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 conducted 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.
  - The ML based tracking is then performed inside the returned neighbors set (bucket).
  - A typical bucket size is 20 neighbors (hits).

The Loss Function

The loss function encodes three actions:

- A pushing action $L_C$: $L_C = \frac{1}{2} \sum_{i,j \in C \setminus \{i,j\}} \left( \frac{1}{\beta_{ij}} - 1 \right)$
- A pulling action $L_I$: $L_I = \sum_{i \in C} \frac{1}{\min(n_i, n_j)}$
- A clustering action $L_C$: $L_C = \frac{1}{2} \sum_{i \in C} \left( \frac{1}{\beta_{ij}} - 1 \right)$

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.

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 (loss separation). We use early stopping to select the appropriate number of epochs.

Once the best model configuration is selected through a grid search, 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.

References


