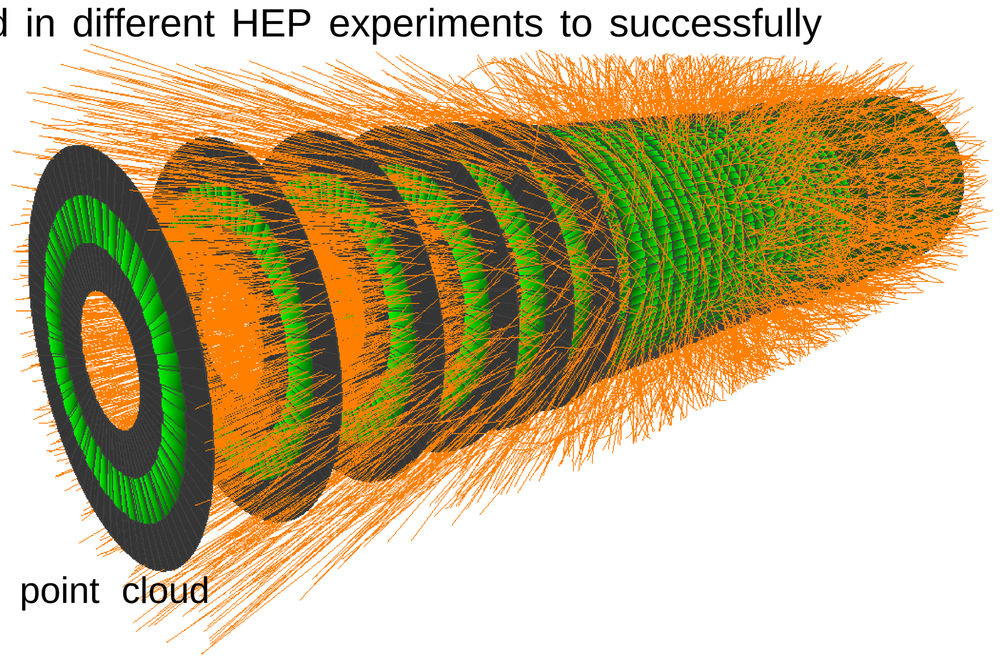
Tracking Aware Metric Learning for Particle Reconstruction

Sabrina Amrouche, T.Golling (UniGe)
N.Calace, M.Kiehn, A.Salzburger (CERN)

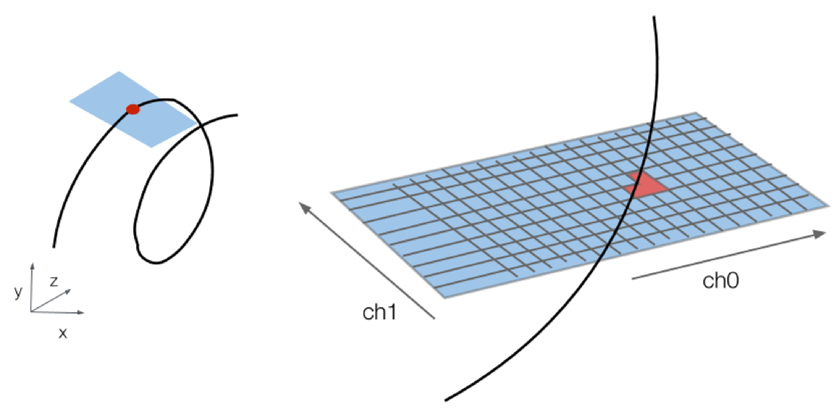
The problem

- Current particle tracking techniques build an excessive amount of **combinatorics**, out of which, only a small fraction represent valid trajectories.
- The **CPU** time spent in ineffectual combinatorics can be invested in increasing the amount of analyzed data, thus increasing the **discovery potential.**
- Novel algorithms for finding charged particle tracks have to optimize speed and accuracy: Find the same trajectories, **faster**.
- A number of studies are currently conduced in different HEP experiments to successfully implement **ML based tracking.**



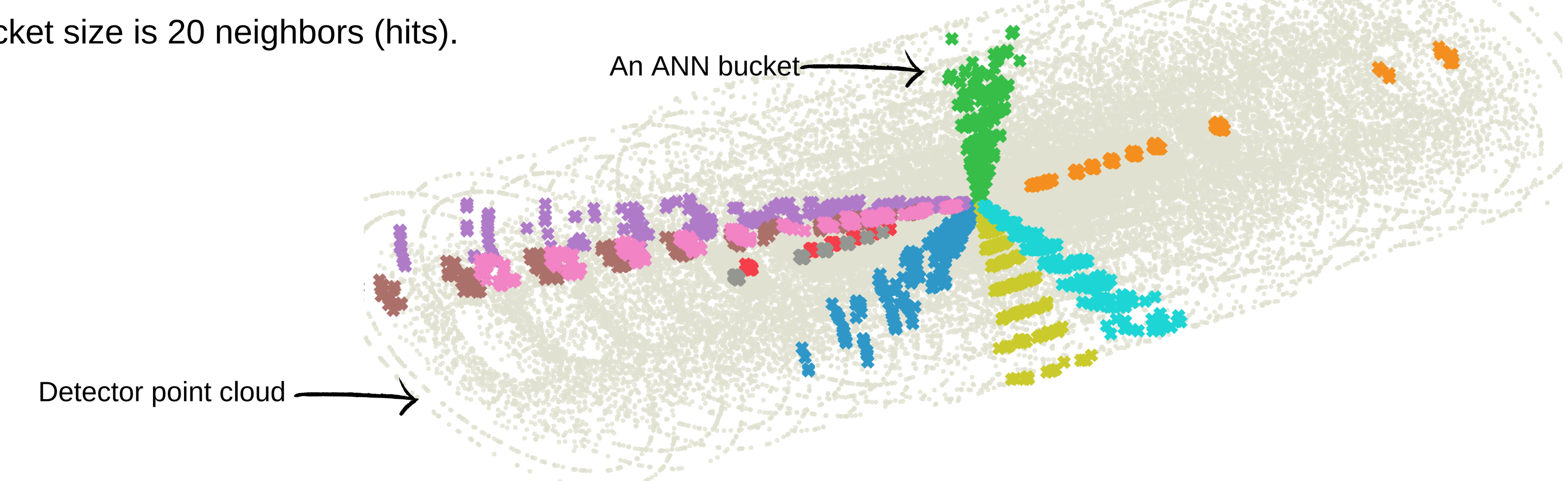
The data

- The simulation of HEP collision produces a point cloud dataset : essentially **3D points**.
- The dataset released for the **TrackML** challenge [1] emulates the future High Luminosity LHC. In HEP experiments, increasing the luminosity is crucial to increase the discovery potential.
- A charged particle produces 10 points (hits) on average and a single event creates **10K** particles.
- Geometry related features are also available per hit but we choose to use only 3D information for **robustness**.

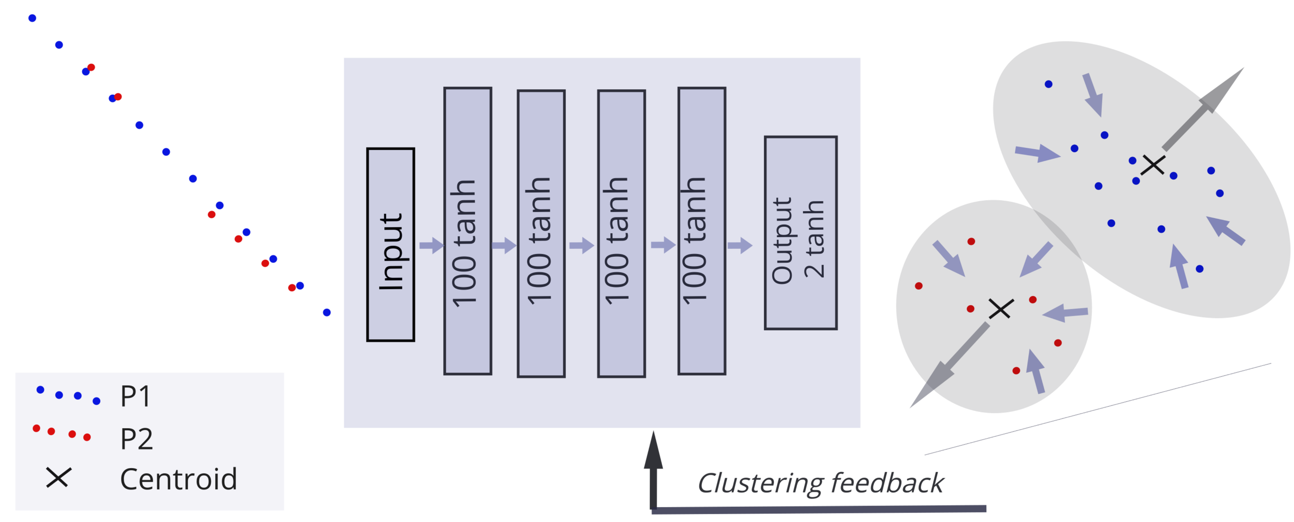


Similarity Search

- **Fast Similarity Search** techniques have been demonstrated to provide high quality tracking bins[2][3].
- For a given data point (query), the idea is to retrieve a set of neighbors that **share similar properties**.
- We use the angular distance between points to construct the search index.
- In practice, an **Approximate Nearest Neighbors** structure is implemented using Annoy[4] for CPU and FAISS [4] for GPU.
- The ML based tracking is then performed inside the returned neighbors set (bucket).
- A typical bucket size is 20 neighbors (hits).



TrackNet



The TrackNet approach : **Maps** a bucket from the ANN search to a new feature space where particles are **maximally separated**.
The model is designed to enable an *intuitive clustering* on the output.

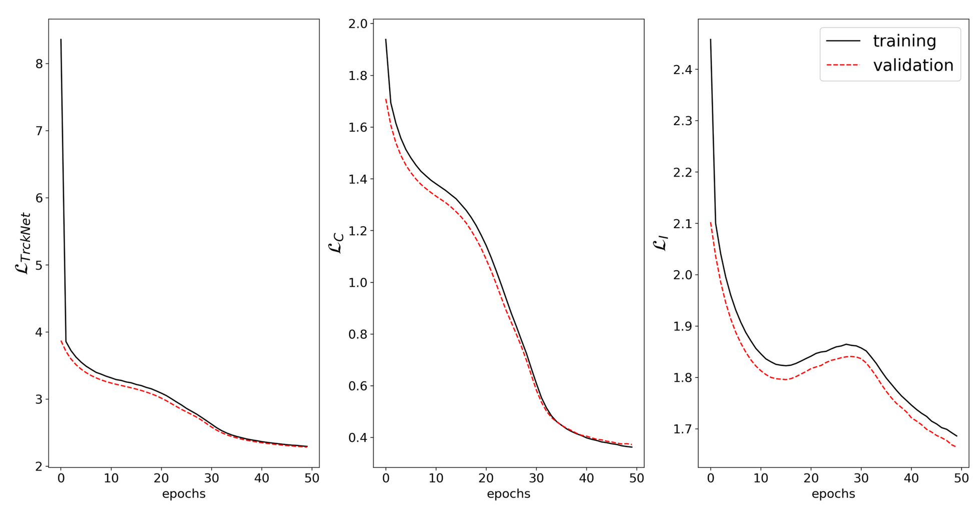
The Loss Function

The Loss function encodes three actions :

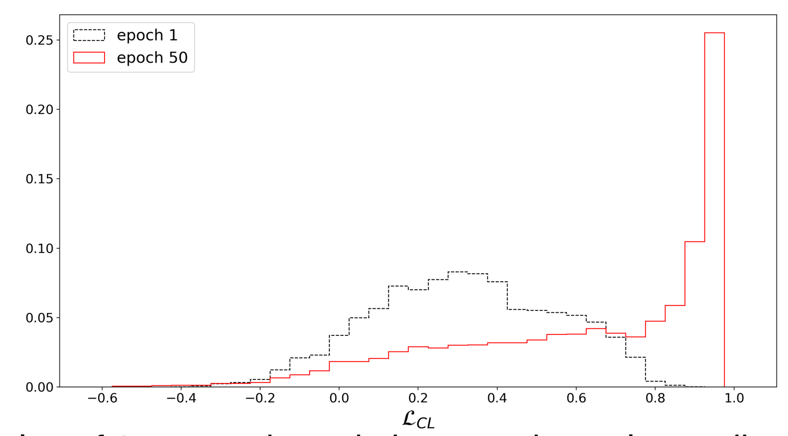
- A **pushing** action $\mathcal{L}_I$
- A **pulling** action $\mathcal{L}_C$
- A **clustering** action $\mathcal{L}_{CL}$

$$\mathcal{L}_{TrackNet} = (\alpha\mathcal{L}_C + \beta\mathcal{L}_I + \gamma\mathcal{L}_{CL})^{\zeta}$$

$$\mathcal{L}_C = \sum_{i=0}^{K} S(c_i) \ ; \ \mathcal{L}_I = \frac{1}{S(\mu_{[1..k]})} \ ; \ \mathcal{L}_{CL} = \sum_{i=0}^{n} \frac{b_i - a_i}{\max(a_i, b_i)}$$

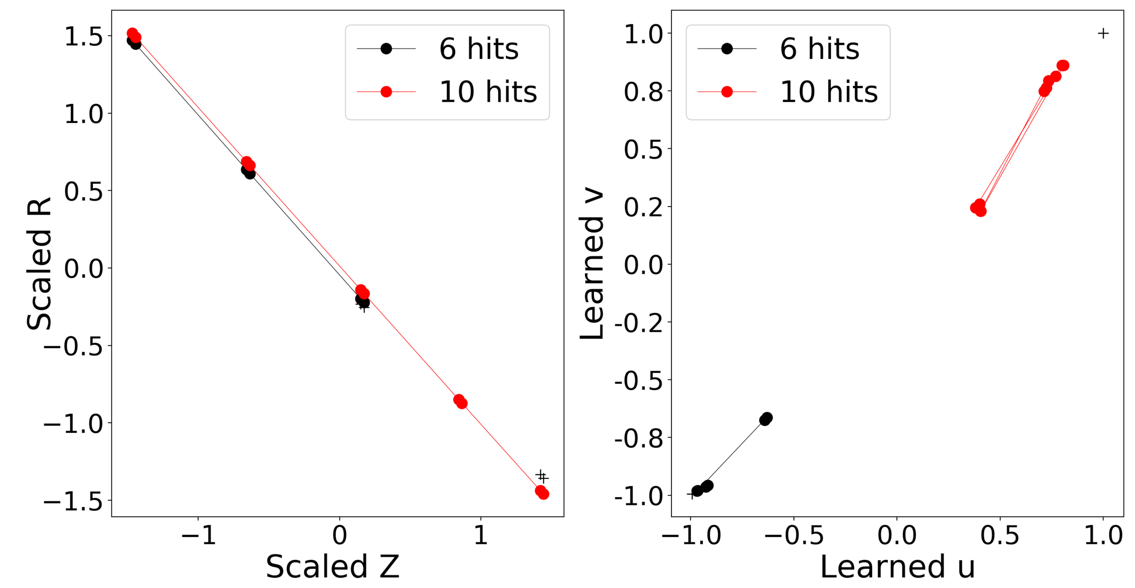


Gradual decrease of the model loss through the decrease of individual terms. Well separated particle **clusters** are formed.



An $\mathcal{L}_{CL}$ value of 1 means the ratio between intra cluster distances and external distance is maximal, i.e. **particles are well separated**.

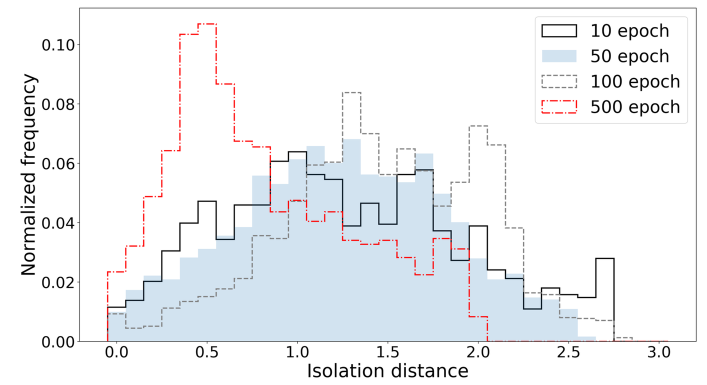
A Tracking Space



Example of a bucket in the original feature space (left) and its mapping through TrackNet (right). The selected bucket contains two particle trajectories of 10 hits (red) and 6 hits (black). **In the mapped space, the two trajectories are well separated.** The distance between the centroid of the largest trajectory and the nearest hit (outside the trajectory) is called *isolation* distance.

Performances

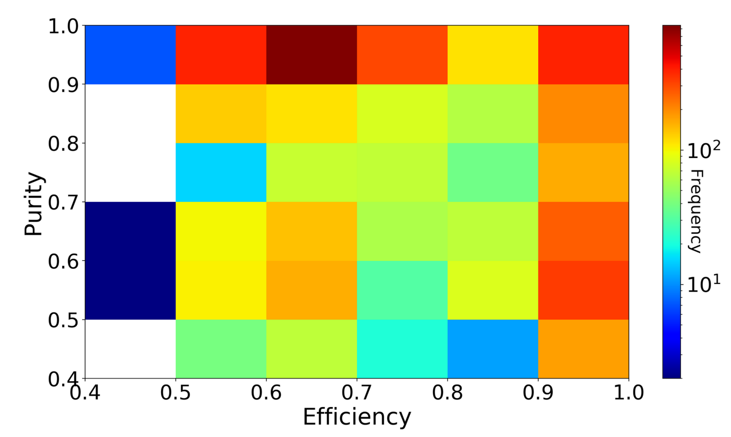
The model task is to maximize the isolation distance. This distance is computed on every model output where the largest trajectory has at least 4 hits. At 500 epochs the model performance is decreased (less separation). We use early stopping to select the appropriate number of epochs.



Once the best model configuration is selected through a grid serch, an **Agglomerative Clustering** (AC) is performed in the new feature space.
The AC merges close-by hits until the *isolation distance* is reached. Clusters with 3 hits or less are discarded. This filters noise hits early on.

Do the obtained clusters contain full particles ? Without outlier noise hits?

- Cluster Efficiency : Proportion of a particle in a cluster over the full particle.
- Cluster Purity: Proportion of a particle in a cluster over the cluster size.



Improvements currently tested: A dynamic physics based clustering algorithm.

References

[1] S. Amrouche, L. Basara, P. Calafiura, V. Estrade, S. Farrell, D. R. Ferreira, L. Finnie, N. Finnie, C. Germain, V. V. Gligorov, et al., "The tracking machine learning challenge: Accuracy phase," in The NeurIPS'18 Competition, pp. 231–264, Springer, 2020.
[2] S. Amrouche, T. Golling, M. Kiehn, C. Plant, and A. Salzburger, "Similarity hashing for charged particle tracking," in 2019 IEEE International Conference on Big Data (Big Data), pp. 1595–1600, IEEE, 2019.
[3] S. Amrouche, M. Kiehn, T. Golling, and A. Salzburger, "Hashing and metric learning for charged particle tracking,"
[4] M. Aumüller, E. Bernhardsson, and A. Faithfull, "Ann-benchmarks: A benchmarking tool for approximate nearest neighbor algorithms," in International Conference on Similarity Search and Applications, pp. 34–49, Springer, 2017.

UNIVERSITÉ DE GENÈVE
CERN