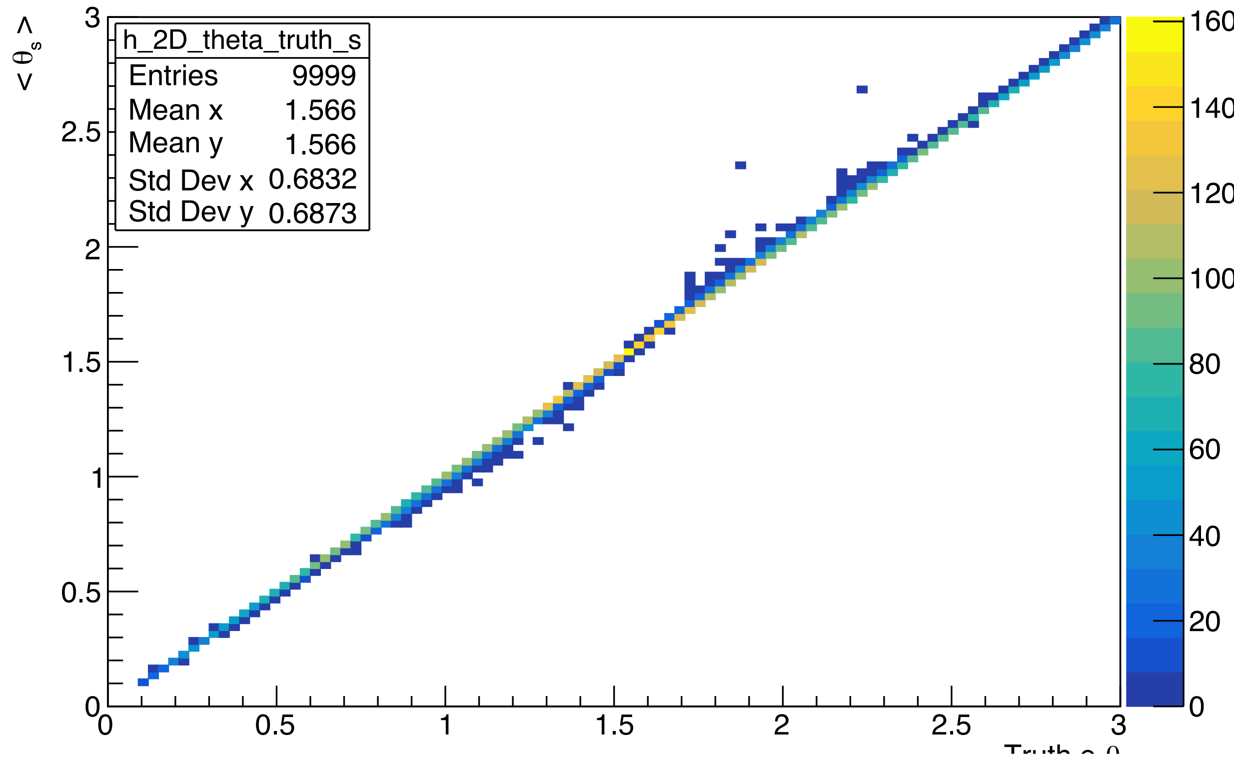


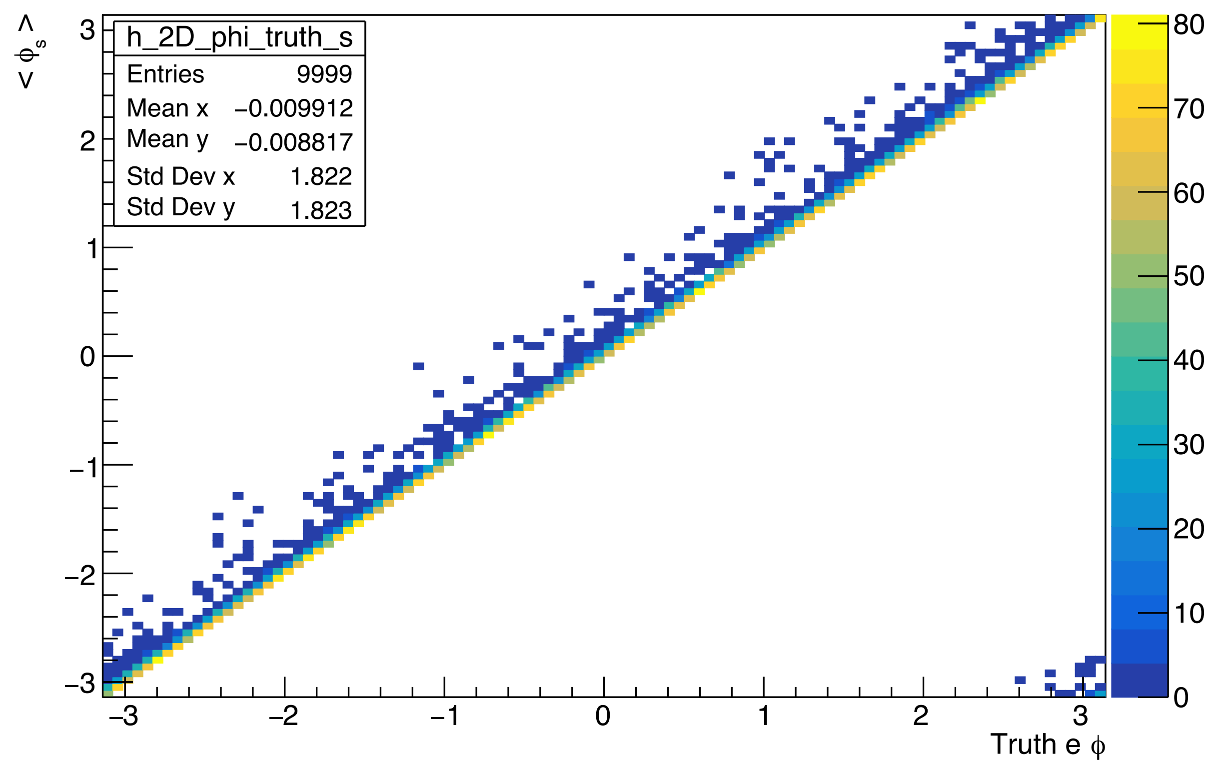
CNN approach
2. VGG-like architecture with proto-clustering





proto-cluster centroid





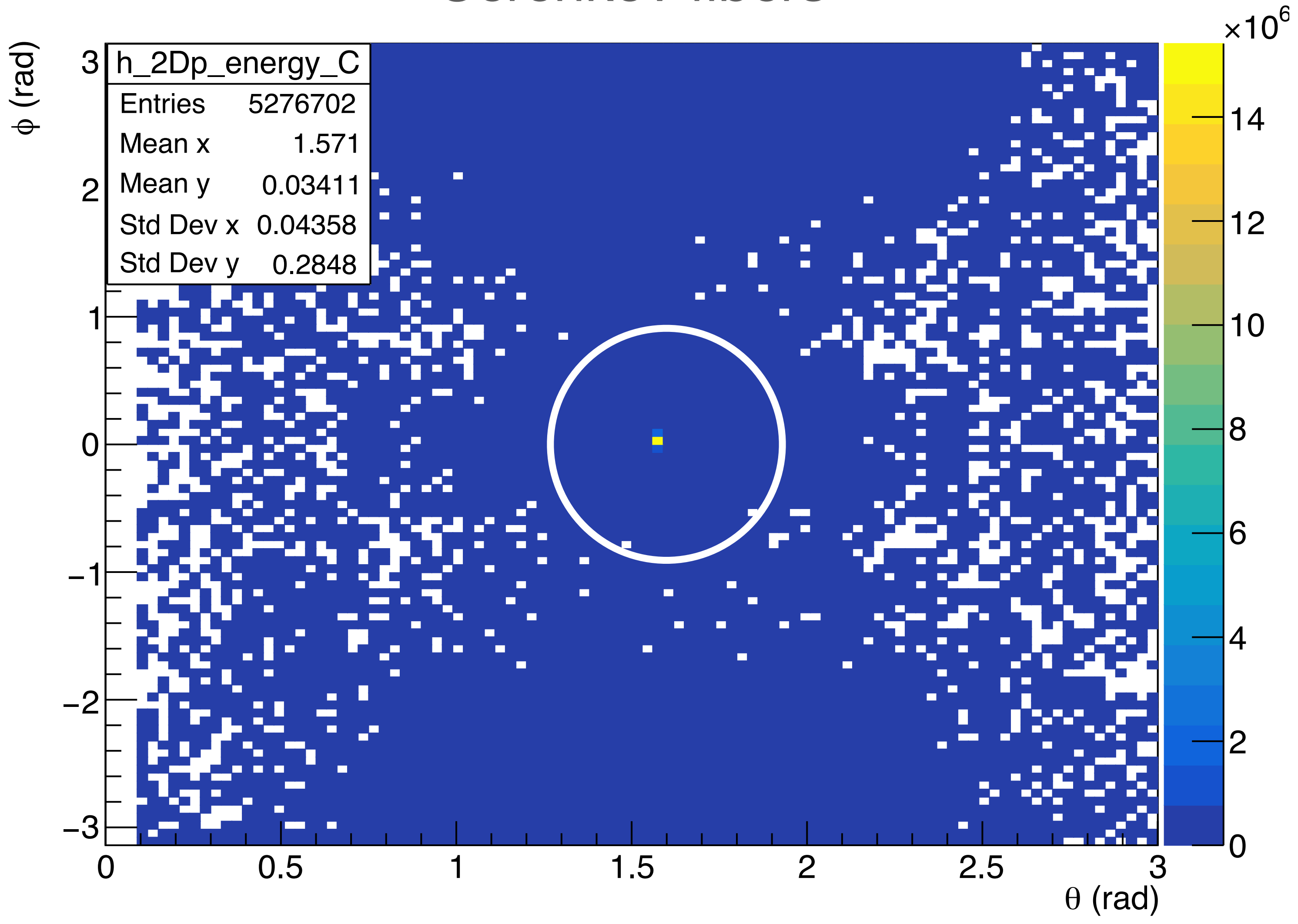
improvement observed, but it doesn't look CNN is adding anything on top of the proto-cluster (it probably deteriorates it)

Simplifying the problem: going to pencil like simulation

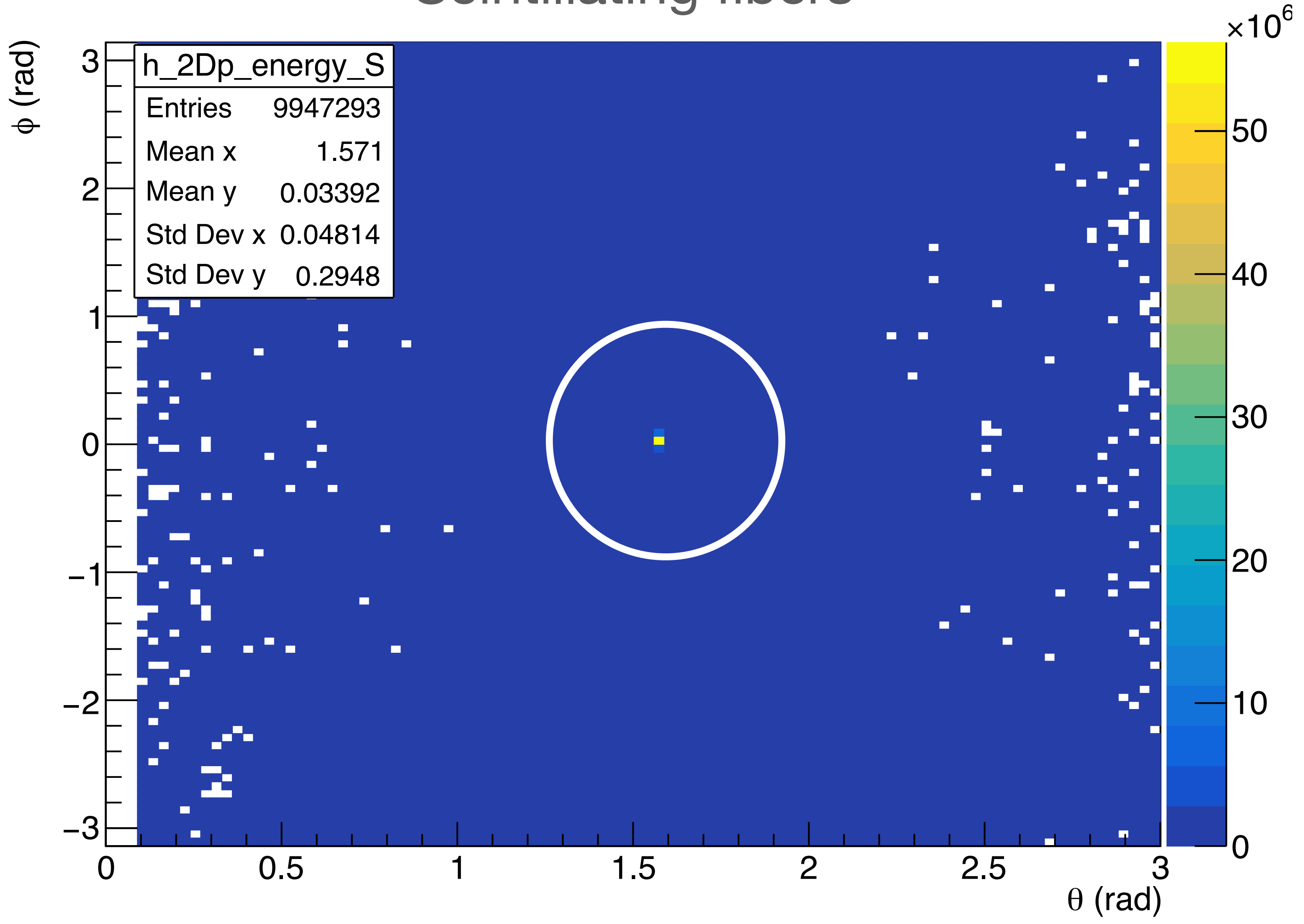
It's clear that the full geometry problem is confusing the NN, trying a simpler approach where electrons are produced in one direction, along x axis.

pencil like simulation

Cerenkov fibers

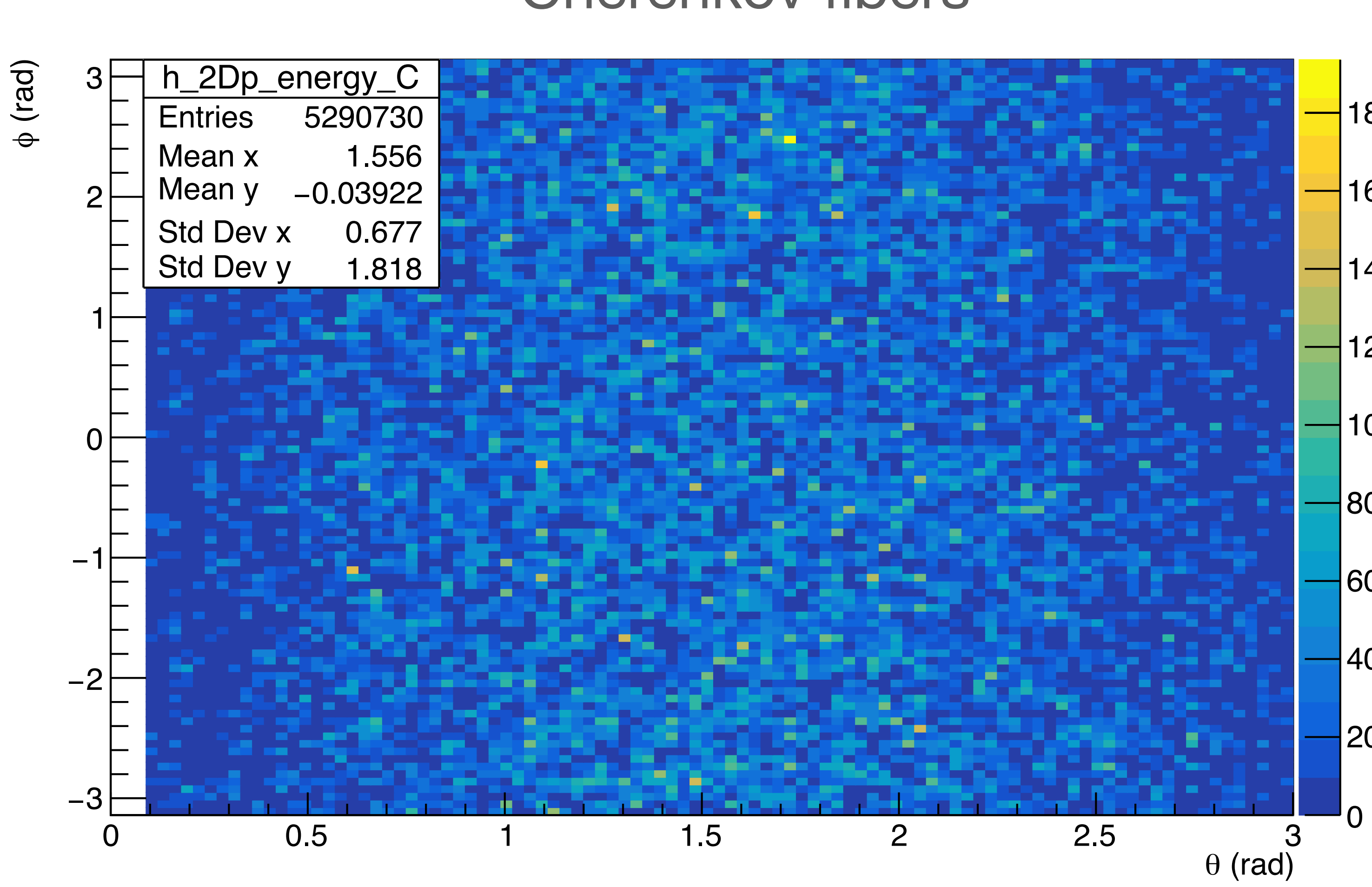


Scintillating fibers

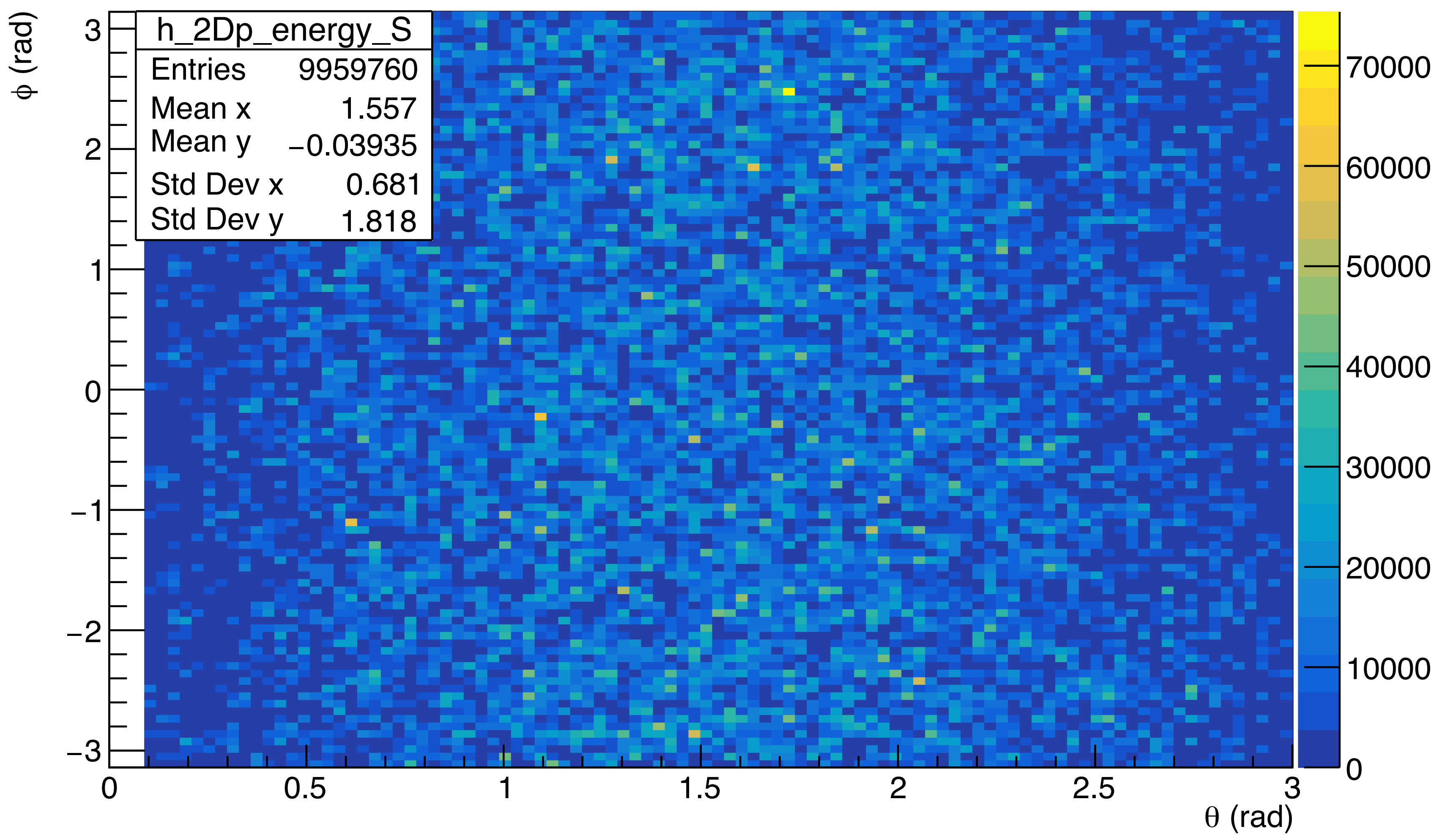


full geometry simulation

Cherenkov fibers

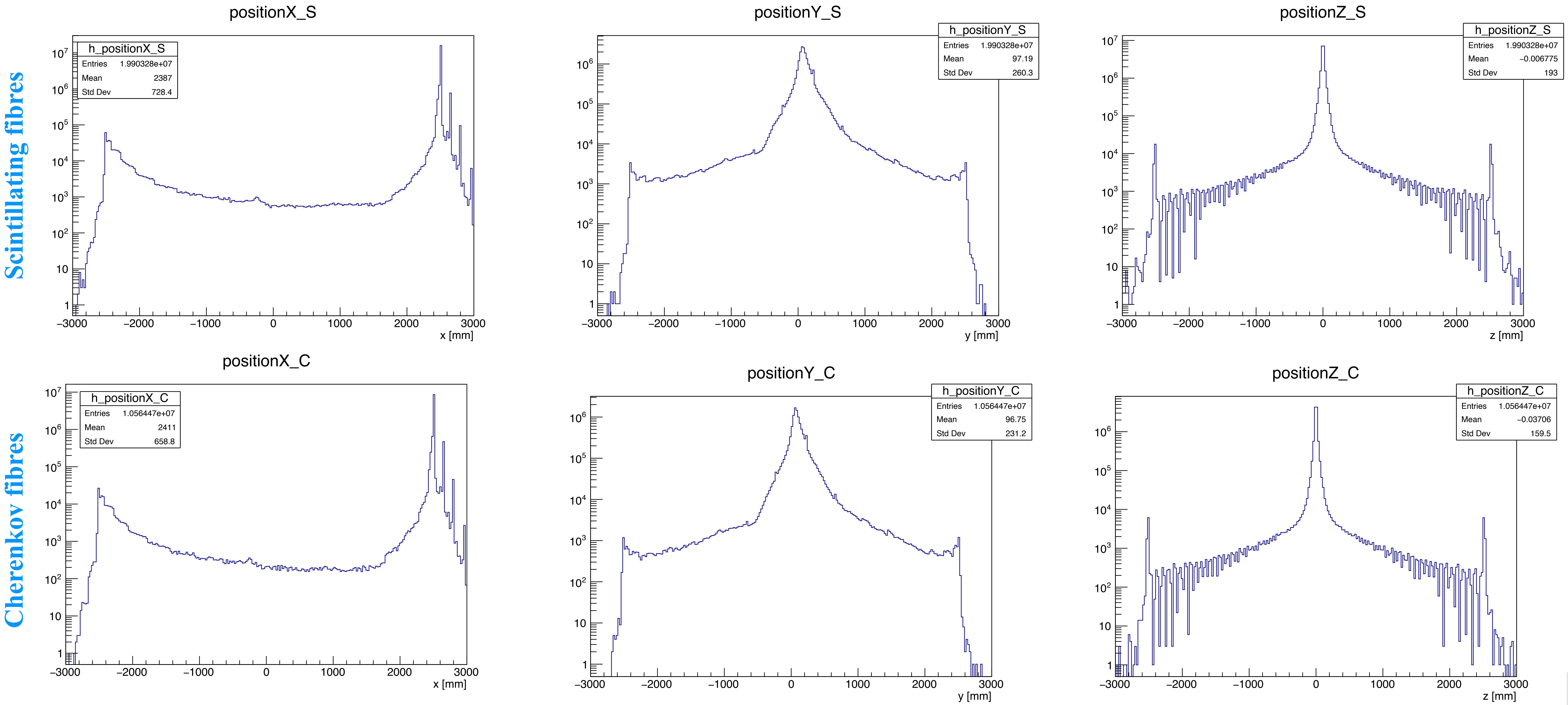


Scintillating fibers



Simplifying the problem: going to pencil like simulation

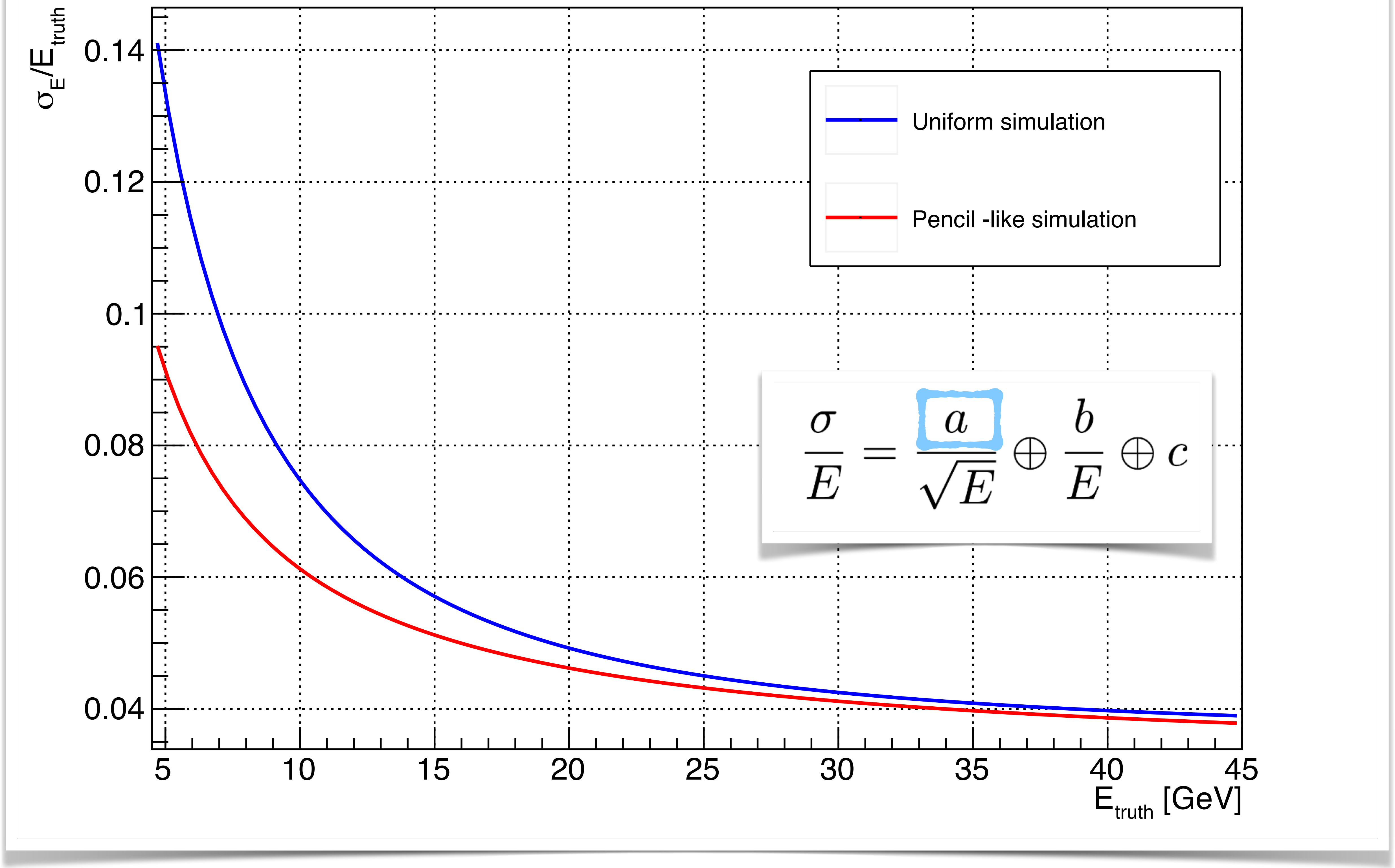
Kinematic checks



- hits are fiber endpoint coordinates, the geometry of the fiber (orientation in space) has an impact;
- tails present very far from the impinging point (shower residuals going around in the detector ?)
- photon radiation can broaden the peak distribution;
- fiber projections can be broader than real energy deposit distributions (broadening the cenral peak)

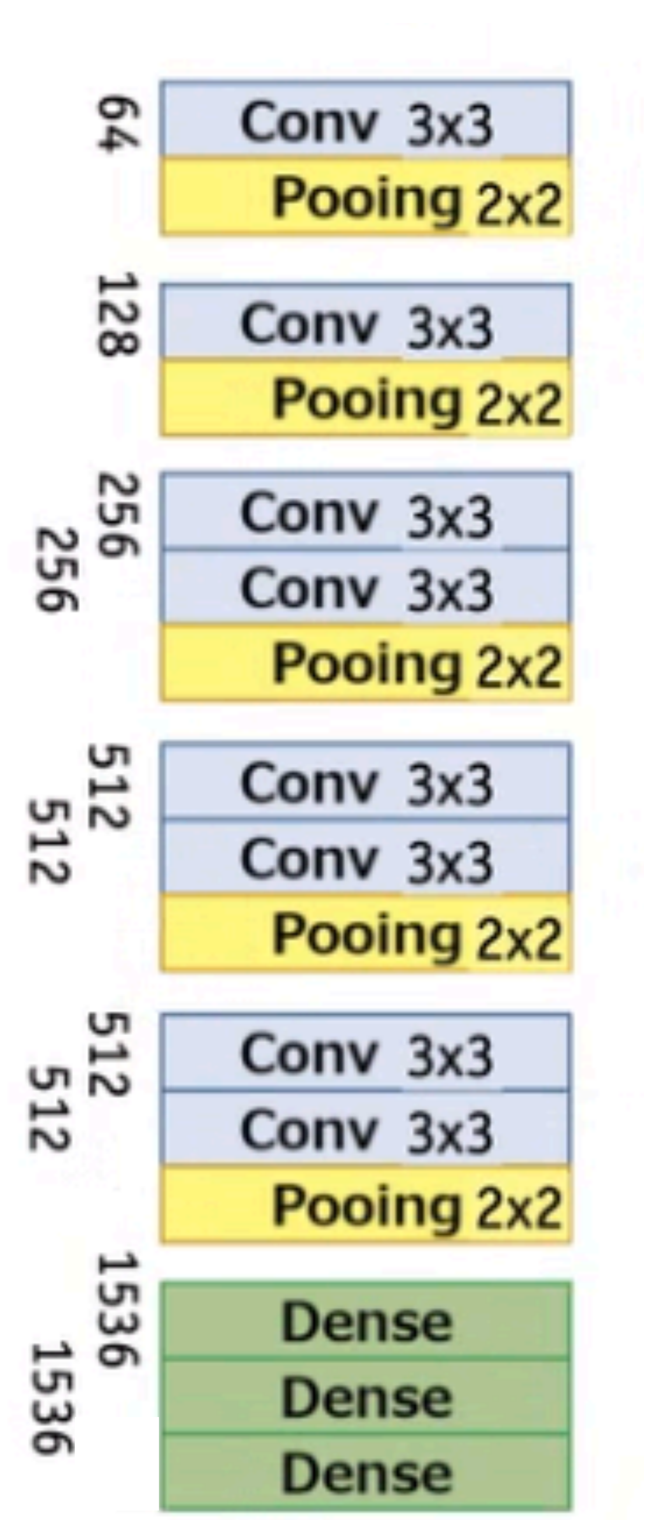
Preliminary results on electron energy resolution

Energy and position info used as inputs to CNN



VGG-like CNN architecture

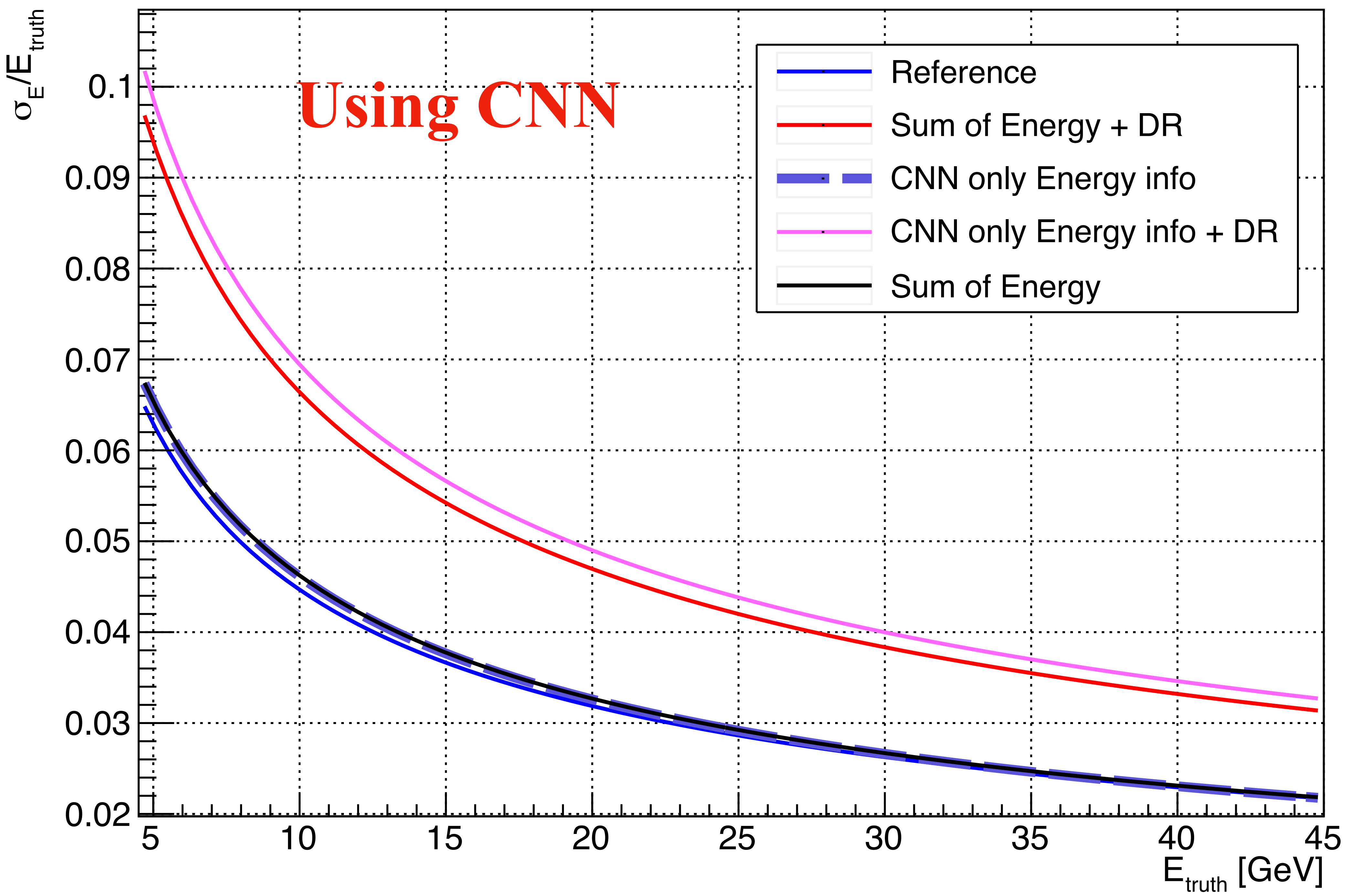
100x100xN
Input tensor



- Same model and set of inputs used for the CNN training, but different simulations—> improvements up to 30% at low E truth
- In the next slides we test the performance of the NN using only the energy information as input as sanity check
  - low energy behaviour is clearly affected by the geometry (we need to check different detector regions)

Preliminary results on electron energy resolution

- The plain sum of energy in the fibres (S&C) is compatible with the reference energy resolution
- We tested the possibility to apply a dR cut R = $\sqrt{d_z^2 + d_y^2}$
- A dR cut at 50 mm degrades the result also in the easy case, cut optimisation ongoing
- If we ONLY use energy information as input for the CNN, we are able to reproduce the plain sum of energy resolution, as a **cross-check**
- If we ONLY use the energy information as input for the CNN and we apply a dR cut at 50 mm, the resolution degrades
- Further tests ongoing switching off the magnetic field to better understand the kinematic and the geometry of the events **(radiation could play a role)**
- **Nest step: clustering in data pre-processing**



Conclusions

1. we were too much optimistic on the NN ability to solve problems for us :-)
2. we need to change approach, add NN on top of a classical reconstructions where known problems and features are solved (radiation, magnetic field bending, fiber geometry)
3. we need to develop a pre-reconstruction algorithm and use it as inputs to the NN together with row infos
4. this is time consuming, difficult to do it for all particles;
5. we need to re-think to the whole project (objectives, descoping)