Machine Learning for Charged Particle Tracking

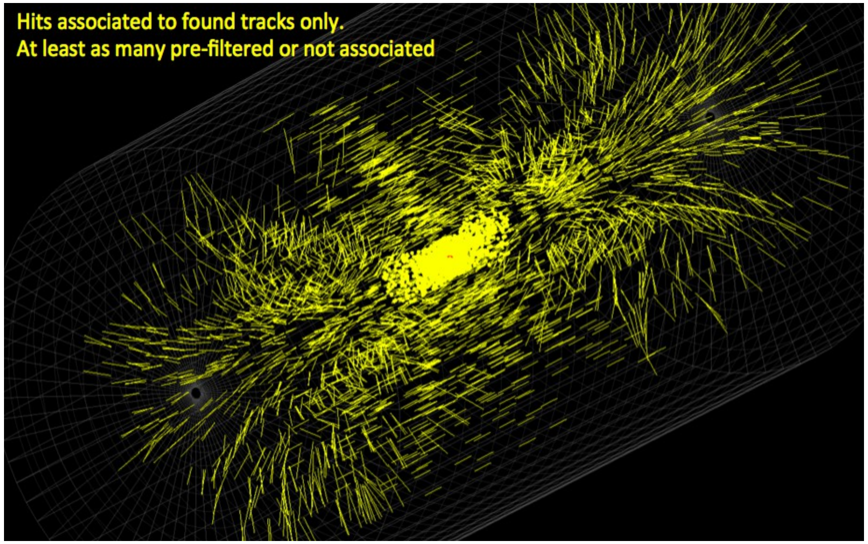
AIDA-2020, Topical Workshop on the Future of Tracking

St Anne's College, 1-2 April 2019

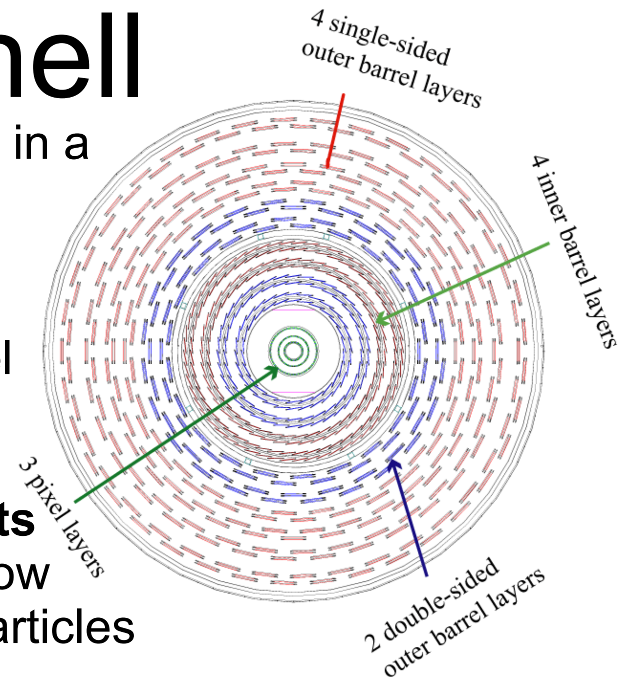
Overview

- Introduce the challenge of charged particle trajectory.
- Collection of applications of machine learning for tracking-like problems.
- Covering some of the technical details, underlying the challenges and prospects.
- Many thanks to LHCb : **P. Seyfert**, ATLAS : **D. Rousseau, R. Jansky, A. Salzburger, S. Amrouche,** ALICE : **R. Shahoyan**, CMS : **V. Innocente, F. Pantaleo** Neutrino : **R. Sulej, A Farbin, HEP.TrkX group** for material and input.
- More at CTD19 https://indico.cern.ch/event/742793/

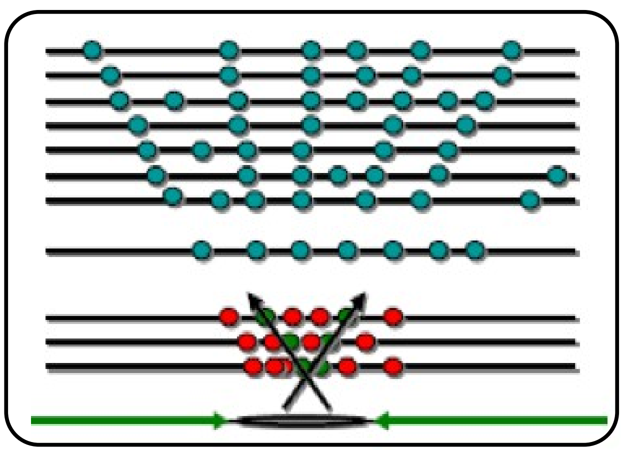
Tracking in a Nutshell



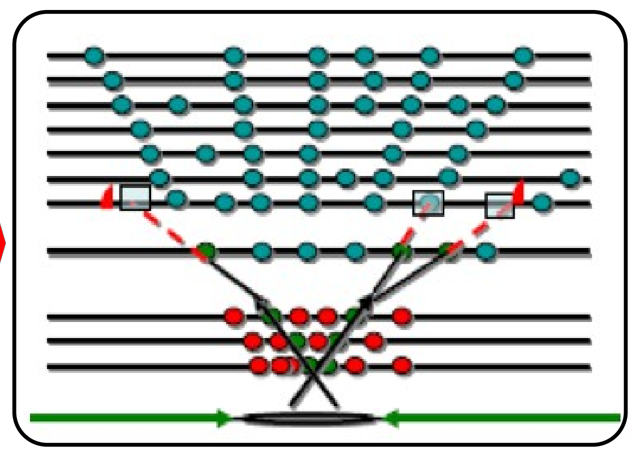
- Particle trajectory bended in a solenoidal magnetic field
- Curvature is a proxy to momentum
- Particle ionize silicon pixel and strip throughout several concentric layers
- **Thousands of sparse hits**
- Lots of hit pollution from low momentum, secondary particles

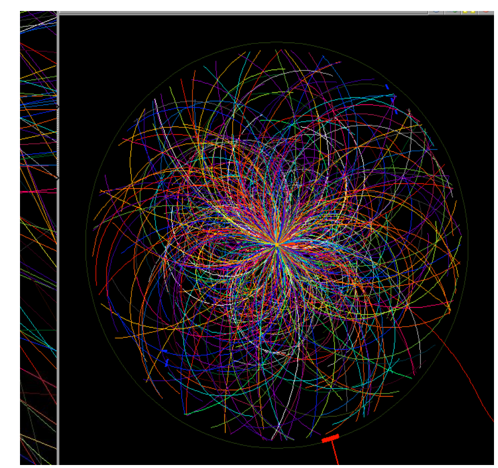


Seeding



Kalman Filter

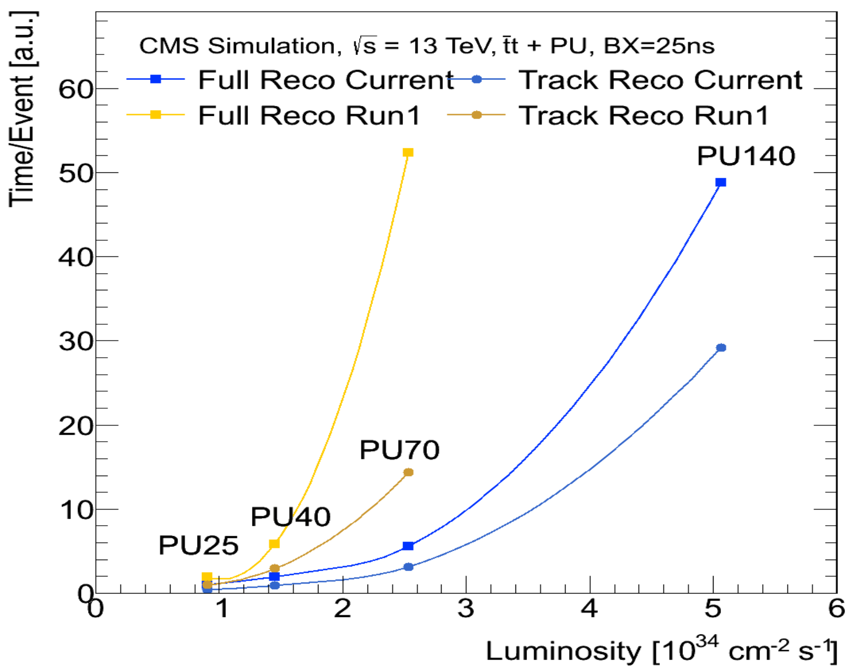


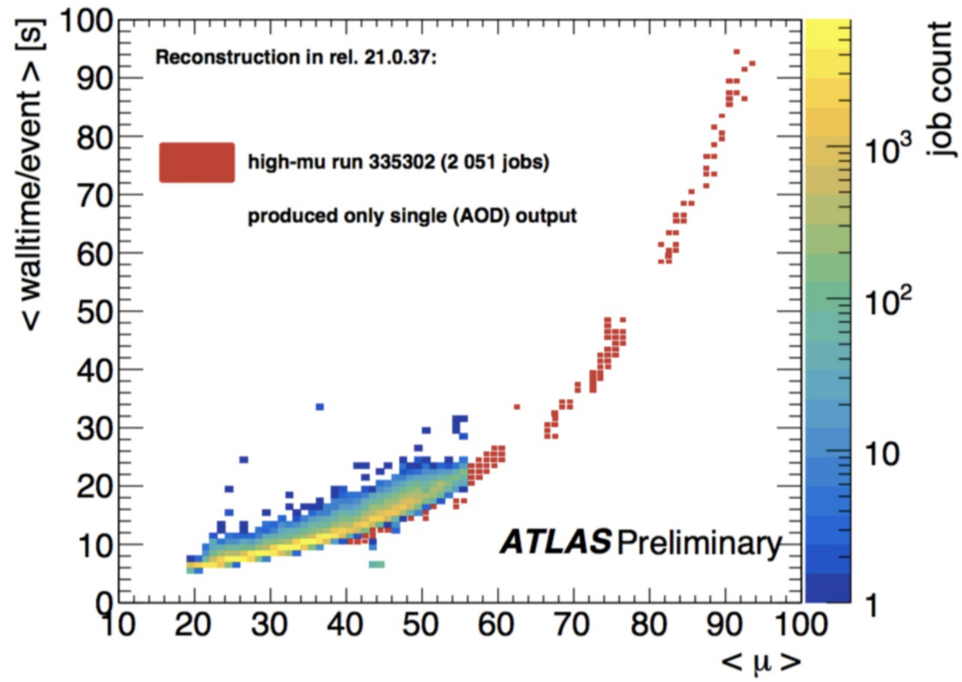


- **Explosion in hit combinatorics** in both seeding and stepping pattern recognition
- **Highly computing consuming task** in extracting physics content from LHC data

Cost of Tracking

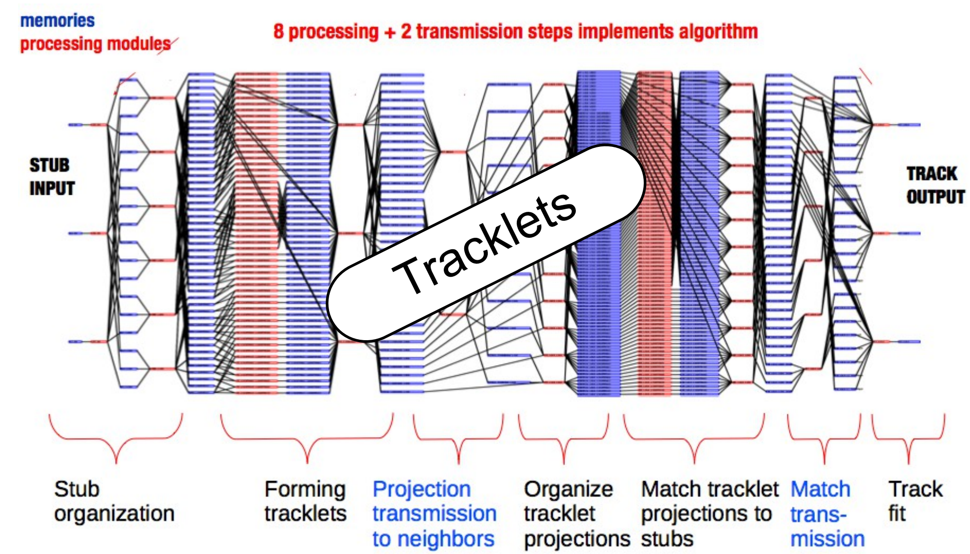
- Charged particle track reconstruction is one of the **most CPU consuming task** in event reconstruction
- **Optimizations (to fit in computational budgets) mostly saturated**
- Large fraction of CPU required in the HLT. **Cannot perform tracking inclusively at CMS and ATLAS.**

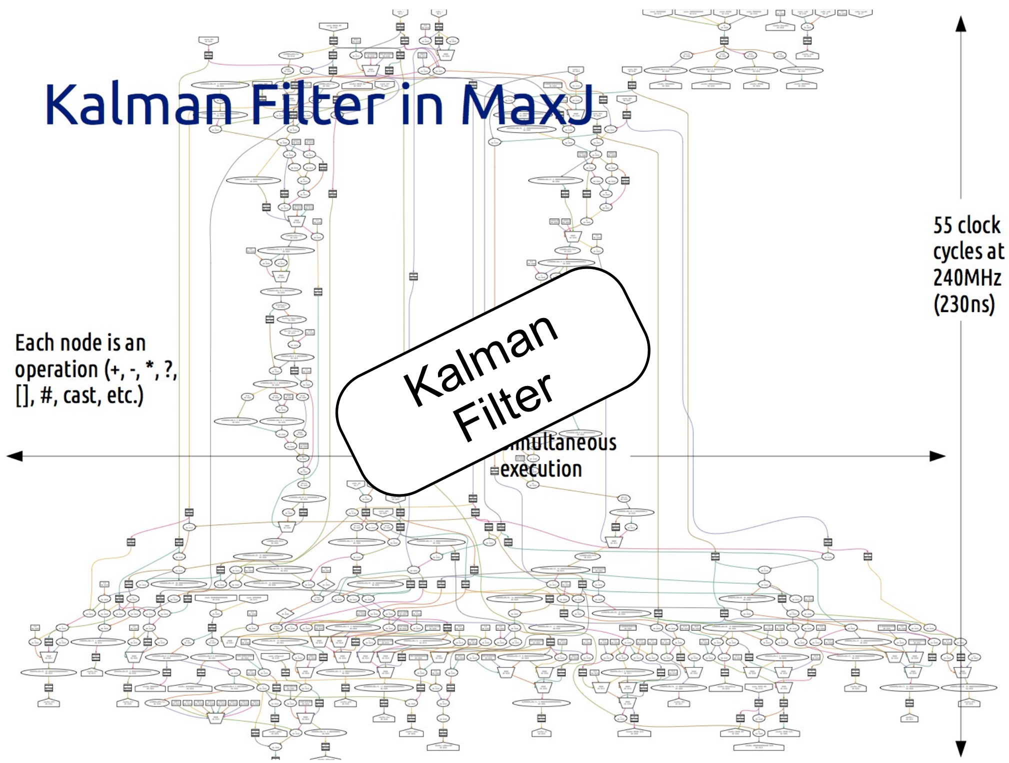


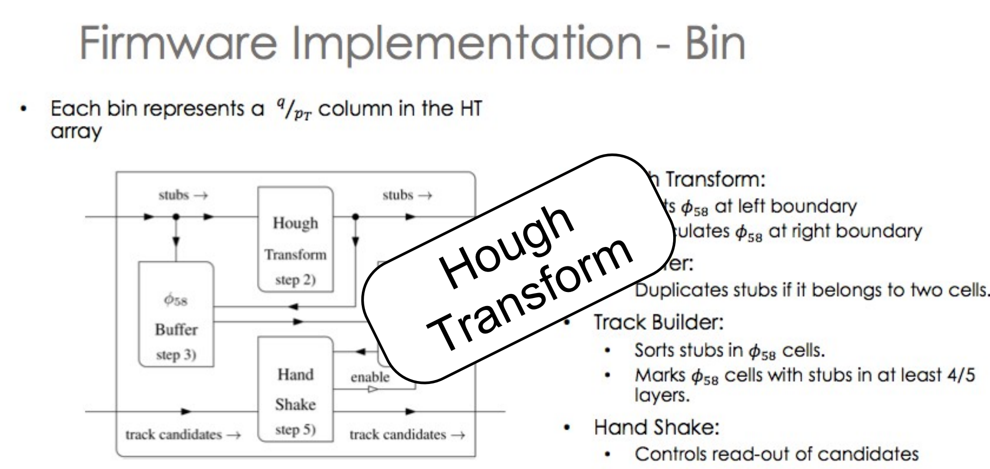


Fast Hardware Tracking

- Track trigger implementation for Trigger upgrades development on-going
- Several approaches investigated
- **Dedicated hardware is the ke**y to fast computation.
- **Not applicable for offline** processing unless by adopting heterogeneous hardware.







See https://ctdwit2017.lal.in2p3.fr/

Outline

I. Challenges and similarities with machine learning

II. Applications of machine learning for tracking

Part I

Similarities and Challenges

- Particle tracking is an active field in data science
- Making a track is called pattern recognition
- Tracking data is much sparser than regular images
- Tracking device may have up to 10M of channels
- Underlying complex geometry of sensors
- Unstable detector geometry ; alignment
- Not the regular type of sequences
- Defining an adequate cost function
- A solution must be performant during inference

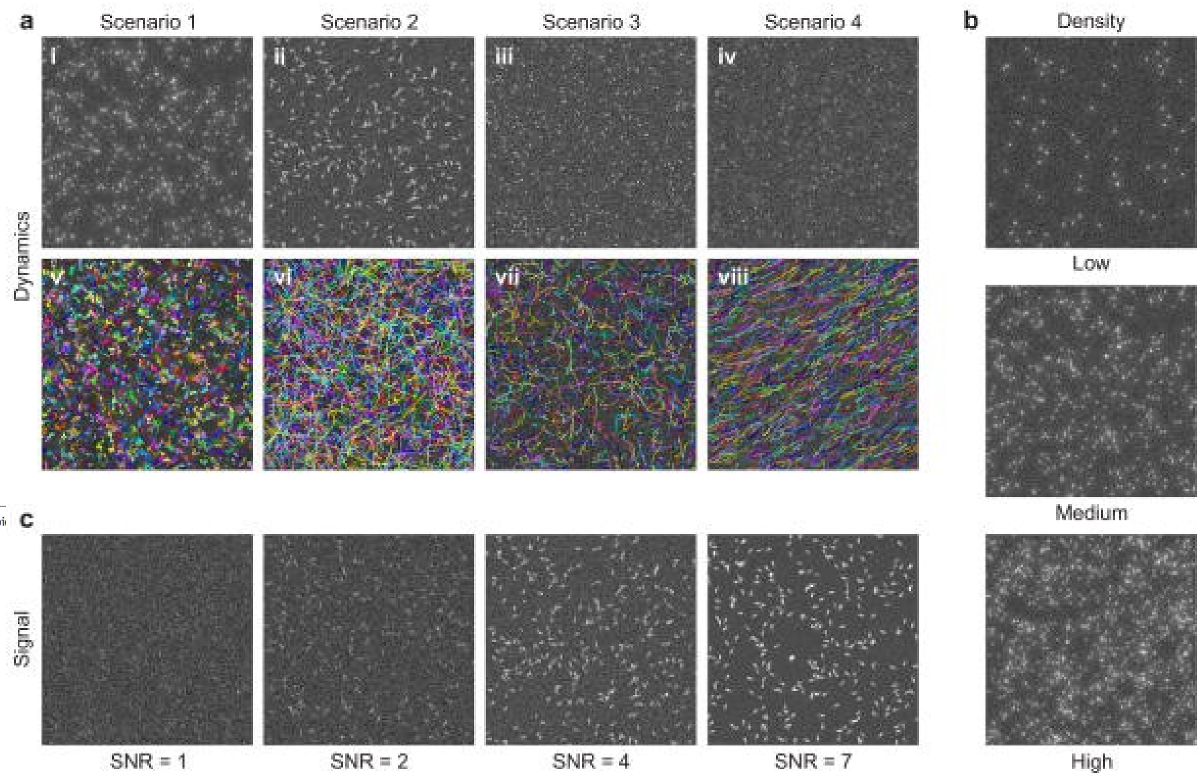
Particle Tracking in Biology

https://www.ncbi.nlm.nih.gov/pubmed/24441936

Table 1 | Participating teams and tracking methods

		Detection			Linking				
Method	**Authors**	**Prefilter**	**Approaches**	**Remarks**	**Principle**	**Approaches**	**Remarks**	**Dim.**	**Refs.**
1	I.F. Sbalzarini Y. Gong J. Cardinale	–	M, C	Iterative intensity-weighted centroid calculation	Combinatorial optimization	MF, MT, GC	Greedy hill-climbing optimization with topological constraints	2D & 3D	32
2	C. Carthel S. Coraluppi	Disk	M, T	Adaptive local-maxima selection	Multiple hypothesis tracking	MF, MT, MM	Motion models are user specified (near-constant position and/or velocity)	2D & 3D	33,34
3	N. Chenouard F. de Chaumont J.-C. Olivo-Marin	Wavelets	M, T	Maxima after thresholding two-scale wavelet products	Multiple hypothesis tracking	MF, MT, MM, GC	Motion models are user specified (near-constant position and/or velocity)	2D & 3D	35–37
4	M. Winter A.R. Cohen	Gaussian, median and morphology	M, T, C	Adaptive Otsu thresholding	Multitemporal association tracking	MF, MT, GC	Post-tracking refinement of detections	2D & 3D	38,39
5	W.J. Godinez K. Rohr	Laplacian of Gaussian or Gaussian fitting	M, T, F, C	Either thresholding + centroid or maxima + Gaussian fitting	Kalman filtering + probabilistic data association	MF, MM	Interacting multiple models using	2D & 3D	29,40
6	Y. Kalaidzidis	Windowed floating mean background subtraction	T, F	Lorentzian function fitting to structures above noise level	Dynamic programming				
7	L. Liang J. Duncan H. Shen Y. Xu	Laplacian of Gaussian	M, T, F	Gaussian mixture model fitting	Multiple hypothesis tracking				
8	K.E.G. Magnusson J. Jaldén H.M. Blau	Deconvolution	M, T, F	Watershed-based clump splitting and parabola fitting	Viterbi algorithm on state-space representation				
9	P. Paul-Gilloteaux	Laplacian of Gaussian or Gaussian filtering	M, T, F	Either maxima with pixel precision (2D) or thresholding + Gaussian fitting (3D)	Nearest neighbor + global optimization				
10	P. Roudot C. Kervrann F. Waharte	Structure tensor	T, F	Histogram-based thresholding and Gaussian fitting	Gaussian template matching				
11	I. Smal E. Meijering	Wavelets	M, F, C	Gaussian fitting (round particles) or centroid calculation (elongated particles)	Sequential multiframe assignment				
12	J.-Y. Tinevez S.L. Shorte	Difference of Gaussian	M, T, F	Parabolic fitting to localized maxima	Linear assignment problem				
13	J. Willemse K. Celler G.P. van Wezel	Gaussian and top hat	T, C	Watershed-based clump splitting	Nearest neighbor				
14	H.-W. Dan Y.-S. Tsai	Gaussian, Wiener and top hat	T, C	Morphological opening–based clump splitting	Nearest neighbor + Kalman filtering				

See **Supplementary Note 1** for further details on methods 1–14. Dim, dimensionality. Detection approaches: M, maxima detection; T, thresholding; F, fitting; C, centroi
GC, gap closing.



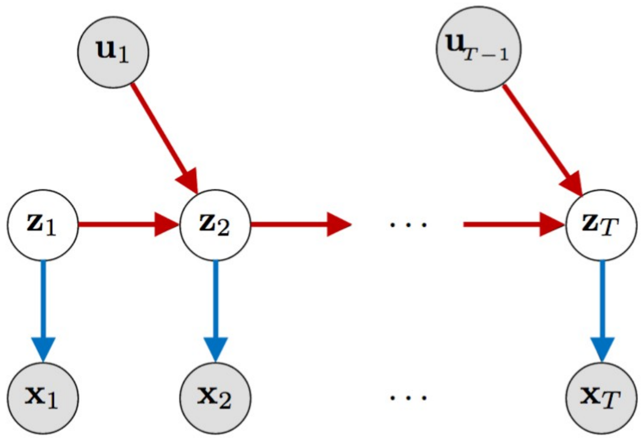
Deep Kalman Filter

Deep Kalman filters

Actions u_t
(e.g., prescribing a medication, performing a surgery)

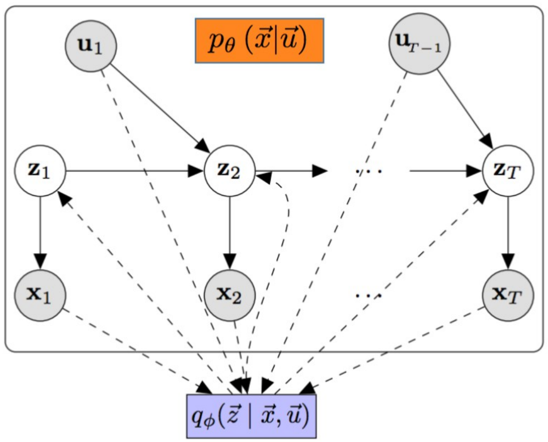
Patient latent state $z_t \in \mathbb{R}^d$

Observations x_t: Lab test results, diagnosis codes, etc.



Optimize *jointly* over generative model $p_\theta(\vec{x}|\vec{u})$ and variational approximation $q_\phi(\vec{z}|\vec{x},\vec{u})$

Stochastic backpropagation
(Rezende et al. 2014, Kingma & Welling, 2014)



Initial state: $z_1 \sim \mathcal{N}(\mu_0, \Sigma_0)$

Action-transition: $z_t \sim \mathcal{N}\left(G_\alpha(z_{t-1}, u_{t-1}), S_\beta(z_{t-1}, u_{t-1})\right)$

Emission: $x_t \sim \Pi(F_\kappa(z_t))$

Uri Shalit at DSHEP2016
https://indico.hep.caltech.edu/indico/conferenceDisplay.py?confId=102

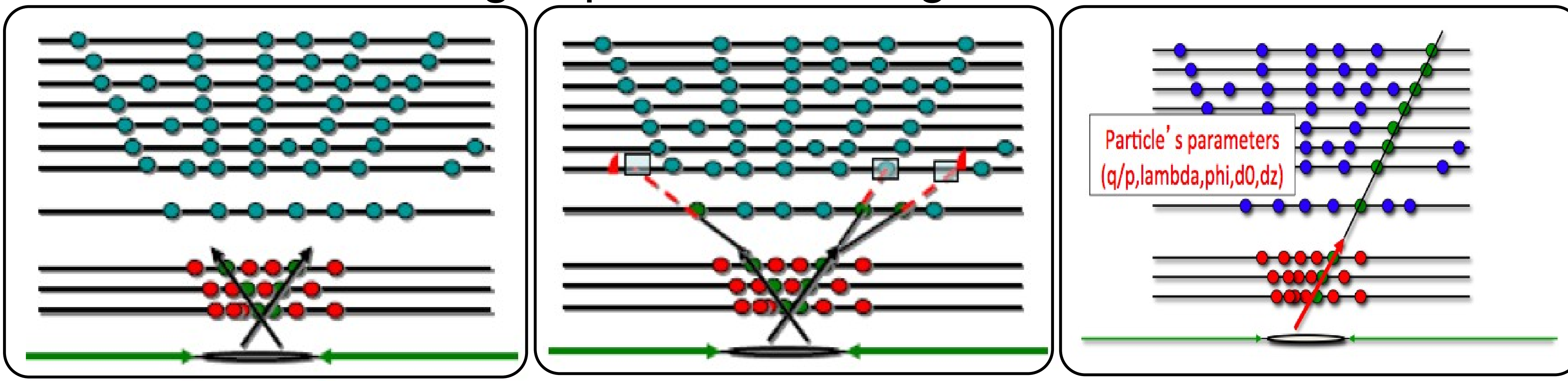
Kalman Filter in Ballistic

- Available methods to track multiple objects using kalman filters
- Deal with “splitting objects”
- Deal with crossing trajetories
- More complexe KF, more computationally intensive ...

Undisclosed contribution during DS@HEP 2016

Pattern Recognition or not

HEP charged particle tracking in a nutshell

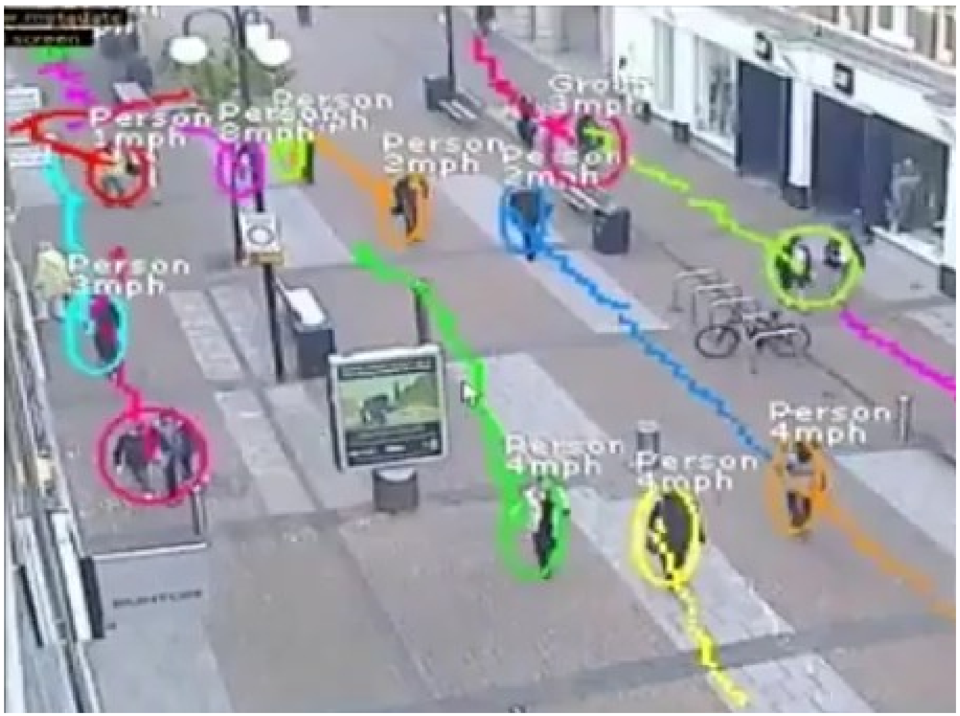


Seeding | Track Building | Track Fitting

- Track building ≡ pattern recognition HEP jargon
- Finding the list of hits belonging to a track ...
- Finding the pattern of hits left by a charged particle in the detector ...

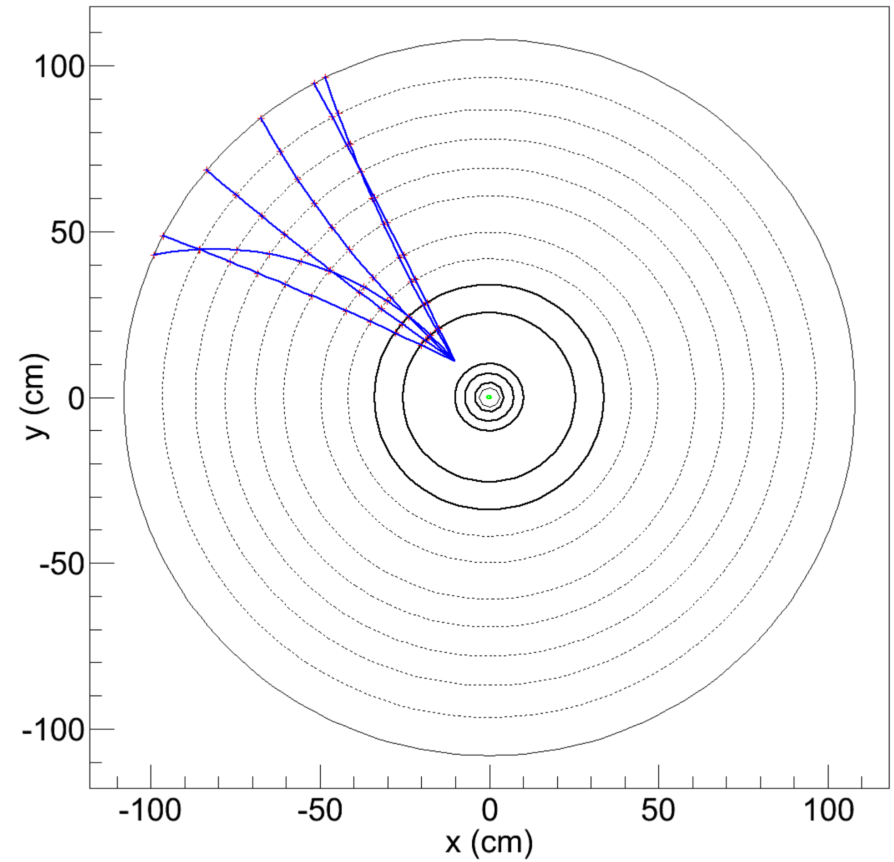
- Not the "usual" data science pattern recognition

Data sparsity

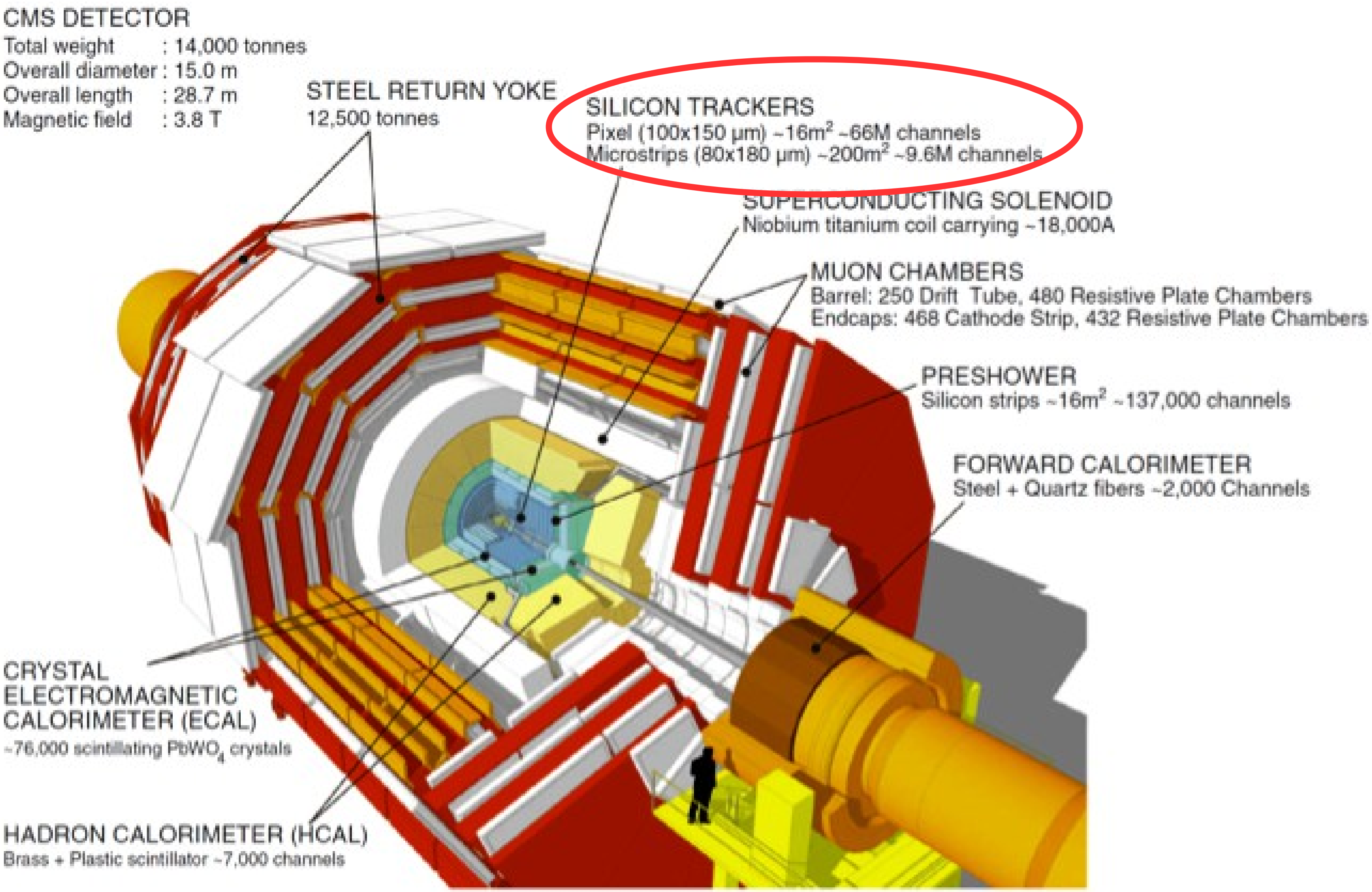


https://privacysos.org/

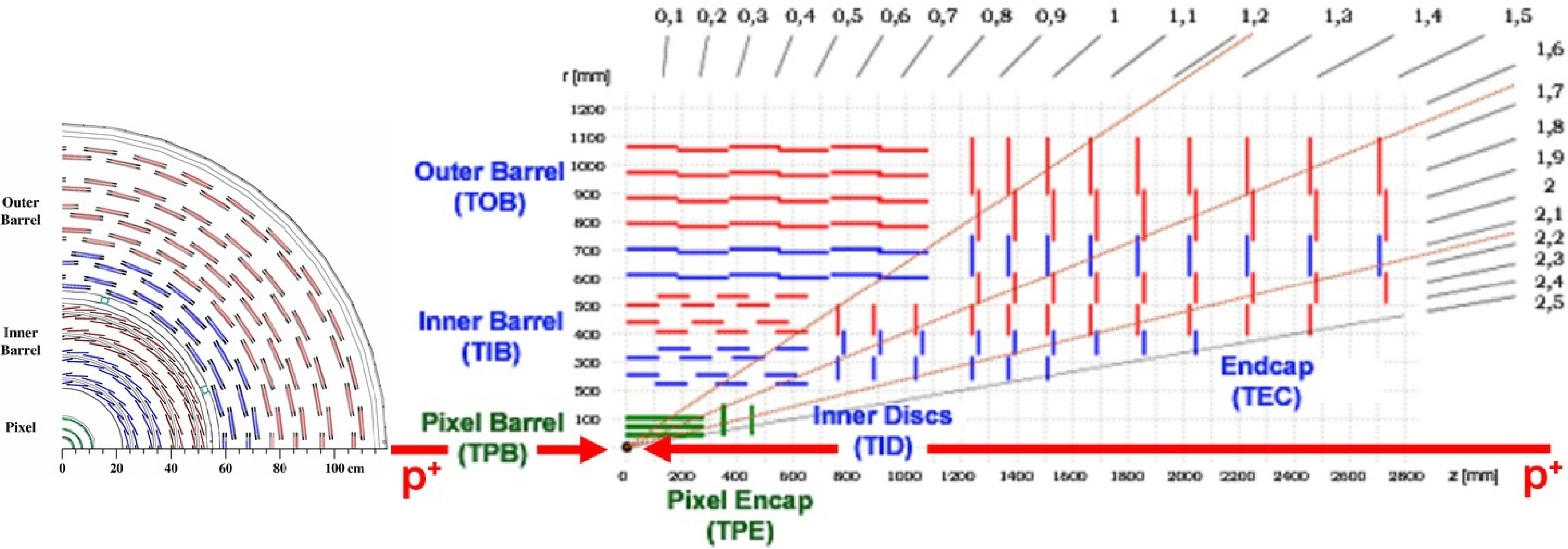
≠



High Dimensionality

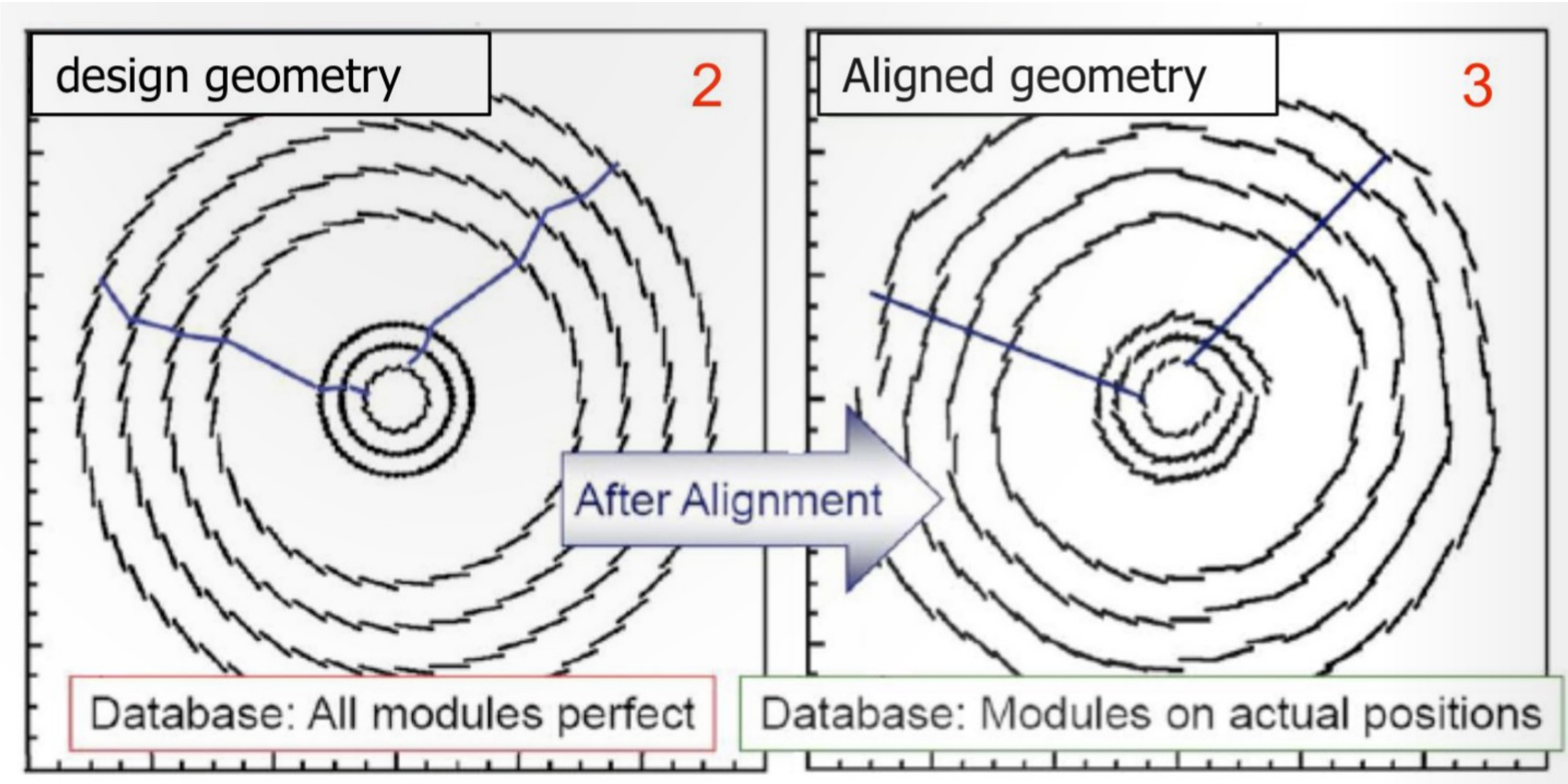


Complex Geometry



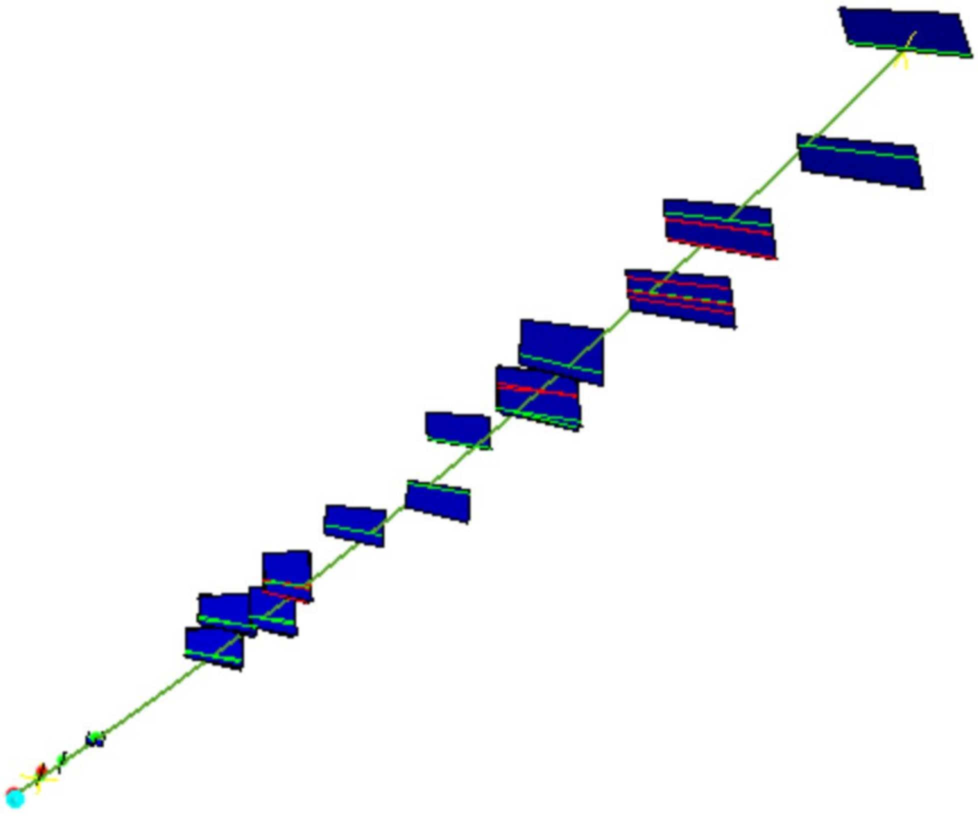
Not the typical data geometry for data science

Mis-aligned Geometry



Mechanical stress (magnetic field, cooling, ...)
does modify the geometry in time

Hit Sequencing



- Hits leave on modules, modules leave on layer, layers are traverse along time.
- “Natural” ordering when trying a hit fitting
- Not so “natural” when doing track building, and hit combinatorics

Figure(s) of Merit(s)

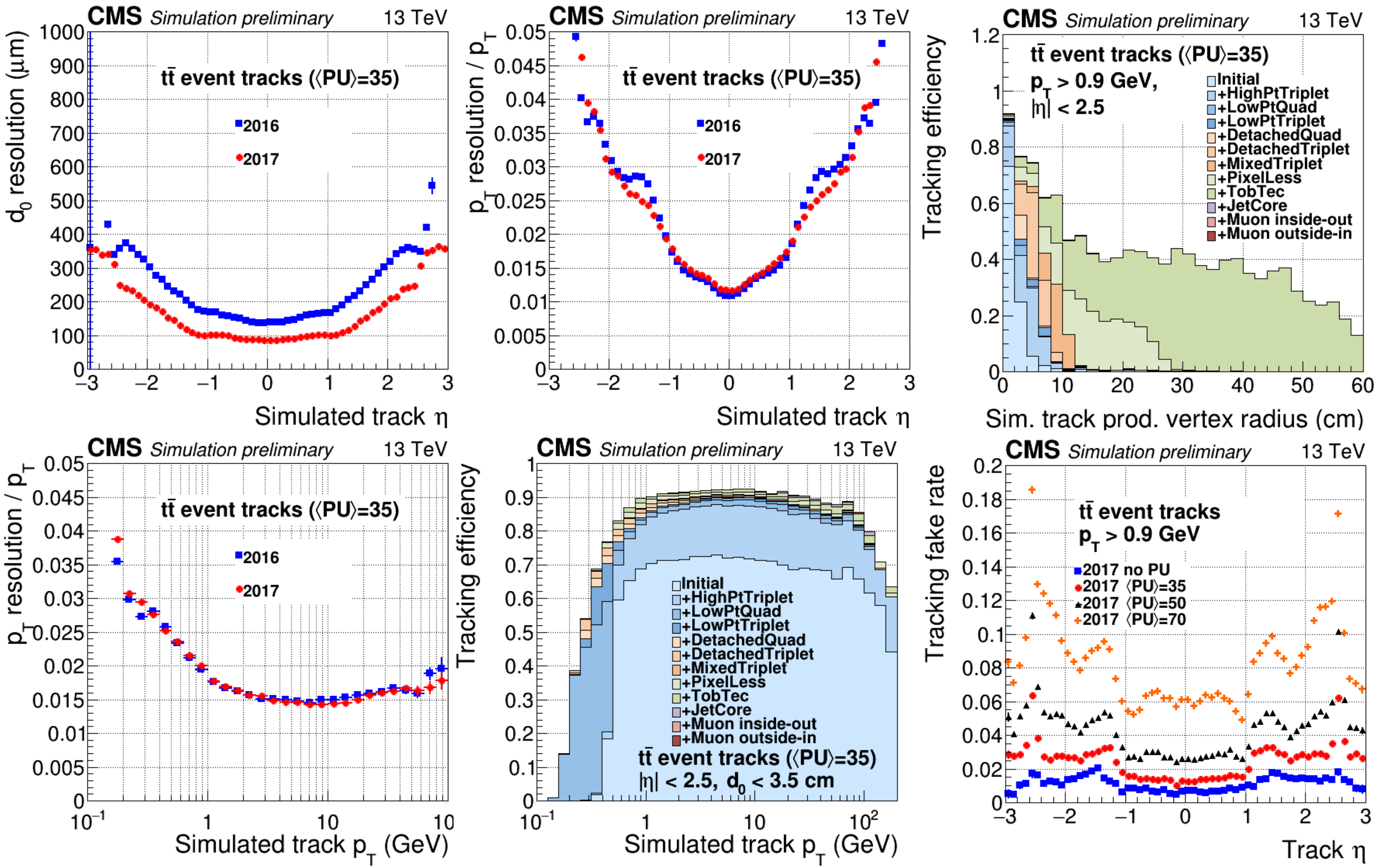


Figure of Merit

- A combination of resolution, fake rate, efficiency, ...

→ Tracking has been improved within a given a method (CKF+CTF) and within processing time constraints

- Not all tracks are equal. Not all features matter

→ High dimensional cost function

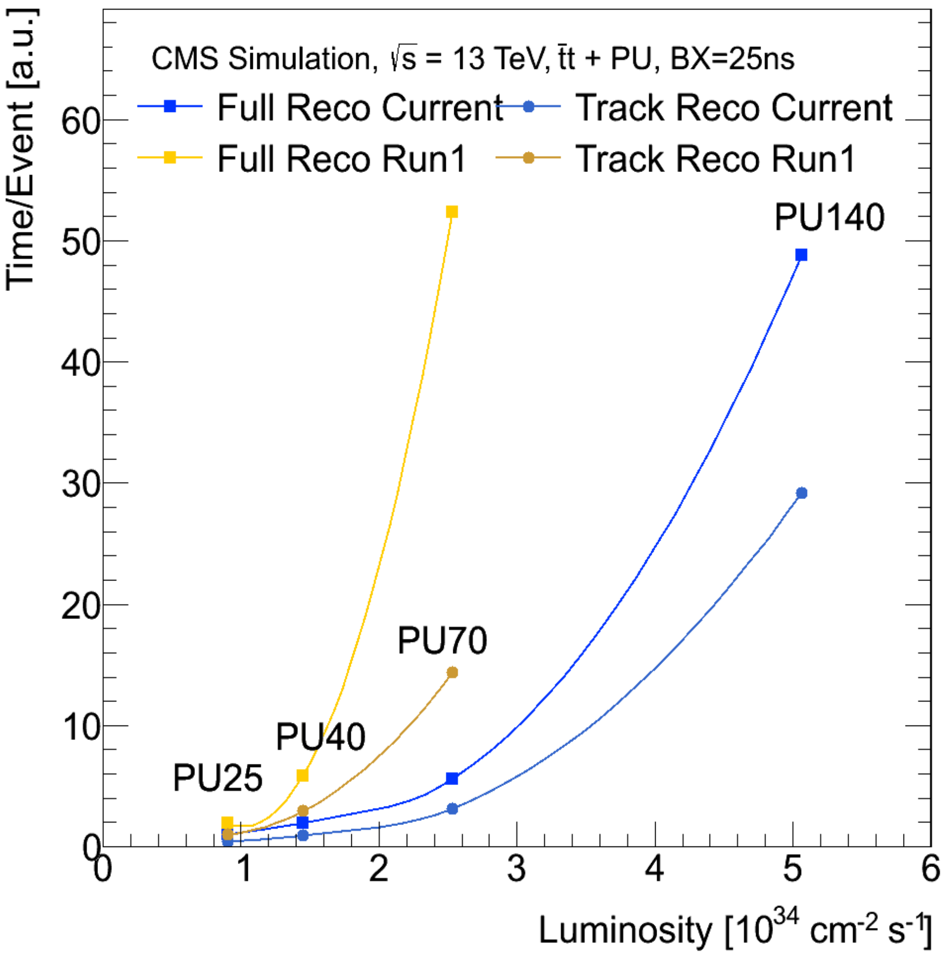
- No golden metric for “tracking” in a general purpose detector

→ Things would be done differently, if the purpose was different

- Remember the breaking point is computation requirement

→ Not something that folds in a cost function ...

Computation Performance



- Worse than quadratic
- PU200 is far off the chart.
- Memory consumption not necessarily an issue

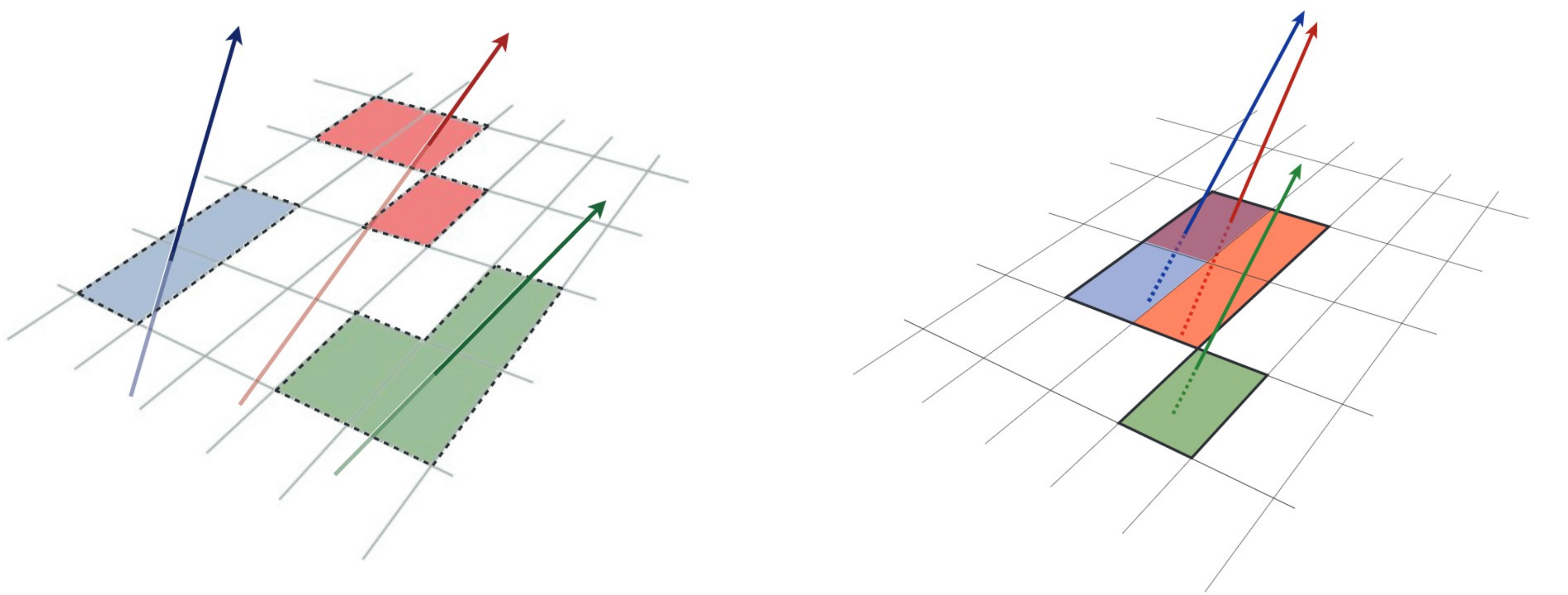
Part II

Machine Learning in Tracking

- Seeding and Clustering
- Pattern recognition
- Track Selection
- Track Parameters
- Vertexing

Seeds and Clusters

Tracking In Dense Environment



(a) Single-particle pixel clusters (b) Merged pixel cluster

Converging tracks are likely in boosted jets and jets dense of charged particles. Degraded performance

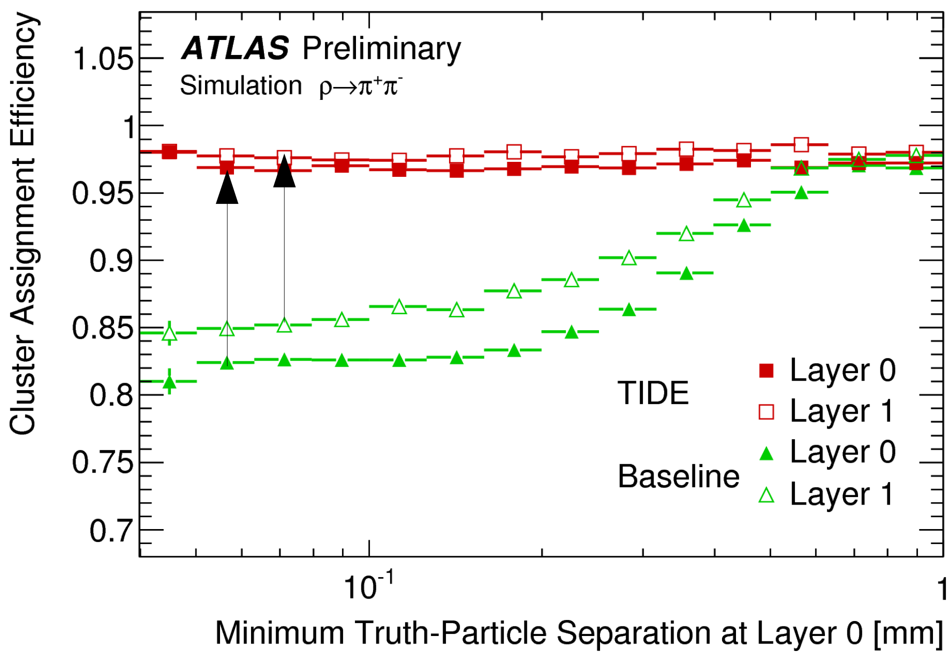
https://arxiv.org/abs/1704.07983
https://link.springer.com/article/10.1140/epjc/s10052-017-5225-7

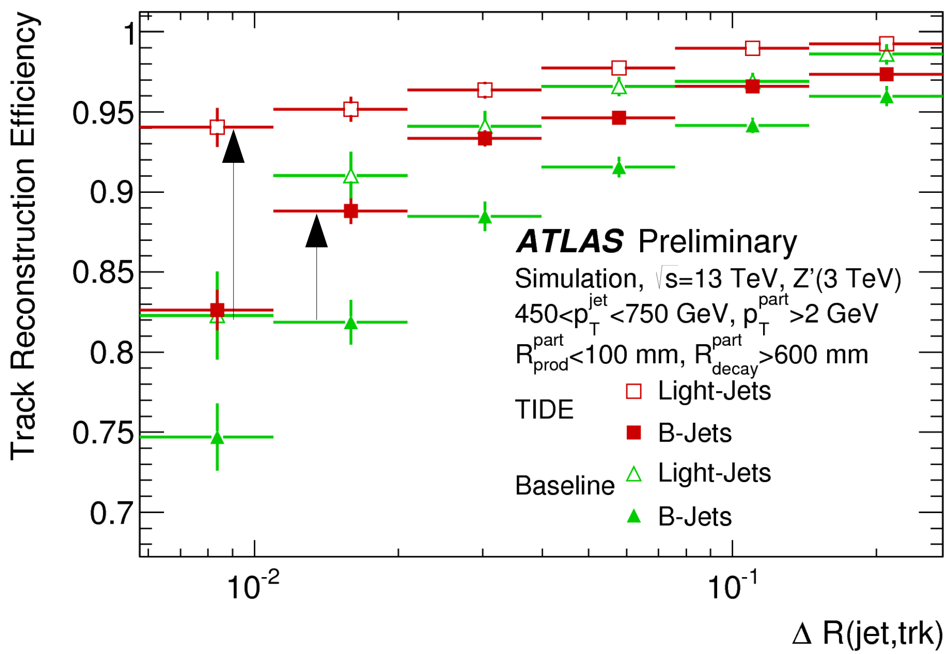
Cluster Splitting

Feed forward NN in three stages

- Determines the category 1-track, 2-tracks, 3-tracks
- Determines the n-crossing positions regression
- Determines the uncertainties as a multi-bin categorization

2 hidden layers fully connected NN with batch norm

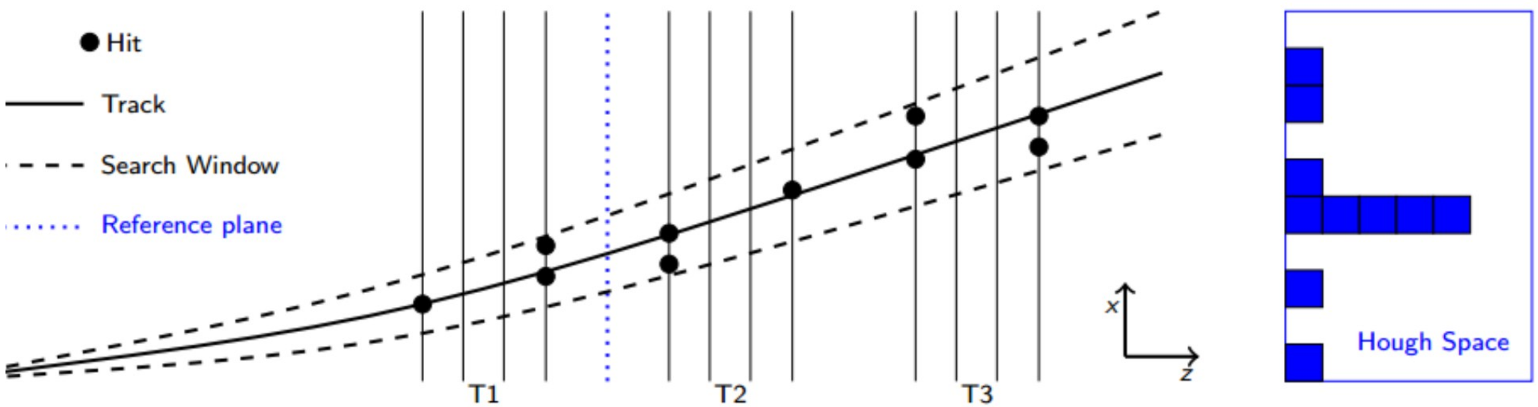




ATL-PHYS-PUB-2015-006

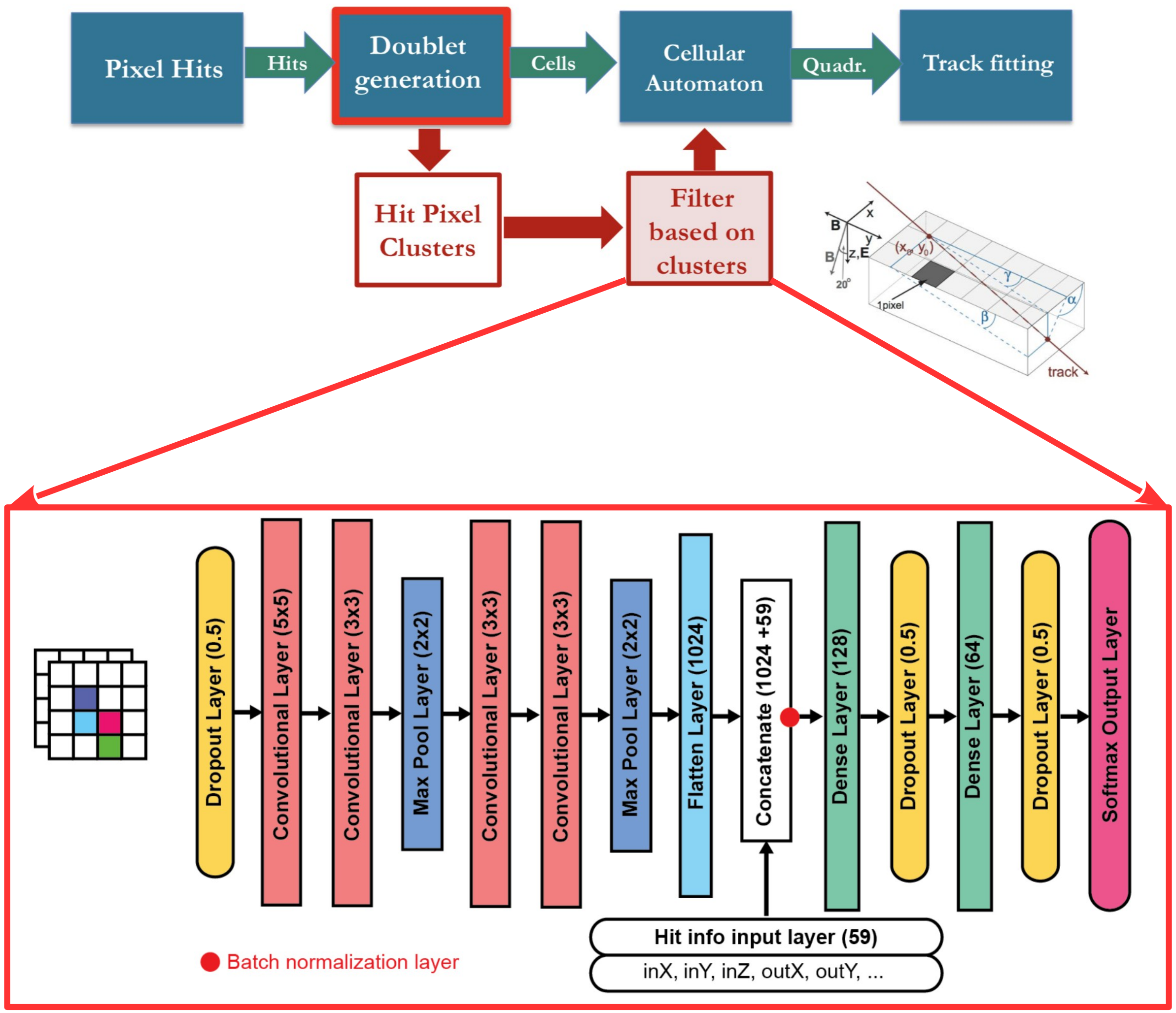
https://link.springer.com/article/10.1140/epjc/s10052-017-5225-7

Seed and Cluster Filtering



- NN classifier to distinguish good and bad clusters in the hough space during forward tracking
- classifier to distinguish good and bad T-seed (Use of the bonsai BDT https://arxiv.org/abs/1210.6861) during downstream tracking

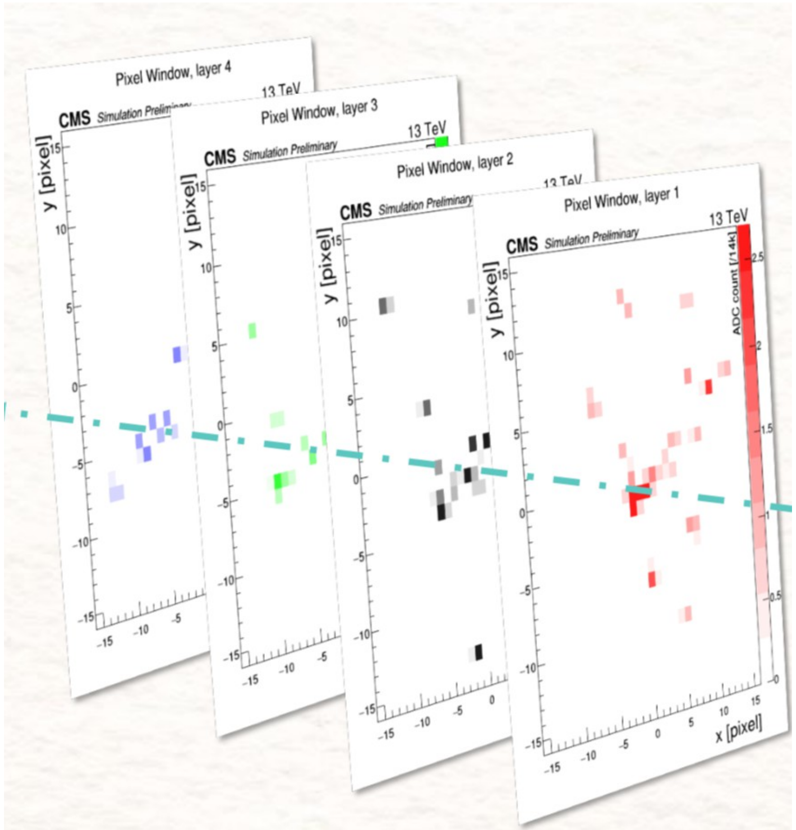
Seed Cleaning

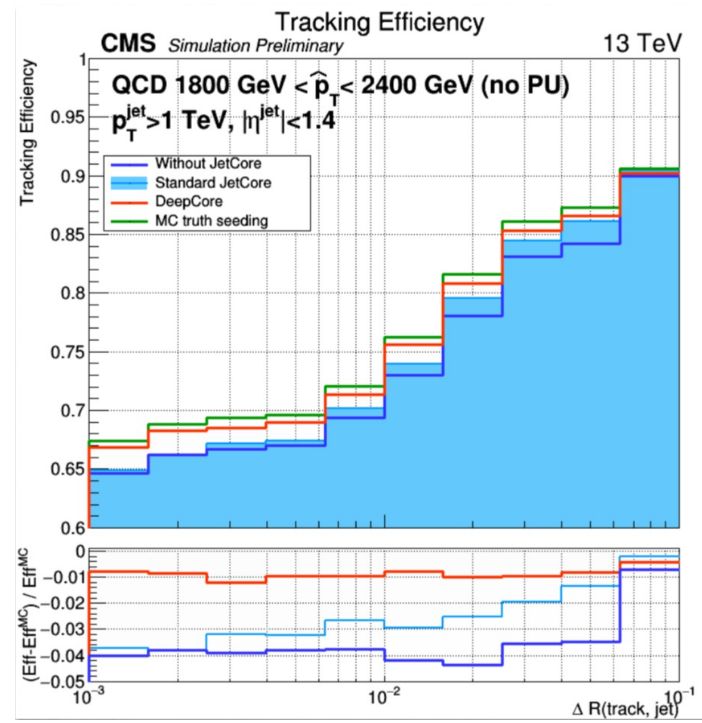


- Categorization of hits doublet using the pixel cluster shapes as input
- Promising at limiting the combinatorial explosion

https://indico.cern.ch/event/567550/contributions/2638698/

Seed Finding in Jets





- Predict tracklets parameters from raw pixels using CNN
- Approaching the maximum performance

https://indico.cern.ch/event/742793/contributions/3274301/

Track Finding

Non Parametric Functional Kernels

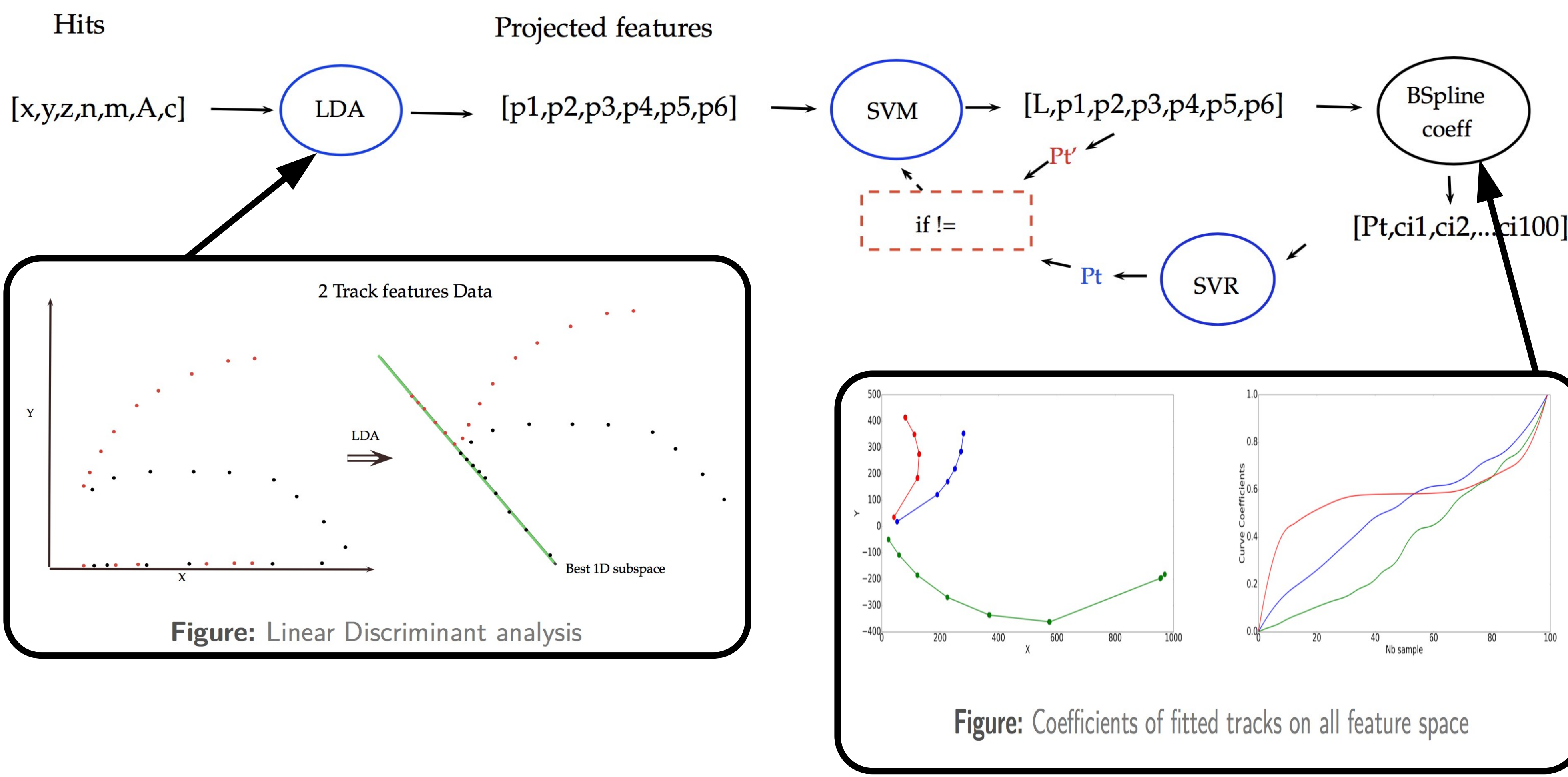
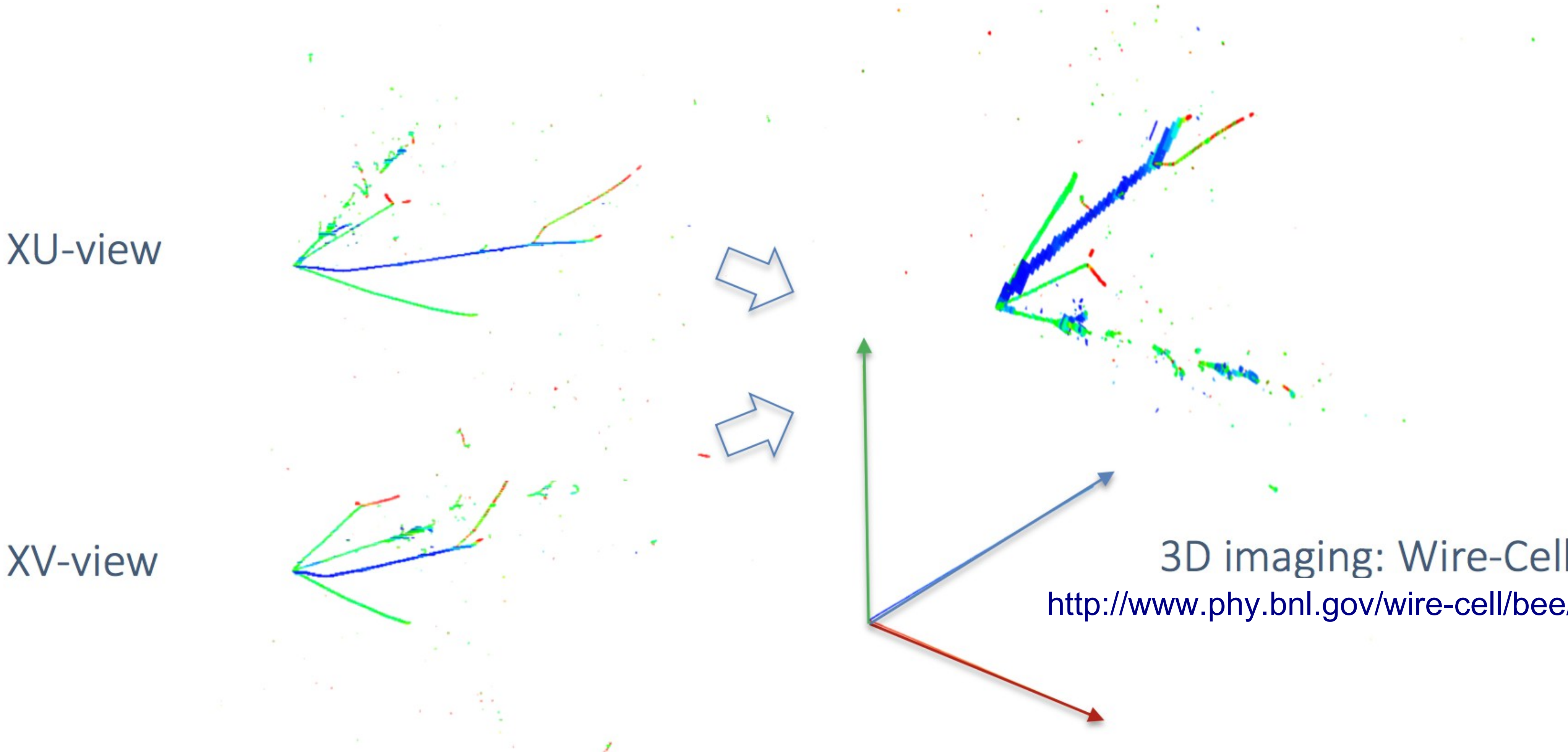


Figure: Linear Discriminant analysis

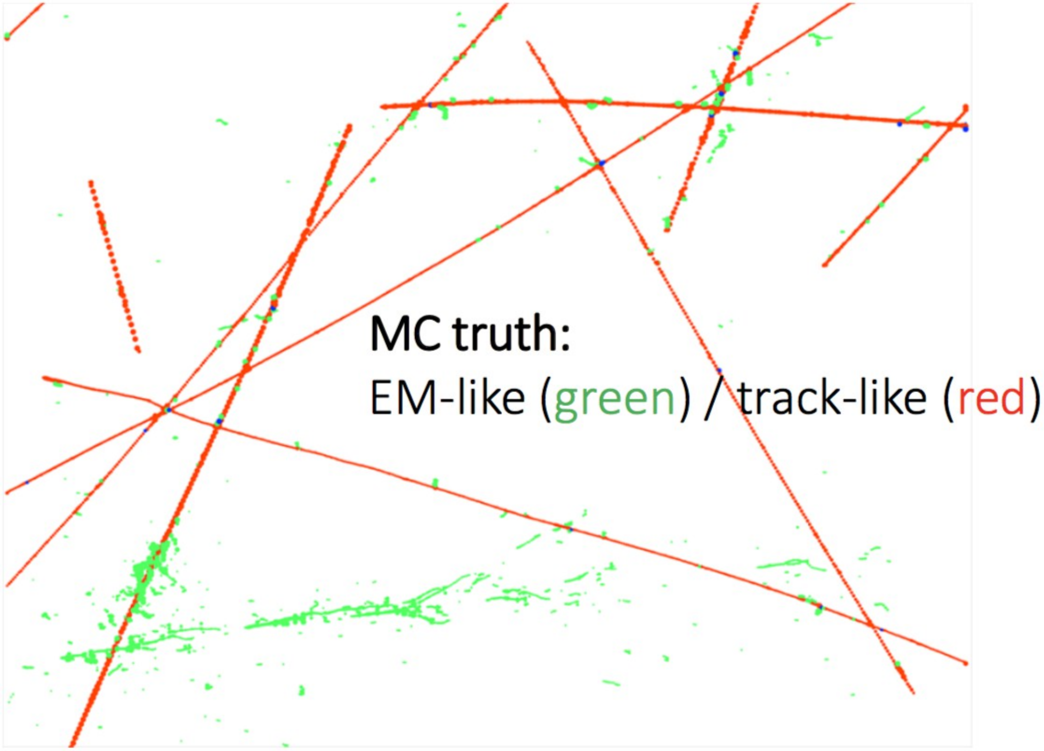
Figure: Coefficients of fitted tracks on all feature space

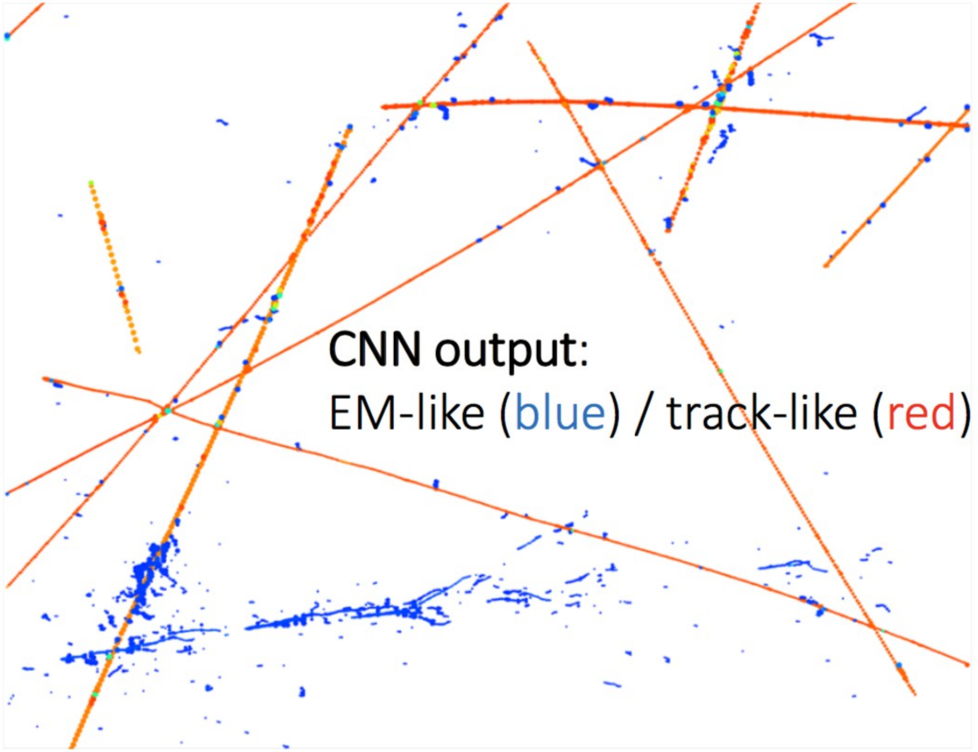
Work in progress : S. Amrouche. T. Golling, A. Salzburger, J. Pilz
https://indico.cern.ch/event/577003/contributions/2444883/

TPC 2x2D to 3D



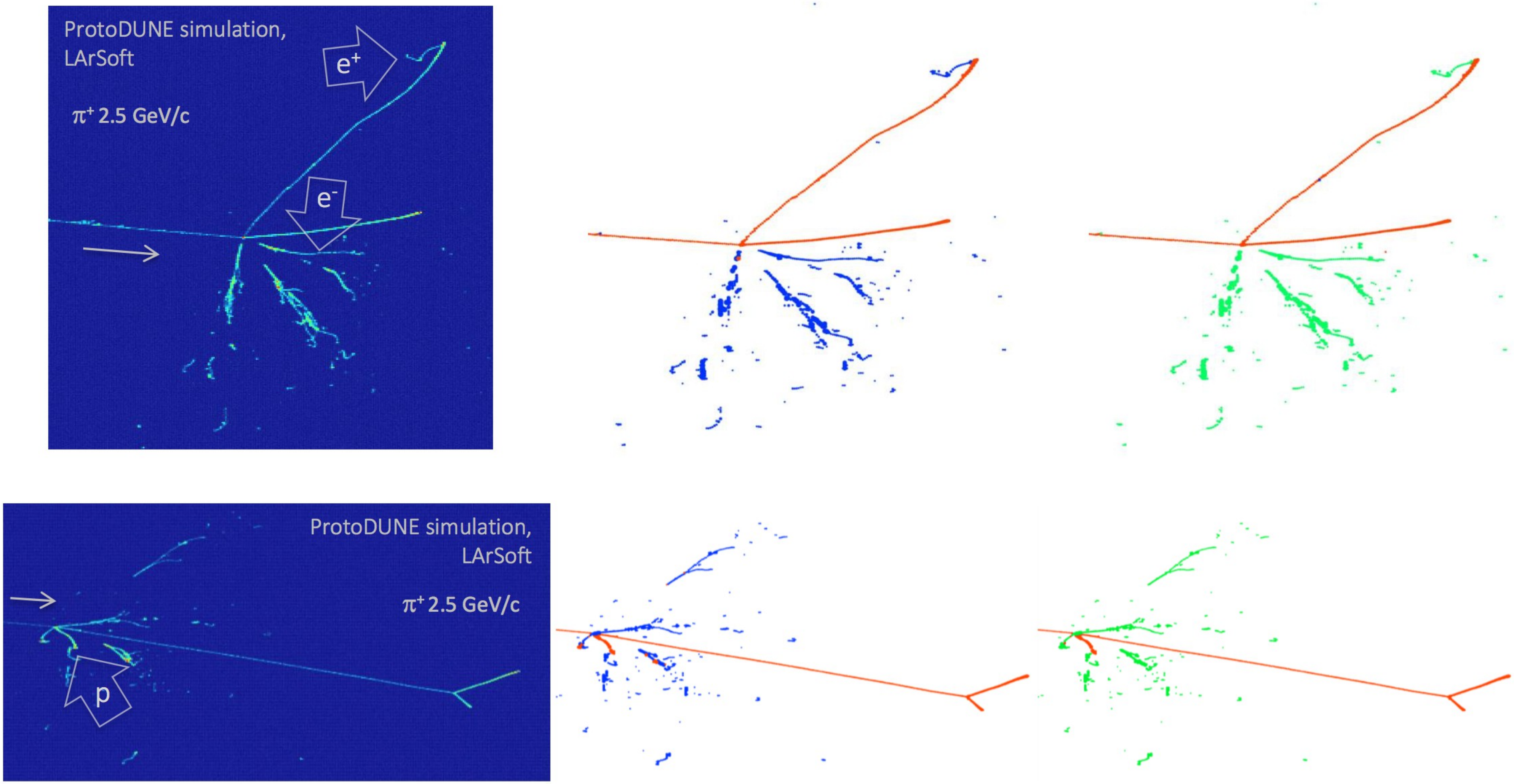
TPC Activity Segmentation





- Challenge to code explicitly
- Almost text-book example of de-noising AE
- Achieved with CNN

Flavor Segmentation

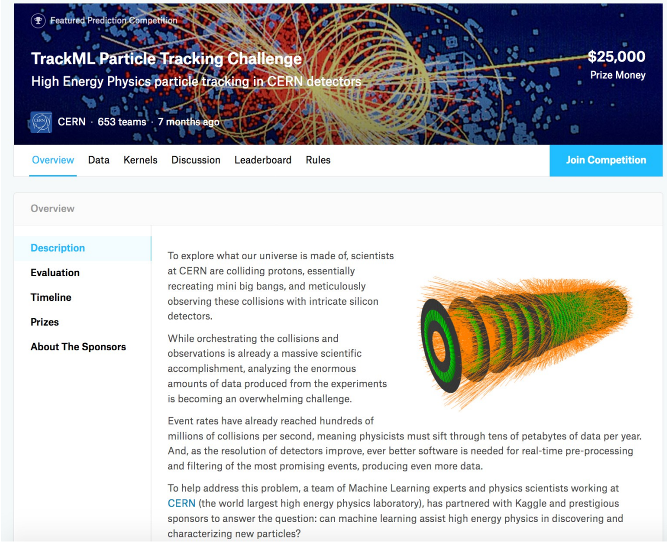


input: 2D ADC

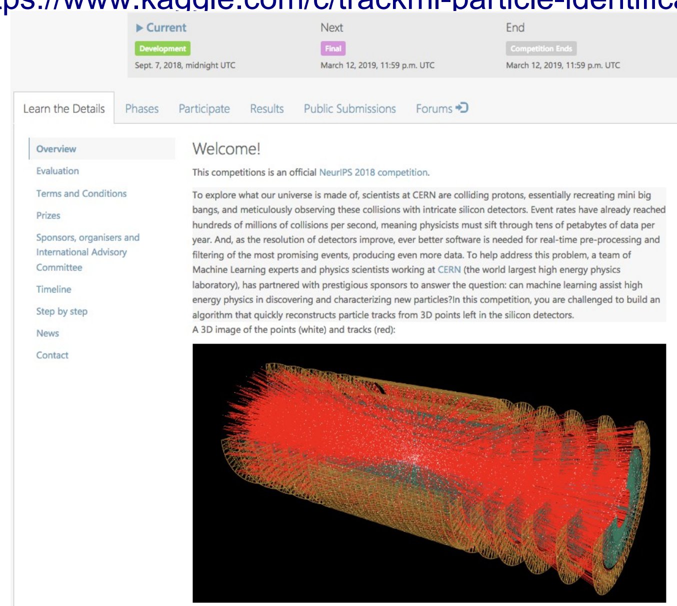
CNN output:
EM-like (blue) / track-like (red)

MC truth:
EM-like (green) / track-like (red)

TrackML Challenge



https://www.kaggle.com/c/trackml-particle-identification

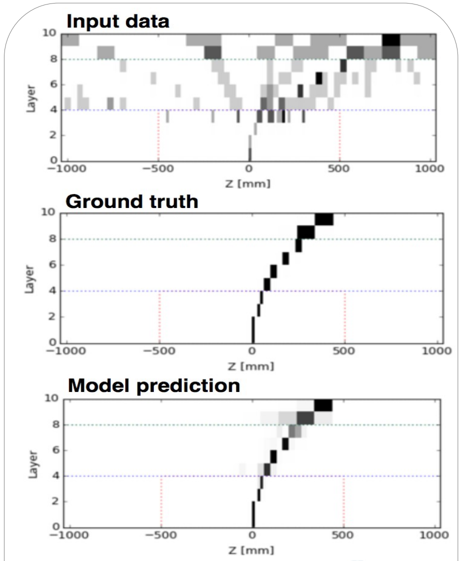


https://competitions.codalab.org/competitions/20112

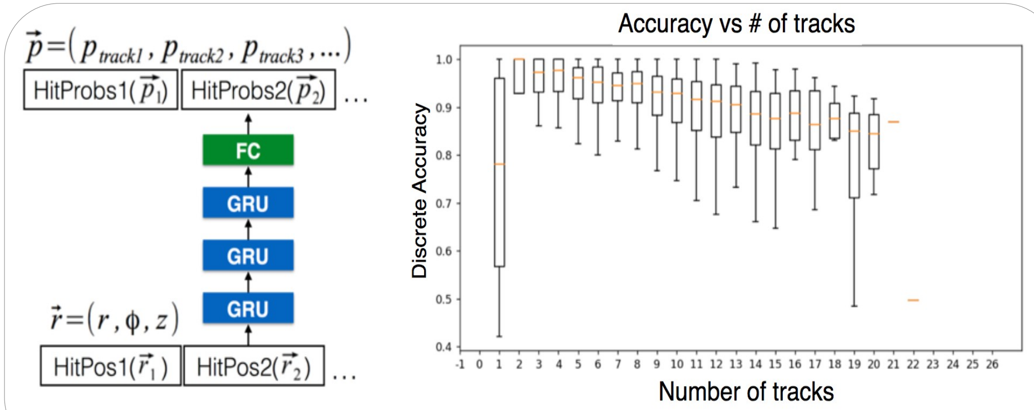
- **First : *Top Quarks***
  - Johan Sokrates is an industrial Mathematics master student
  - **Pair seeding, triplet extension, trajectory following, track cleaning, all with machine learning for quality selection**
- **Second :**
  - Pei-Lien Chou is a software engineer in image-based deep learning in Taïwan
  - **Machine learning to predict the adjacency matrix**
- **Thirds :**
  - Sergey Gorbunov is a p**hysicist, expert in tracking**
  - **Triplet seeding, trajectory following**
- **Jury Innovative prize**
  - Yuval Reina is an electronic engineer and Trian Xylouris is an entrepreneur
  - Marginalized Hough transform with **machine learning classifier**
- **Jury Clustering prize**
  - Jean-François Puget CPMP is a software engineer at IBM. He is both competition and discussion Kaggle grandmaster
  - **DBSCAN clustering** with iterative Hough transform
- **Jury Deep Learning prize**
  - Nicole and Liam Finnie are software engineers
  - DBSCAN seeding, **trajectory following with LSTM**
- **Organization pick**
  - Diogo R. Ferreira is a professor/researcher, focusing on data science and nuclear fusion
  - **Pattern matching**

→ Workshop at CERN in Spring this year with presentation of full details

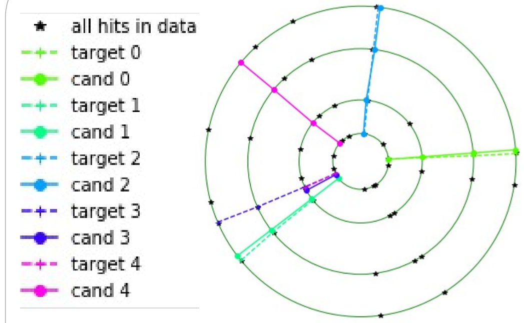
HEP.TrkX Approaches



Track following with RNN



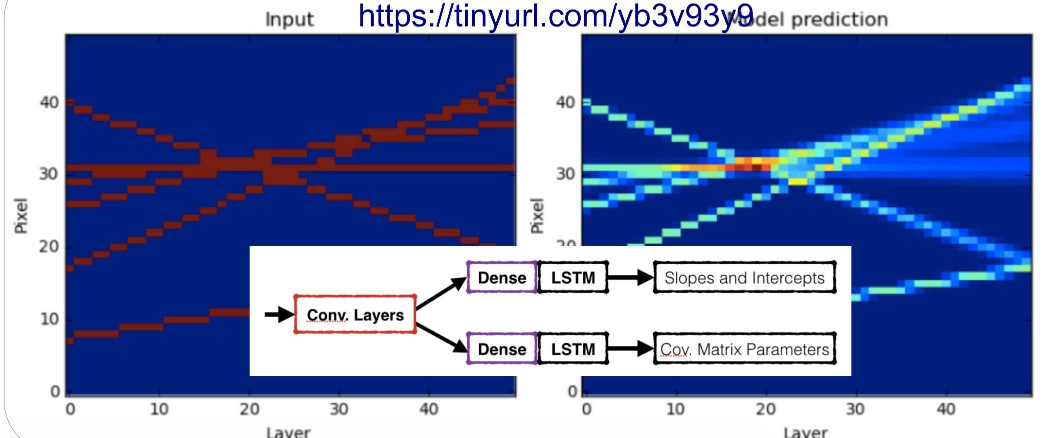
End-to-end hit assignment

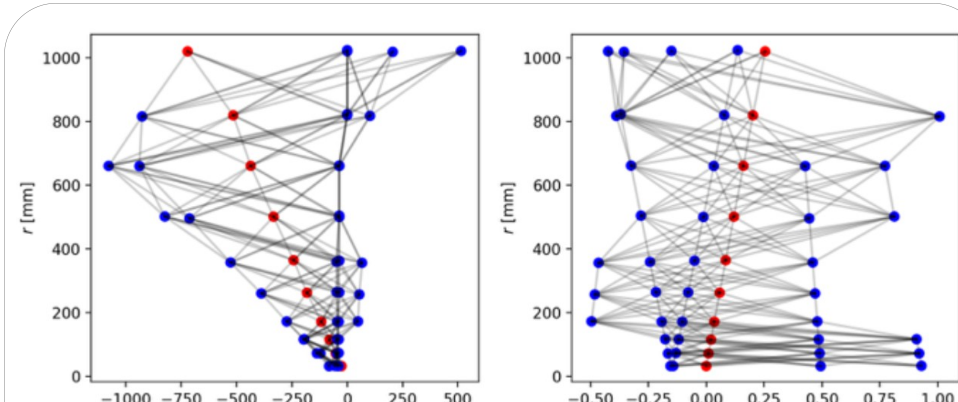


Seq-to-seq track finding

https://heptrkx.github.io/

https://tinyurl.com/yb3v93y9

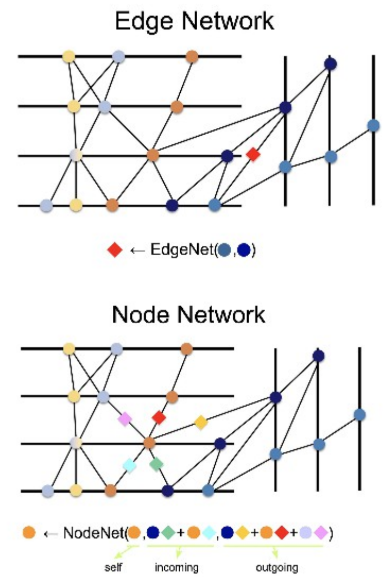


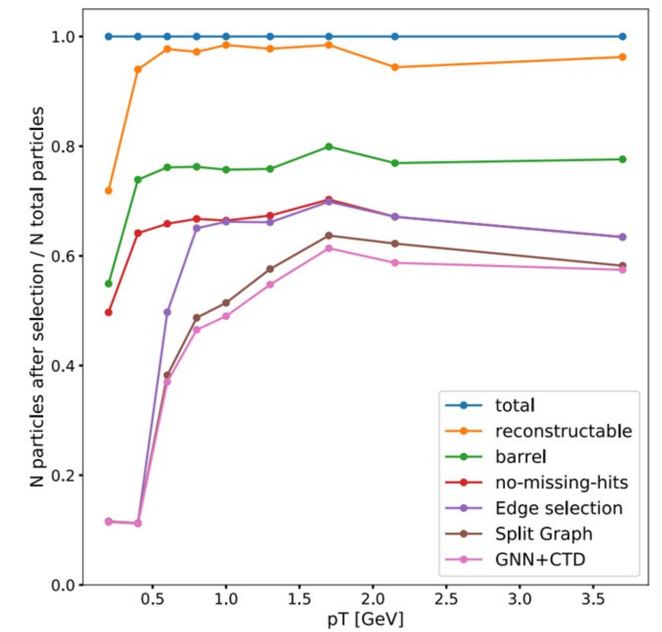


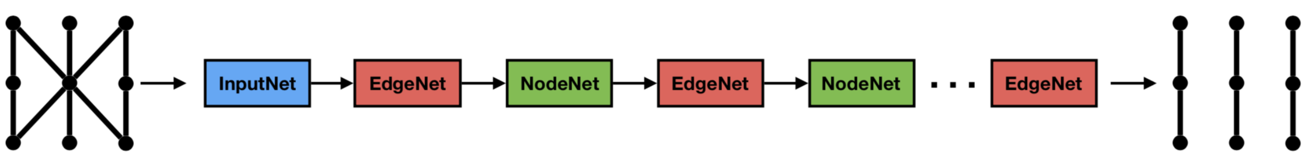
https://tinyurl.com/y87saehf

Graph Network

- **Input Network**
  - Transforms from hit features (r,φ , z) to the node latent representation (N for 8 to 128)
  - Dense : 3→...→N
- **Edge Network**
  - Predicts an edge weight from the node latent representation at both ends
  - Dense : N+N→...→1
- **Node Network**
  - Predicts a node latent representation from the current node representation, weighted sum of node latent representation from incoming edge, and weighted sum
  - Dense : N+N+N→...→N





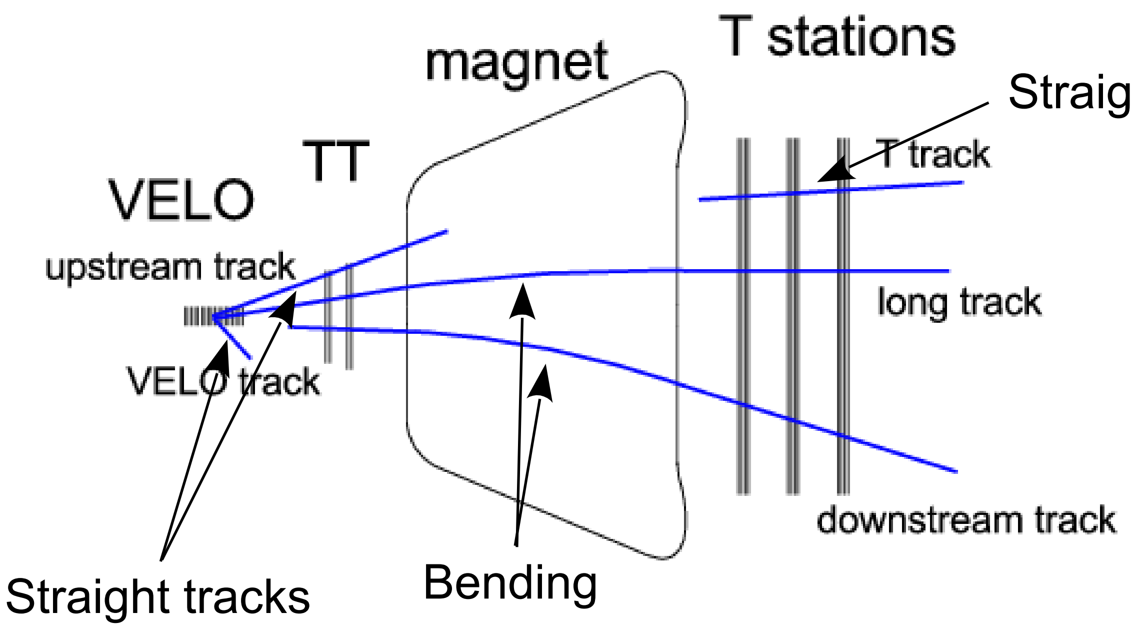


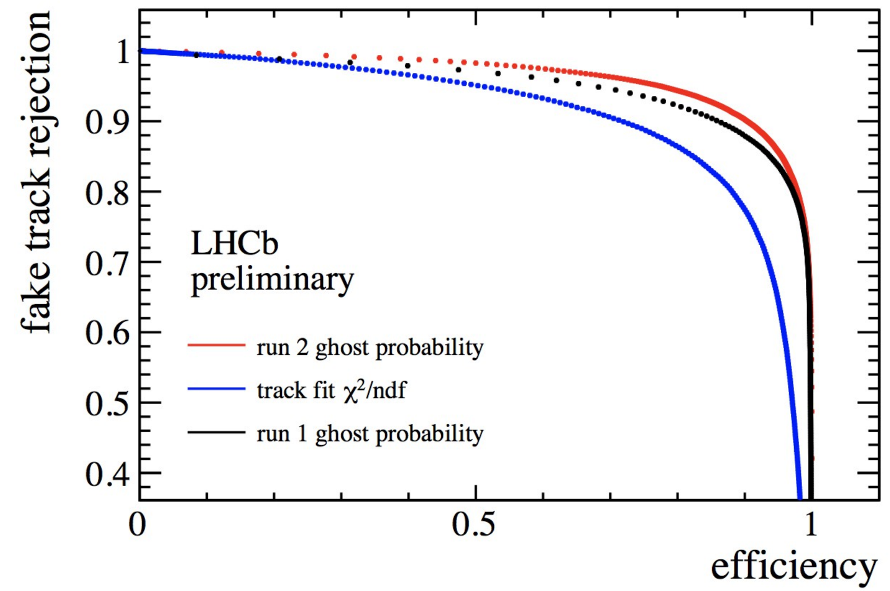
- Tracker hits form graph, using simple geometrical constraints
- Graph neural network and message passing network achieve classification of good edges
- Promising approach on TrackML dataset at 200PU

https://indico.cern.ch/event/742793/contributions/3274328/

Track Selection

Track Selection

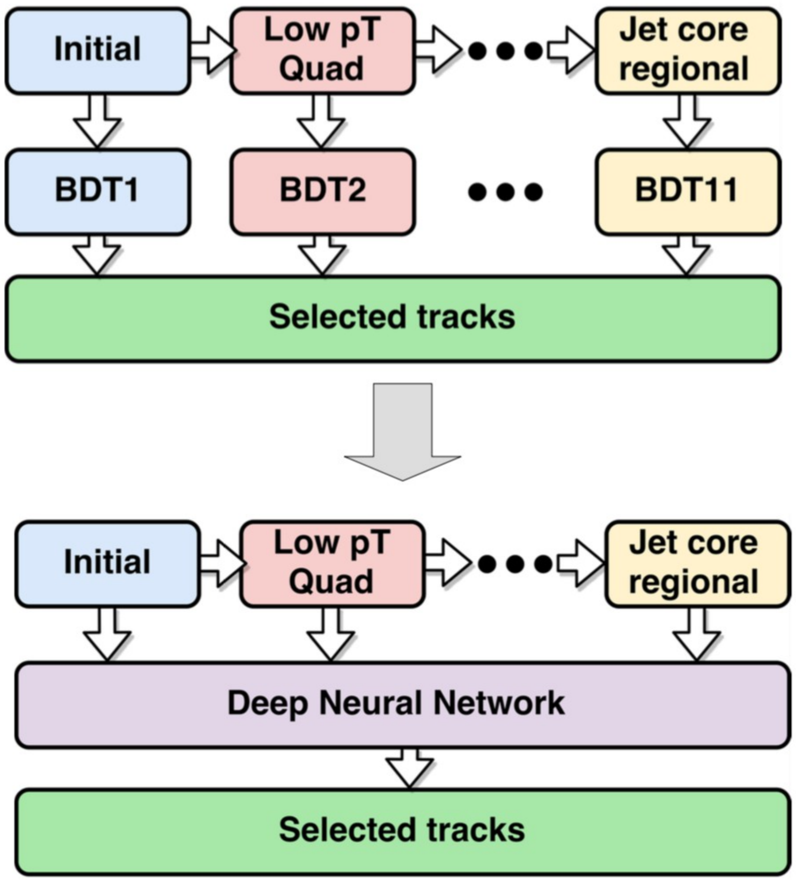


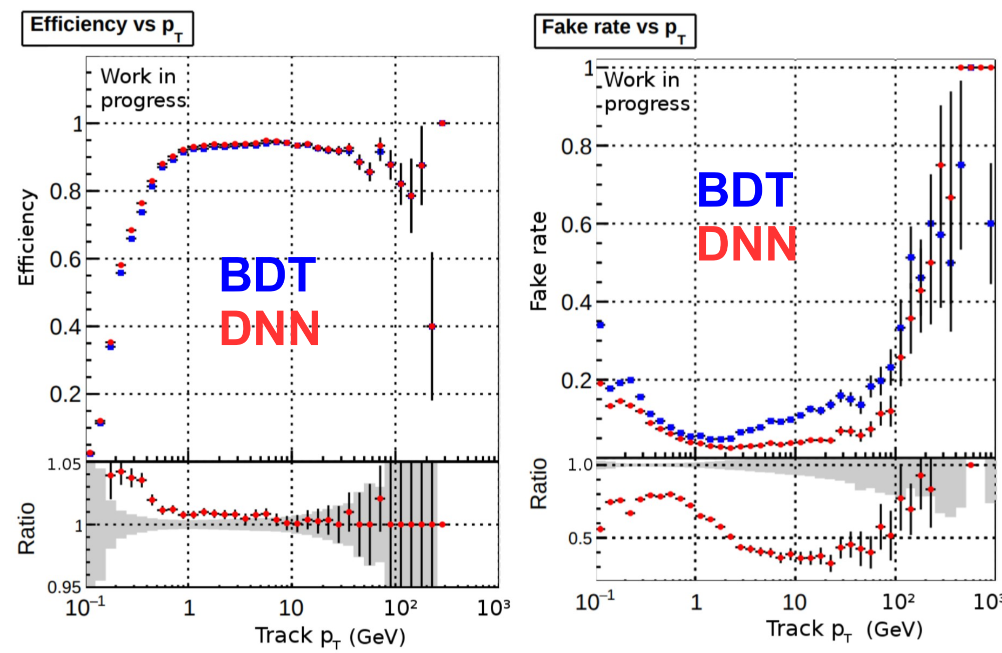


NN classifier implemented to select good from bad tracks in forward tracking and downstream tracking

http://cds.cern.ch/record/2255039

Track Quality with DNN



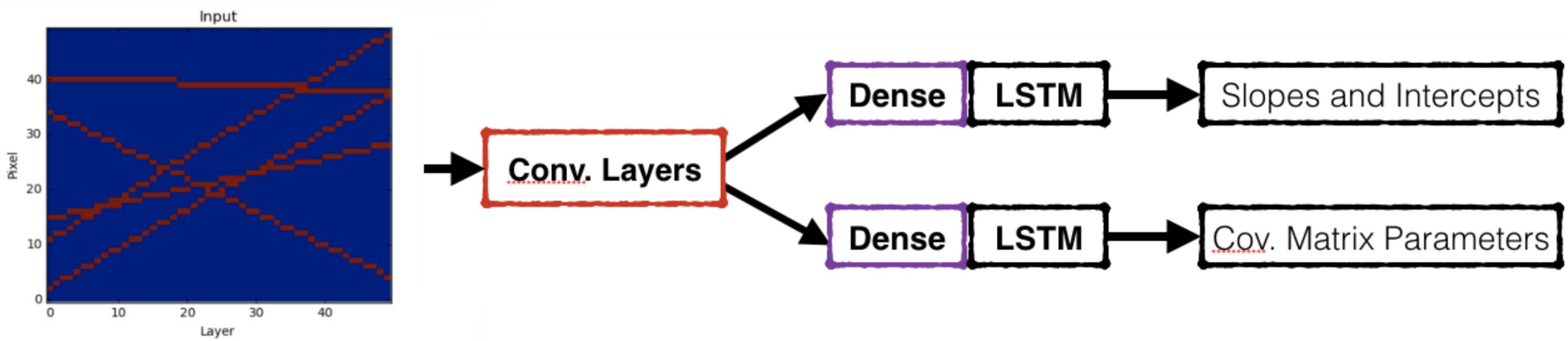


Simplifies and improves track selection within the scope of CMS iterative tracking

https://indico.cern.ch/event/658267/contributions/2813693/

Track Parameters

Track Uncertainties

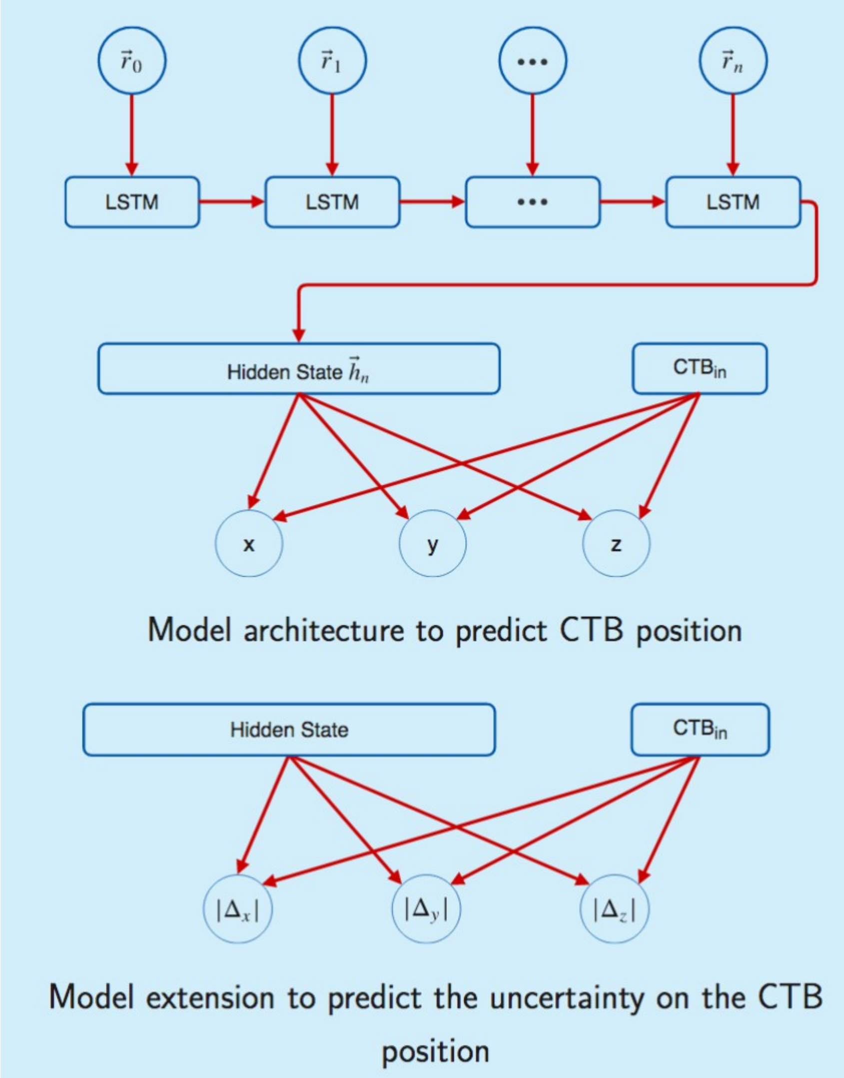


- LSTM predicts many track candidates
- Model predicts a covariance matrix for which there is no ground truth, but is used with the modified loss function

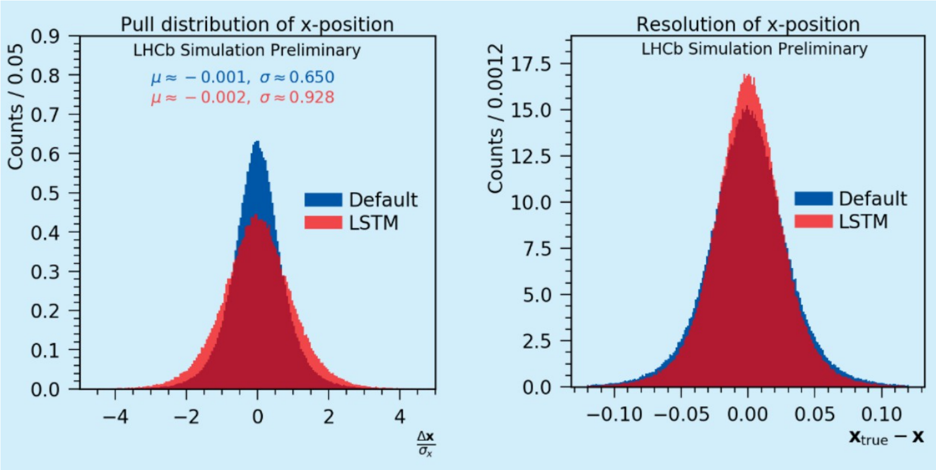
$$L(\boldsymbol{x},\boldsymbol{y}) = \log|\boldsymbol{\Sigma}| + (\boldsymbol{y}-\boldsymbol{f}(\boldsymbol{x}))^T \boldsymbol{\Sigma}^{-1} (\boldsymbol{y}-\boldsymbol{f}(\boldsymbol{x}))$$

https://heptrkx.github.io/

Impact Parameters



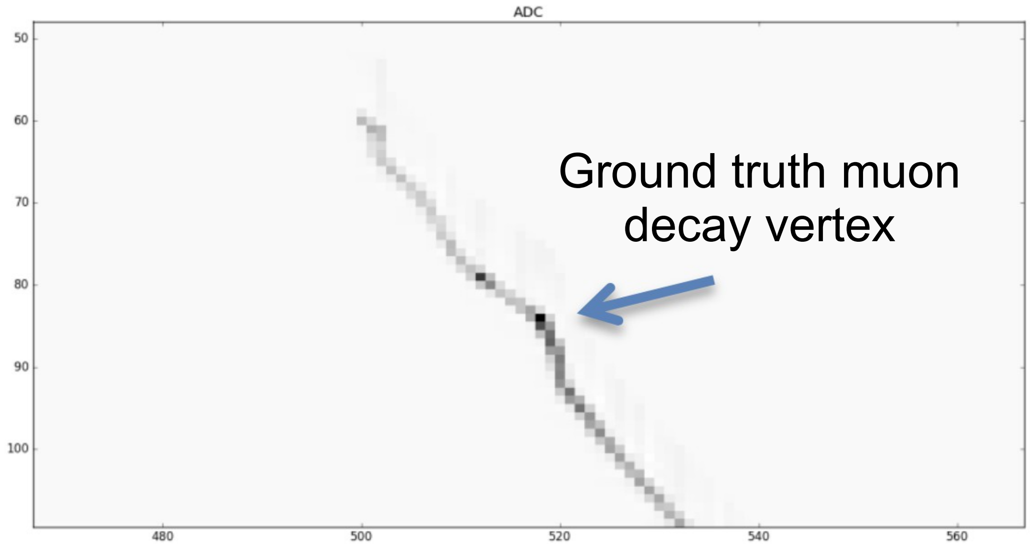
- LSTM model supplements a Kalman Filter approach
- Improve resolution and estimation of track impact parameters in LHCb

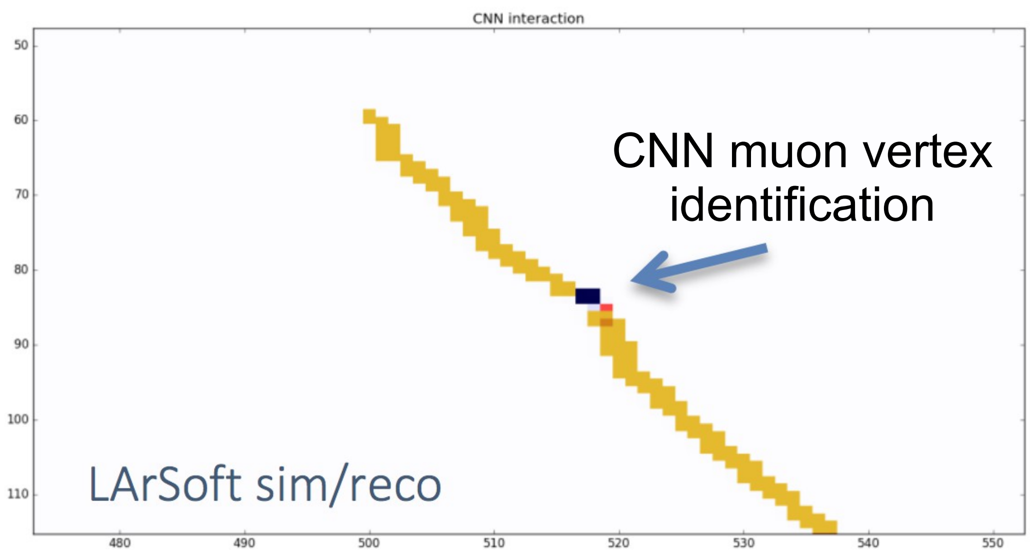


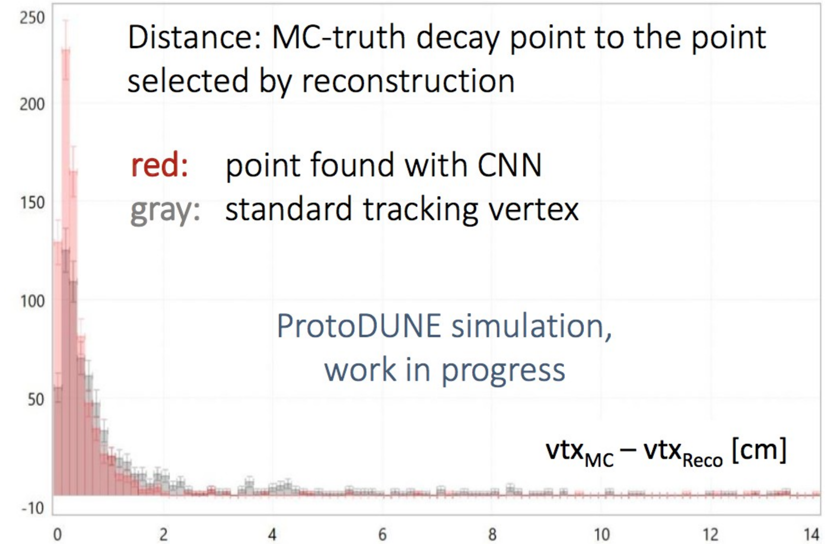
https://indico.cern.ch/event/587955/contributions/2935754/

Vertexing

Decay Point Identifier

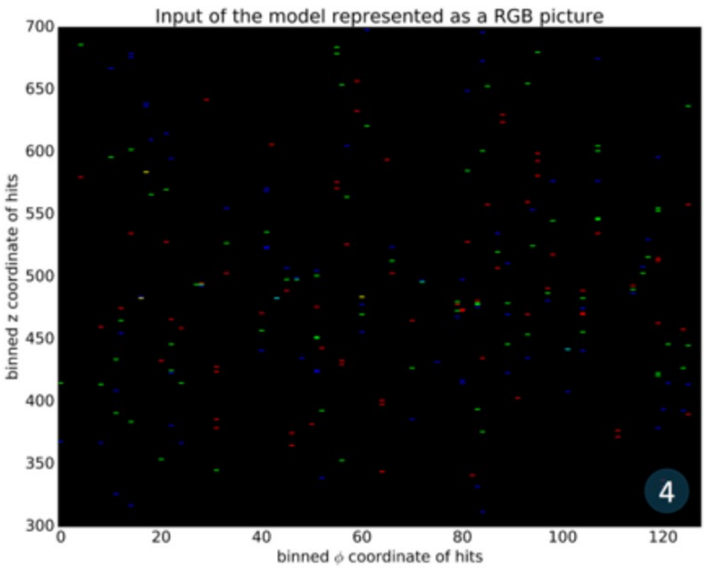


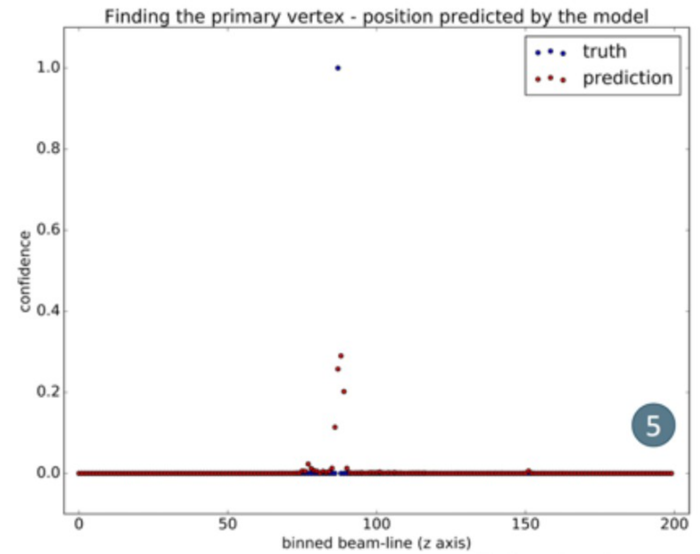




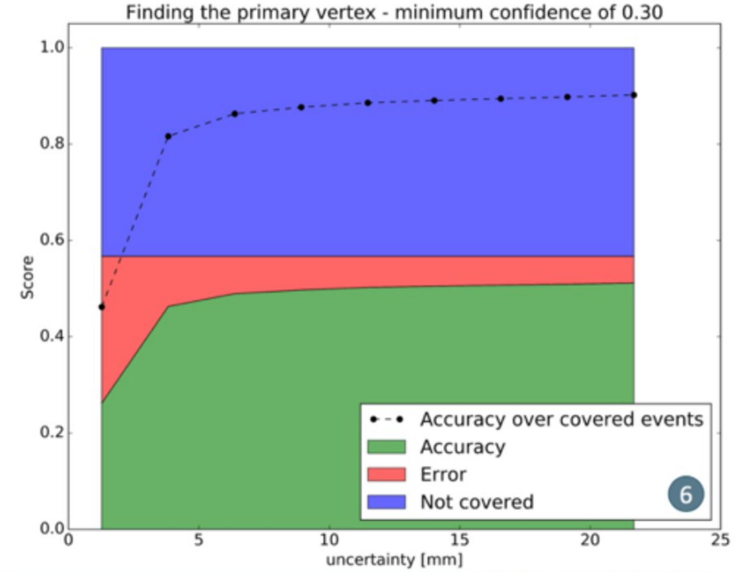
- CNN slightly outperform the classical approach
- Much less complication in programming the vertex finding

Vertexing with CNN



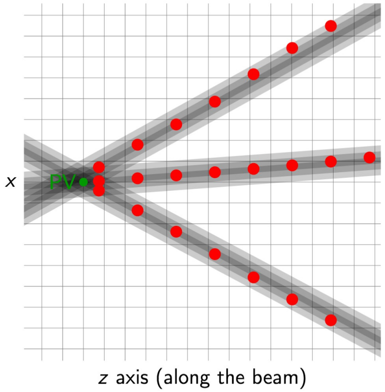


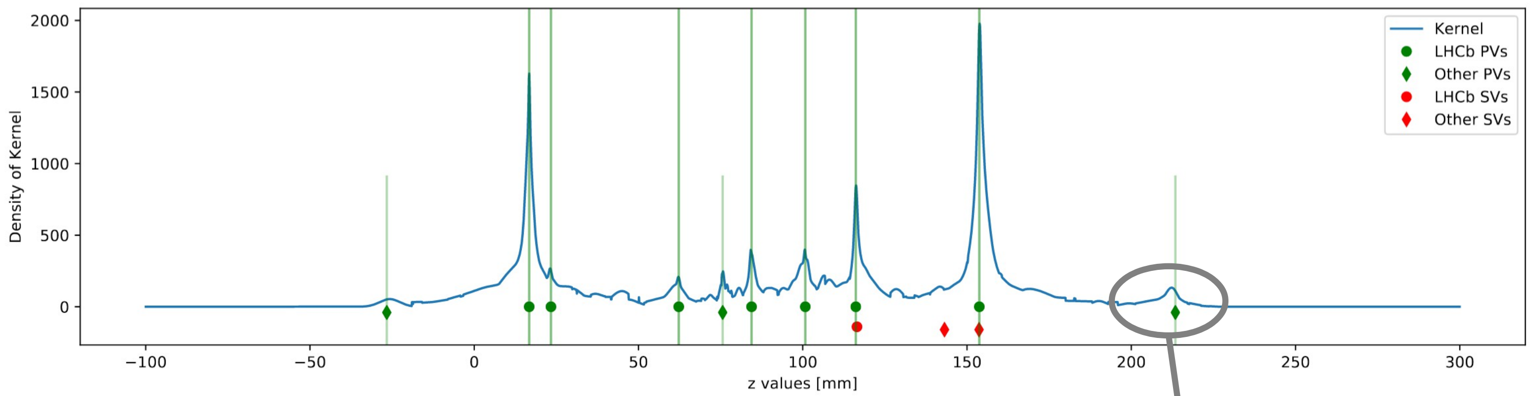
- Using hits binned (η, φ) map in input for a regression of the primary vertex position
- Modest success

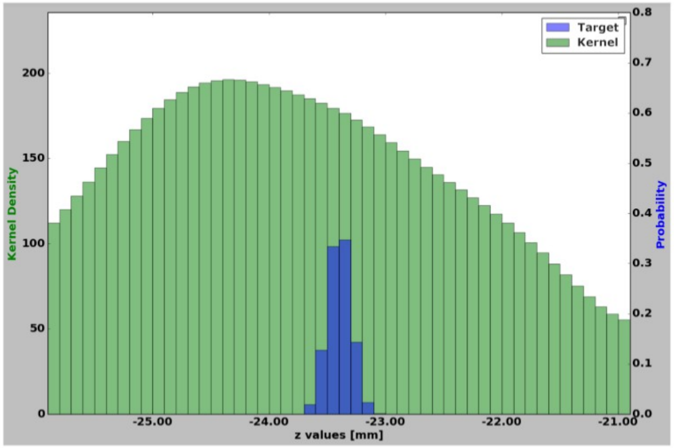


https://indico.cern.ch/event/567550/contributions/2629737/

Hybrid Vertexing







- Form a track density over longitudinal axis using Gaussian kernels
- Learn vertex position from local longitudinal density
- Similar performance with traditional approach.
- Advantage of ML in deployment

https://indico.cern.ch/event/708041/contributions/3269692/

Summary

- Charged particle tracking is a computationally intensive task
- Specific challenges in applying machine learning in High Energy Physics

➢ Machine learning is already applied at several levels to cope with the task complexity.

➢ Active R&D in tracking & vertexing using machine learning

Extra Material

Where ML Can Fit

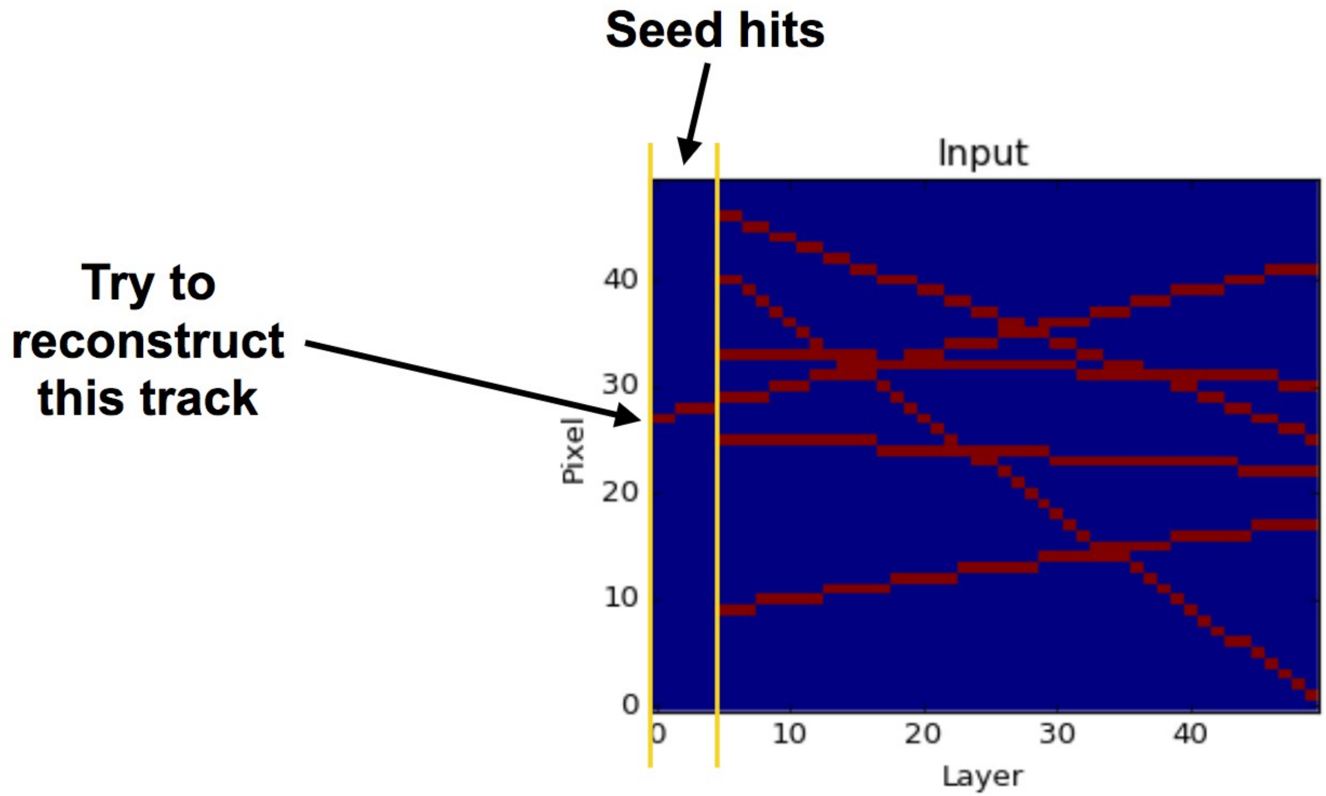
- Signal de-noising (less hit, less combinatorics)
- Making of clusters of hits (less merged, less ambiguity)
- Hits quality (less noise, less combinatorics)
- Seed making (faster composition of tracklets)
- Seed cleaning (less seed, less track making)
- KF pattern recognition
  - In the transport, the update to the new state: deep KF
  - Selecting the best hit candidate
- Pattern recognition
  - Seeded track making
  - Un-seeded track making
- Track fitting
  - Track parameters regression
  - Track parameter reconstruction
- Any combination with other alternative methods (see next slides)
- Any new idea from this workshop ...

ML in Tracking

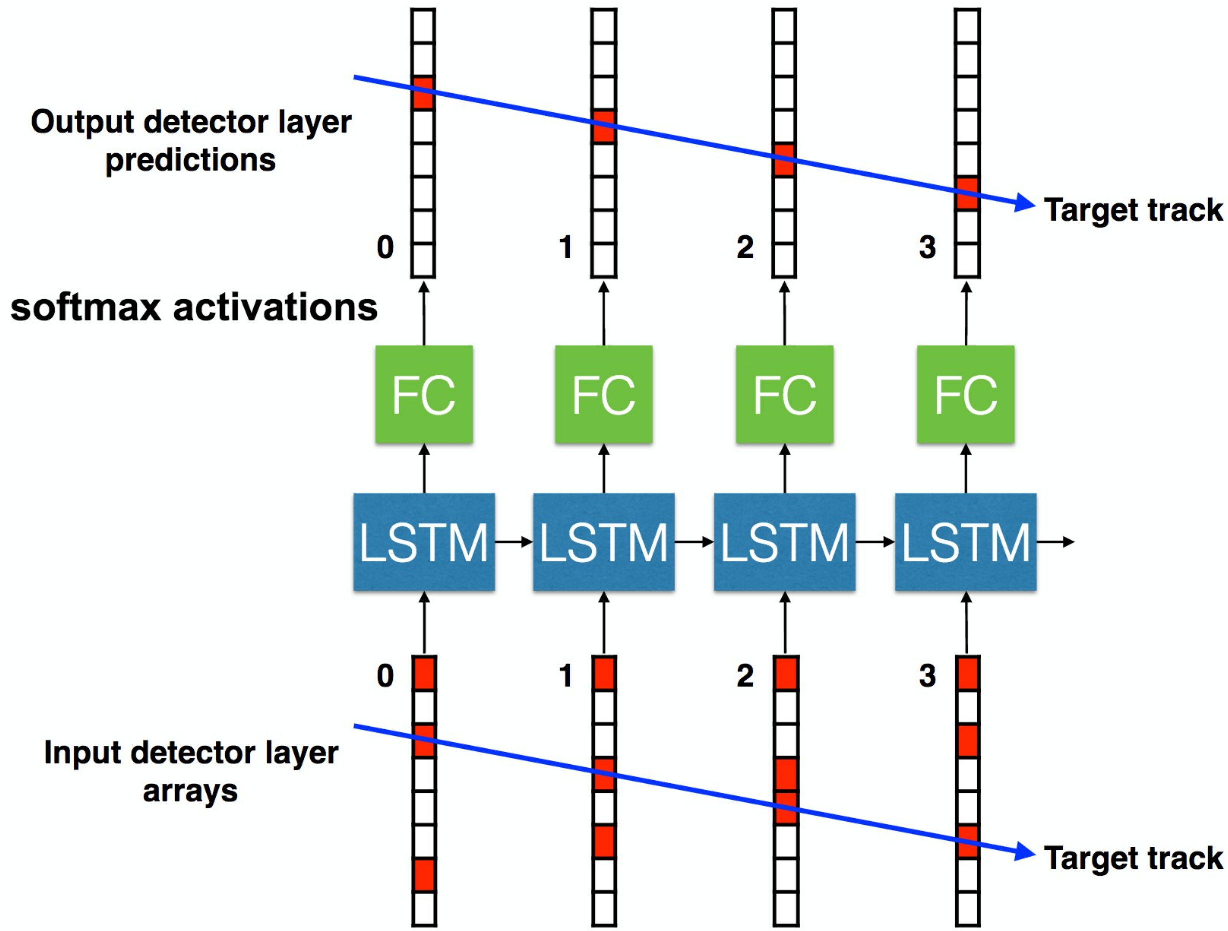
- Hopfield network
- Tracking in dense environment in ATLAS
- Seed cleaning in CMS
- Activity segmentation in TPC test beam
- Activity segmentation for neutrino flavor ID
- Muon decay point identifier
- Track selection in LHCb and in CMS
- Tracker hit cluster selection in LHCb
- Non-parametric functional regression
- Seeded track finding in simplified model
- Track parameters estimation using LSTM
- Pattern recognition with sequence-2-sequence
- Pattern recognition with graph network
- ...

Seeded Pattern Prediction

- Hits on first 3 layers are used as seed
- Predict the position of the rest of the hits on all layers



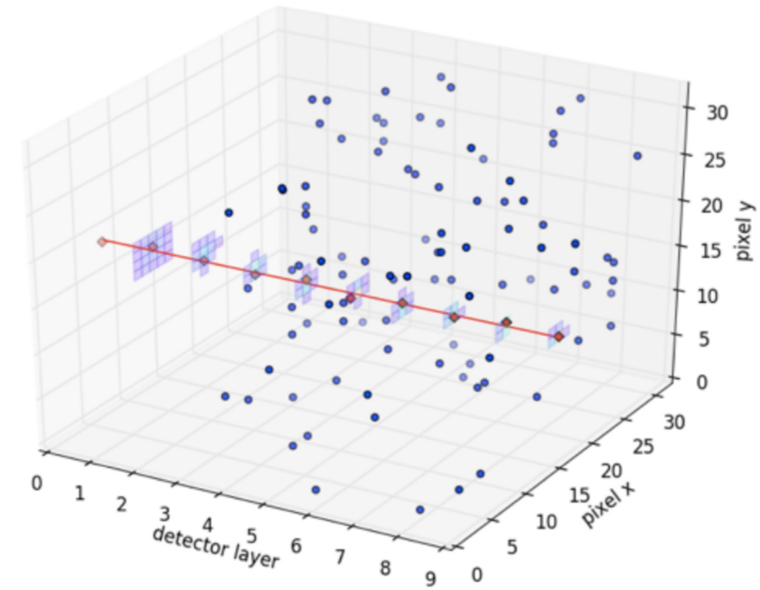
LSTM ≡ Kalman Filter

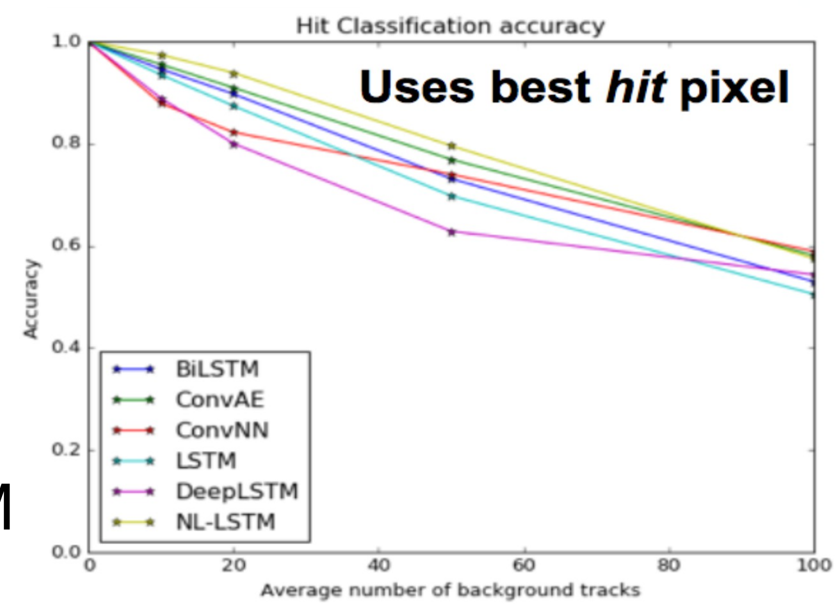


Seeded Pattern Recognition Insights

- For a simplified track models, predicting the track pattern from the seed works
  - In 2D and 3D
  - With some level of noise
  - With other tracks present
  - On layers with increasing number of pixels

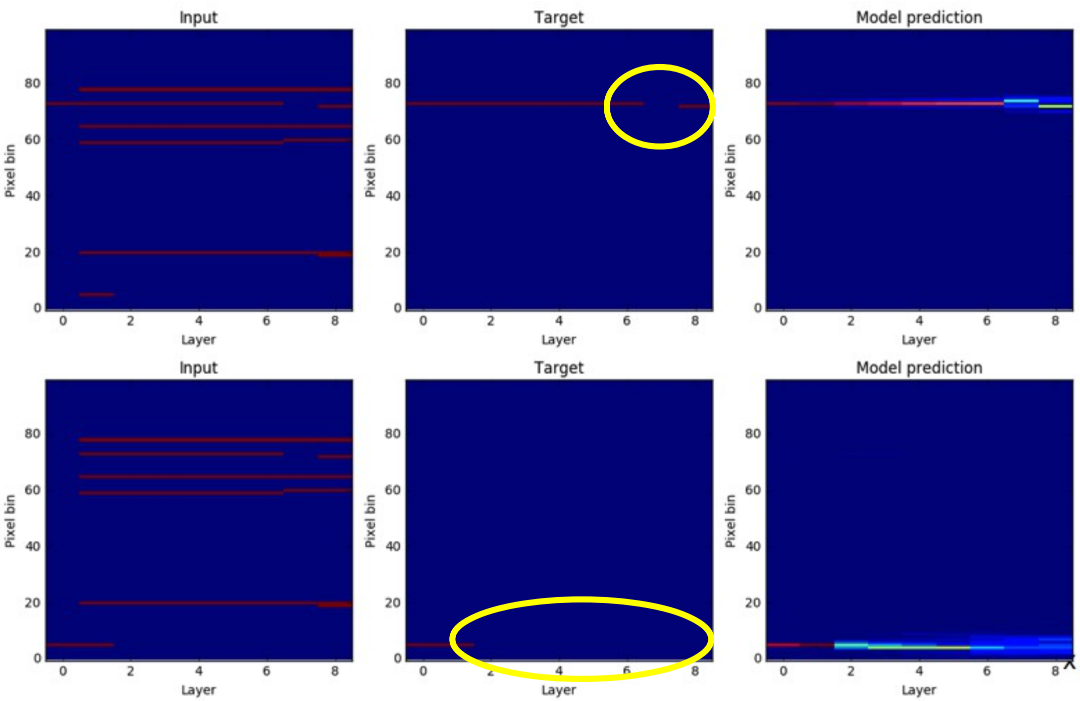
- Several other architectures tried
  - Convolutional neural nets (no LSTM)
  - Convolutional auto-encoder
  - Bi-directional LSTM
  - Prediction on next layer with LSTM



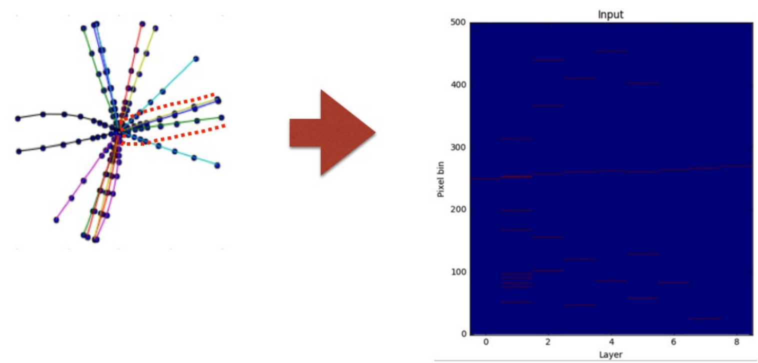


Tracking RAMP at CtD

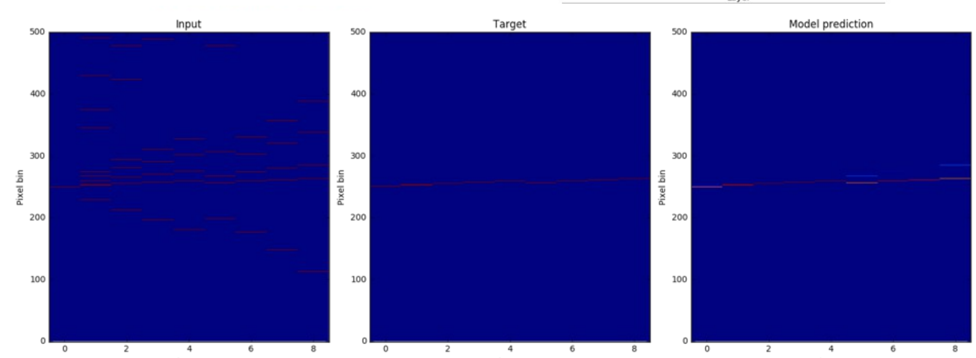
S. Farrell : Best solution in the Machine Learning category
https://indico.cern.ch/event/577003/contributions/2509988/



- Increased granularity in “road”
- LSTM for hit assignment
- 95% efficiency



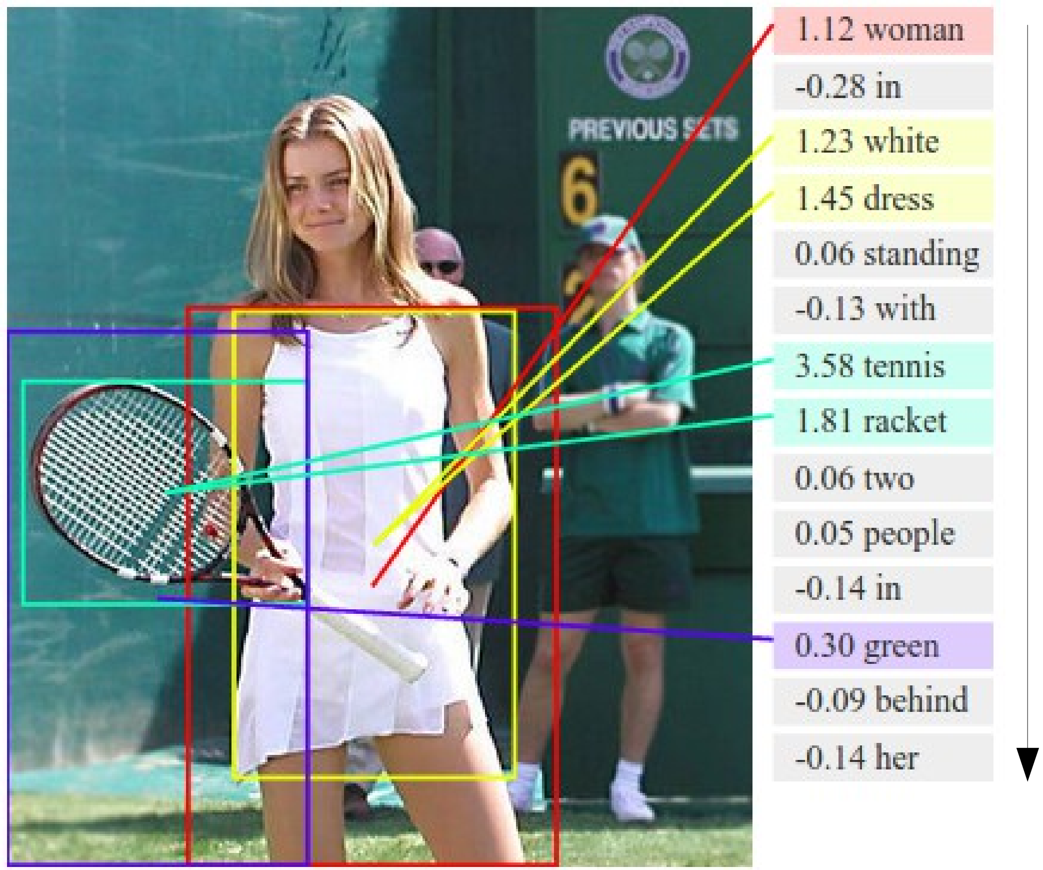
- Down-sampling layer to 100 bins
- LSTM for hit assignment
- 92% efficiency
- Robust to holes and missing hits



Track Parameters Measurement

https://heptrkx.github.io/

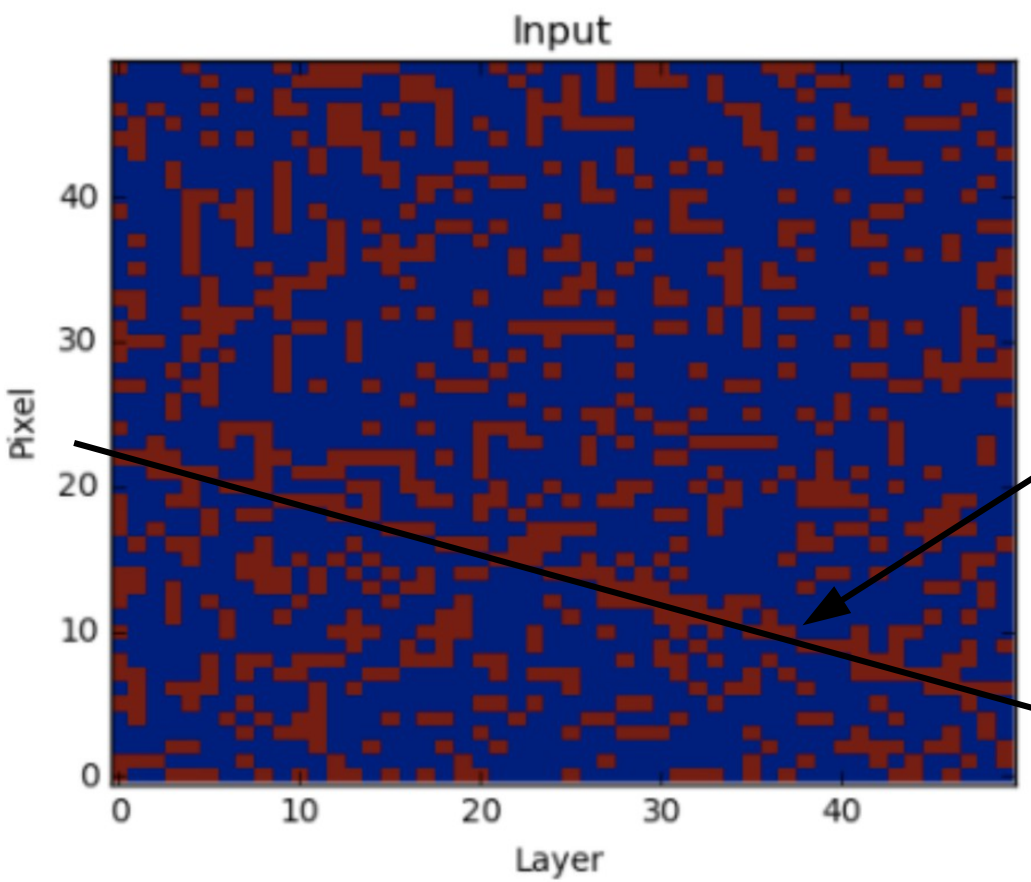
Scene Captioning



Karpathy, Fei-Fei, CVPR 2015

- Compose tracks explanation from image

Track Parameter Estimation

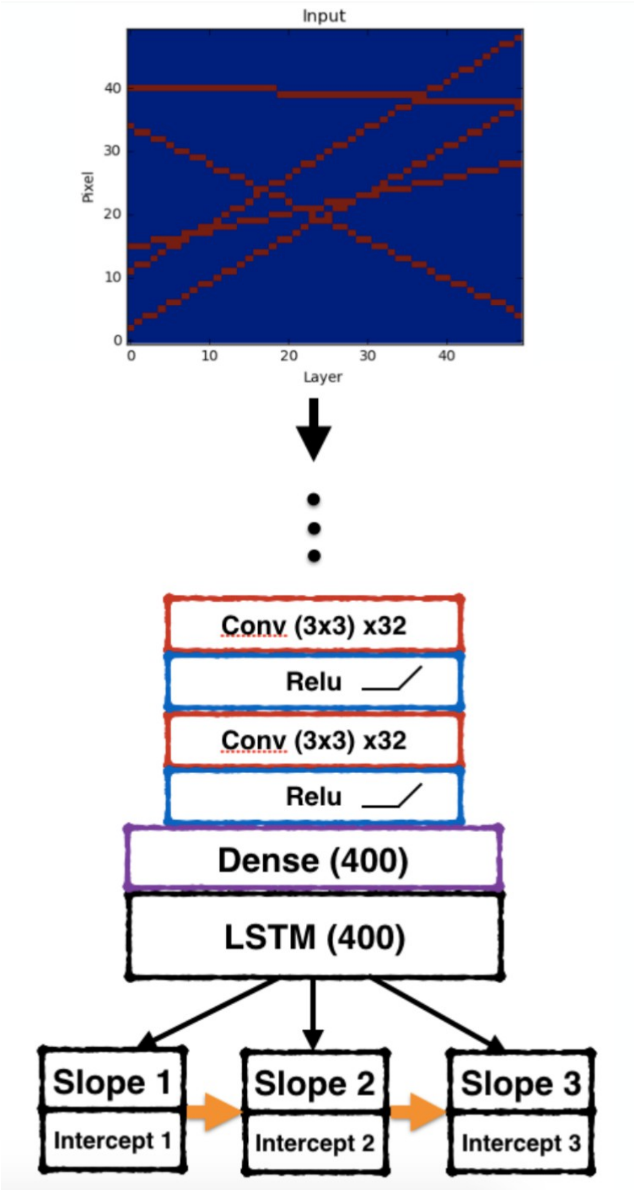


Try to predict the slope and intersect of this track

Multi-Track Prediction with LSTM

- Hit pattern from multiple track processed through convolutional layers

- LSTM Cell runs for as many tracks the model can predict.

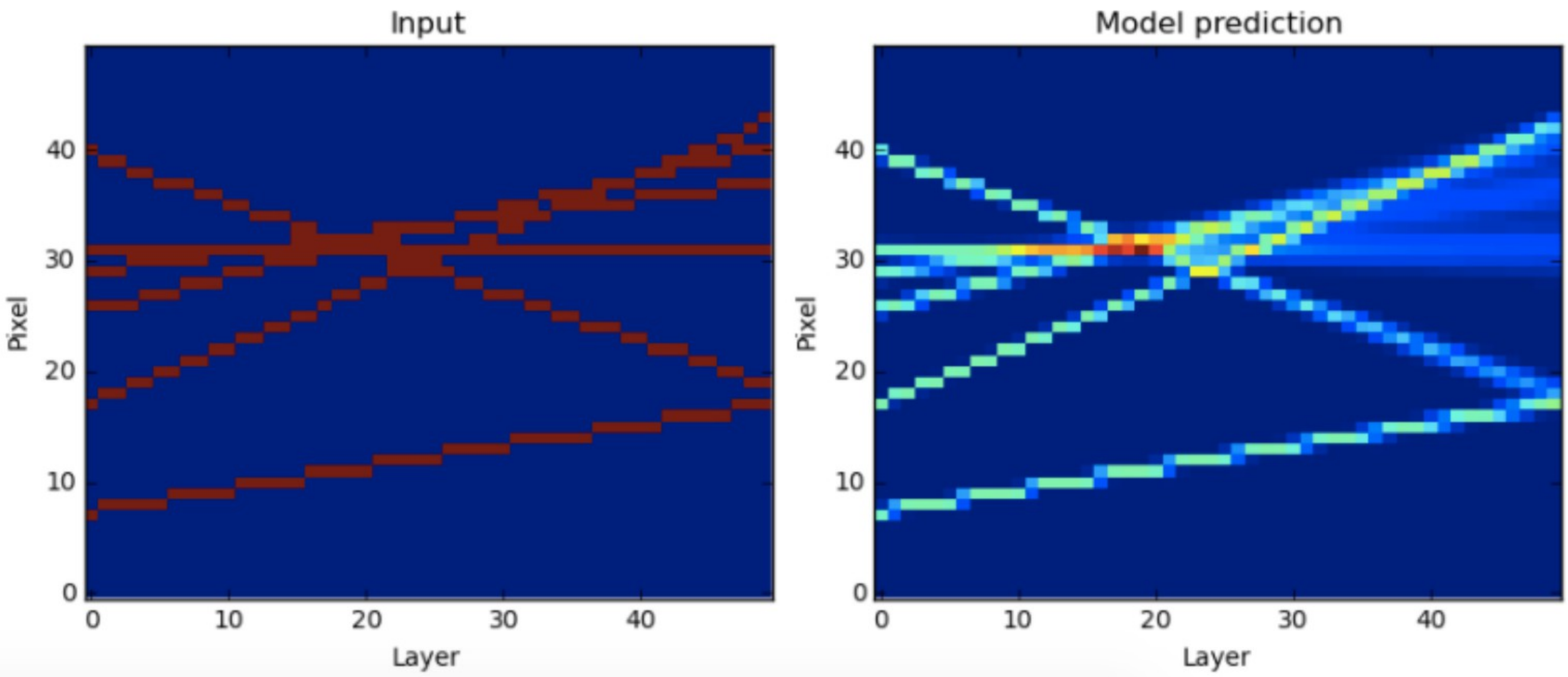


Prediction Track Covariance

Model is modified to predict a covariance matrix for which there is no ground truth, but is used with the modified loss function

$$L(\boldsymbol{x}, \boldsymbol{y}) = \log |\boldsymbol{\Sigma}| + (\boldsymbol{y} - \boldsymbol{f}(\boldsymbol{x}))^T \boldsymbol{\Sigma}^{-1} (\boldsymbol{y} - \boldsymbol{f}(\boldsymbol{x}))$$

Track Parameters Uncertainty



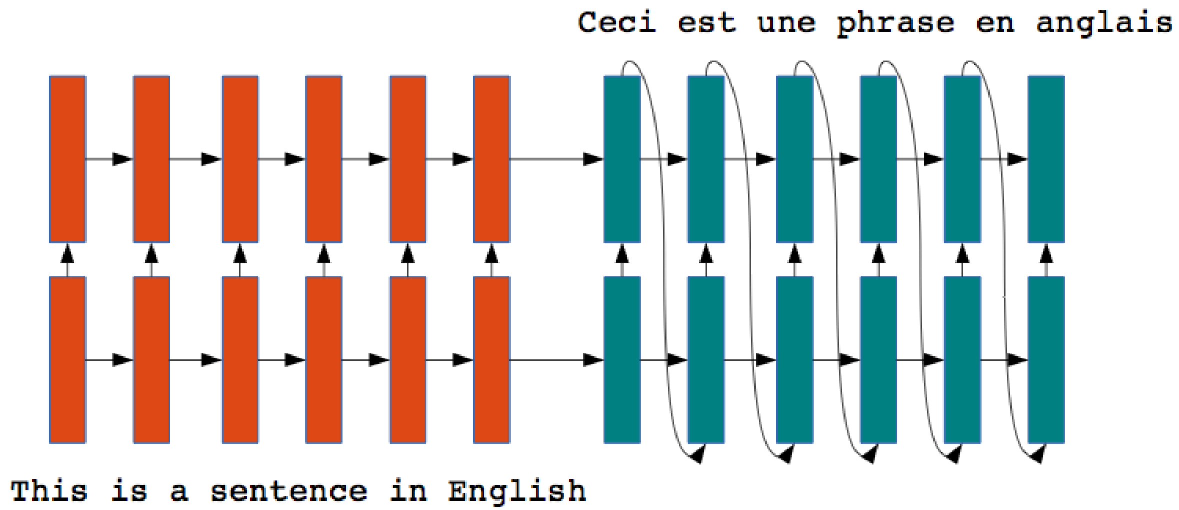
Representation of track slope, intersect and respective uncertainties

Pattern Recognition / Seeding

https://heptrkx.github.io/

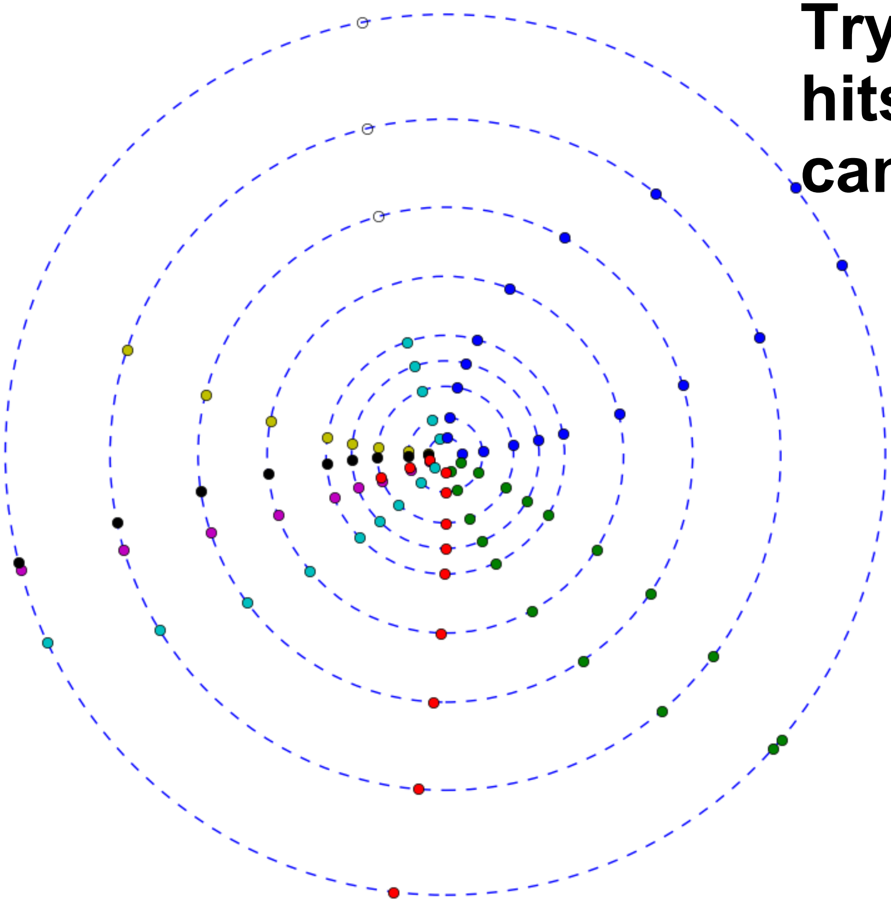
Text Translation

- [Sutskever et al. NIPS 2014]
  - Multiple layers of very large LSTM recurrent modules
  - English sentence is read in and encoded
  - French sentence is produced after the end of the English sentence
  - Accuracy is very close to state of the art.



→ From sequence of hits on layer to sequence of hits on track

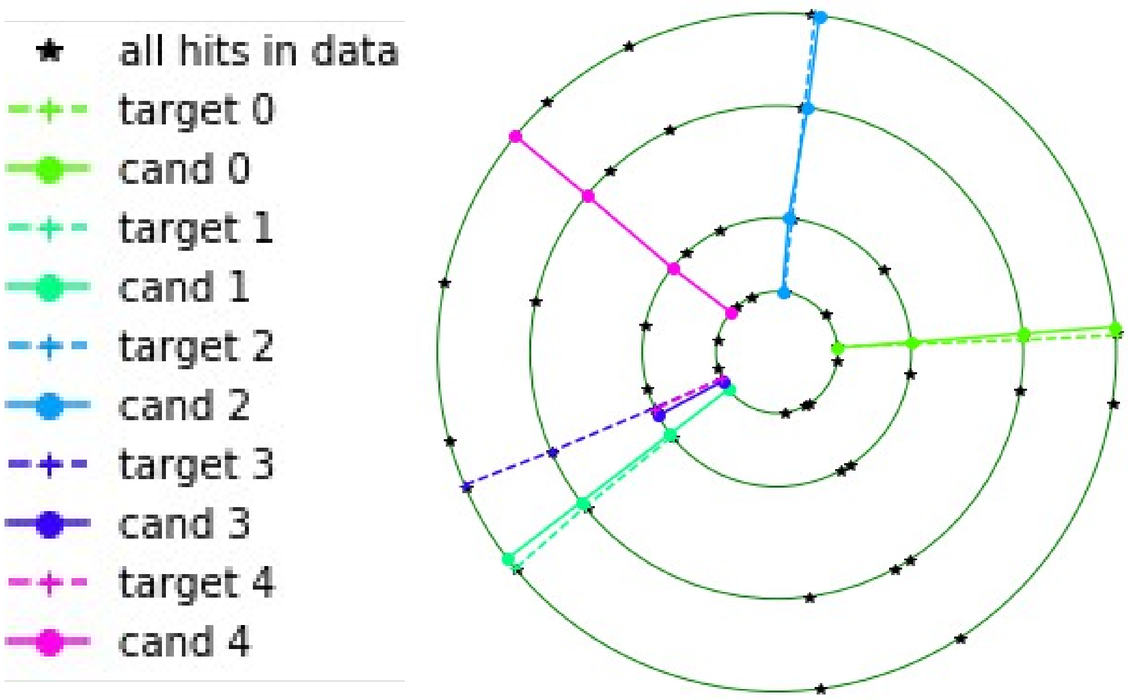
Pattern Recognition



Try to assemble hits into track candidates.

Pattern Recognition with LSTM

- Input sequence of hits per layers (one sequence per layer)
  - One LSTM cell per layer
- Output sequence of hits per candidates
  - Final LSTM runs for as many candidates the model can predict



- Still work in progress
- Restricted to 4 layers (with seeding in mind)
- Work to some extend