WEIZMANN INSTITUTE OF SCIENCE

Università di Genova

DIFI DIPARTIMENTO DI FISICA

Accelerating Graph-Based Tracking with Symbolic Regression

2406.16752
accepted to MLST

F. di Bello, E. Gross, C. Schiavi, **N. Soybelman**

INFN

ML4Jets, 06.11.2024

Motivation

- Tracking data is crucial for trigger decisions
- New ML methods improve tracking but are slow
- Implementing advanced models on FPGAs is complex
- Faster execution mean higher event handling rate
- Can we accelerate these models effectively?

Symbolic Regression on FPGAs for Fast Machine Learning Inference

arXiv: 2305.04099

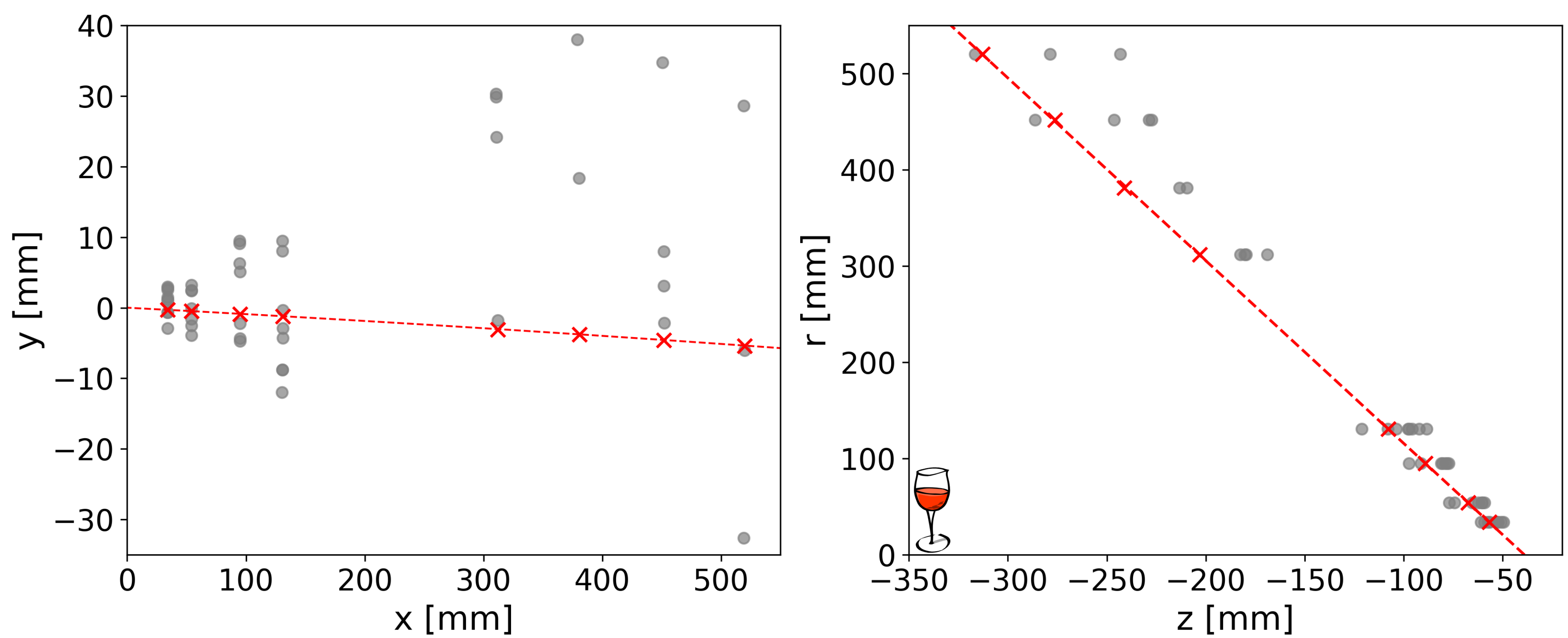
Ho Fung Tsoi[1*], *Adrian Alan* Pol[2], *Vladimir* Loncar[3,4], *Ekaterina* Govorkova[3], *Miles* Cranmer[2,5], *Sridhara* Dasu[1], *Peter* Elmer[2], *Philip* Harris[3], *Isobel* Ojalvo[2], and *Maurizio* Pierini[6]

- Approximate MLP with symbolic expression
- Used for jet classification
- Easy to implement on FPGA and fast inference

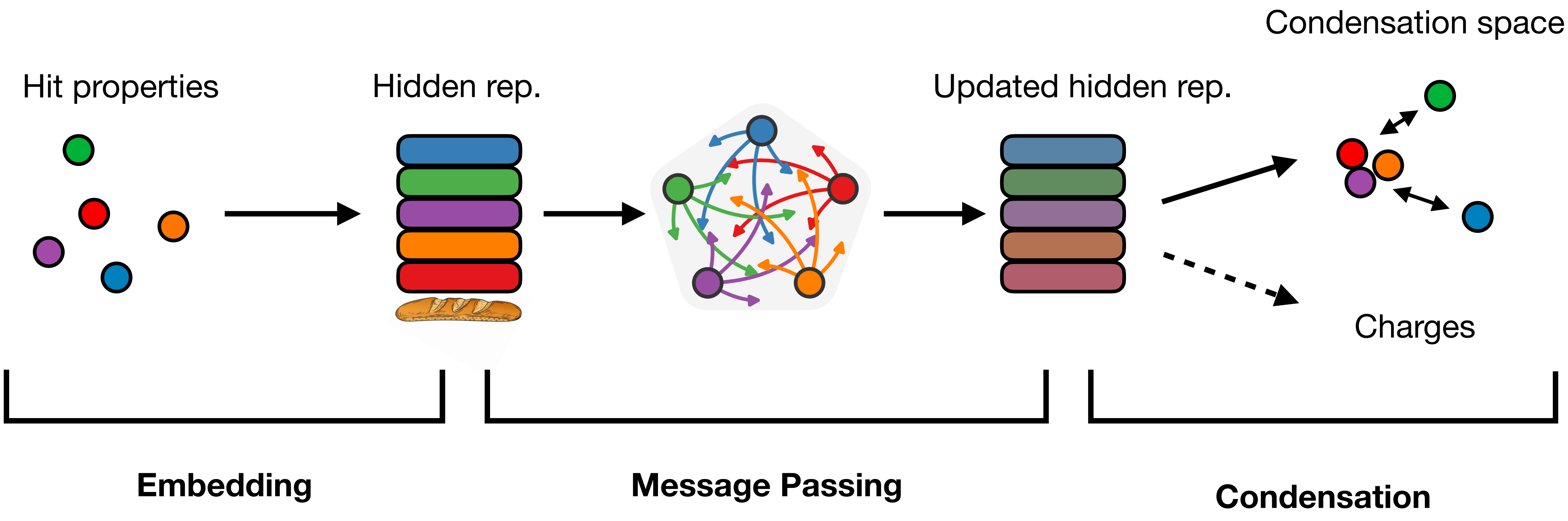
Can be generalized to more complex problems

Dataset

- **Events:** Single track, $p_T > 20$ GeV
- **Background:** Pile-up $\mu = 25$, detector noise (LHC Run 3)
- **Detector:** Simplified cylindrical, matching ATLAS ID radii
- **Smearing:** ϕ-, z-smearing based on ATLAS Pixel and SCT (no r-smearing)
- **Hit selection:** Preselected within 0.1×0.1 $\eta - \phi$ wedge, ± 5 mm around PV, fully contains track



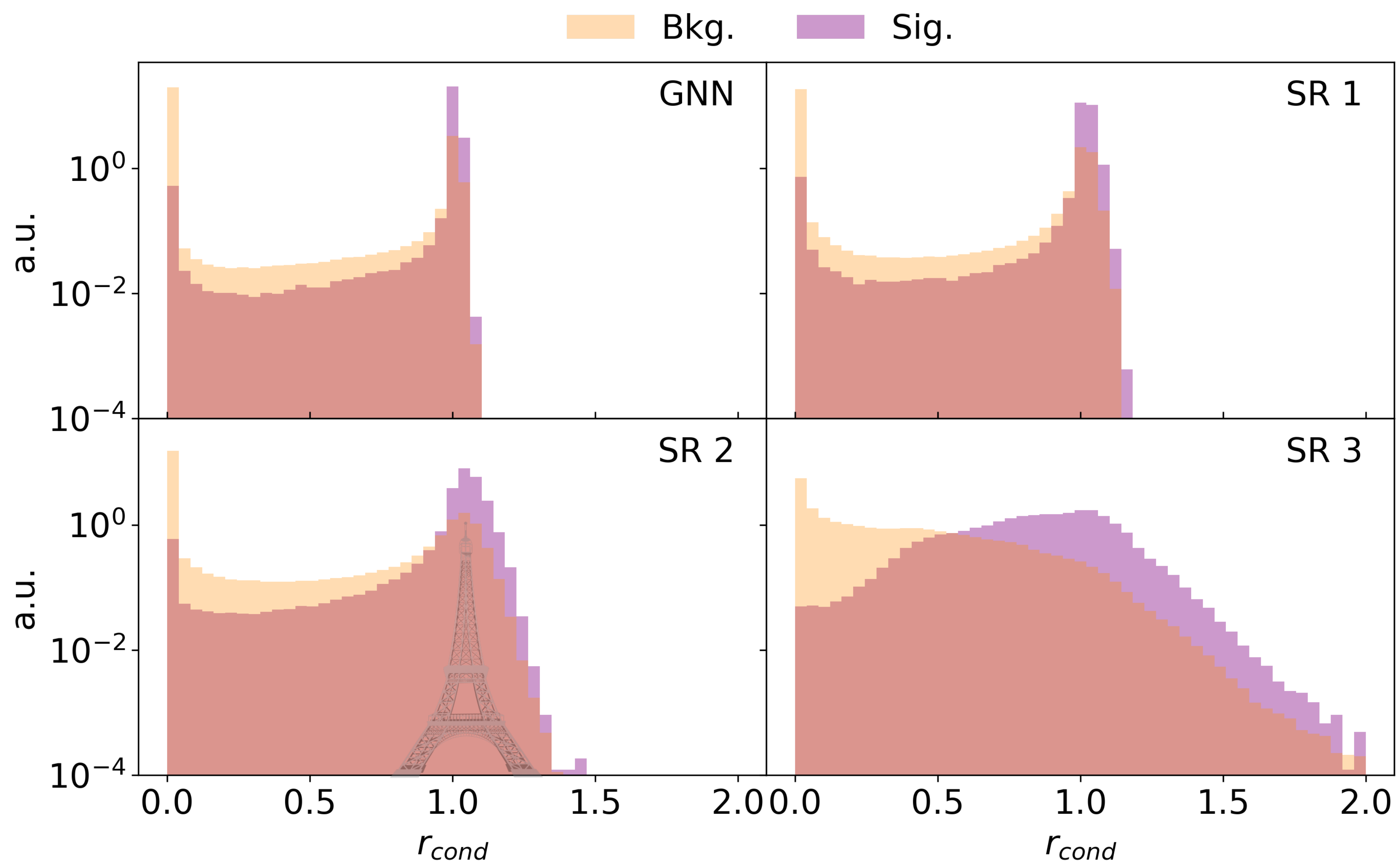
GNN with Symbolic Regression



- **Inspired by object condensation** (2002.03605)
- **Loss function:** Clusters signal hits, and maps noise to origin
- **Architecture:** 3MLPs; replaceable with symbolic expressions, preserving graph structure
- **PySR** for Symbolic Regression
- **Retrain** after each replacement to maintain performance

N. Soybelman — Weizmann Institute

Seperation in condensation space



SR1 - Embedding replaced

SR2 - Embedding + MP

SR3 - Full net

- Performance decreases with each step
- Good performance for SR1 and SR2, major performance loss in SR3
- No retraining possible after last replacement

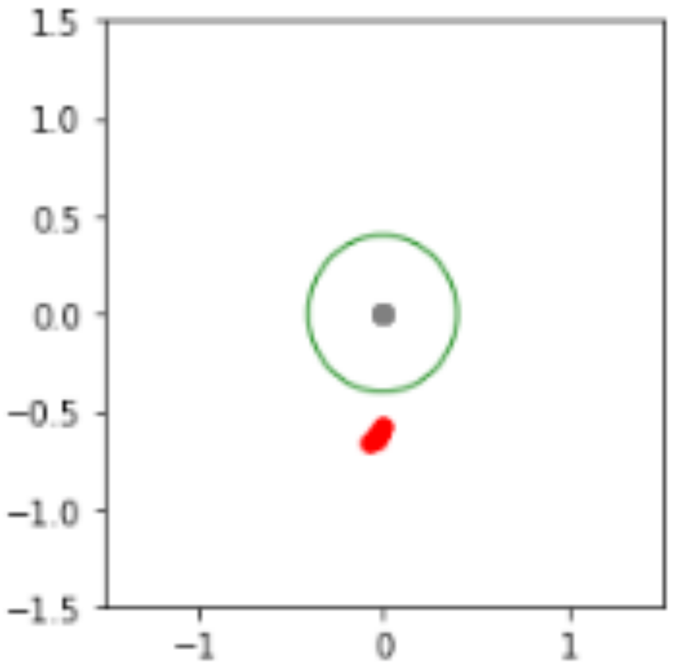
Postprocessing

1. Clustering

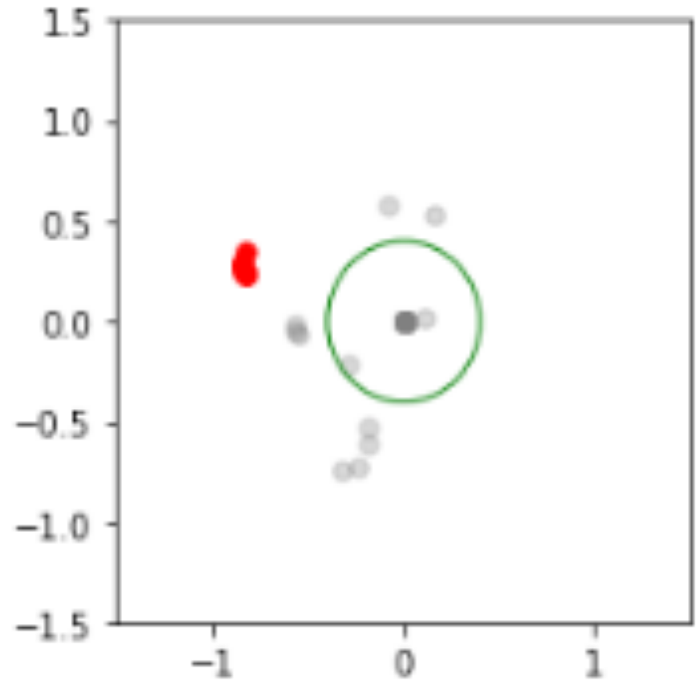
2. Triplet construction

3. Track fitting

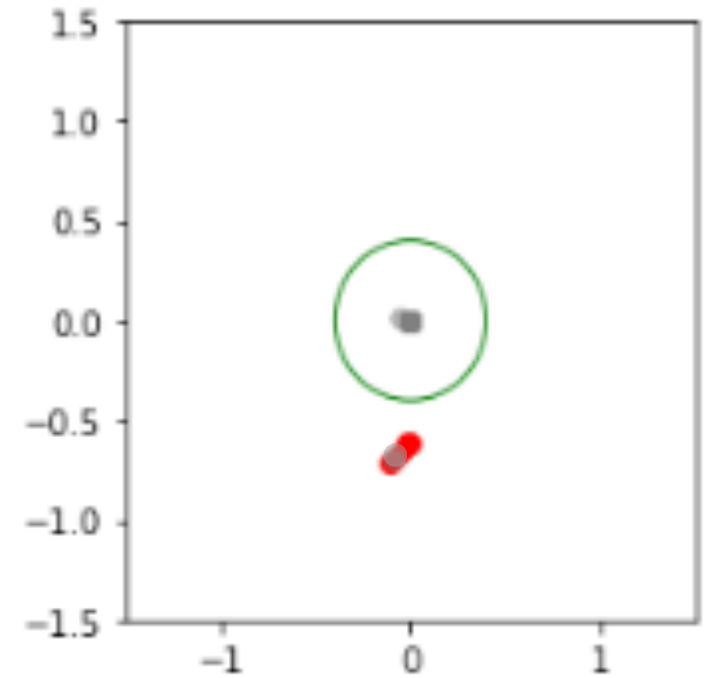
1. Clustering



Ideal case



Needs clustering



Contamination

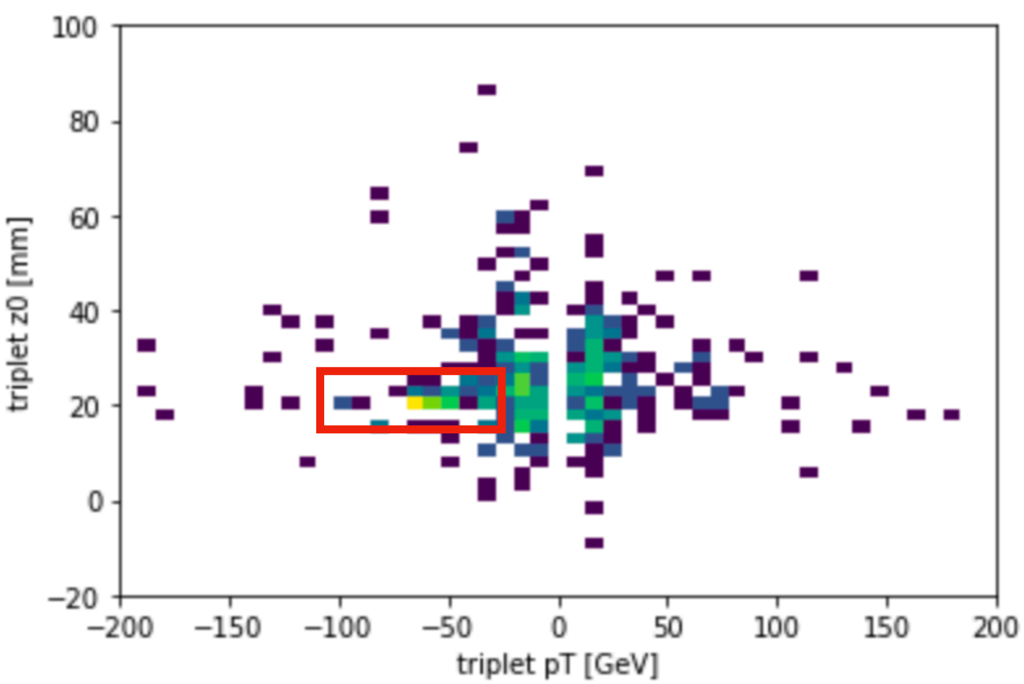
1. **Radius cut** around origin
2. **MeanShift** clustering to isolate track hits

Compare signal efficiency and background rejection

	GNN	SR1	SR2	SR3
r cut	0.05	0.2	0.3	0.4
bandwidth	0.7	0.7	0.6	0.7
ε_s cluster	96%	95%	93%	86%
r_b cluster	94%	94%	93%	71%

2. Triplet construction

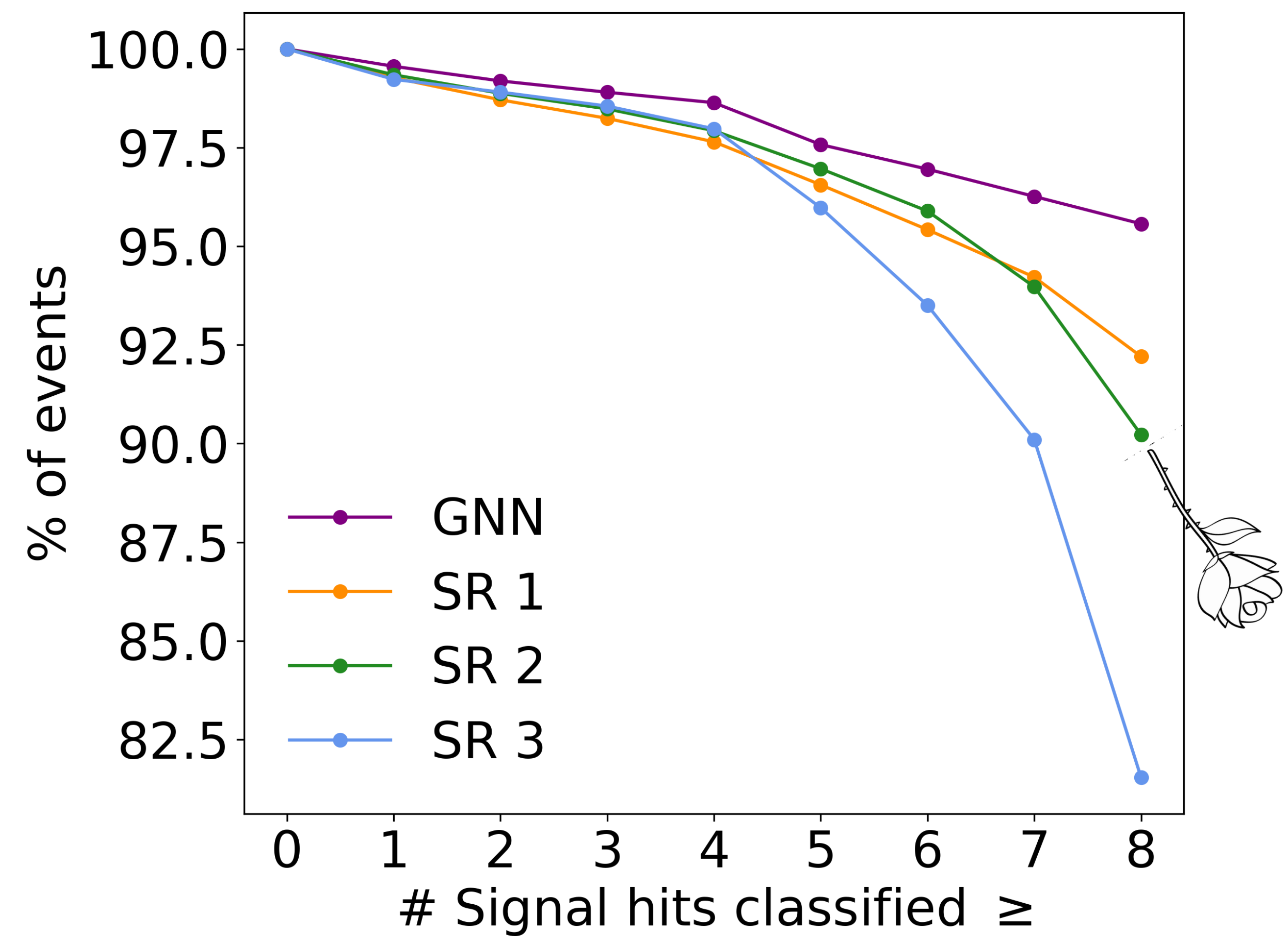
1. Get all combinations of 3 hits
2. Fit line and circle to obtain p_T and z_0
3. Fill 2D histogram in $p_T - z_0$
4. Select triplets in window around maximum
5. Keep hits contained in selected triplets



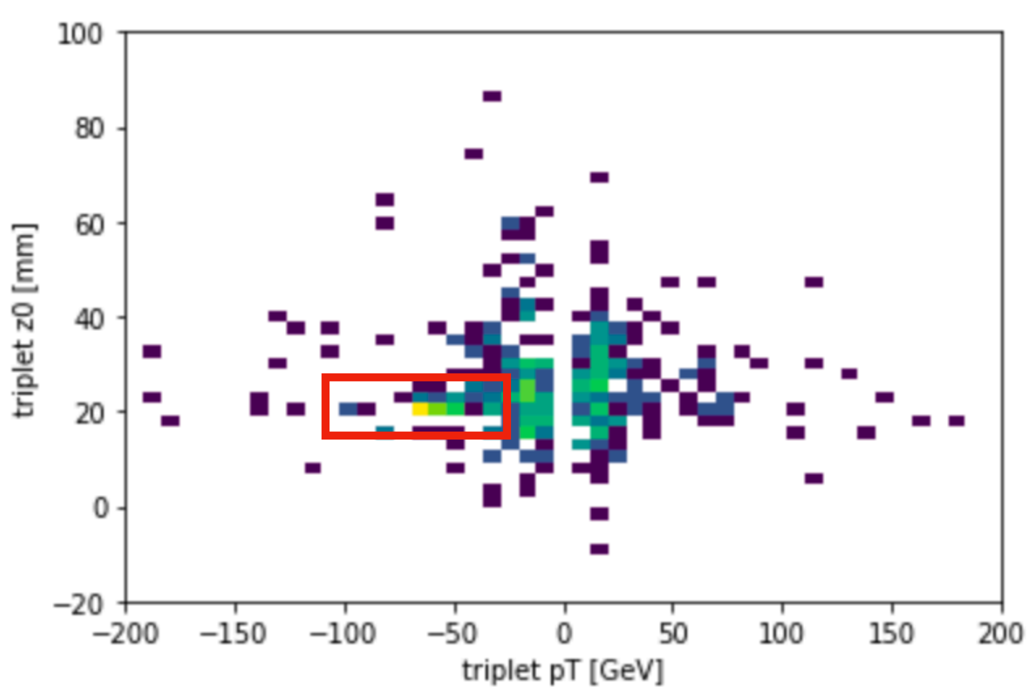
	GNN	SR1	SR2	SR3
r cut	0.05	0.2	0.3	0.4
bandwidth	0.7	0.7	0.6	0.7
ε_s cluster	96%	95%	93%	86%
r_b cluster	94%	94%	93%	71%
ε_s triplets	95%	94%	91%	82%
r_b triplets	99%	98%	98%	90%

2. Triplet construction

1. Get all combinations of 3 hits
2. Fit line and circle to obtain p_T and z_0
3. Fill 2D histogram in $p_T - z_0$
4. Select triplets in window around maximum
5. Keep hits contained in selected triplets

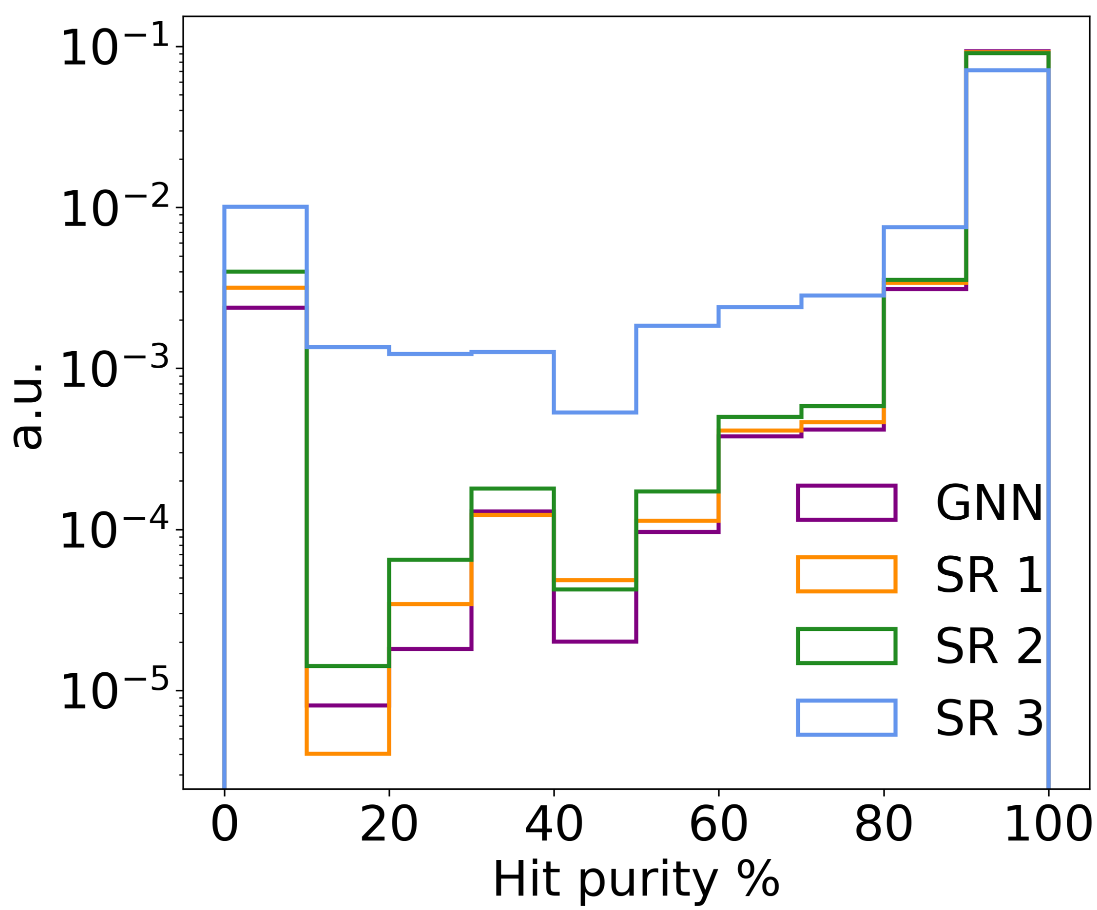


	GNN	SR1	SR2	SR3
r cut	0.05	0.2	0.3	0.4
bandwidth	0.7	0.7	0.6	0.7
ε_s cluster	96%	95%	93%	86%
r_b cluster	94%	94%	93%	71%
ε_s triplets	95%	94%	91%	82%
r_b triplets	99%	98%	98%	90%

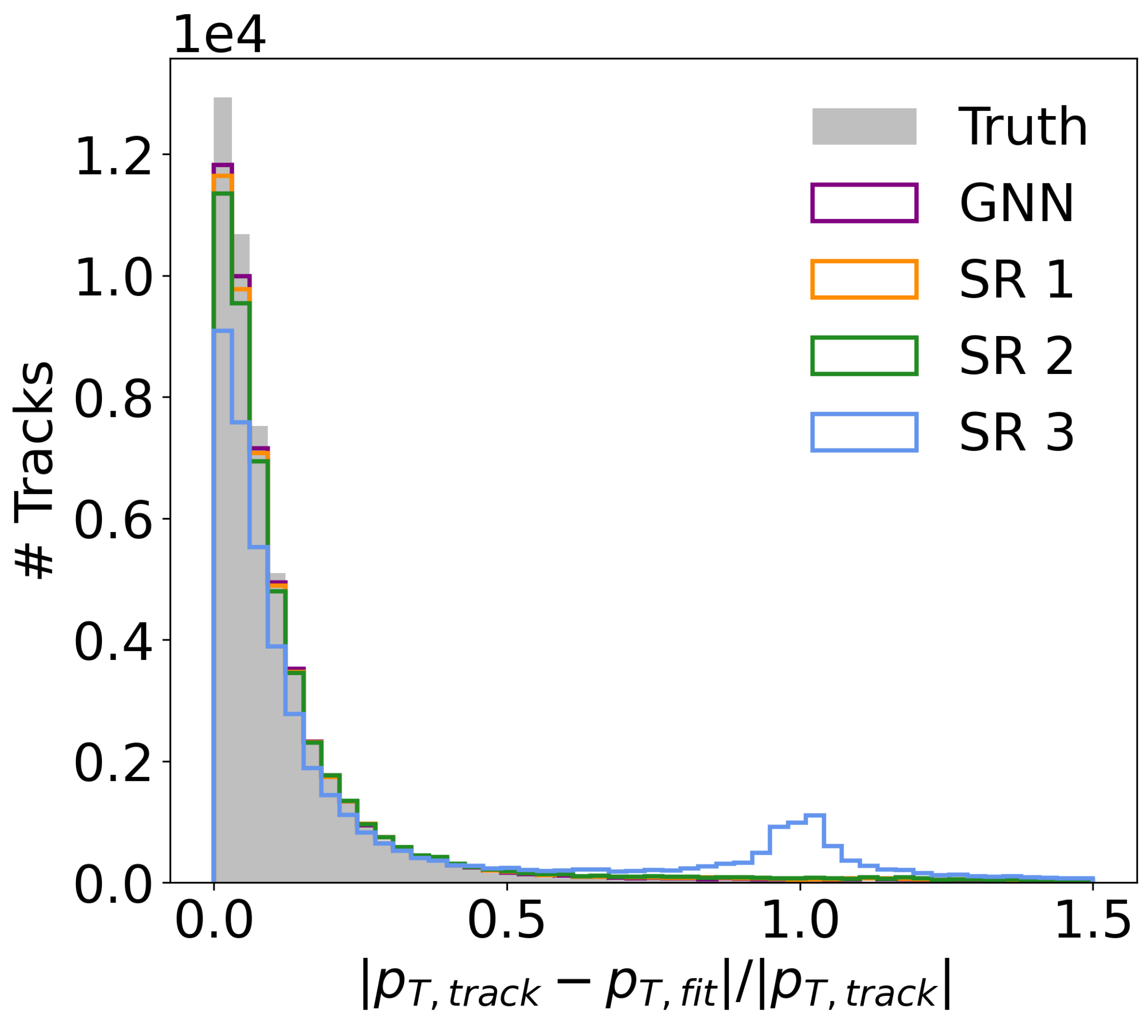


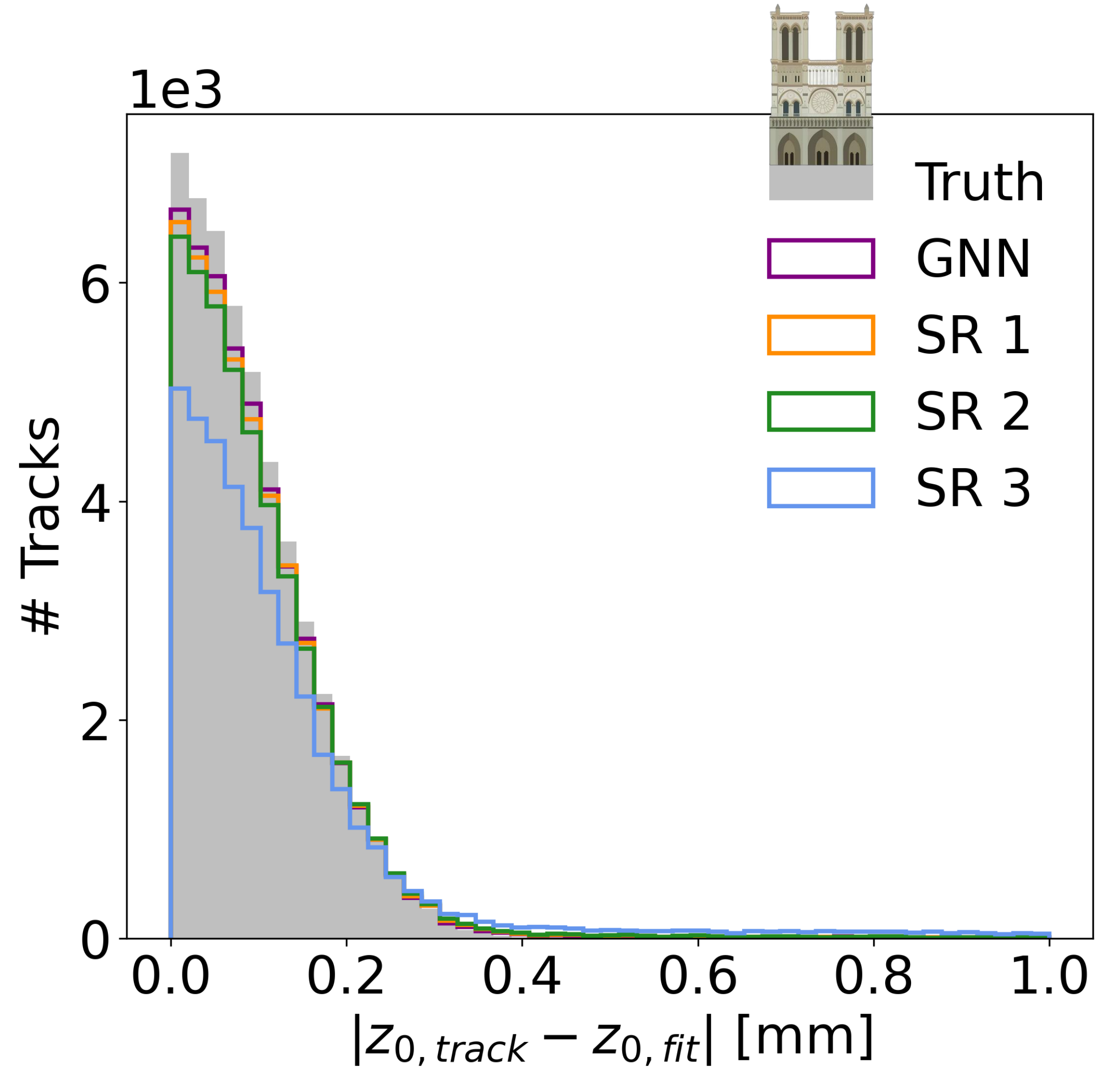
3. Track fitting

- If only 1 hit per layer, perform final fits for p_T and z_0
- Else, iteratively remove hits based on the χ^2 of fit



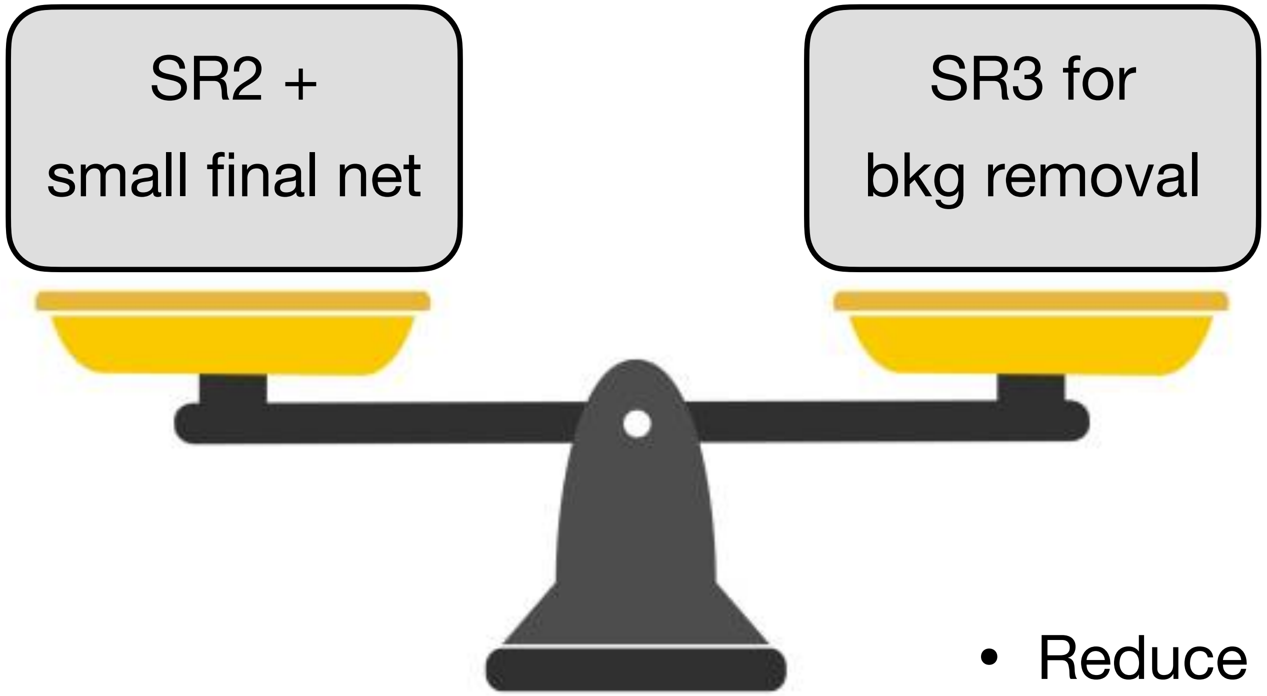
Residuals





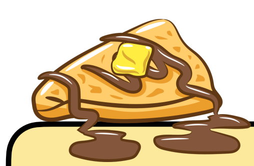
- Truth fit refers to fit using only signal hits
- Very good performance for SR1 and SR2
- Failed circle fits give nearly 0 p_T leading to the bump at 1 in SR3

One more thing …



- Good perfomance
- Few step postprocessing

- Reduce goal to plain bkg removal
- 1k triplets for SR3 after clustering (~13k without filter)
- Even 70% bkg removal would help a lot!



Test on more realistic dataset …






Did you find them all?