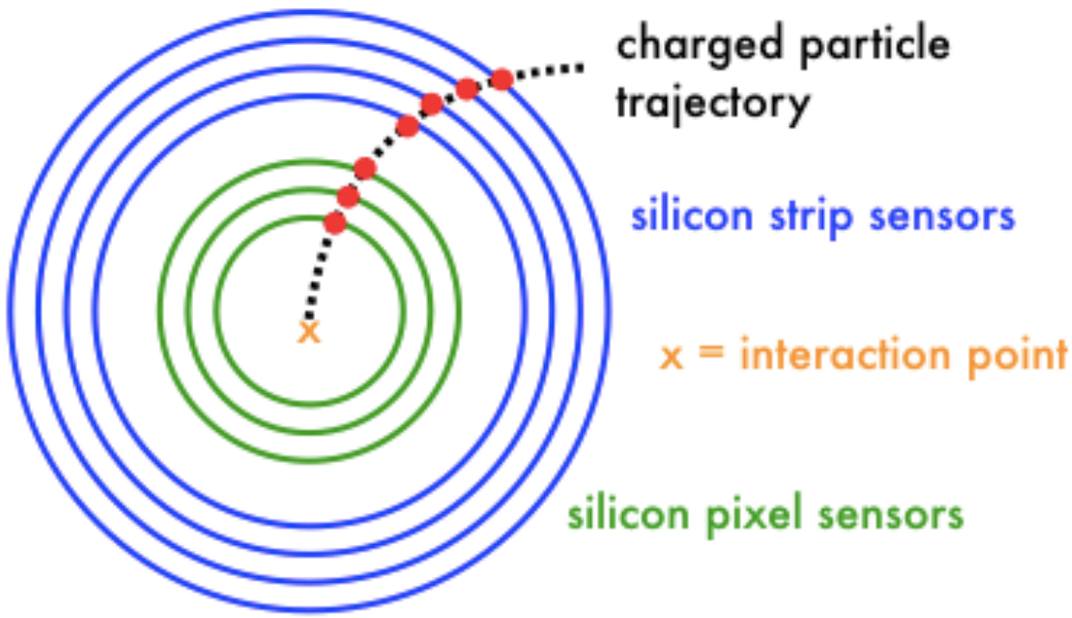
Graph Neural Networks for Reconstruction

Lindsey Gray et al.

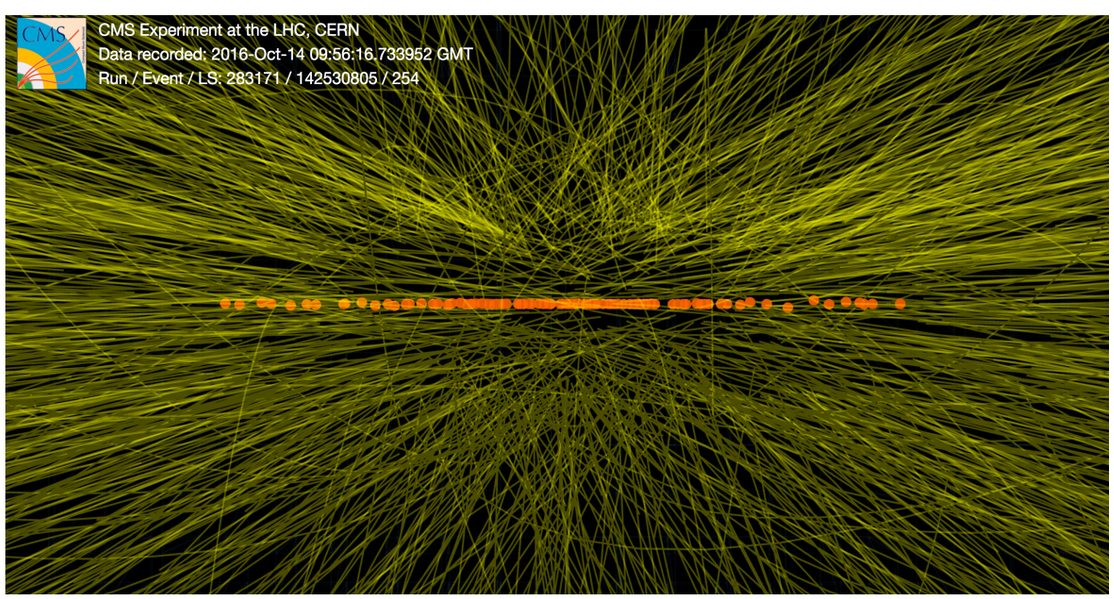
11 September 2019

HL-LHC: Two Big Problems

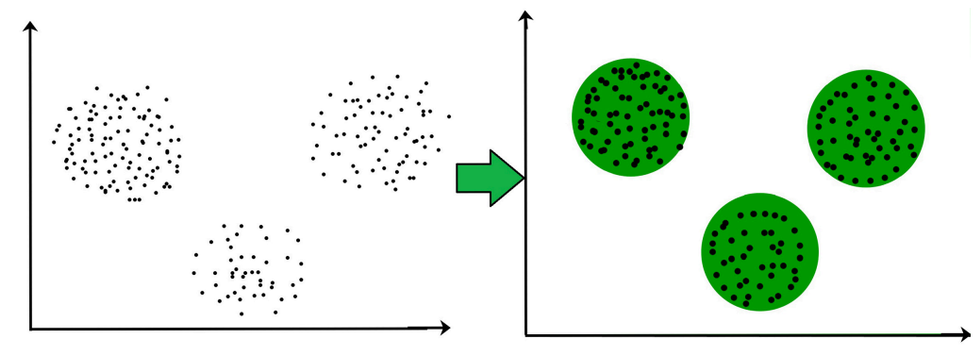
Tracking in 200 PU



1000s of tracks at once



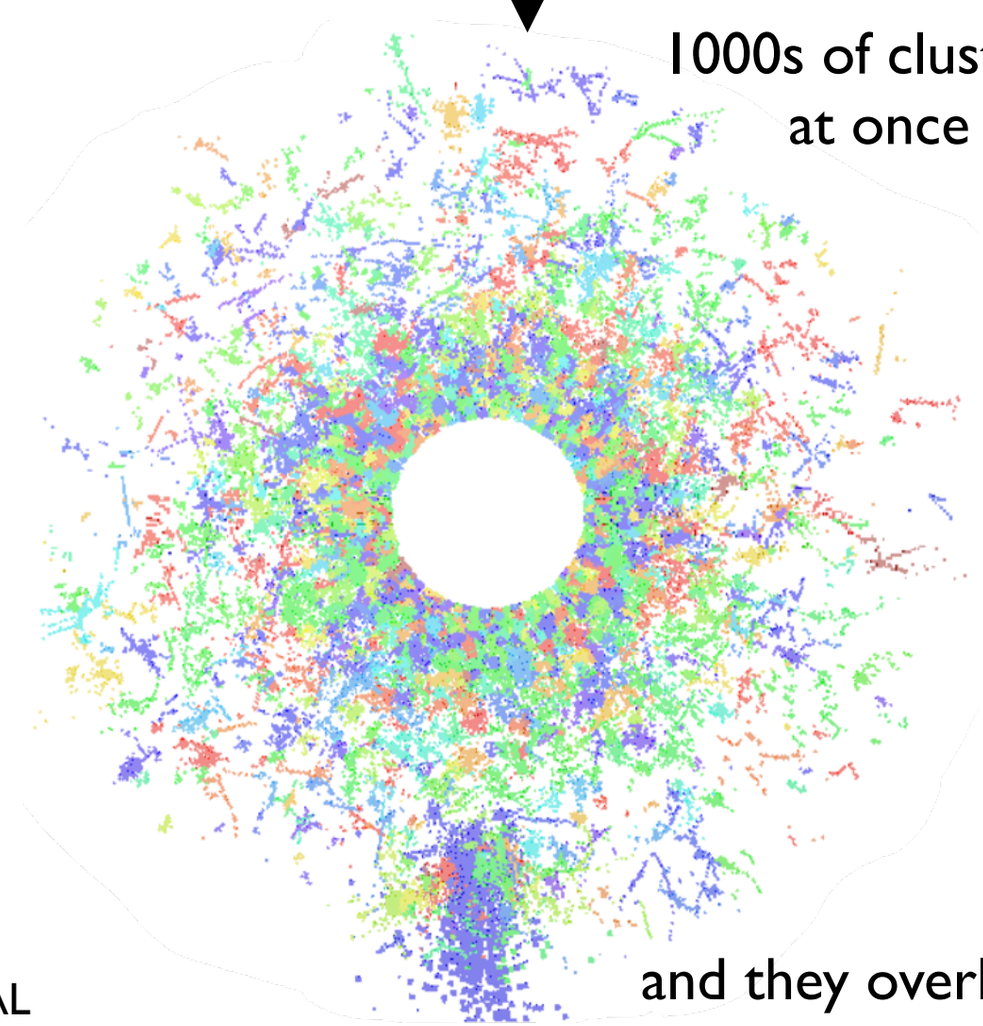
High-granularity calorimetry in 200PU



unlabelled hits

groups hits

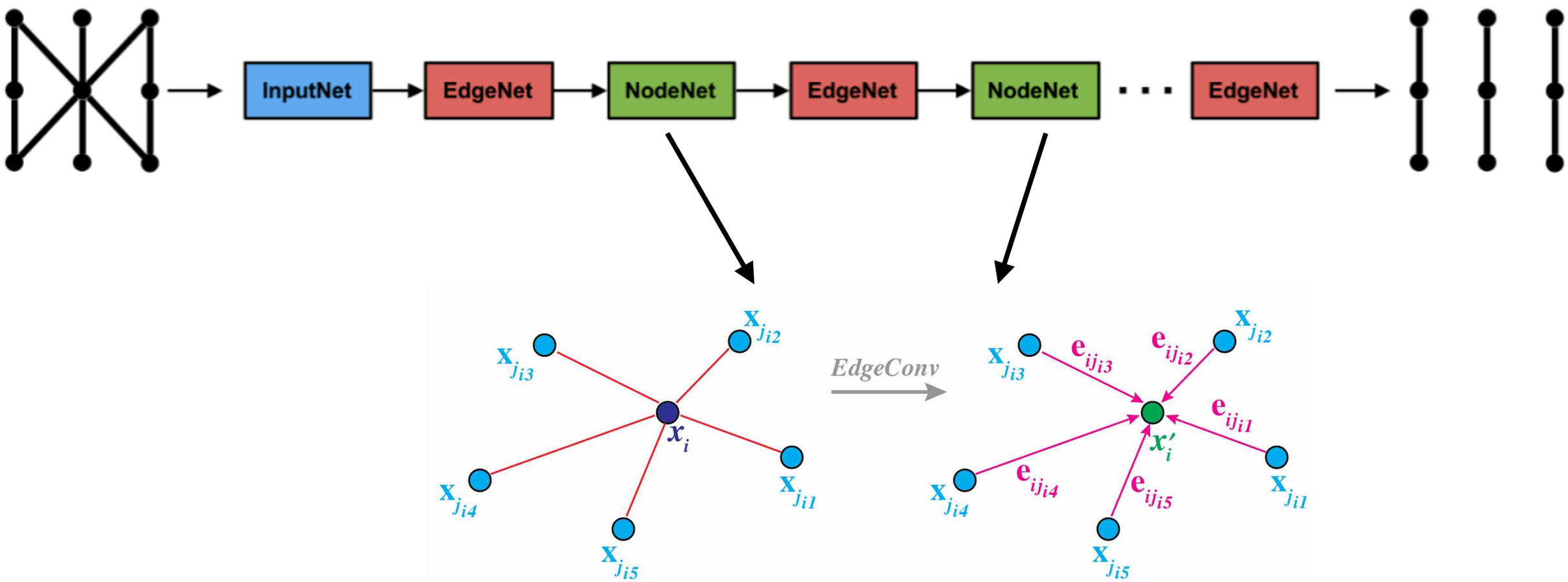
1000s of clusters at once



and they overlap :-)

A Path Forward with Hep.TrkX

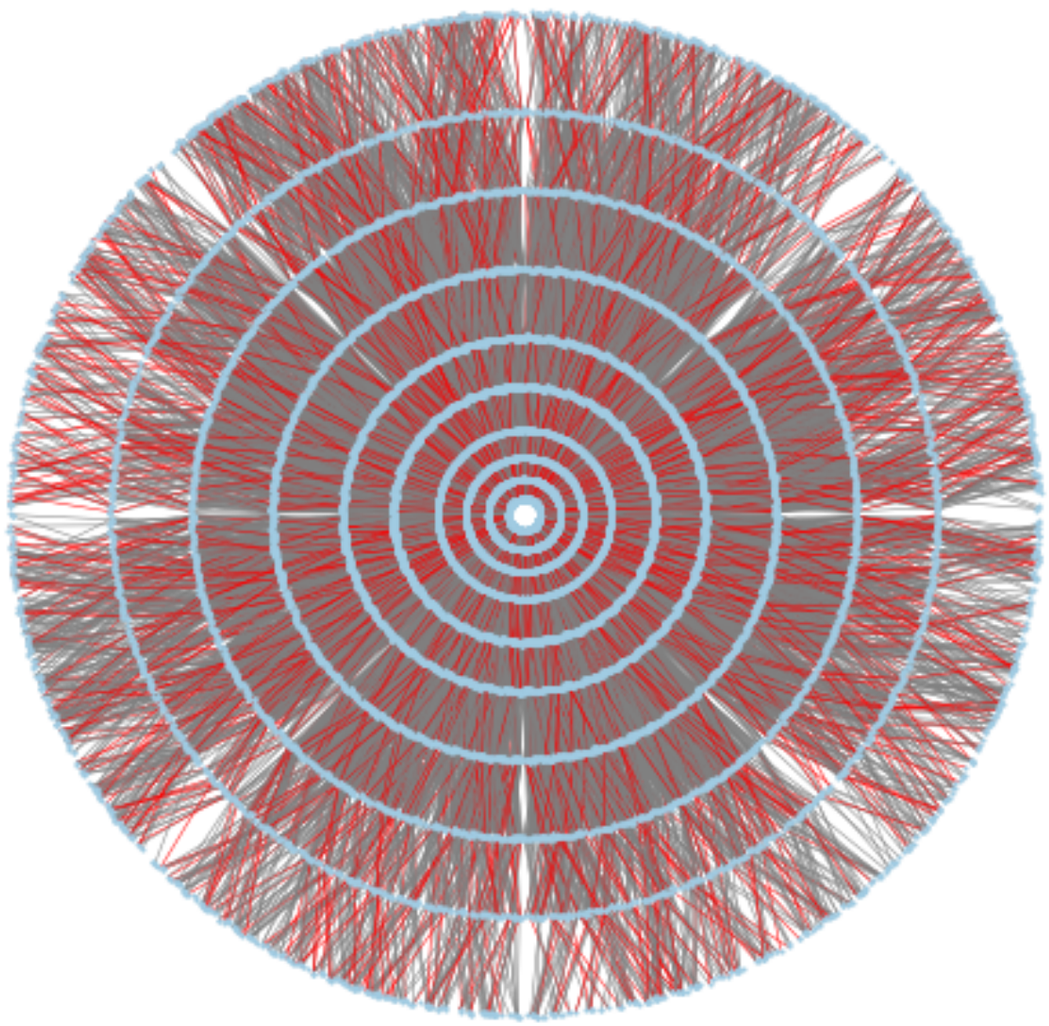
- Aim of project was to discover NN track finders
  - Very promising solution in graph neural networks
  - Particular flavor: graph convolutional edge classifiers

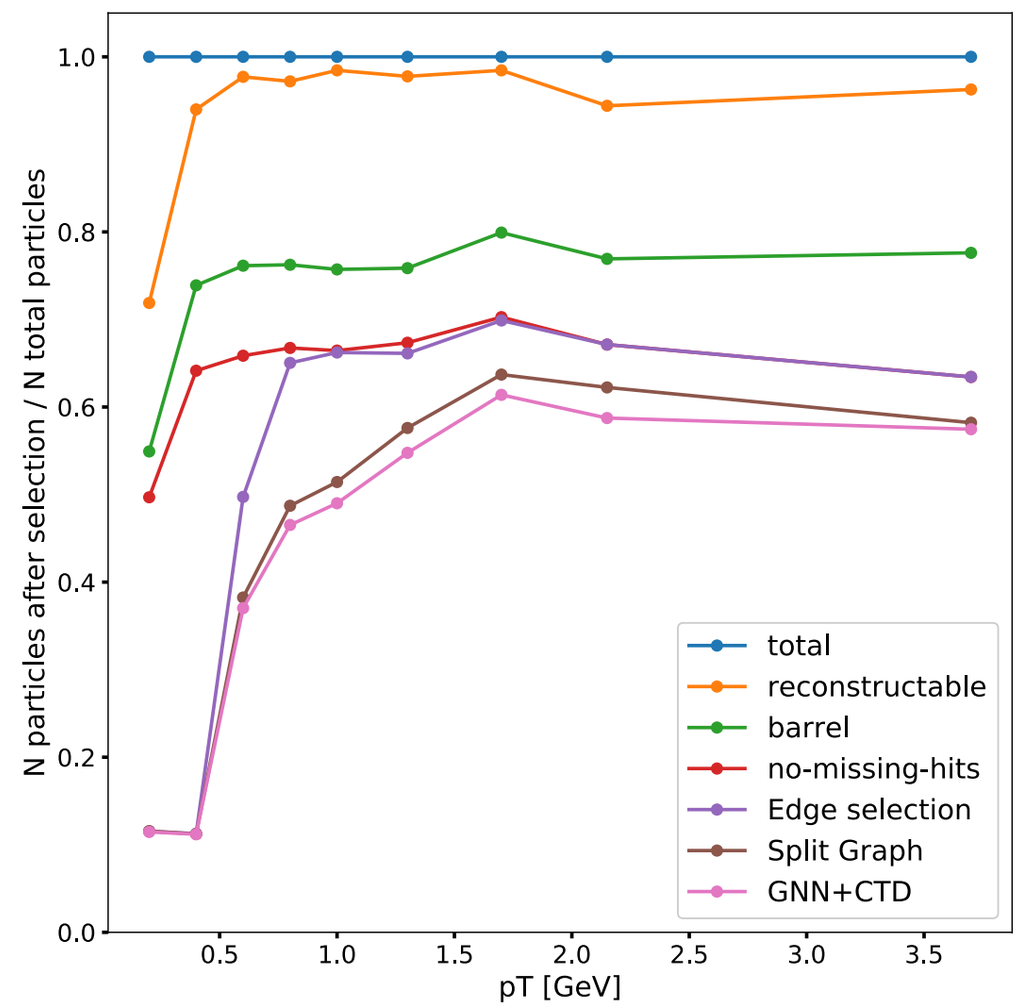


 https://arxiv.org/abs/1810.06111

First Performance and Promise

- Many selections applied to yield training set
  - Important: sectorization and no missing hits
  - These are "easy" tracks but this also early days for the these kinds of network in HEP
- Applying GNN, assembling tracks -> 97% efficient relative to preselection
  - Track-segment selection GNN executes significantly faster than Kalman filter

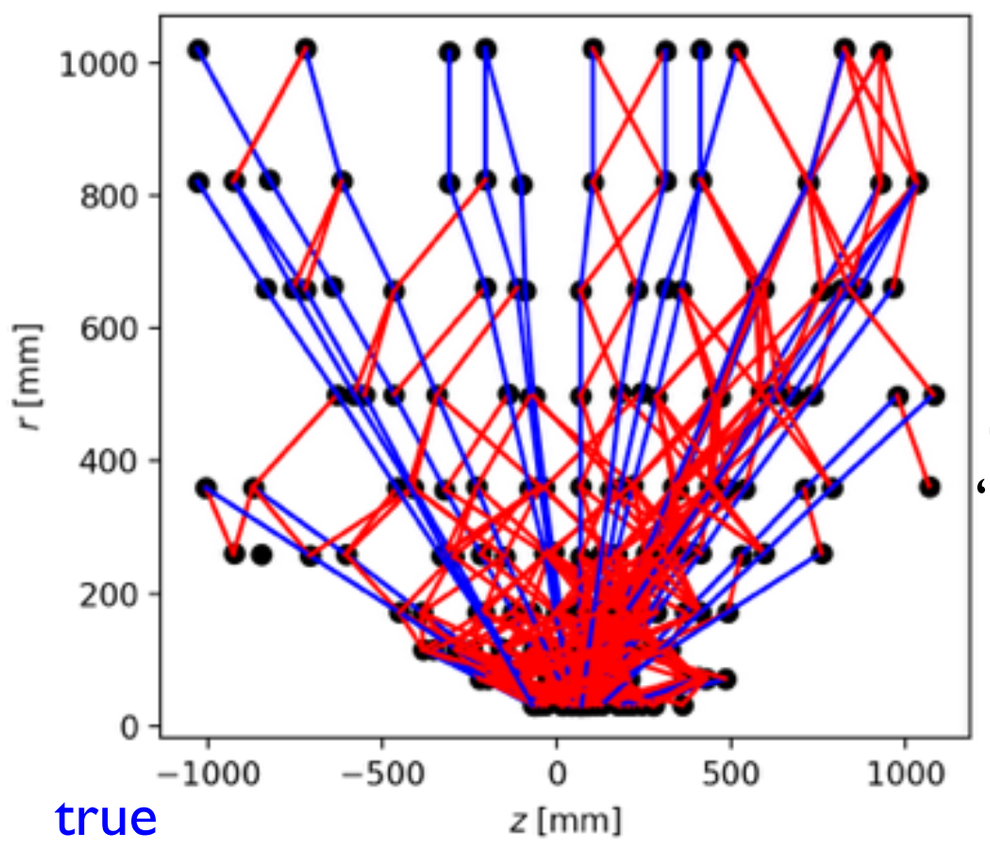




X. Ju @ CTD 2019

From Tracks to Clusters

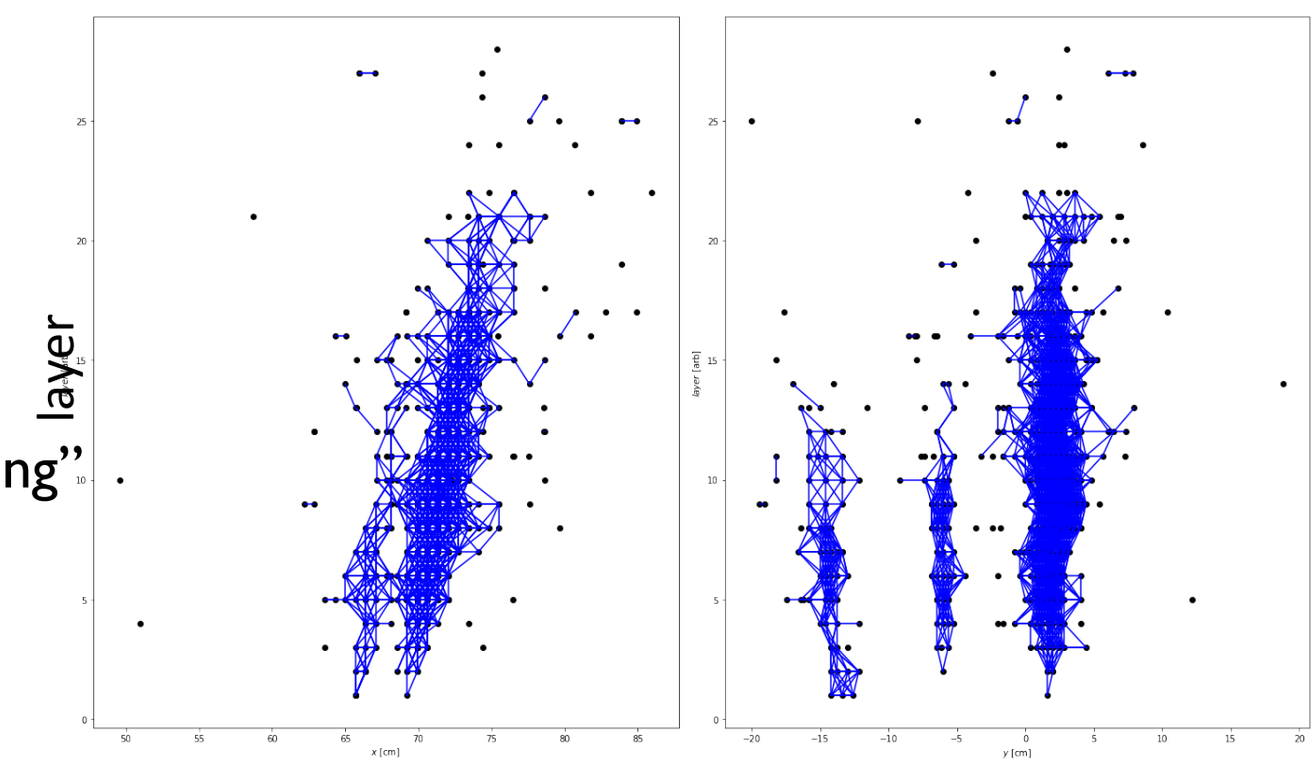
- Imagine a calorimeter cluster as a clump of edges
  - In terms of abstract of relationships between hits this is the same object as a track



true

false

"same thing"



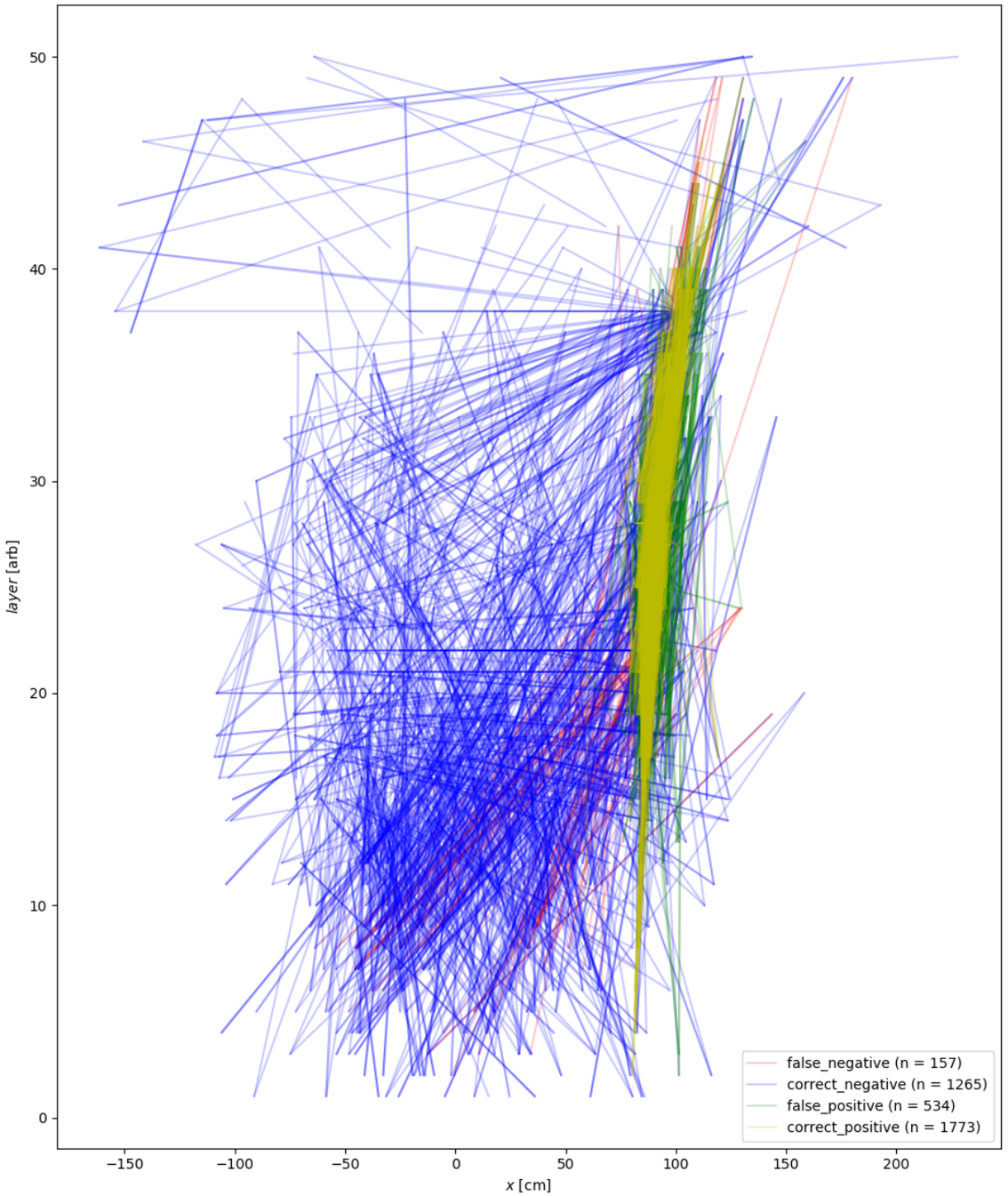
false edges not shown (too many)

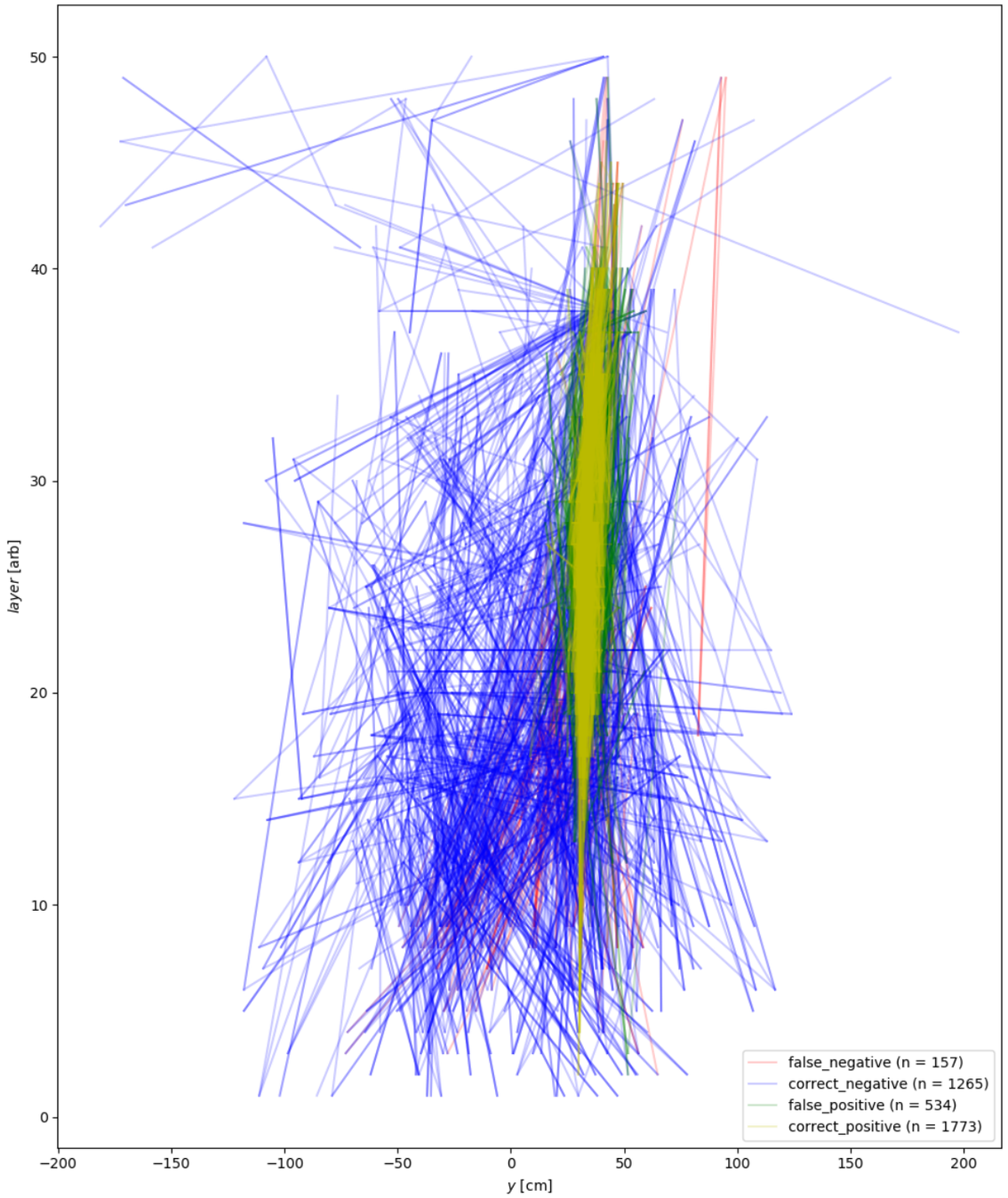
Pion Classification in HGCal

true negatives
true positives
false positives
false negatives

Pion shower identified in a sea of noise

Yes, this is an event display: I can give you statistics if you're on CMS



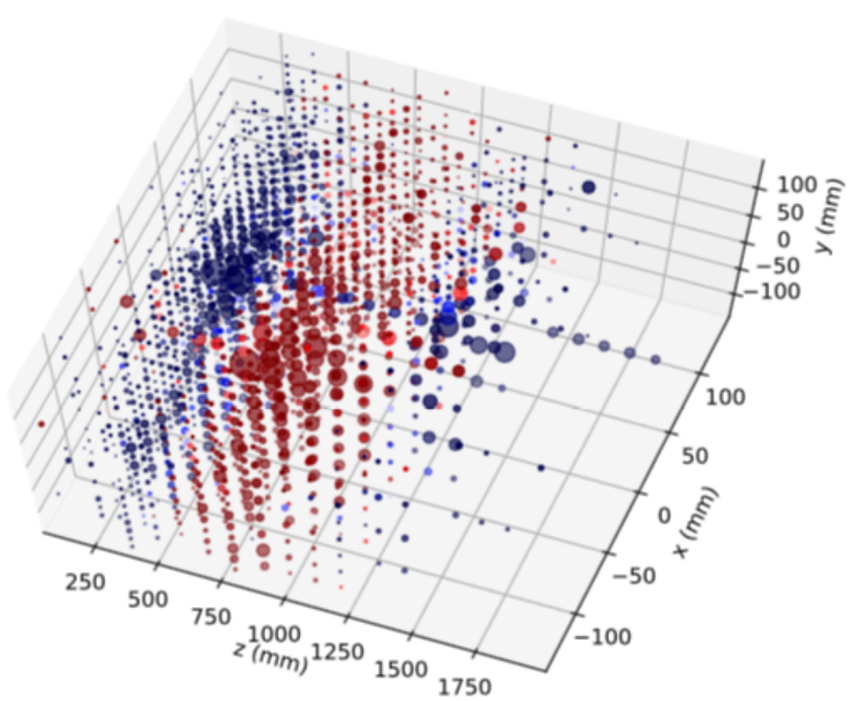


Lindsey Gray, FNAL

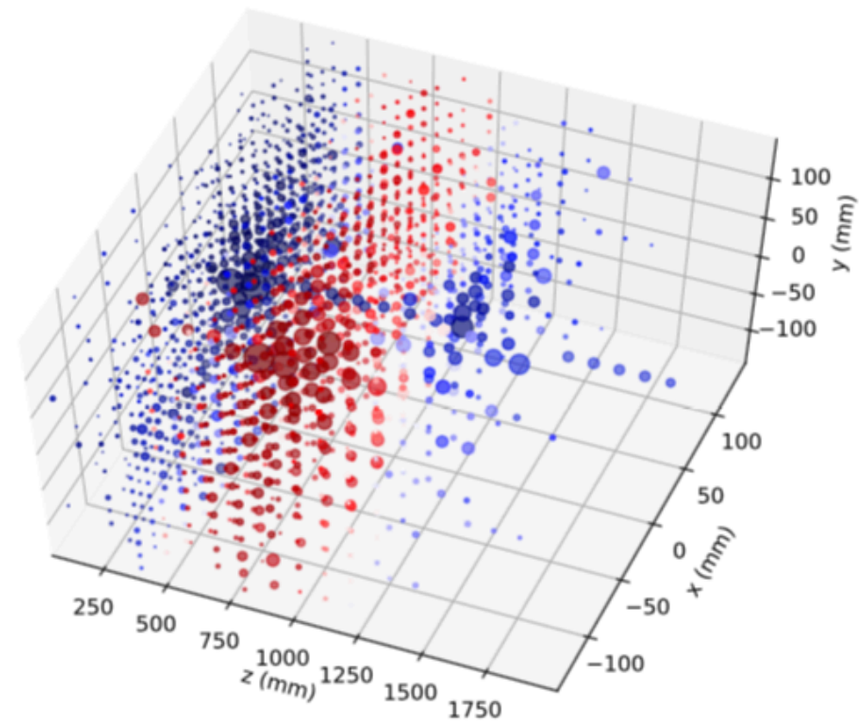
T. Klijnsma / S. An

Finally: Towards Actual Clustering

- Finding correct edges in a static graph is a well characterized learning problem
  - And you can use simple algorithms to keep your data processing quick!
- Clustering is better characterized as "finding the correct graph" rather than finding the correct subgraph
  - It's possible to learn the the latent space which best clusters the data
  - But! Right now, the number of clusters / categories needs to be known beforehand



(a) Truth



(b) Reconstructed

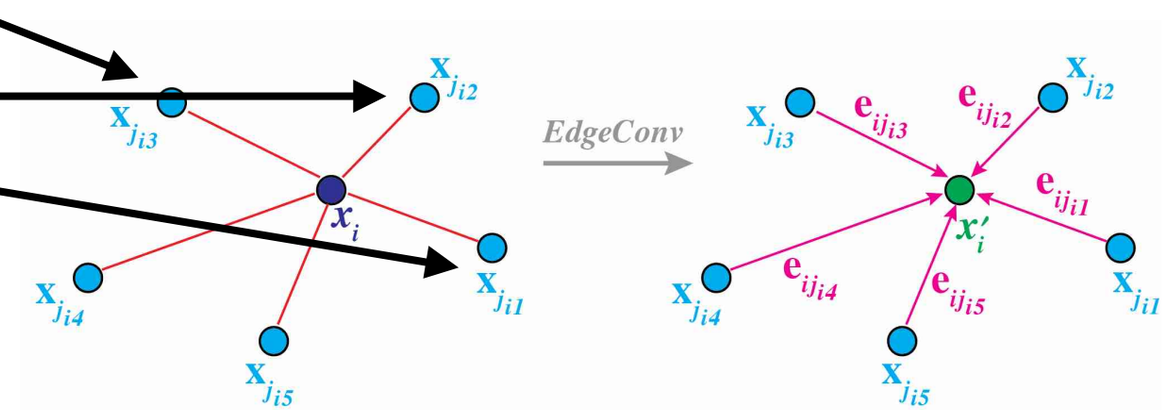
https://arxiv.org/abs/1902.07987
https://arxiv.org/abs/1801.07829

K. Pedro, LG, et al. working towards unbounded clustering

GNN Scaling and Memory Access

- Graph convolutional operations require huge local memories
  - 200 pile up (HL-LHC) ~300k nodes - 60 MB
  - Millions of edges, considering nearest neighbors only
  - Each edge accesses a fairly random address in the node features
  - The *whole* graph needs to be loaded in order to do the convolution
  - In zero pileup 500k edges in tau reconstruction prototype -> 13 GB gradient matrix
- Scattered operations are not great for modern processors
  - How easy/hard is this for FPGAs? How fast can we go here?
  - Small networks shown in hls4ml, but what sort of scale up is effective?

[[feature 1, feature 2, …],
[feature 1, feature 2, …],
[feature 1, feature 2, …],
[feature 1, feature 2, …],
…]



Let's not even talk about dynamic graphs…

Concluding Remarks

- Graph neural networks are promising for particle physics reconstruction and tackling the problems of the HL-LHC casted as static graph segmentation
  - Good efficiency in predicting edges for tracks
  - Similar concepts are immediately applicable to calorimeter clustering and are already promising
  - Sparse, scattered operations at these scales sound challenging (at least to me)
- As these techniques are proven, we should understand how to bring them to the current LHC
  - Muon triggering, pixel seed finding, particle flow
    - All of these can be thought of in terms of graph segmentation problems!

Extras

Collaborators

- Hep.TrkX & Exa.TrkX
- N. Tran, K. Pedro, J. Duarte, J. Hirschauer, T. Klijnsma, B. Kreis
- S. An, S. Gleyzer
- J. Kieseler, M. Pierini, et al.
- P. Harris, et al.

Going Further: Simultaneous Particle ID

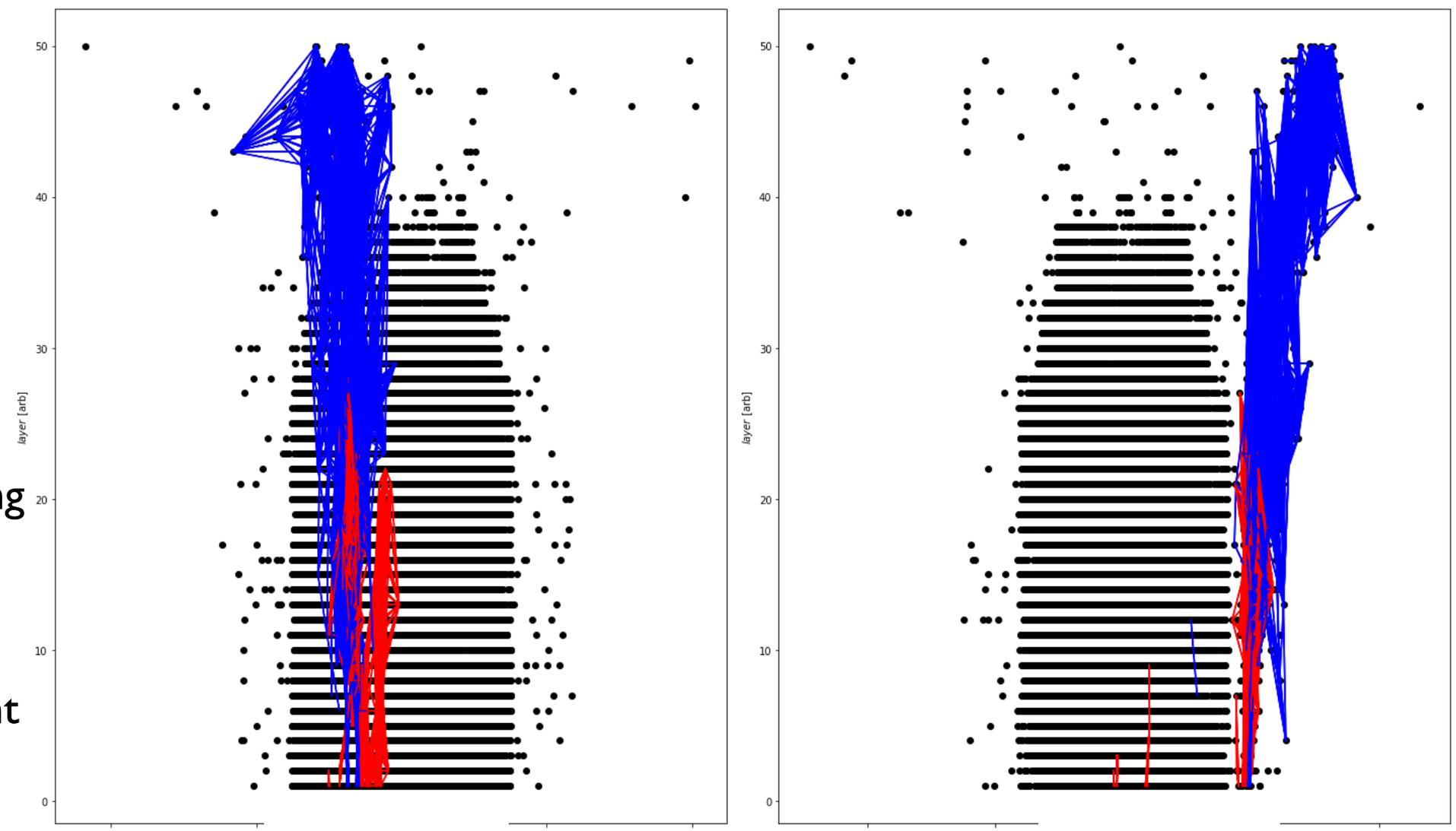
- Instead of simply true / false - classify energy flow type
  - Harder problem, but a classic machine learning application

tau -> pi + pi0 truth

hadronic
electromagnetic
false edges not shown

Networks currently being trained.

Initial tests indicated ~90% correct assignment efficiency on ~3000 events (kNN, k=26)



x projection

y projection

Lindsey Gray, FNAL