CERN TrackML accuracy

Deep Learning Approach

Liam Finnie & Nicole Finnie

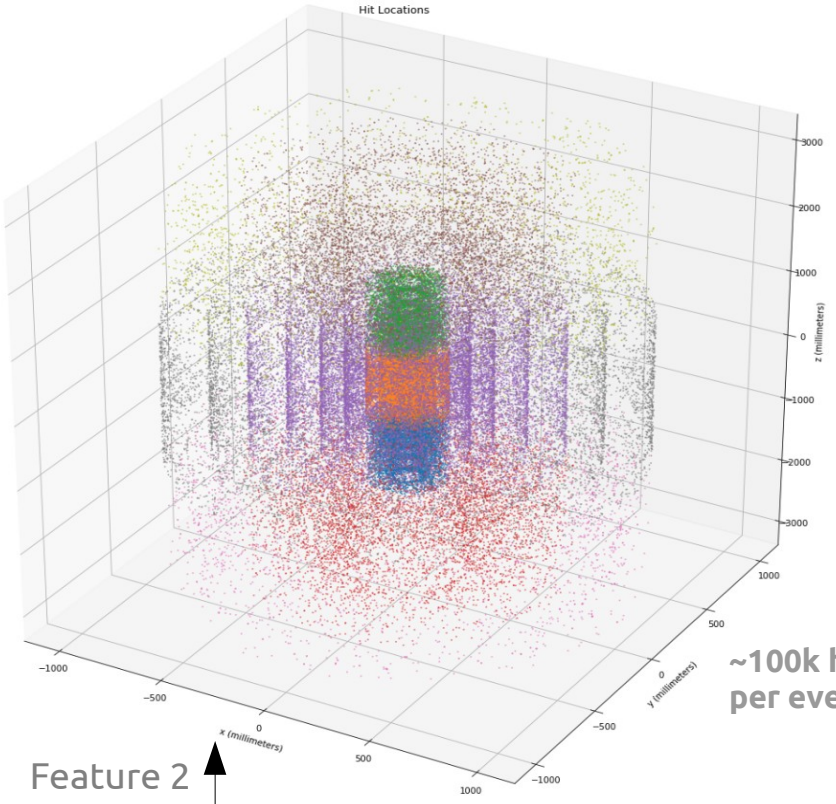
IBM Germany R&D
Bosch Centre for AI

Jul 2nd, 2019

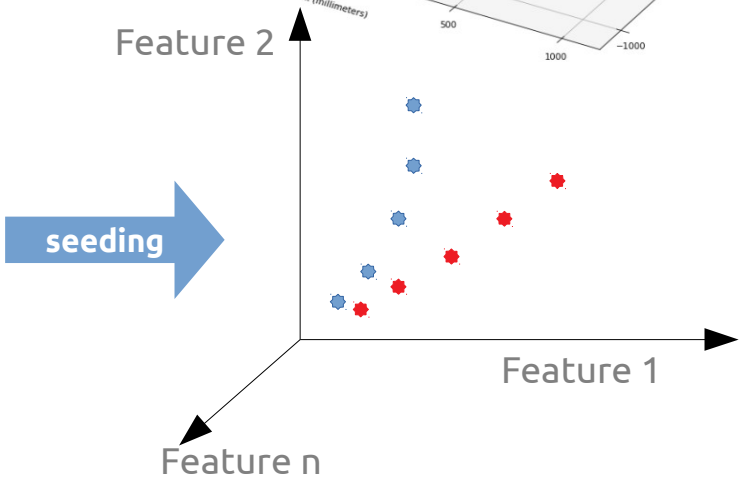
Correlation != Causation

Deep Learning vs. Laws of Physics

Solution Pipeline



~100k hits (~10k tracks) per event



seeding

inference

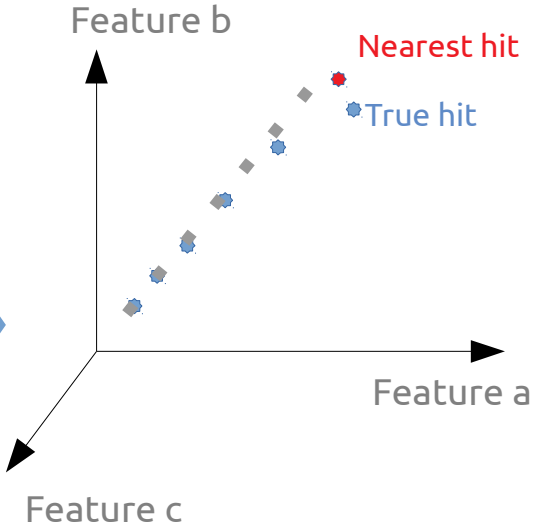
Bi-LSTM

LSTM

LSTM

LSTM

fitting

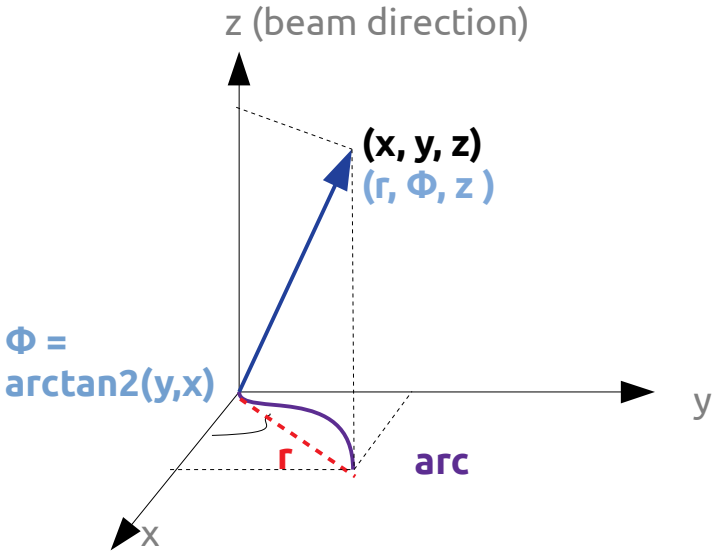


Track seeding
(clustering)

Inference & Ensembling

Track fitting
(k-nearest neighbour)

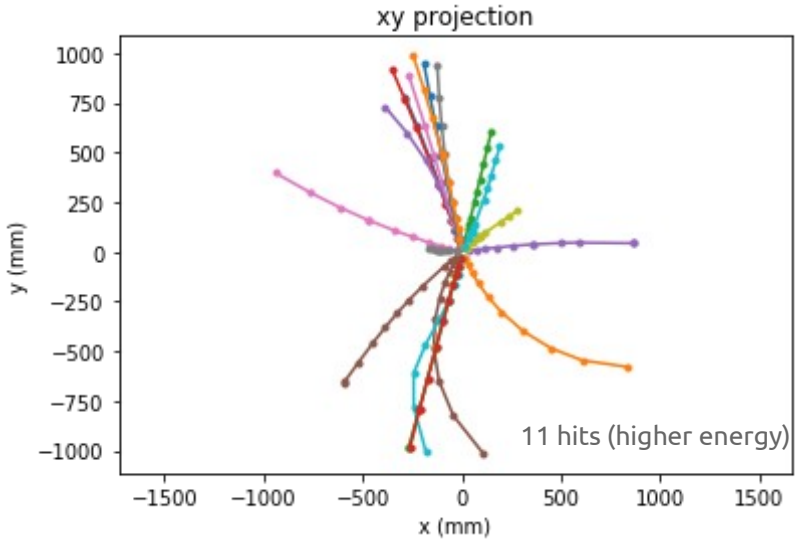
Feature Engineering... for people who don't know physics :D



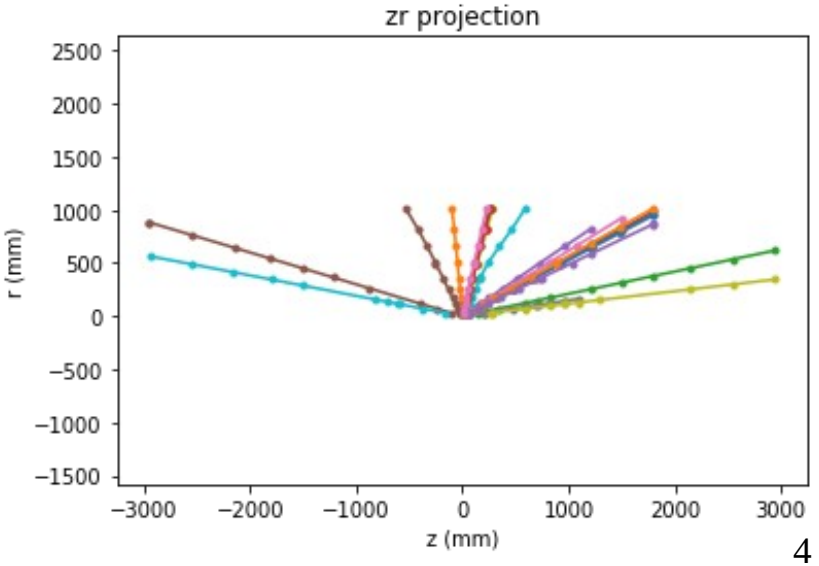
Data we use: (x, y, z) coordinates of hits

For clustering: sin(Φ), cos(Φ), z/arc
(new feature: generate possible arcs using train data)

For LSTM: Φ, r, z, z/r

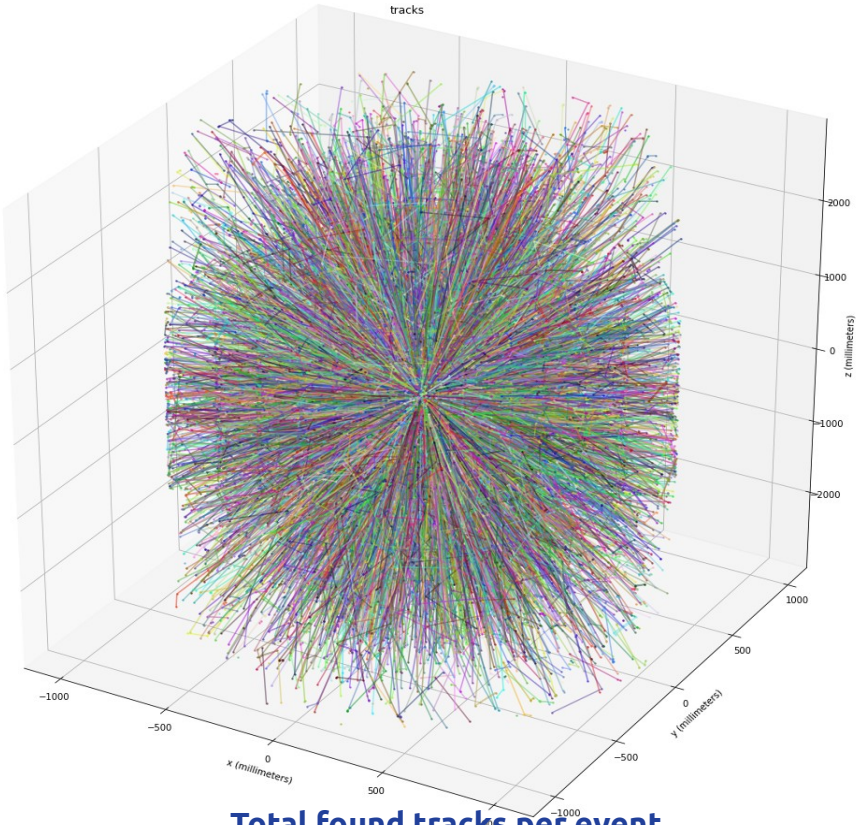


project



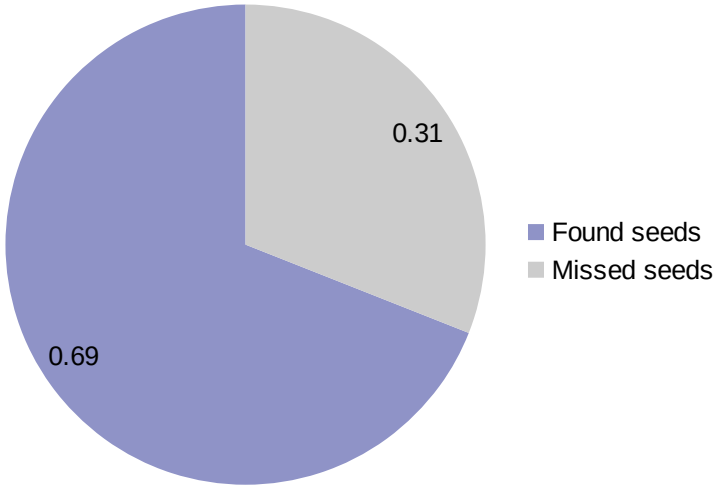
Cartesian -> Polar coordinates: **easier for LSTM to learn**

Track Seeding

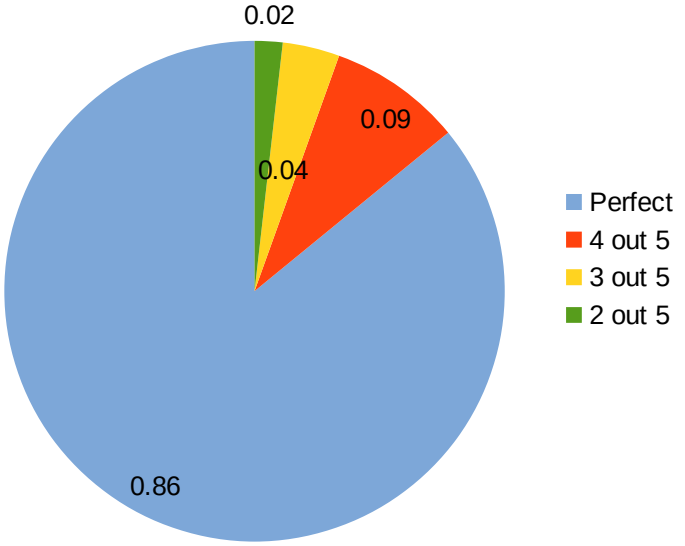


- DBSCAN clustering + outlier removal
- **Seeds: first 5 hits** of each found track
- 69% total seeds per event found within 1 min
- Seed quality: **85.8%** perfect

Total found tracks per event



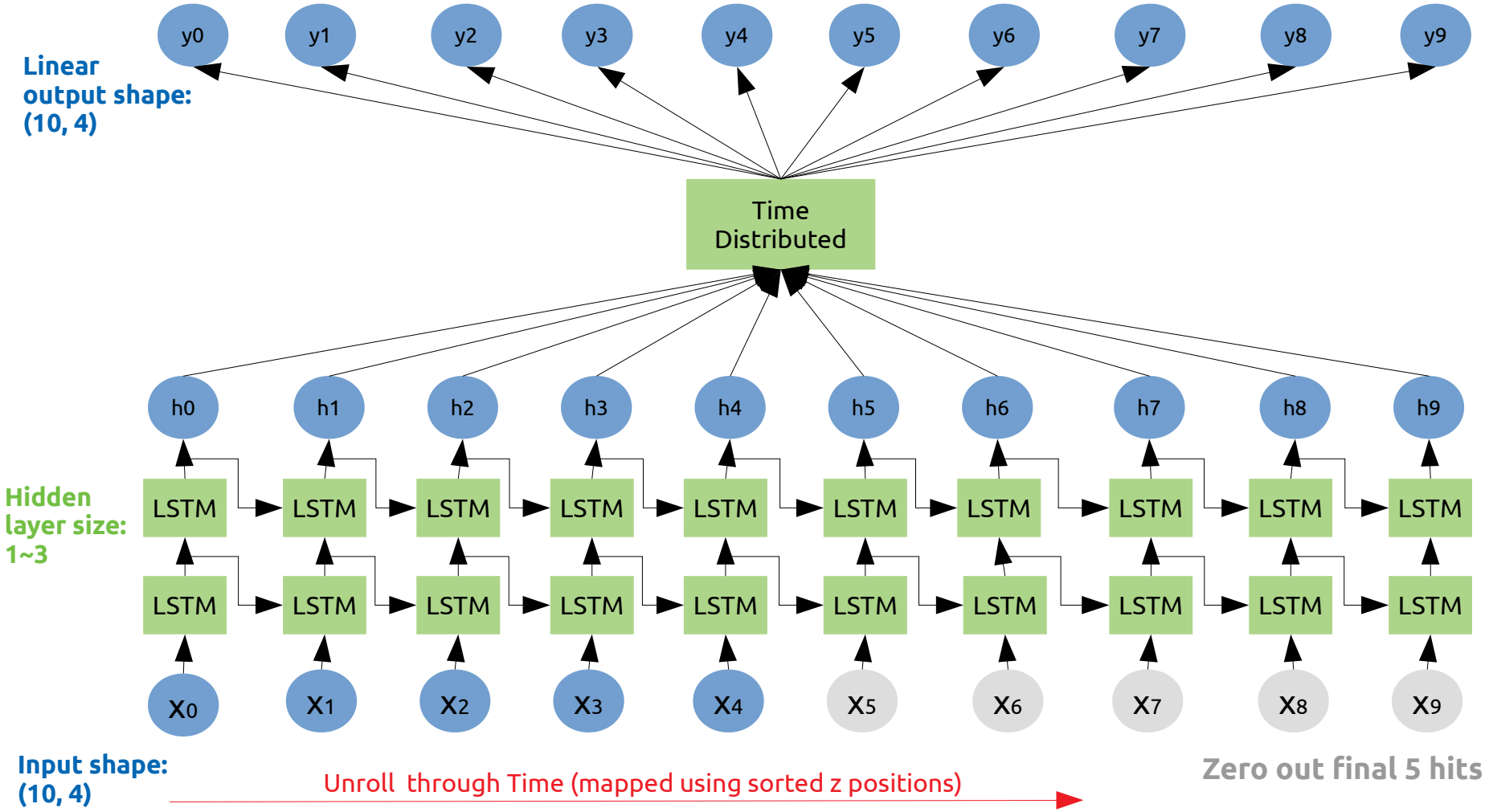
Seed quality



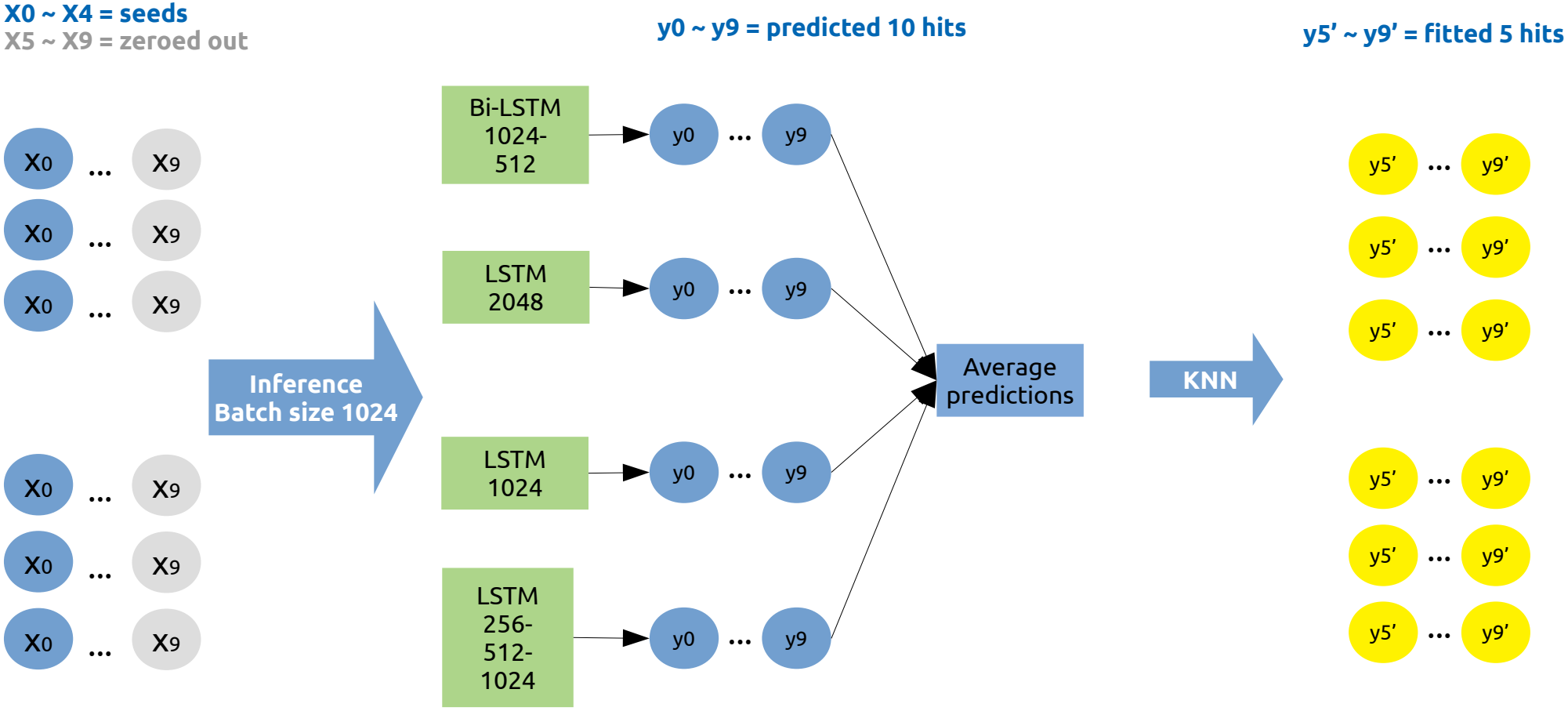
This can totally be replaced by CERN's superior seeding algorithm.....

Track Fitting – LSTM training

Train: 8300 events (limited on x, y, z > 0, track length >= 10, ~6.5% of total tracks)
Validation: 500 events
Test: 100 events
Input/Output: Φ, r, z, z/r



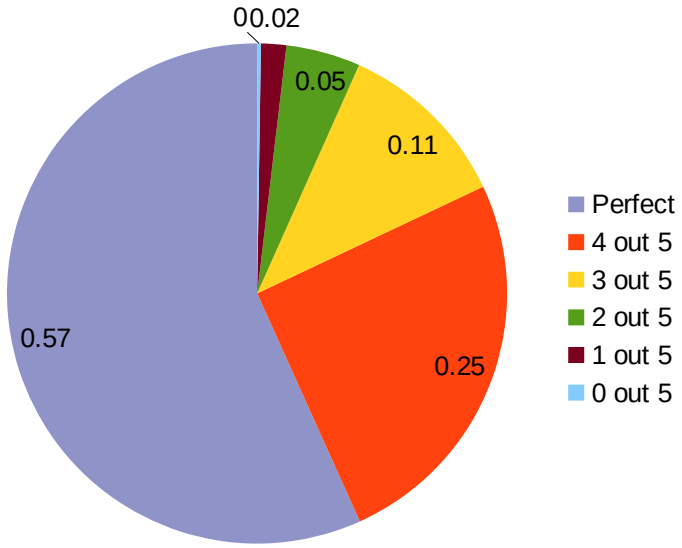
Track Fitting – LSTM inference



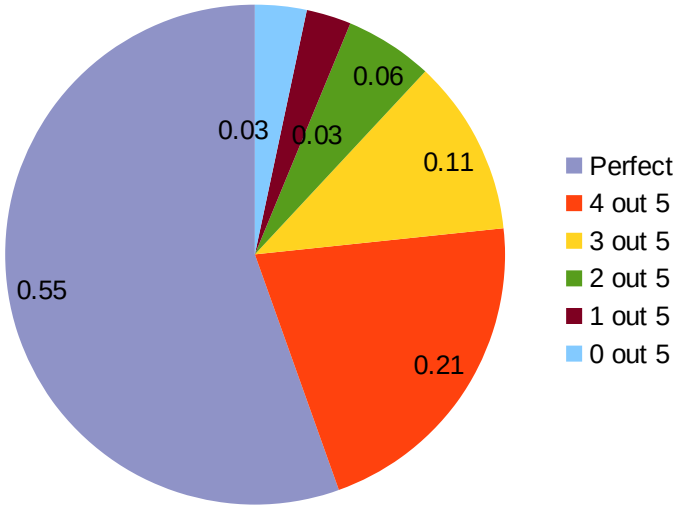
Empirical Results (tf.keras)

- Over 100 test events - 63810 true tracks, 43820 seeded tracks in total
- 2015 Macbook Pro - Intel i7-4870HQ 2.5 GHz – ~~crappy~~ **CPU only**
- 2018 Dell 9570 - Intel i7-8750H 2.2 GHz - **GTX 1050 Ti GPU**

	Seeding	Bi-LSTM	LSTM 2048	LSTM 1024	LSTM-256-512-1024	Fitting	Total
Average time per 10k tracks (s) - Mac	60 (per event)	**49**	**42**	**12**	**21**	8	192
Average time per 10k tracks (s) - Dell	30 (per event)	**3**	**2**	**1**	**2**	6	44

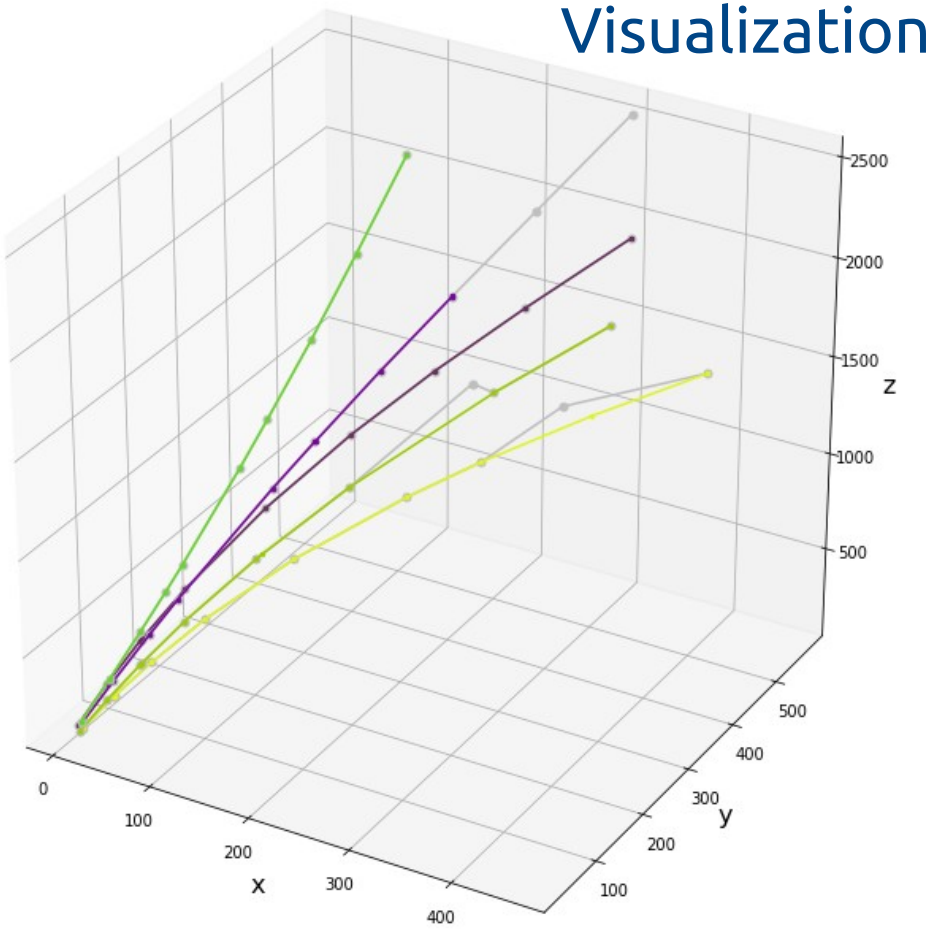


Accuracy of predicted hits (with true seeds)



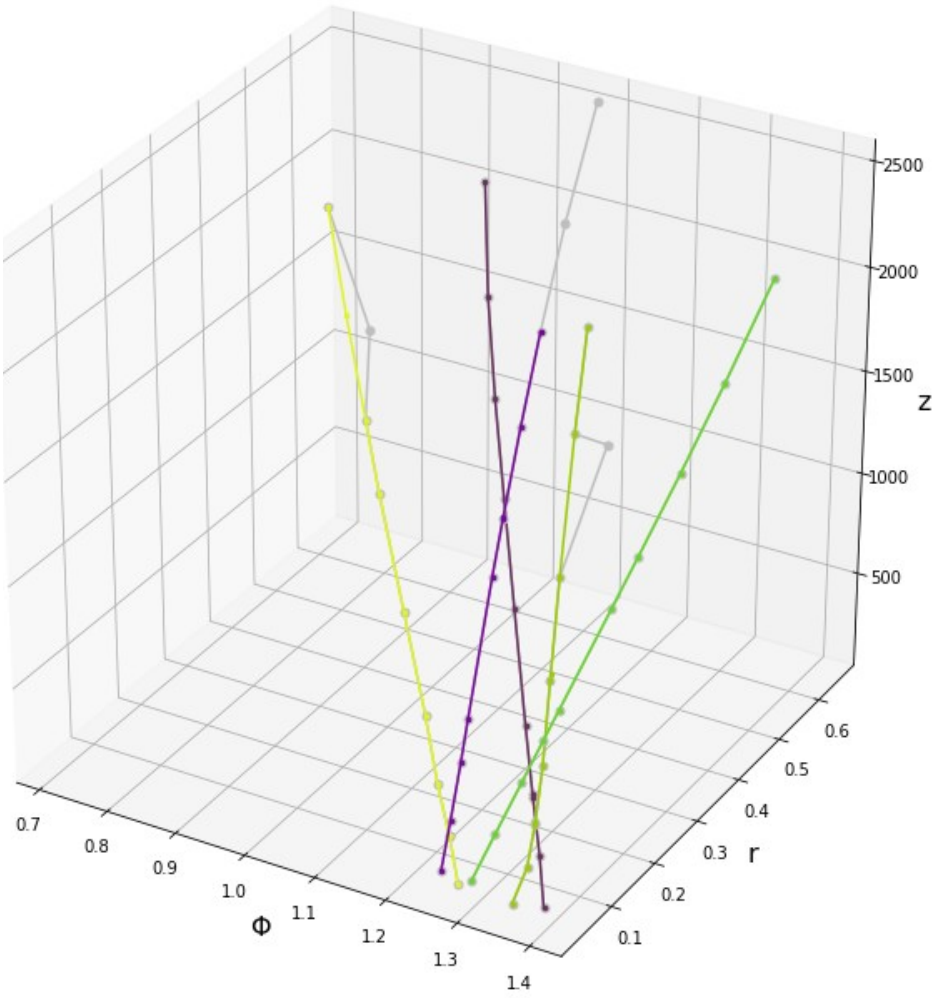
Accuracy of predicted hits (with clustered seeds)

Visualization after fitting



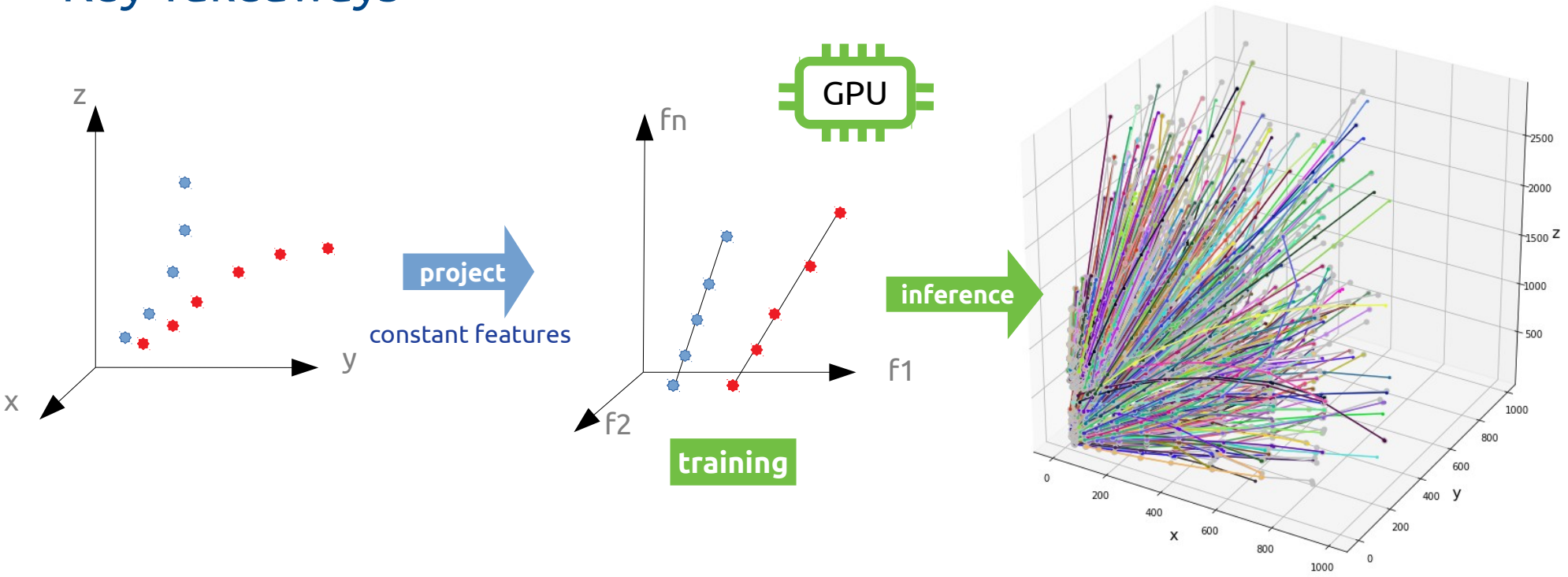
Cartesian coordinates

--- predicted (grey)
--- ground truth (colour)



Polar coordinates

Key Takeaways



- Easy for LSTM to learn **constant features** of a track
- Low **inference time** with **data parallelization (batch)** has its potential to tackle the bottleneck
- **Room for improvement**
  - Train different models with different types of tracks (low energy/curvy, irregular)
  - Train models with tracks in all Cartesian dimensions + directions in cell data
  - Replace K-nearest neighbour for fitting

Our source code: https://github.com/jliamfinnie/kaggle-trackml