Application of NN to tracking and particle ID

I. Yeletskikh

Motivations

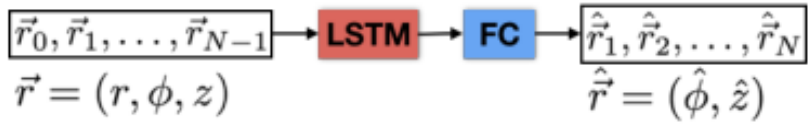
- PreFit analysis: using hit patterns to distinguish between different types of events…
  - technically similar to image recognition…
  - can be used for, e.g., triggering particular events…

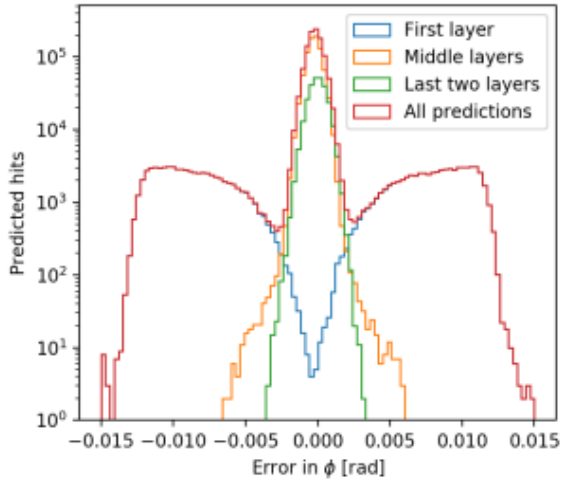
- GNN and RNN for track finding and track reconstruction
  - Recurrent Neural Networks allow prediction of the next hit (with uncertainties) of track given set of previous hits, give score to track candidates. May be more efficient compared to, e.g., Kalman filter.
  - Graph Neural Networks can preselect hits that are likely to belong to same track. Efficient in particular in case of high multiplicity events.

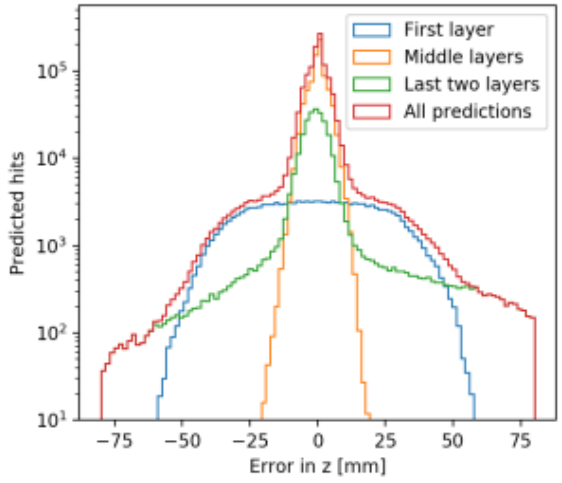
- PostFit: Particle ID using high-level variables NN
  - Using parameters of the reconstructed (by other algorithms) track (momentum, dE/dx, track geometry) to perform identification... Make ID possible in absence of dedicated detector facilities;

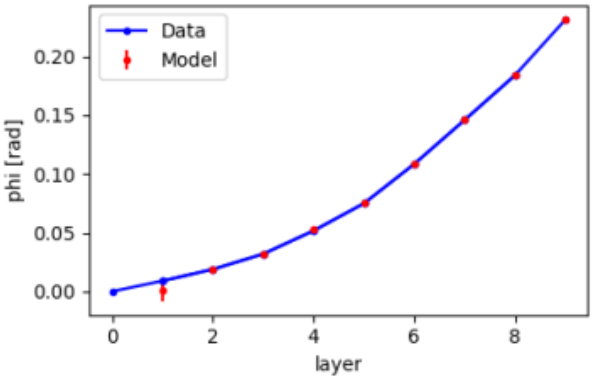
Applications to tracking at LHC and HL-LHC

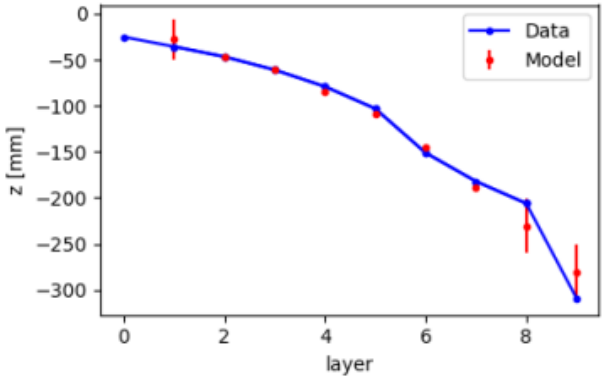
- RNN applications to track reconstruction.
  RNN is used to predict next track hit given a set of previous hits











Uncertainty of the prediction for a hit coordinated in subsequent layers of the detector
RNN is able to predict coordinate and it's uncertainty to score different track candidates.

Few RNN may be trained on different types of tracks which allows to compare score of the same track assumed to be muon, pion, kaon etc.

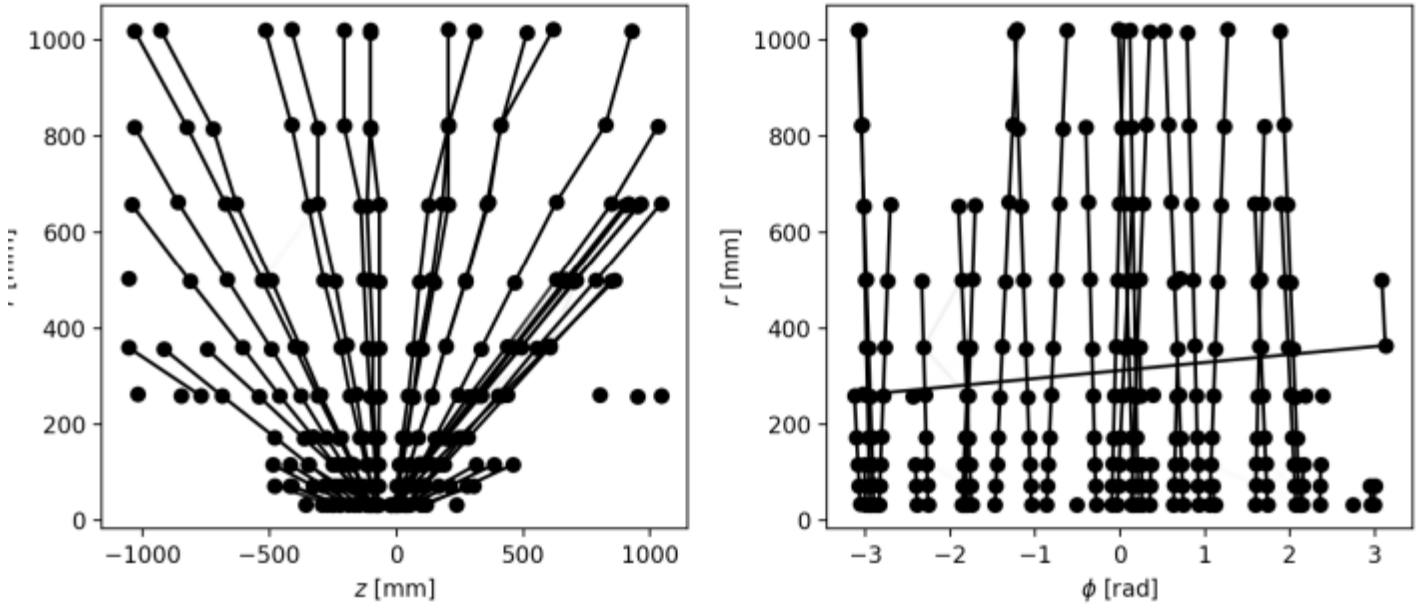
Applications to tracking at LHC and HL-LHC

- GNN applications to track building.

  In case of higher multiplicity building tracks may become an untrivial task

  GNN provide connections between hits that are likely or less likely belong to the same track segment with a modest CPU consumption

  Fake hits of hit holes may be accounted for. Prelimilnaruy studies show high efficiency of track finding and building.



Status and plans

- 10K muon and hadron tracks generated with Pgun with hits coordinated in tracker. Converted into plain ntuple format;
- Straightforward application of NN (pykeras) to hit coordinates as input is probably too ambitious… One of the problem is variable number of inputs for different tracks… Another problem is combinations of different track types and contaminations in real data… PreFit analysis looks not too promising..

- Currently try to apply RNN to the hits to build track candidates and give them scores. In particular, mixed events are used with several tracks hits put together…

- In case events with reconstructed tracks are available (where parameters like momentum, dE/dx, <ToT> etc. are availeble) it would be a simple task to apply multiclassification NN to combine this information and output scores for given track to be consistent with a given particle type.