ML4Jets 2023, Hamburg, 6/11/23

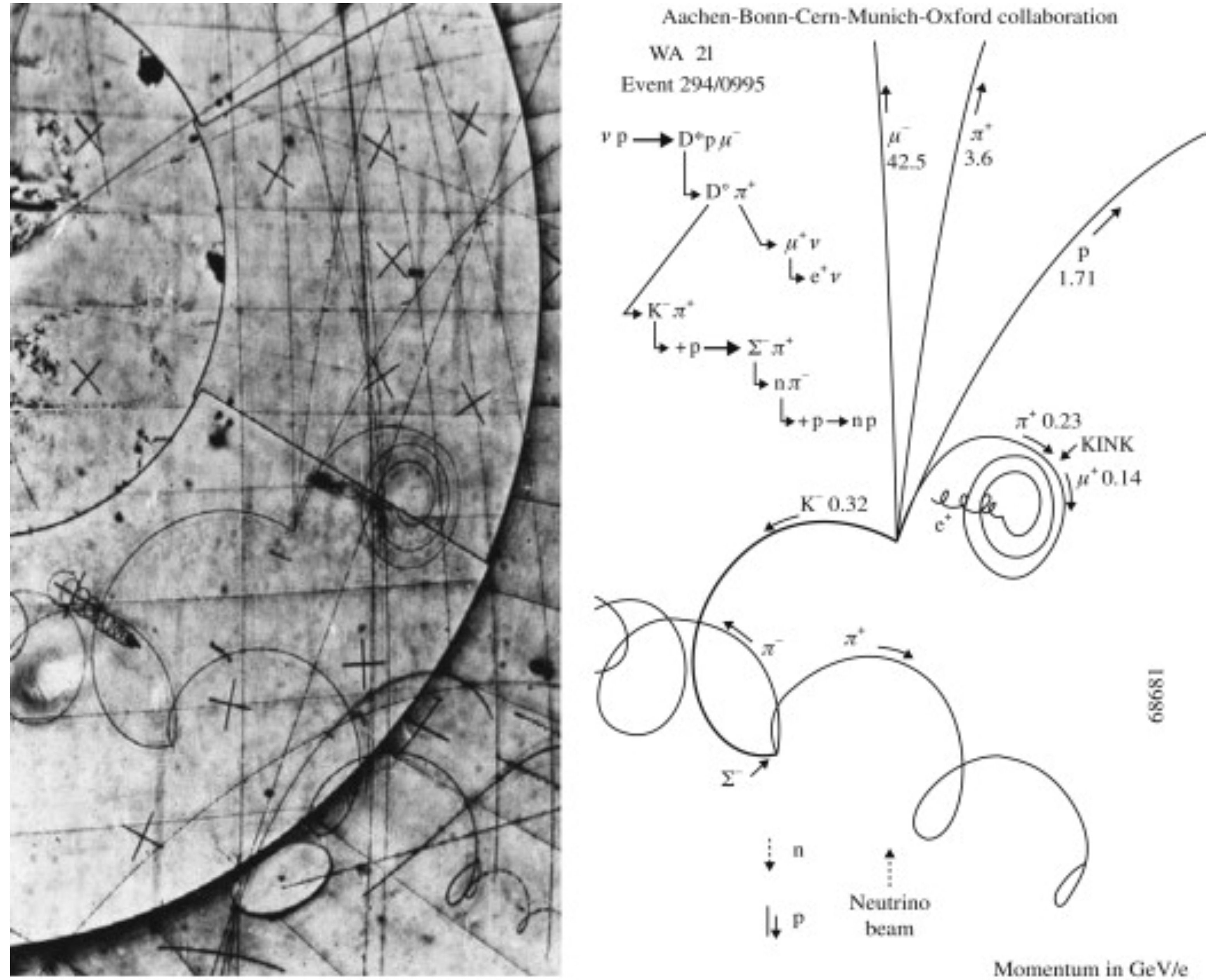
Tracking with Transformers and U-net Models

Zef Wolffs

Antonio Ferrer, Jose Martin, Jose Salt, Matouš Vozák, Nadezhda Dobreva, Roberto Ruiz de Austri Bazan, Sascha Caron, Uraz Odyurt, Yue Zhao

Introduction

Big European bubble chamber at CERN (operation in 1970s) [1]

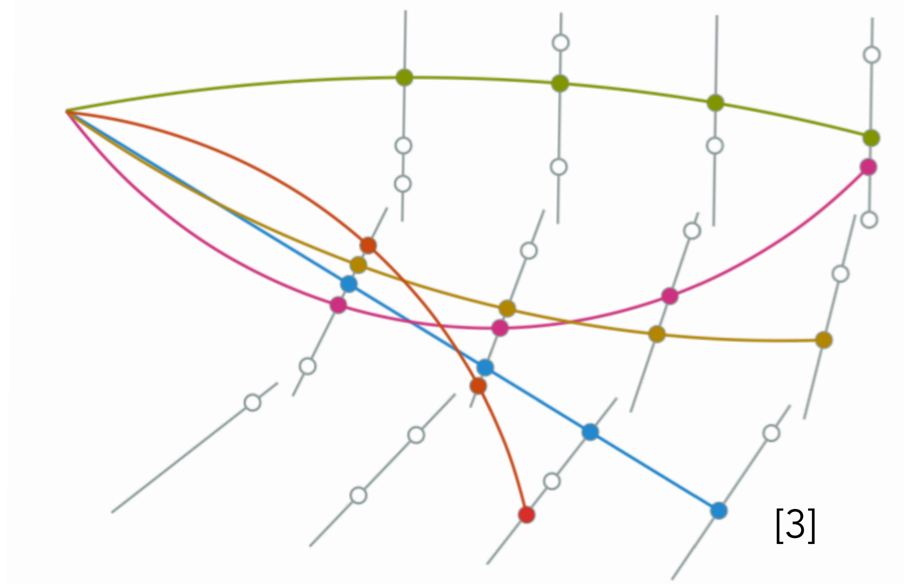


Tracking by meticulously following visible tracks [2]

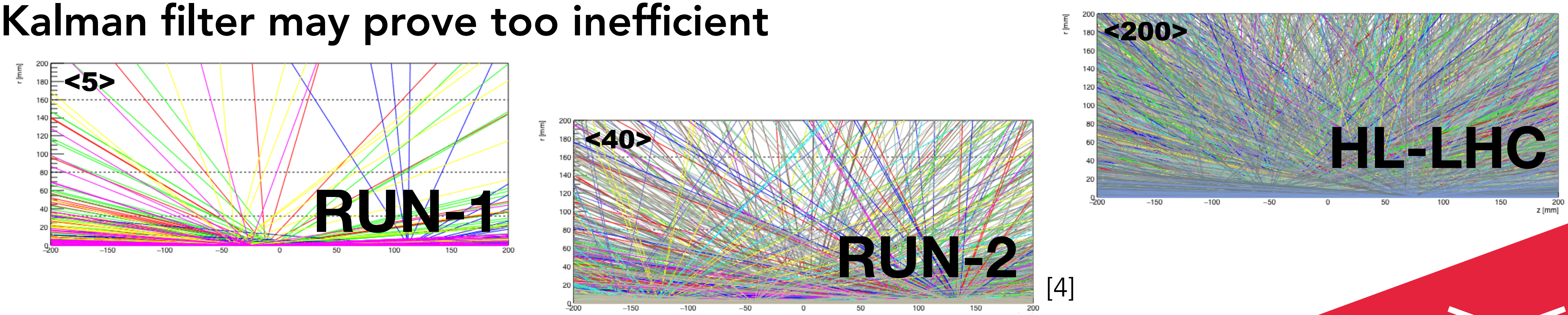
[1] Arpad, Horvath. "Big European Bubble Chamber" 2005. *Wikimedia.org*. Wikimedia Commons. Web. 4 November 2023.
[2] Ahmed, Syed Naeem. Physics and engineering of radiation detection. Academic Press, 2007.

Introduction

- With modern **layered detector design** and **electronic readout systems** the situation looks rather different


[3]

- However, **constructing tracks by hand** has for a while been **completely unfeasible**, and so computational techniques such as the **Kalman filter** have been adopted
- But with **upcoming HL-LHC** even traditional computational techniques such as the **Kalman filter may prove too inefficient**

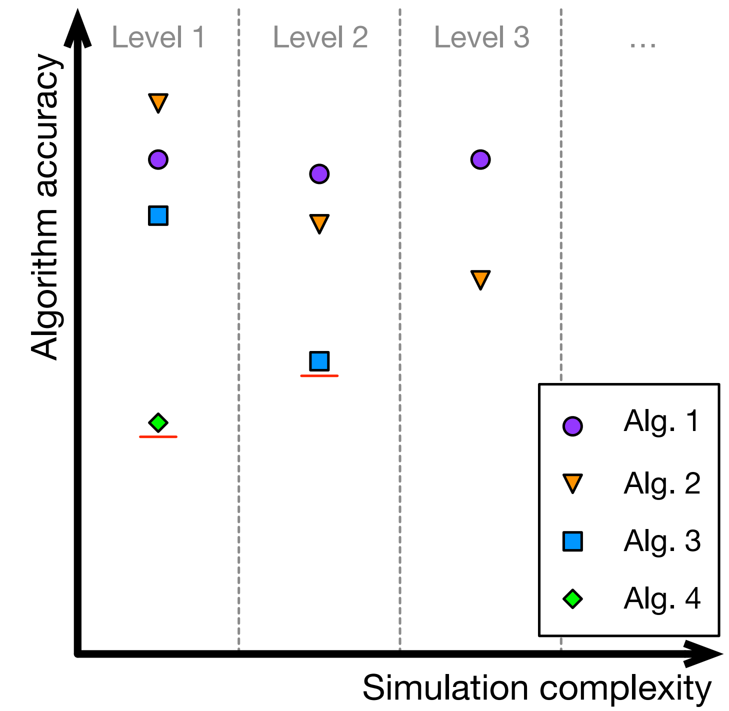


[4]

[3] Amrouche, Sabrina, et al. "The tracking machine learning challenge: throughput phase." Computing and Software for Big Science 7.1 (2023): 1.
[4] Vlimant, J.-R., Innocente, V., Salzburger, A., & Guyon, I. (n.d.). TrackML: a Tracking Machine Learning Challenge [Slide show]. ACAT 2019, Saas-fee.

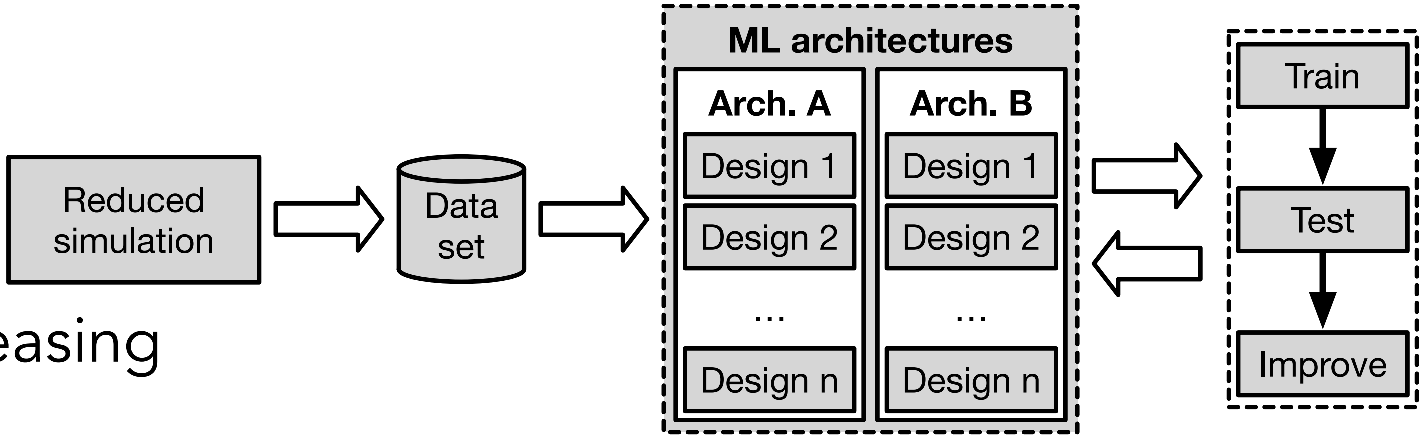
Introduction

- This is a very **active field of research** with multiple directions of development
  - TrackML challenge [5]
  - Graph Neural Networks (GNN)



- Our goal is to construct a **systematic way of finding an optimal algorithm** for the task at hand
  - Increase efficiency
  - Reduce designer bias

- The requirements for this are two-fold
  - **Synthetic detector data** with increasing complexity levels
  - Multiple **ML(-assisted) strategies** for tracking

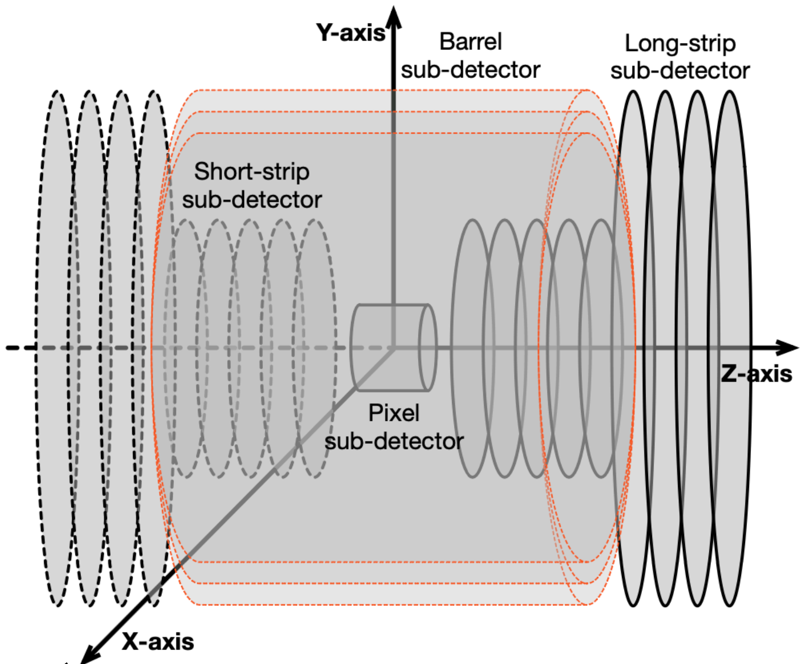


[5] Calafiura, Polo, et al. "Trackml: a high energy physics particle tracking challenge." 2018 IEEE 14th International Conference on e-Science (e-Science). IEEE, 2018.

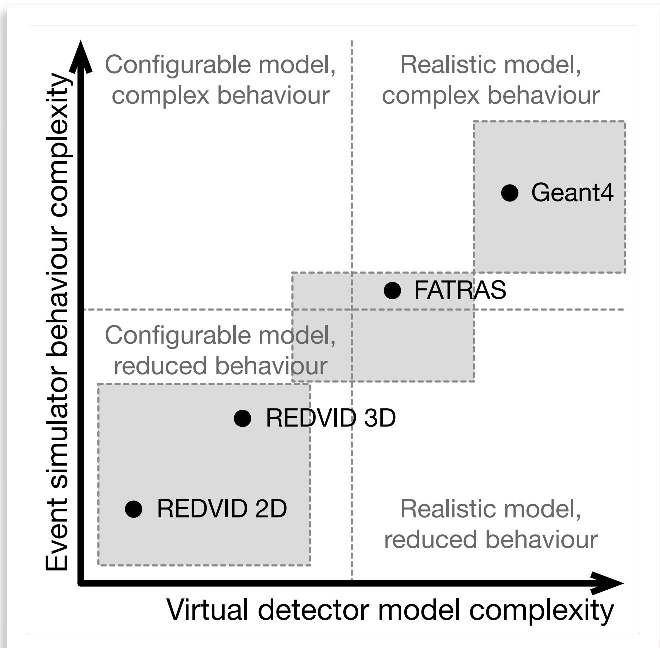
Data Generation

Data generation

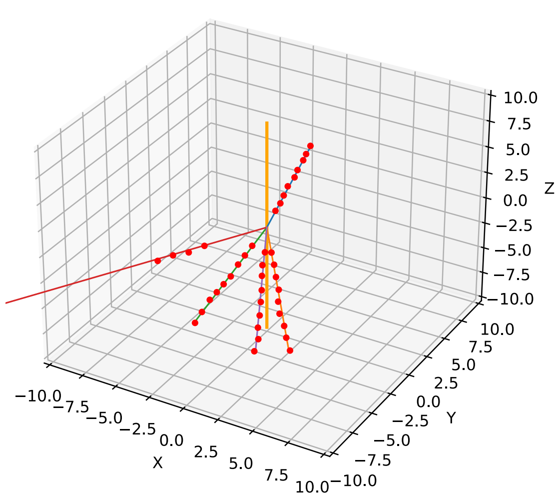
- For the purpose of data generation at various levels of complexity a lightweight simulator for HEP collision data was developed: **REDuced Virtual Detector (REDVID)** [6]



- Trades in physics accuracy for increased **configurability and modularity**
  - Physics-accurate simulators do already exist (FATRAS, Geant)
  - High configurability is necessary for the first complexity layers in the aforementioned systematic algorithm search



- **Layers of complexity achievable** with this dataset at the current state include
  - Parametrised linear tracks
  - Parametrised helical tracks
  - Noise levels
  - Origin smearing

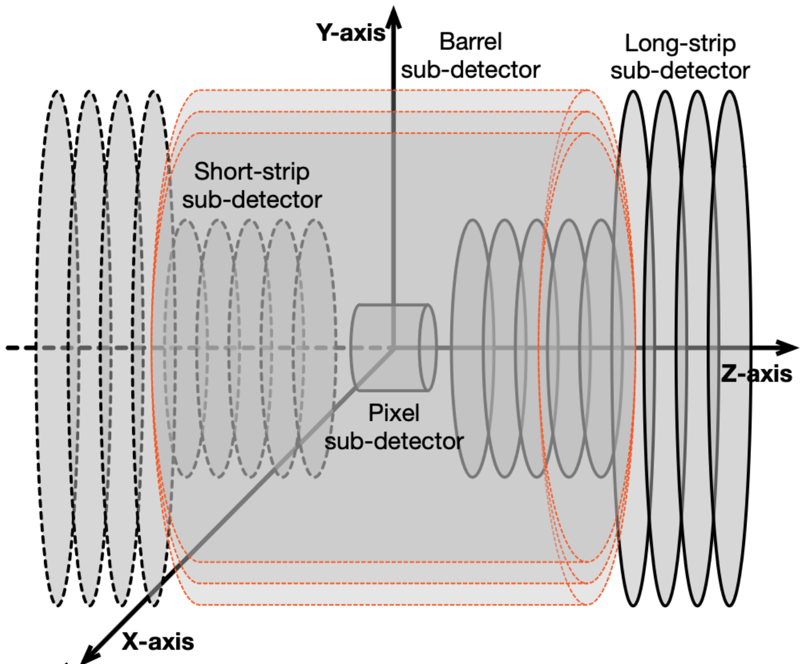


[6] U. Odyurt and S.N. Swatman and A.L. Varbanescu and S. Caron, "Reduced Simulations for High-Energy Physics, a Middle Ground for Data-Driven Physics Research," 2023, DOI: 10.48550/arXiv.2309.03780.

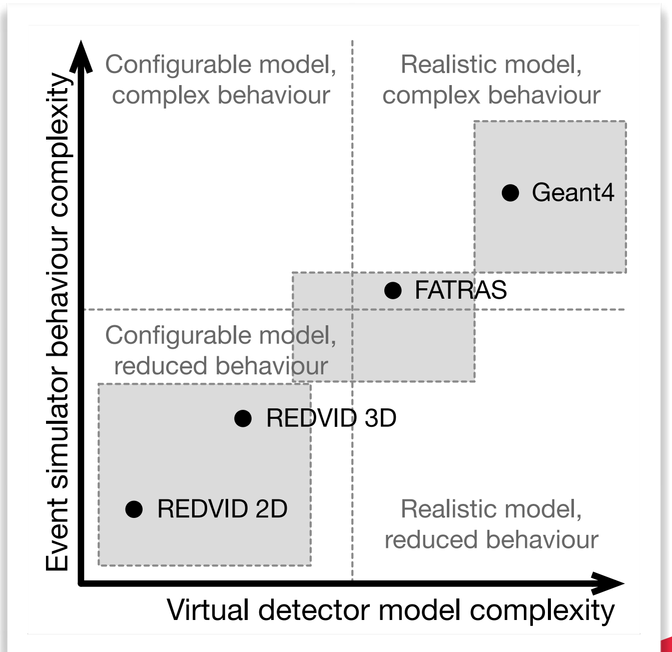
Nikhef

Data generation

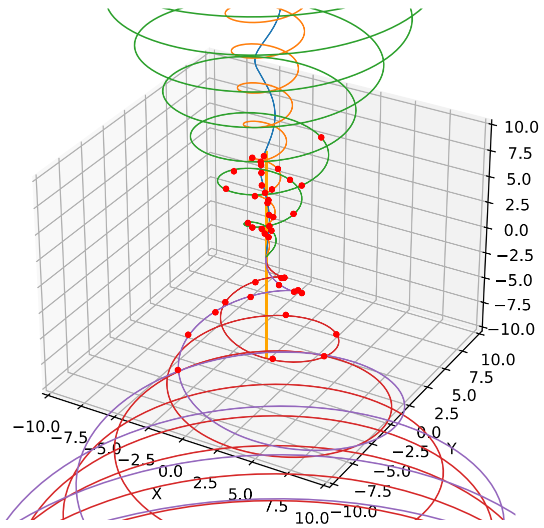
- For the purpose of data generation at various levels of complexity a lightweight simulator for HEP collision data was developed: **REDuced Virtual Detector (REDVID)** [6]



- Trades in physics accuracy for increased **configurability and modularity**
  - Physics-accurate simulators do already exist (FATRAS, Geant)
  - High configurability is necessary for the first complexity layers in the aforementioned systematic algorithm search



- **Layers of complexity achievable** with this dataset at the current state include
  - Parametrised linear tracks
  - Parametrised helical tracks
  - Noise levels
  - Origin smearing



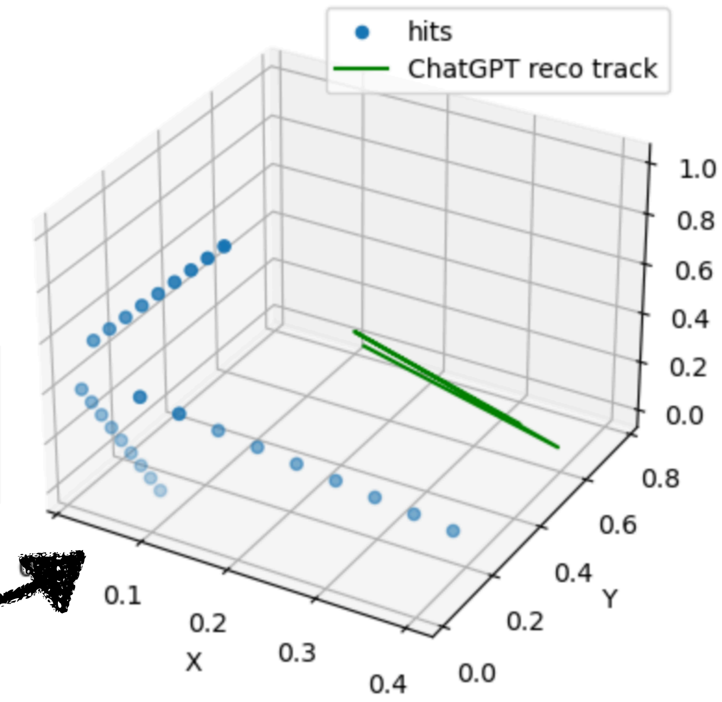
[6] U. Odyurt and S.N. Swatman and A.L. Varbanescu and S. Caron, "Reduced Simulations for High-Energy Physics, a Middle Ground for Data-Driven Physics Research," 2023, DOI: 10.48550/arXiv.2309.03780.

Algorithm Designs

Transformers

Z Hi ChatGPT, could you convert for me the following matrix of hits in three dimensions (n_hits, 3) to tracks?

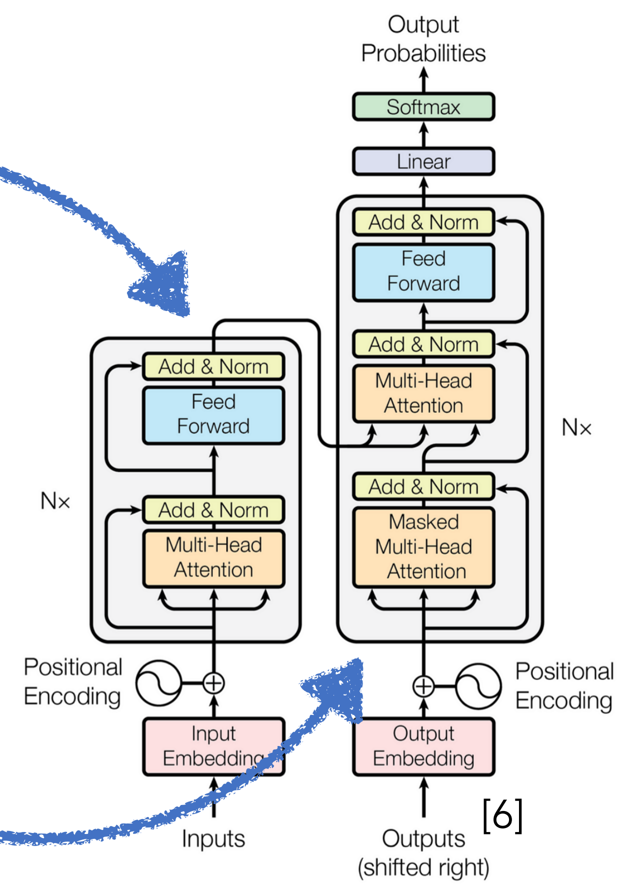
Certainly! Here are the hits that are part of one track based on the calculated z-coordinates:



- Asking ChatGPT is not the way…
  - We have to do something more sophisticated

The encoder model **takes some input and encodes it** into some latent space

The decoder model **takes as input the output of the encoder and some input sequence**, and **outputs the next item in the sequence**



[6]

Advantages

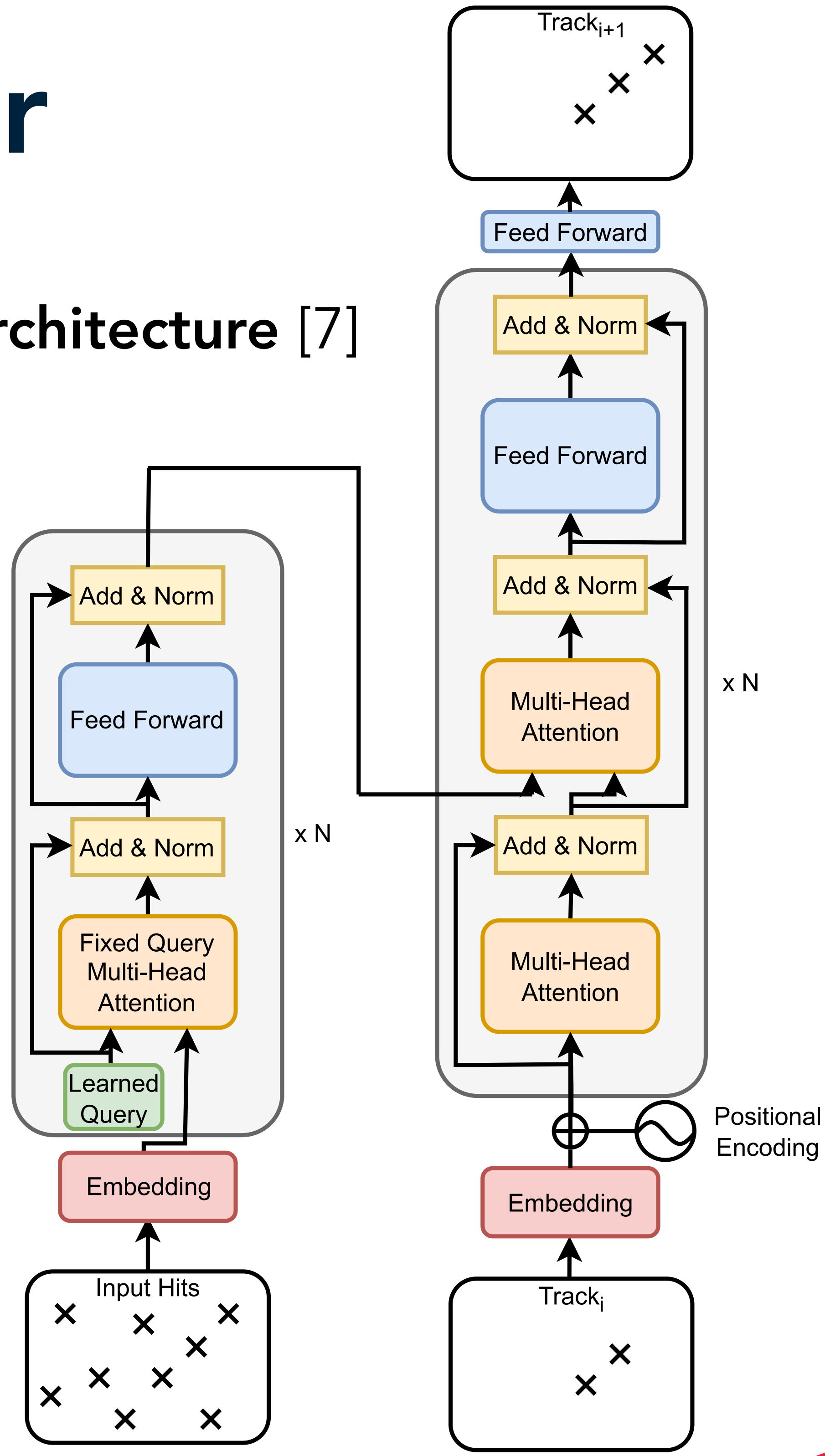
- Parallelizable training
- O(n^2) complexity, developments for efficient transformers
- Good at capturing complex nonlinear dynamics

[7] Vaswani, Ashish, et al. "Attention is all you need." Advances in neural information processing systems 30 (2017).

Nikhef

Transformers - Trackformer

- This model resembles closely the **original transformer architecture** [7]
- Translating, e.g. English to Spanish, is a typical task for transformer models
  - This model in similar fashion **translates hits to tracks**
- **Encoder:** Encodes full event hits
  - **No positional embedding** as hits have no particular order
  - **Fixed-query attention** [8] to achieve full positional invariance of inputs
- **Decoder:** Predicts next hit in track
  - **Autoregressively builds the full track**, starting from a given seed

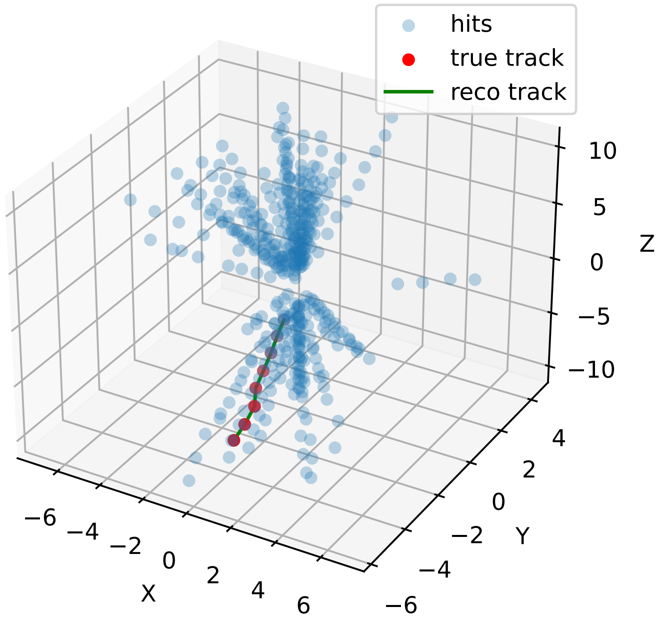


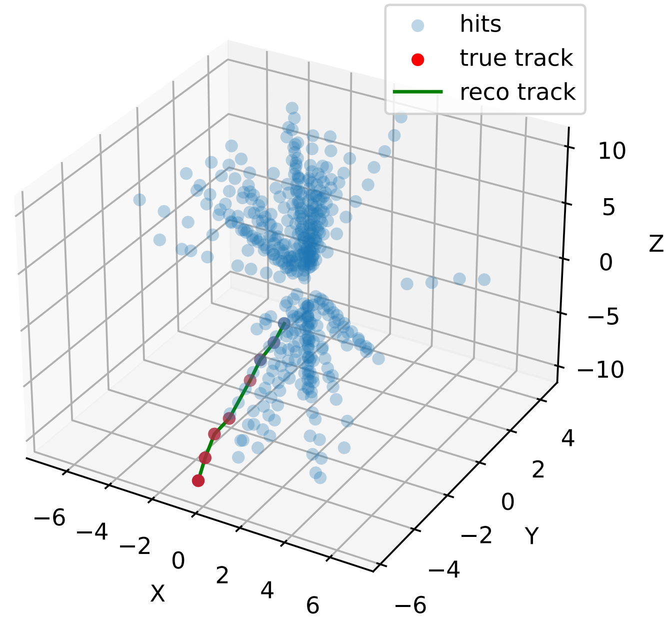
[7] Vaswani, Ashish, et al. "Attention is all you need." Advances in neural information processing systems 30 (2017).
[8] Lee, Juho, et al. "Set transformer: A framework for attention-based permutation-invariant neural networks." International conference on machine learning. PMLR, 2019.

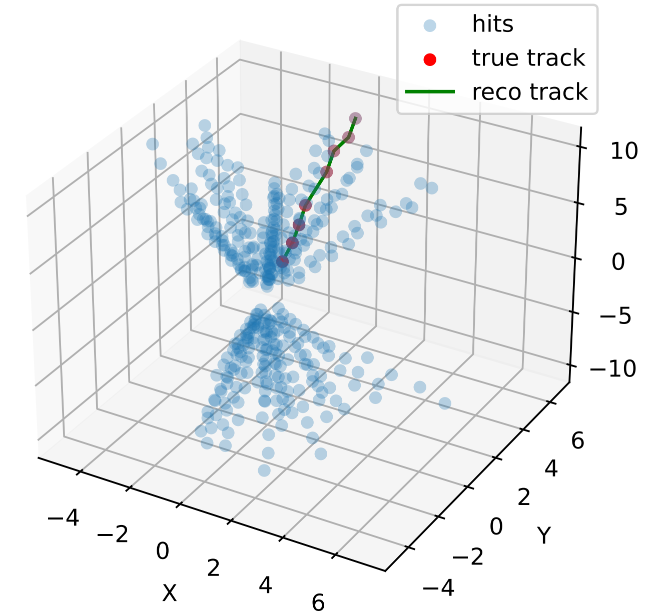
Nikhef

Transformers - Trackformer

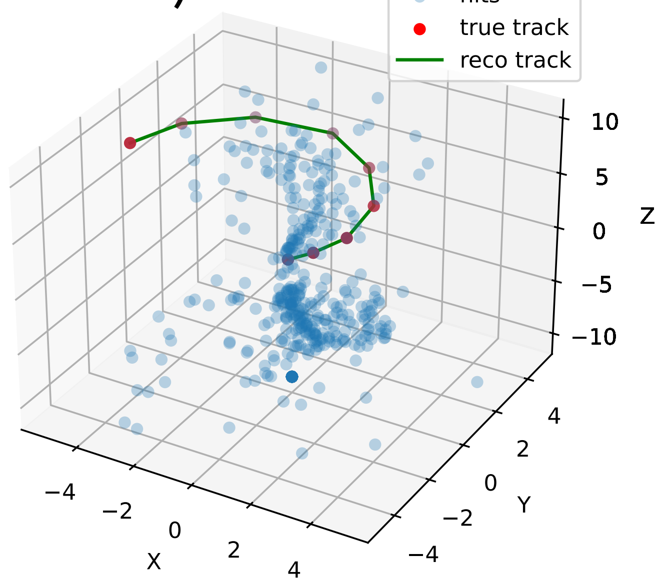
- Model achieves **92% accuracy** on REDVID data with 1-20 (17 average, 2.8 standard deviation) **straight tracks with noise** per event

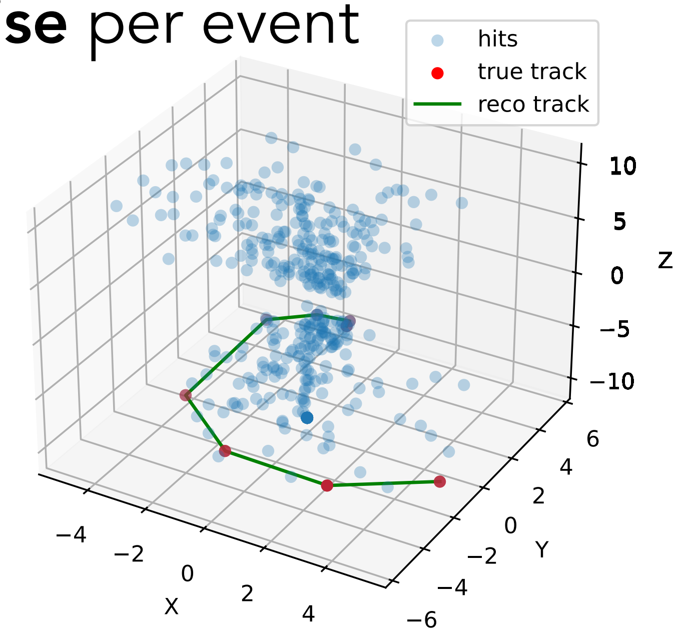


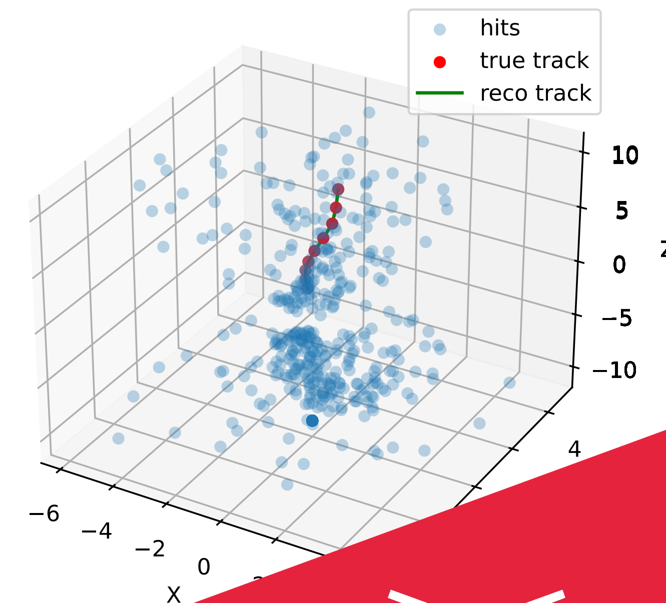




- Model achieves **85% accuracy** on REDVID data with 10-50 (42 average, 6.7 standard deviation) **curved tracks with noise** per event

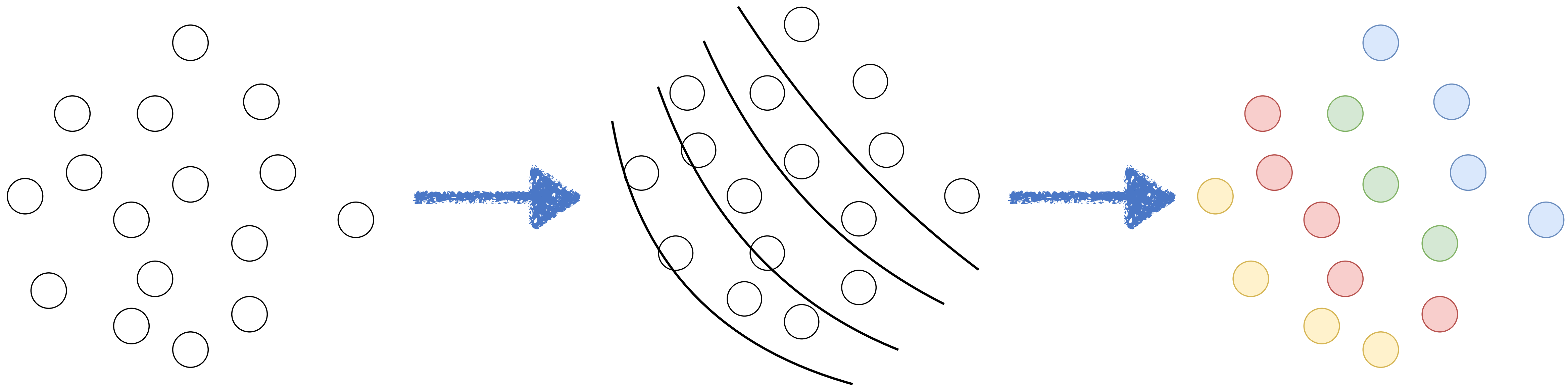






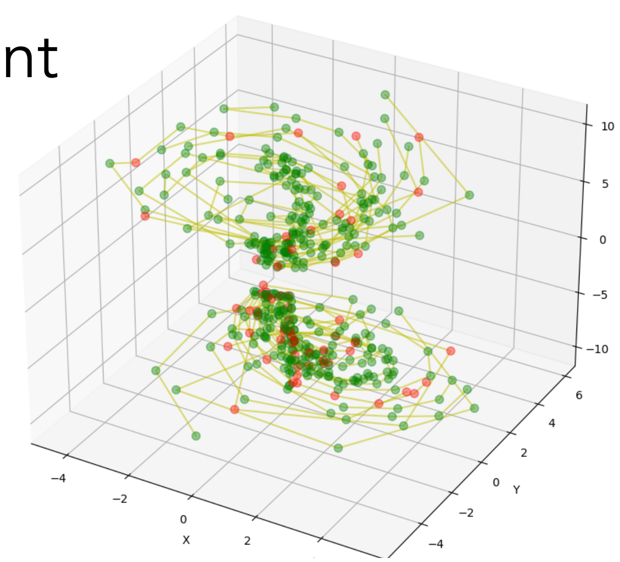
Transformers - Encoder-only Classifier

- This architecture only uses an only an **encoder as a classifier** (sequence to sequence)
  - Not autoregressive, advantage being **one-shot classification** of full hits in event
  - Classification bins defined in track parameter dimensions, e.g. radius, pitch, …



- Model achieves **88% accuracy** on REDVID data with 10-50 (42 average, 6.7 standard deviation) **curved tracks with noise** per event

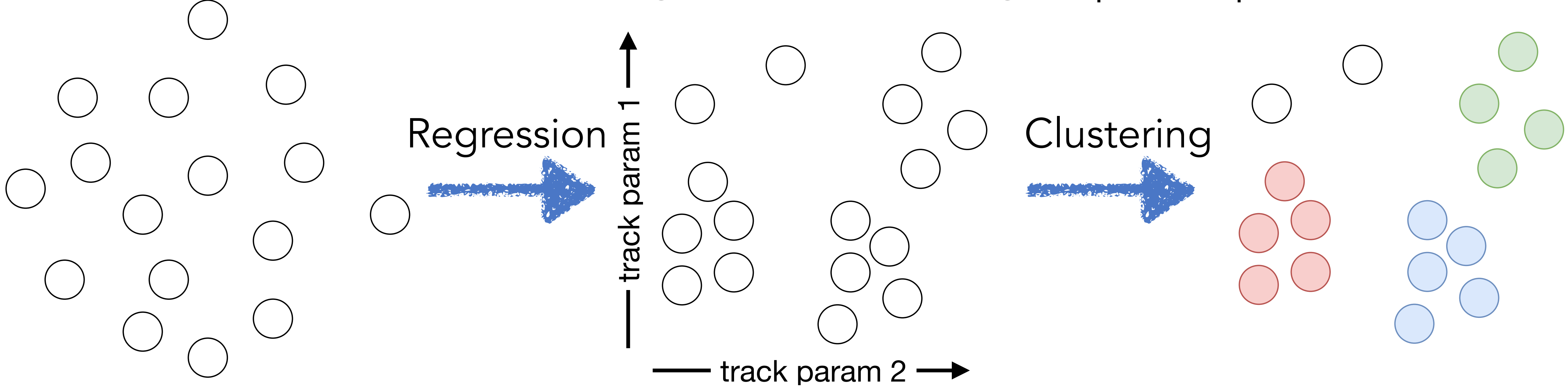
Green: Correctly classified hits
Red: Incorrectly classified hits
Lines connect true tracks



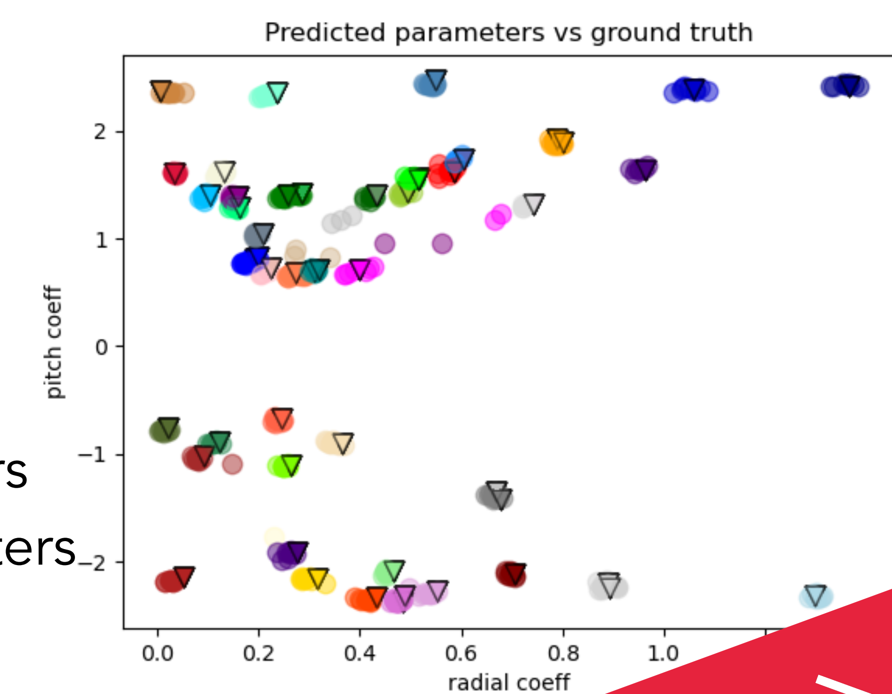
Nikhef

Transformers - Encoder-only Regressor

- This architecture only uses an only an **encoder as a regressor** (sequence to sequence)
  - Regresses track parameters, followed by agglomerative clustering
  - Also a one-shot approach, although extra clustering step is required



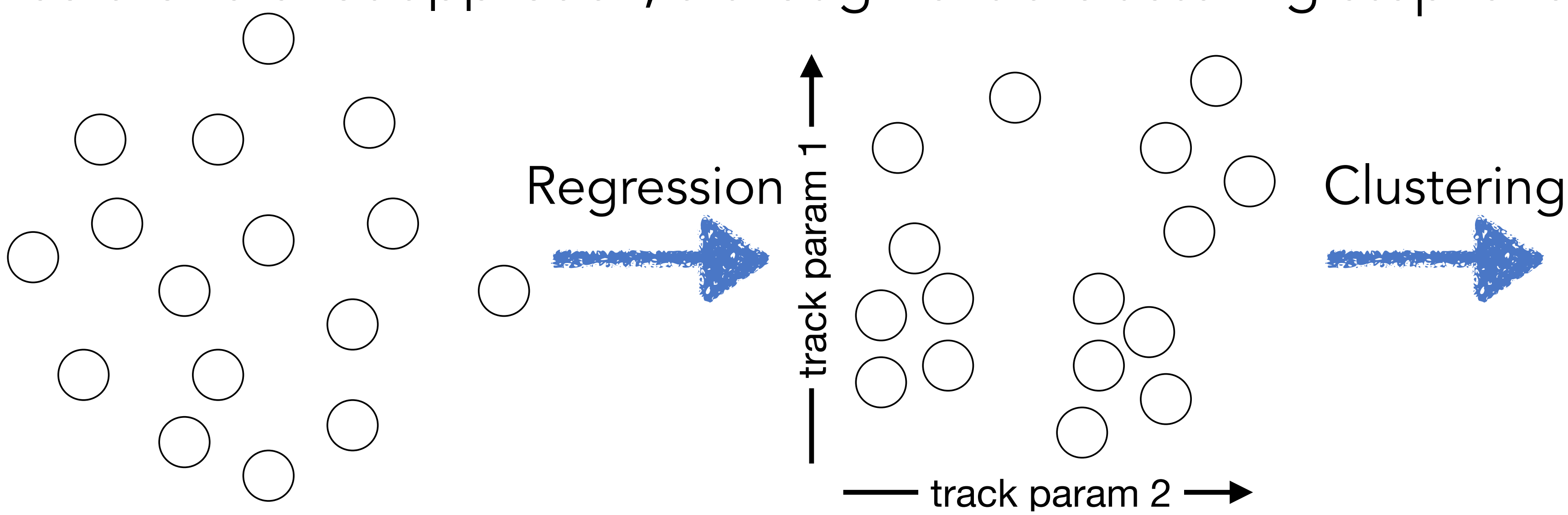
- Model achieves **87% accuracy** on REDVID data with 10-50 (42 average, 6.7 standard deviation) **curved tracks with noise** per event



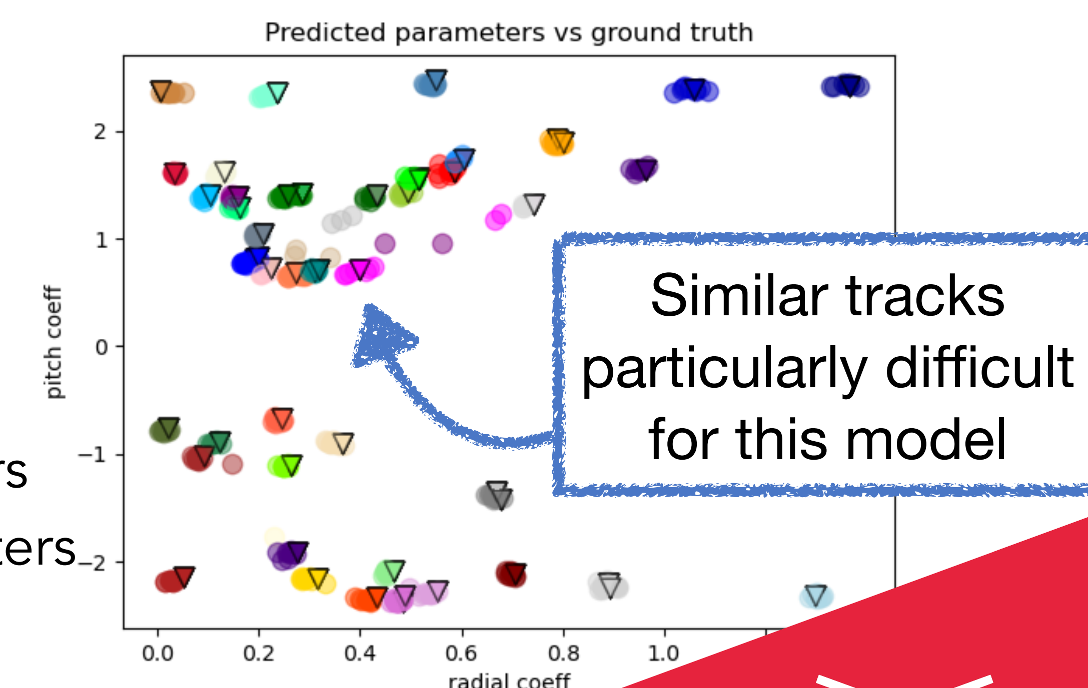
Triangles: True track parameters
Circles: Regressed track parameters
Colors indicate clusters

Transformers - Encoder-only Regressor

- This architecture only uses an only an **encoder as a regressor** (sequence to sequence)
  - Regresses track parameters, followed by agglomerative clustering
  - Also a one-shot approach, although extra clustering step is required

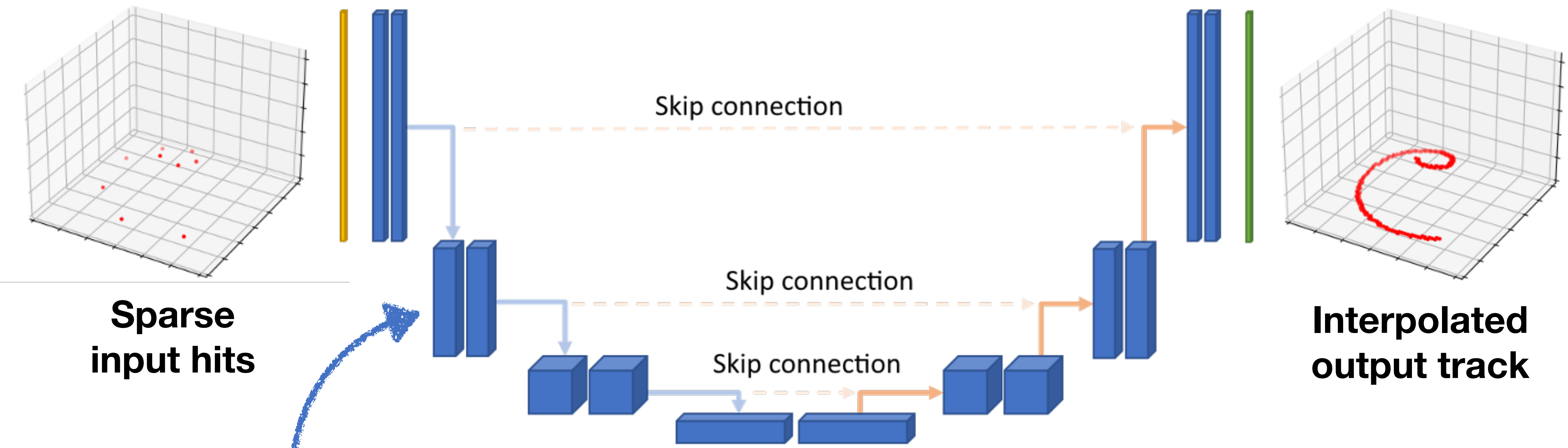


- Model achieves **87% accuracy** on REDVID data with 10-50 (42 average, 6.7 standard deviation) **curved tracks with noise** per event



Triangles: True track parameters
Circles: Regressed track parameters
Colors indicate clusters

U-net Model



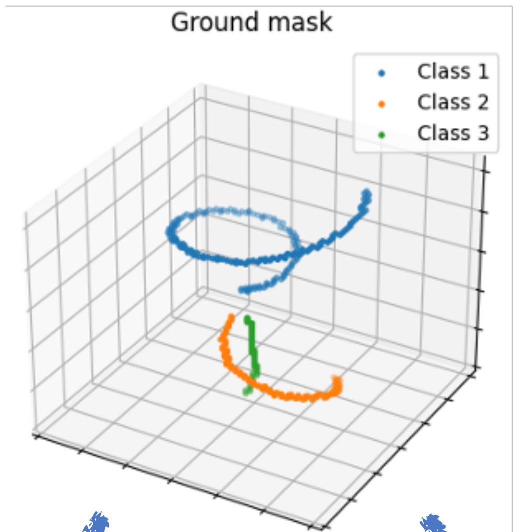
Vanilla convolutions are replaced by **submanifold sparse convolutions** [10]**.** This ensures that convolution operations are only executed in places where there is information in the input image

Binary cross entropy with weight for classes

$$\mathcal{L}_{\omega\text{-BCE}} = -\frac{1}{N}\sum_{i=1}^{N}\left[\omega_1\, p_i \log p_i + \omega_2\,(1-p_i)\log(1-p_i)\right]$$

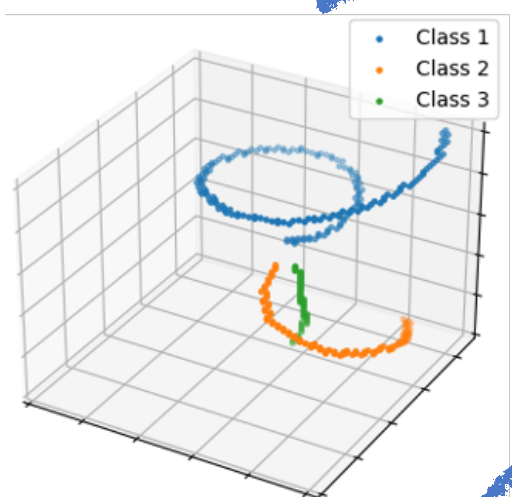
[9] Ronneberger O, Fischer P, Brox T (2015). "U-Net: Convolutional Networks for Biomedical Image Segmentation". arXiv:1505.04597
[10] Graham, Benjamin, and Laurens Van der Maaten. "Submanifold sparse convolutional networks." arXiv preprint arXiv:1706.01307 (2017).

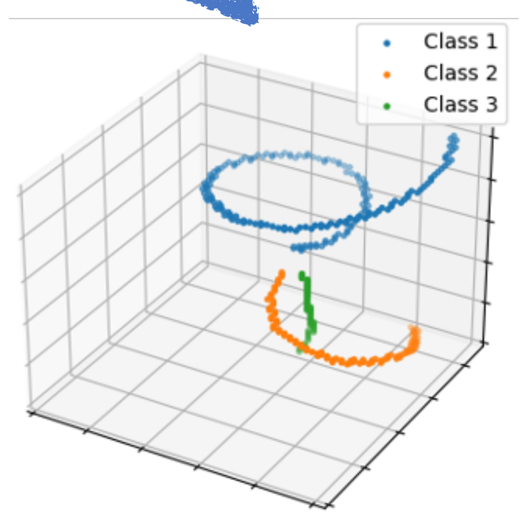
U-net Model

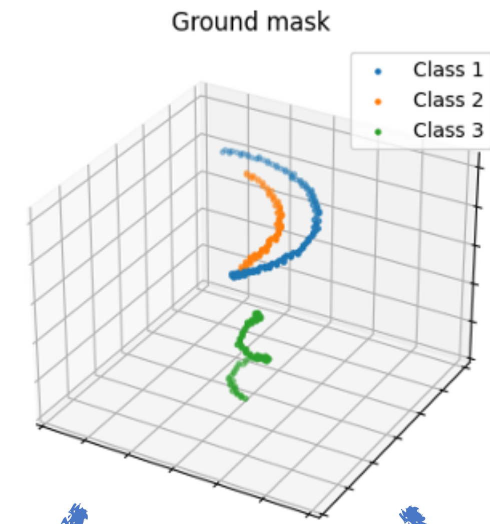


DBSCAN

Spectral clustering

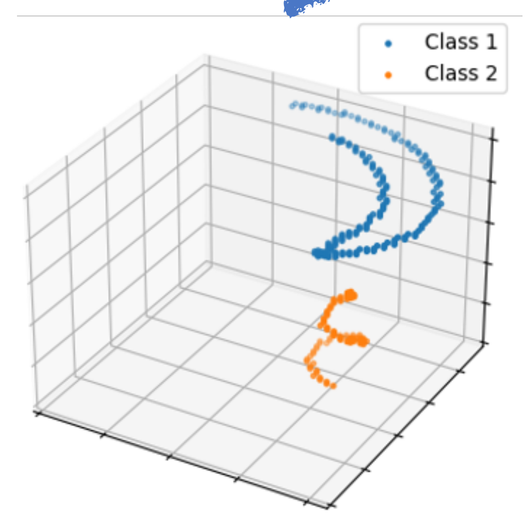


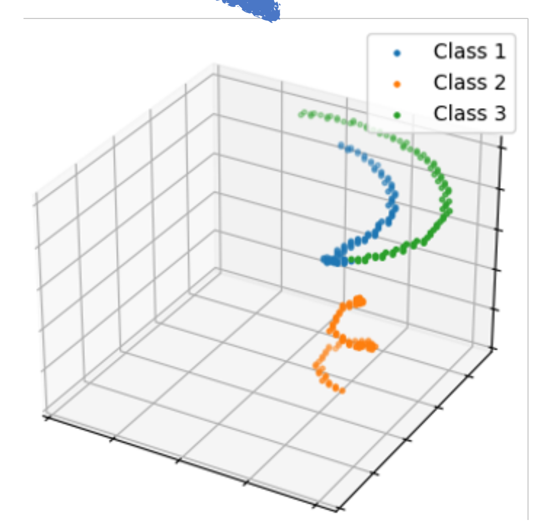


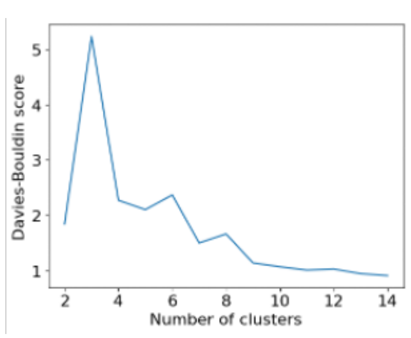


DBSCAN

Spectral clustering







Optimal number of clusters can be determined using different metrics even for those algorithms that are not strictly based on densities!

DBSCAN struggles when borders of tracks are not well defined. Its performance can improved by increasing resolution of tensor

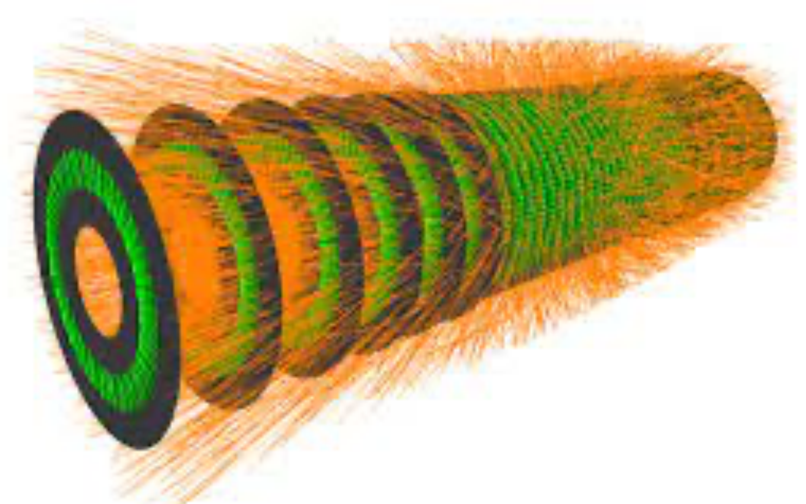
Conclusions and Future Developments

Conclusions and Future Developments

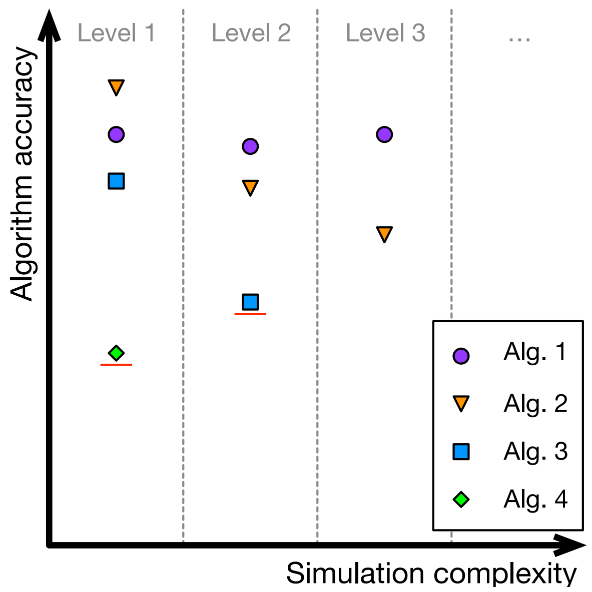
- Developing a systematic approach to optimal ML(-assisted) model finding in the context of charged particle tracking
  - Models currently considered showing promising results in ~50 tracks REDVID data

Future Developments

- Can fill out table for first few complexity steps in systematic approach soon for all models, most developments now ongoing in U-net approach
- For the transformer models developments have started on trackML dataset, challenges largely computational



TrackML [5] simulated event

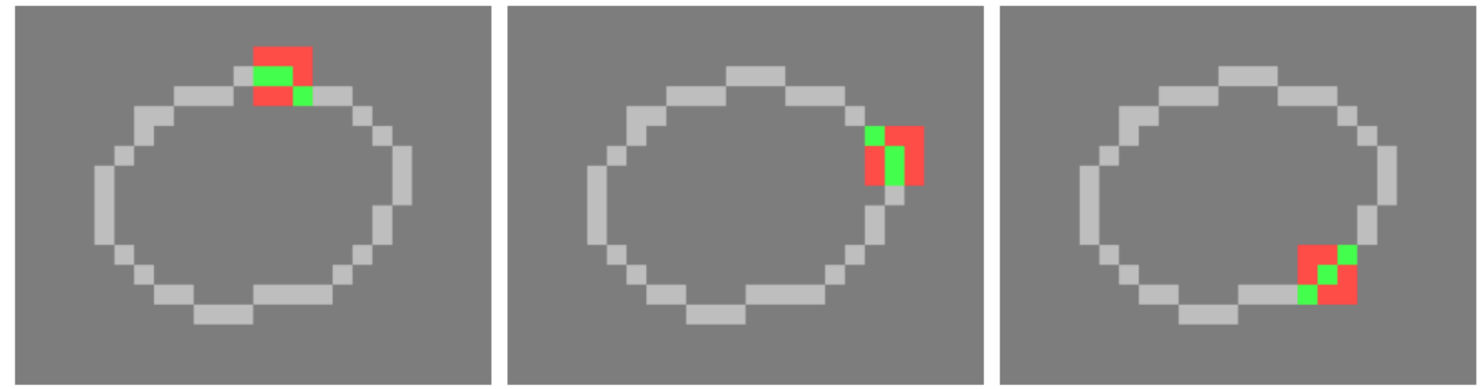


[5] Calafiura, Polo, et al. "Trackml: a high energy physics particle tracking challenge." 2018 IEEE 14th International Conference on e-Science (e-Science). IEEE, 2018.

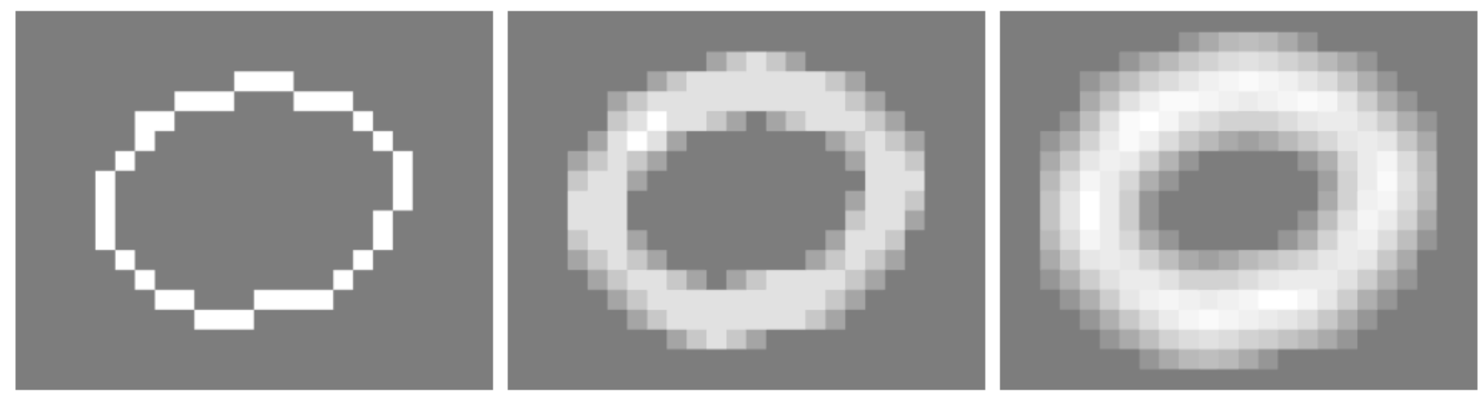
Backup

Submanifold Sparse Convolutions

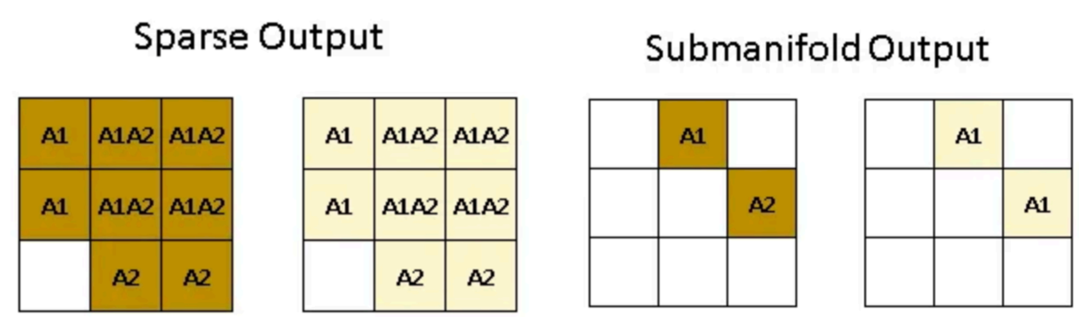
Sparse convolutions only consider input "active sites" and the kernel does process the entire image



This still causes sub manifold dilation



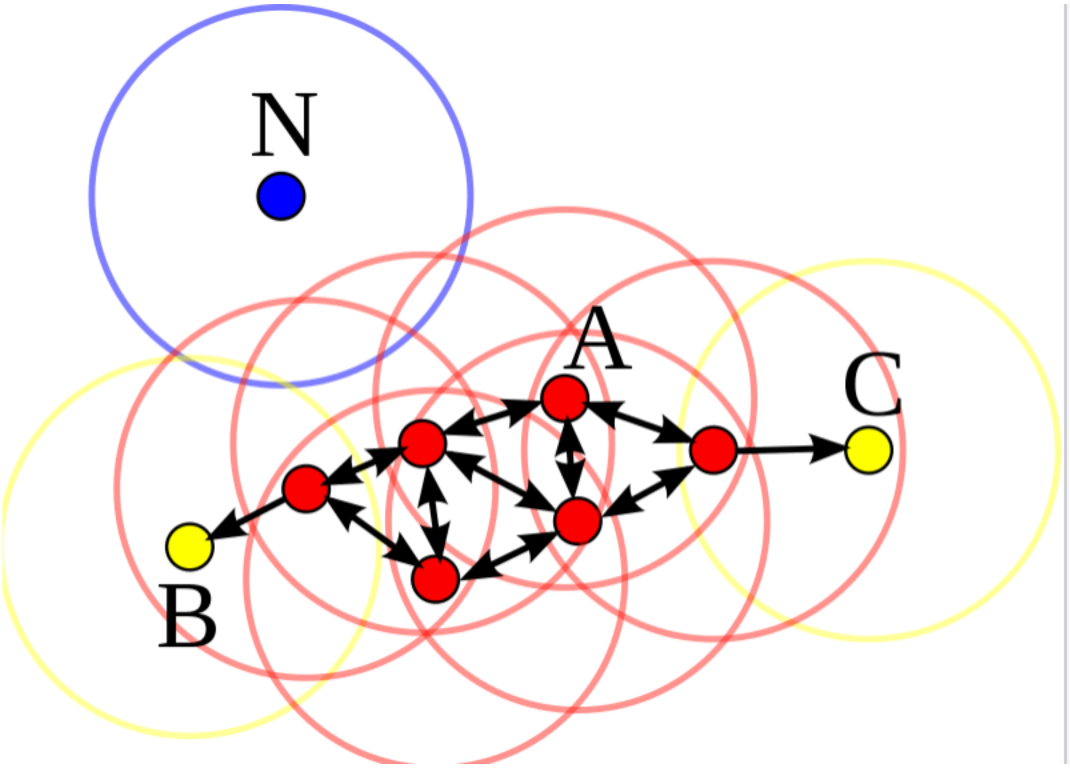
To remedy this, submanifold convolutions are proposed, which only calculate outputs for active input sites, i.e. no dilation



by Zhiliang Zhou

DBSCAN

- A point p is a *core point* if at least minPts points are within distance ε of it (including p).
- A point q is *directly reachable* from p if point q is within distance ε from core point p. Points are only said to be directly reachable from core points.
- A point q is *reachable* from p if there is a path $p_1, \ldots, p_n$ with $p_1 = p$ and $p_n = q$, where each p_{i+1} is directly reachable from p_i. Note that this implies that the initial point and all points on the path must be core points, with the possible exception of q.
- All points not reachable from any other point are *outliers* or *noise points*

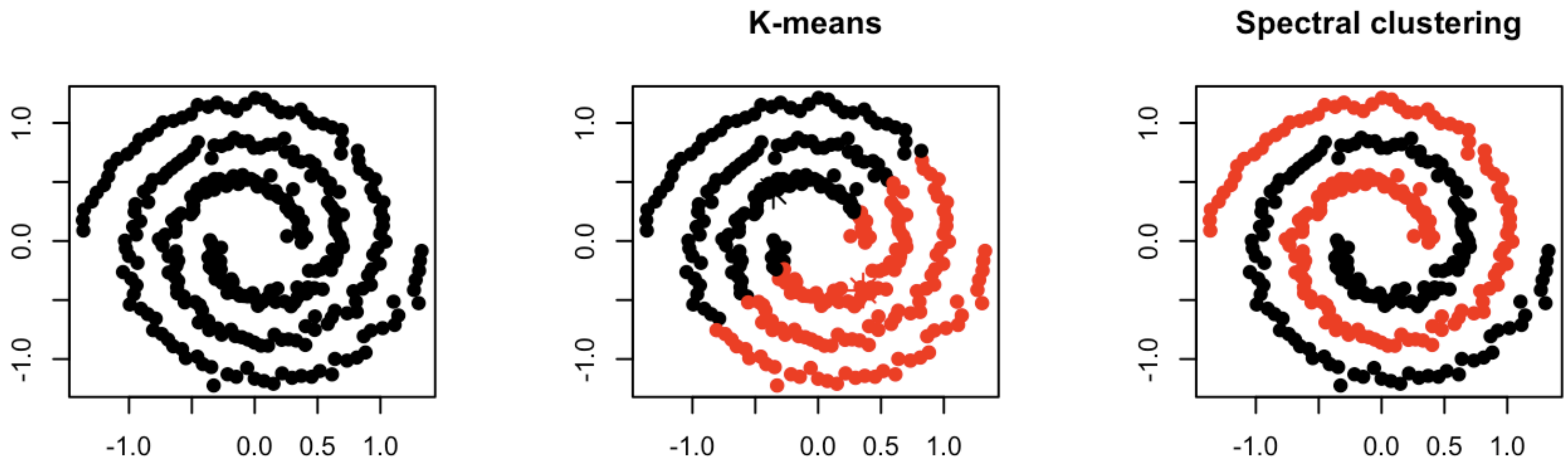


In this diagram, minPts = 4. Point A and the other red points are core points, because the area surrounding these points in an ε radius contain at least 4 points (including the point itself). Because they are all reachable from one another, they form a single cluster. Points B and C are not core points, but are reachable from A (via other core points) and thus belong to the cluster as well. Point N is a noise point that is neither a core point nor directly-reachable.

Spectral Clustering

Clusters uses connectivity between datapoints to create clusters.

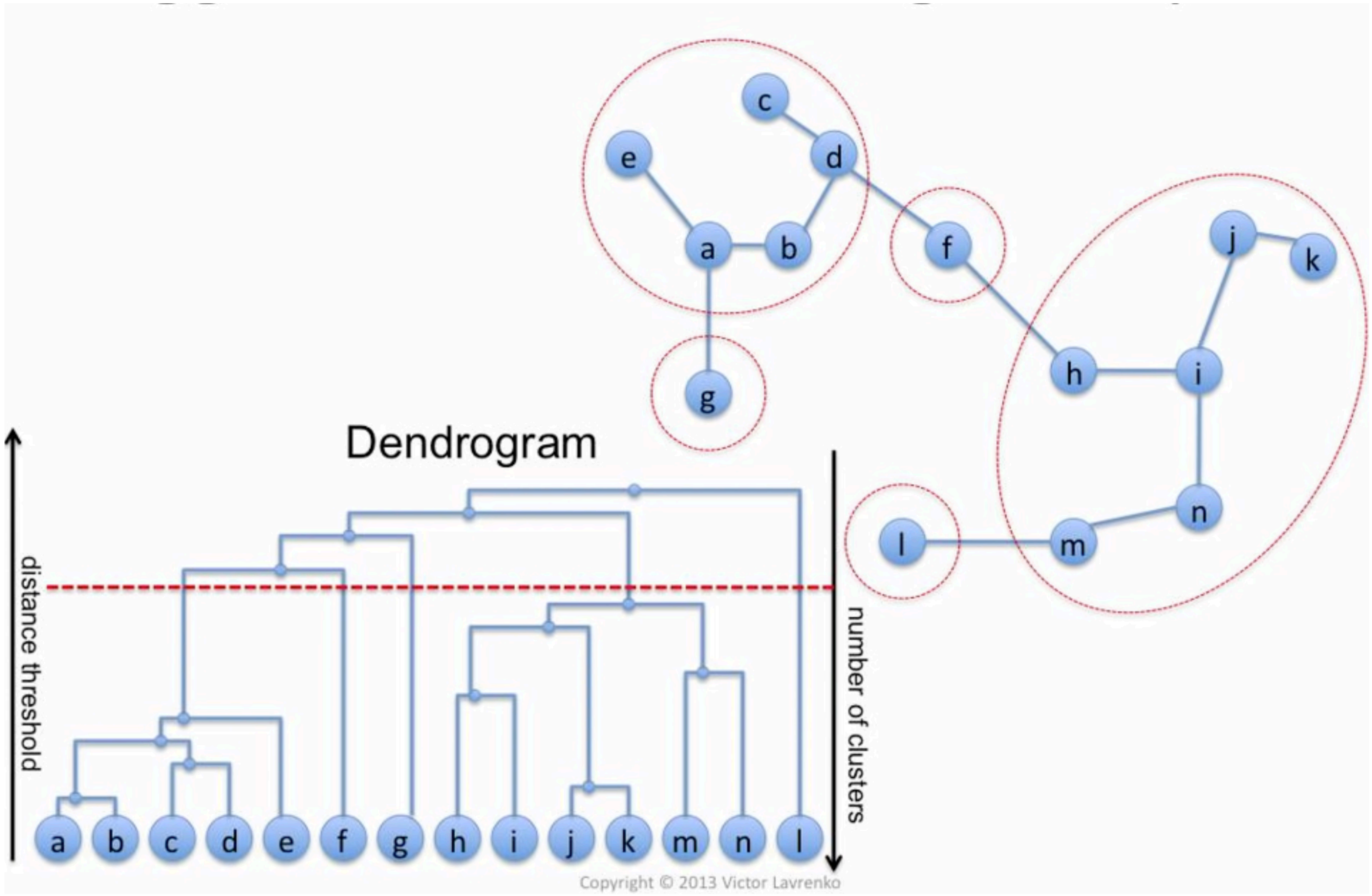
Uses eigenvalues and eigenvectors of the data matrix to forecast the data into lower dimensions space to cluster the data points. Based on the idea of a graph representation of data where the data point are represented as nodes and the similarity between the data points are represented by an edge.



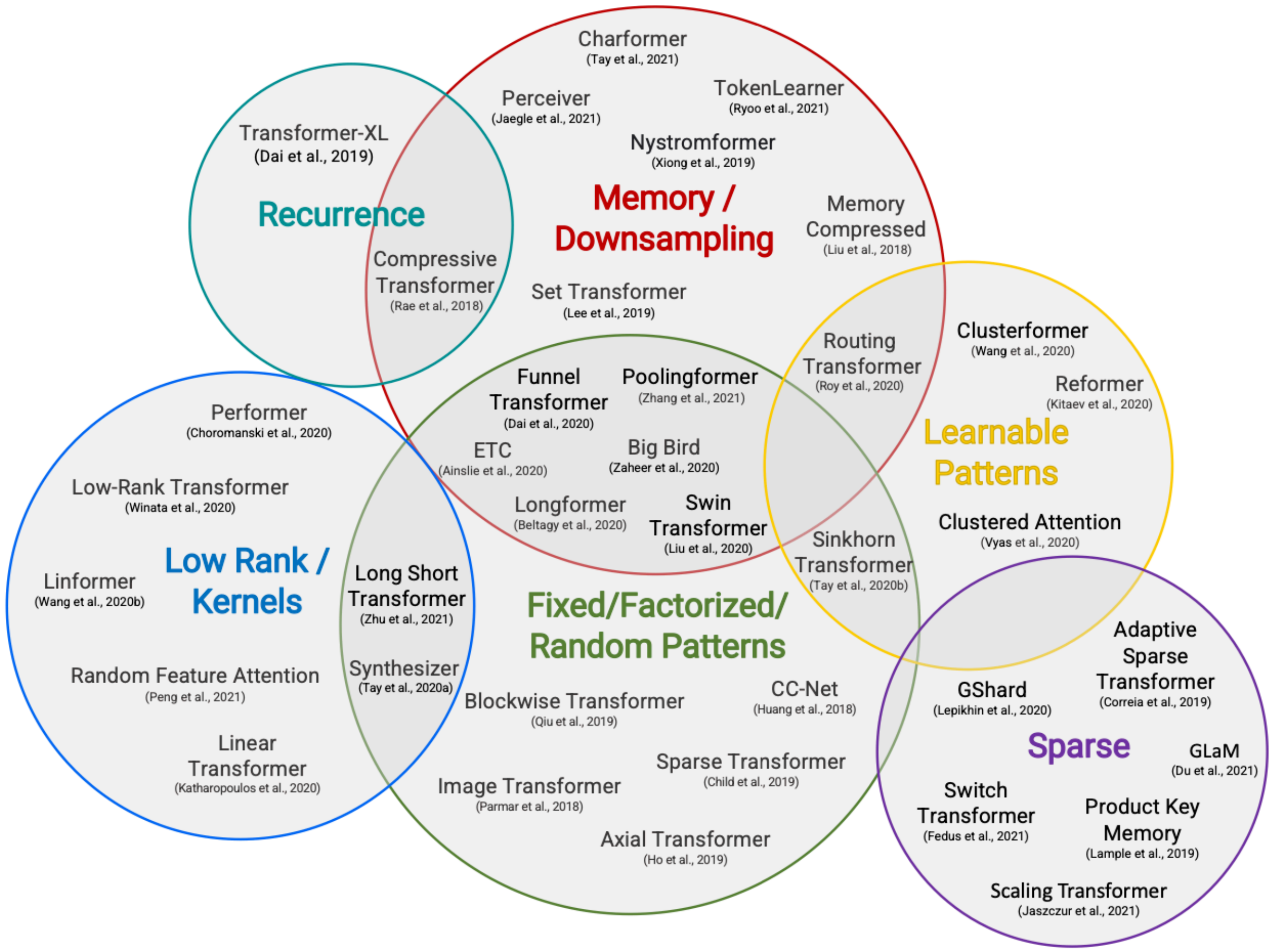
The Davies-Bouldin score is defined as the average similarity measure of each cluster with its most similar cluster, where similarity is the ratio of within-cluster distances to between-cluster distances. Thus, clusters which are farther apart and less dispersed will result in a better score.

Agglomerative Clustering

Agglomerative clustering iteratively adds closest points to clusters, starting with all points as singleton clusters, until all points are connected, at which state a cut in the distance results in a corresponding number of clusters



Efficient Transformers



Model / Paper	Complexity	Decode	Class
Memory Compressed (Liu et al., 2018)	$\mathcal{O}(N_c^2)$	✓	FP+M
Image Transformer (Parmar et al., 2018)	$\mathcal{O}(N.m)$	✓	FP
Set Transformer (Lee et al., 2019)	$\mathcal{O}(kN)$	✗	M
Transformer-XL (Dai et al., 2019)	$\mathcal{O}(N^2)$	✓	RC
Sparse Transformer (Child et al., 2019)	$\mathcal{O}(N\sqrt{N})$	✓	FP
Reformer (Kitaev et al., 2020)	$\mathcal{O}(N \log N)$	✓	LP
Routing Transformer (Roy et al., 2020)	$\mathcal{O}(N\sqrt{N})$	✓	LP
Axial Transformer (Ho et al., 2019)	$\mathcal{O}(N\sqrt{N})$	✓	FP
Compressive Transformer (Rae et al., 2020)	$\mathcal{O}(N^2)$	✓	RC
Sinkhorn Transformer (Tay et al., 2020b)	$\mathcal{O}(B^2)$	✓	LP
Longformer (Beltagy et al., 2020)	$\mathcal{O}(n(k+m))$	✓	FP+M
ETC (Ainslie et al., 2020)	$\mathcal{O}(N_g^2 + NN_g)$	✗	FP+M
Synthesizer (Tay et al., 2020a)	$\mathcal{O}(N^2)$	✓	LR+LP
Performer (Choromanski et al., 2020a)	$\mathcal{O}(N)$	✓	KR
Funnel Transformer (Dai et al., 2020)	$\mathcal{O}(N^2)$	✓	FP+DS
Linformer (Wang et al., 2020c)	$\mathcal{O}(N)$	✗	LR
Linear Transformers (Katharopoulos et al., 2020)	$\mathcal{O}(N)$	✓	KR
Big Bird (Zaheer et al., 2020)	$\mathcal{O}(N)$	✗	FP+M
Random Feature Attention (Peng et al., 2021)	$\mathcal{O}(N)$	✓	KR
Long Short Transformers (Zhu et al., 2021)	$\mathcal{O}(kN)$	✓	FP + LR
Poolingformer (Zhang et al., 2021)	$\mathcal{O}(N)$	✗	FP+M
Nyströmformer (Xiong et al., 2021b)	$\mathcal{O}(kN)$	✗	M+DS
Perceiver (Jaegle et al., 2021)	$\mathcal{O}(kN)$	✓	M+DS
Clusterformer (Wang et al., 2020b)	$\mathcal{O}(N \log N)$	✗	LP
Luna (Ma et al., 2021)	$\mathcal{O}(kN)$	✓	M
TokenLearner (Ryoo et al., 2021)	$\mathcal{O}(k^2)$	✗	DS
Adaptive Sparse Transformer (Correia et al., 2019)	$\mathcal{O}(N^2)$	✓	Sparse
Product Key Memory (Lample et al., 2019)	$\mathcal{O}(N^2)$	✓	Sparse
Switch Transformer (Fedus et al., 2021)	$\mathcal{O}(N^2)$	✓	Sparse
ST-MoE (Zoph et al., 2022)	$\mathcal{O}(N^2)$	✓	Sparse
GShard (Lepikhin et al., 2020)	$\mathcal{O}(N^2)$	✓	Sparse
Scaling Transformers (Jaszczur et al., 2021)	$\mathcal{O}(N^2)$	✓	Sparse
GLaM (Du et al., 2021)	$\mathcal{O}(N^2)$	✓	Sparse

Yi Tay, Mostafa Dehghani, Dara Bahri, & Donald Metzler. (2022). Efficient Transformers: A Survey.