Music, **Neighbors** and **tracking**

Sabrina Amrouche, Tobias Golling,
Andreas Salzburger, Moritz Kiehn
c.amrouche@cern.ch

ATLAS EXPERIMENT

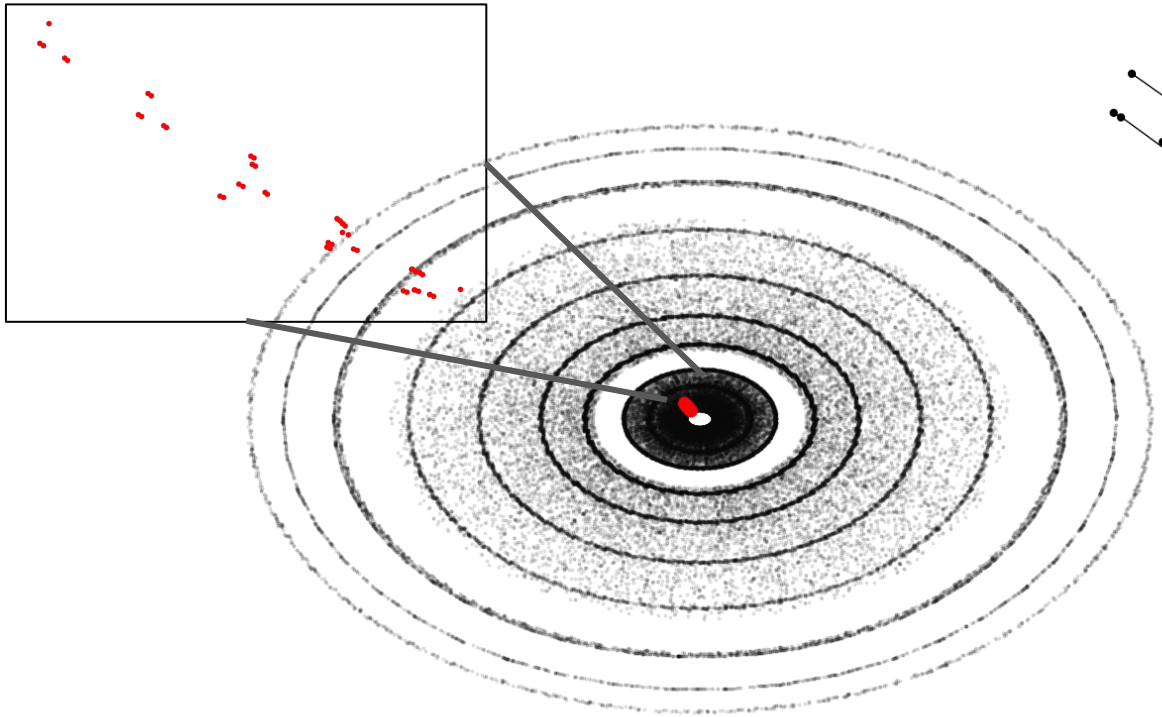
Motivation

- High-luminosity LHC (and future FCC) will bring very high pile-up scenarios
- Average 200 proton-proton collisions for HL-LHC (up to 1000 proton-proton collisions for FFC)
- Combinatorics becomes very challenging

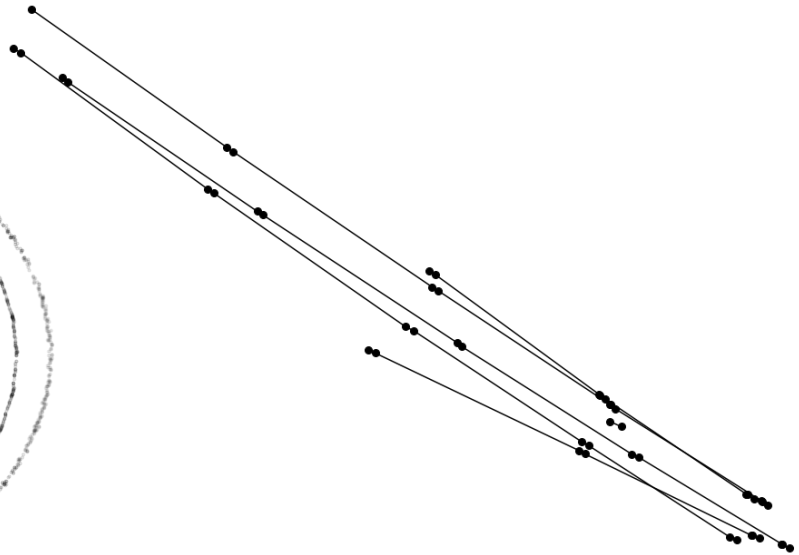
The approach

- Reduce combinatorial complexity : define regions
- Cluster inside regions

(1) Translate dataset to small sets

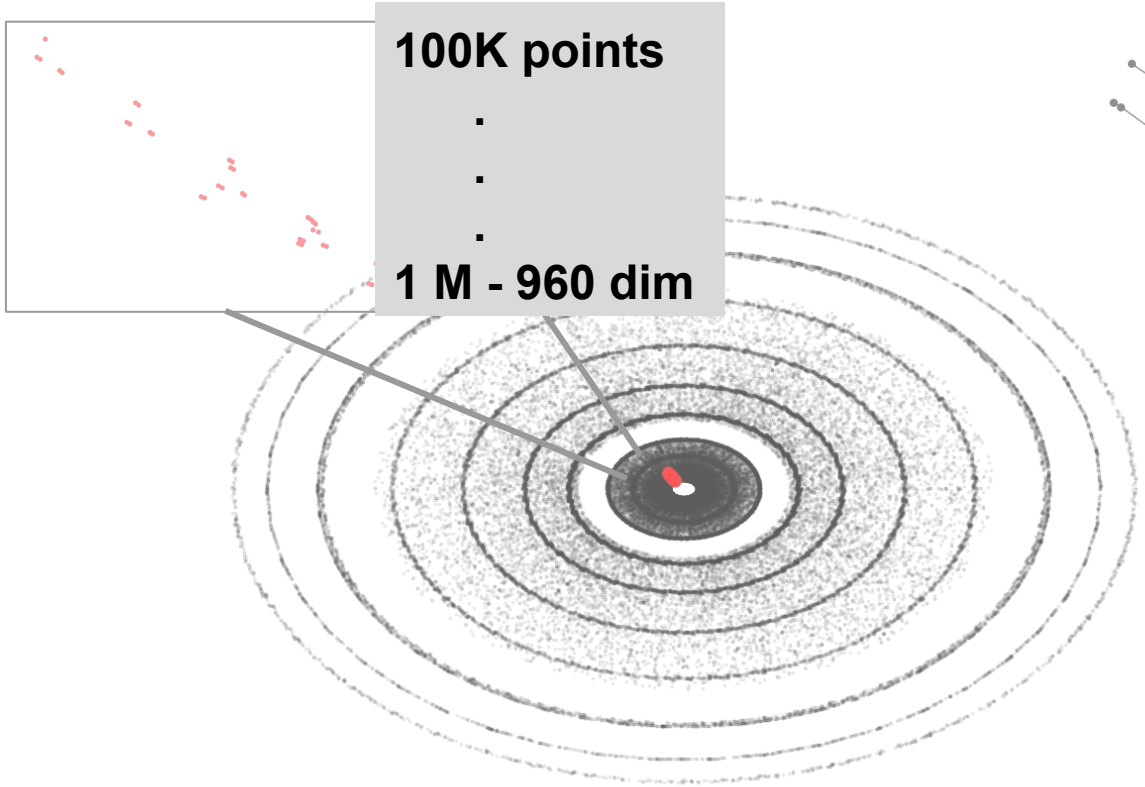


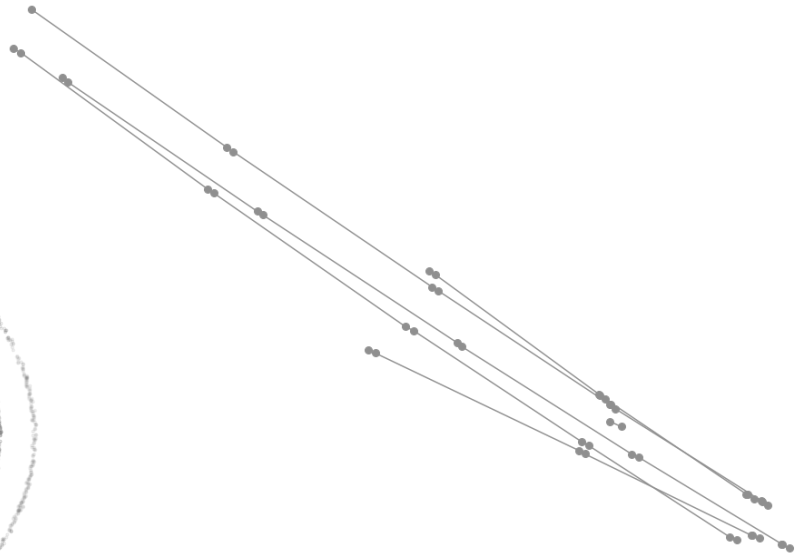
(2) Tracking in small set



(1) Translate **dataset** to small sets

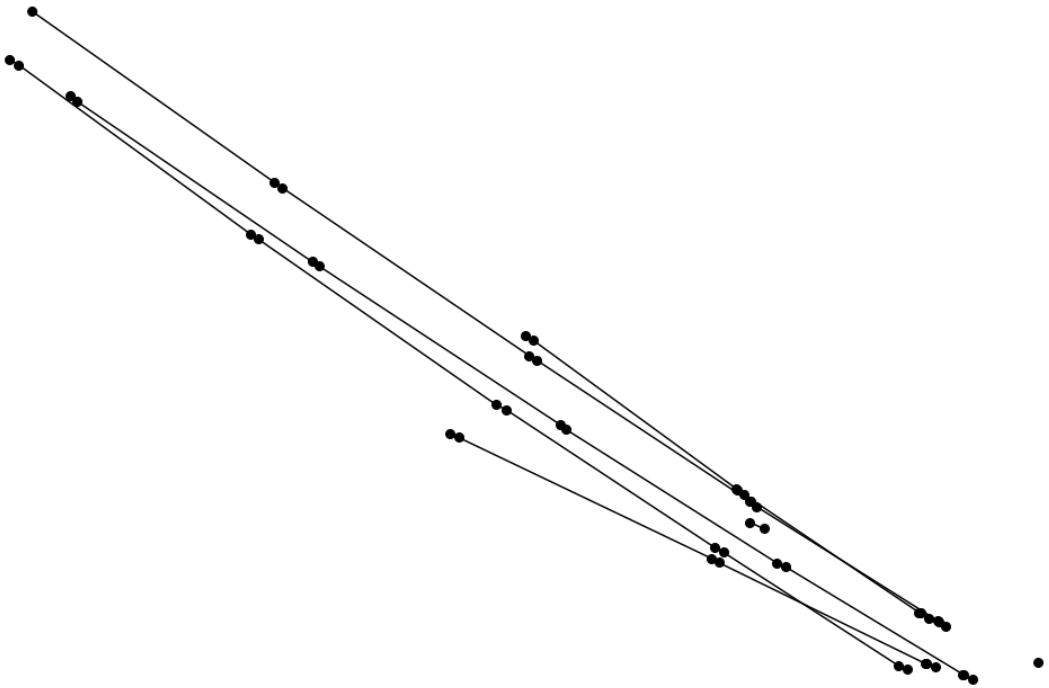
(2) Tracking in small set





(2) Tracking in Small set

(2) Tracking in Small set



Spotify

- Accueil
- Rechercher
- Bibliothèque

ÉCOUTÉ RÉCEMMENT

Valencià.
PLAYLIST

Valencia
ALBUM

Calm Vibes
PLAYLIST

Deep House Relax
PLAYLIST

Pop Klassiker
PLAYLIST

Installer l'appli

greysab

Valencià.

irnemercury

LECTURE

588 TITRES

El Meu Poble	Txarango • Som Riu	4:24
Governant	Txarango • Som Riu	4:02
Corazón Viajero	Txarango • Som Riu	3:50
Camp de Batalla	Txarango • Som Riu	3:42
Com Dues Gotes d'Aigua	Txarango • Som Riu	3:59
Tant de Bo	Txarango • Som Riu	3:13
No t'Adormis	Txarango • Som Riu	4:01
La Dansa del Vestit	Txarango • Benvinguts al Llarg Viatge	3:37
Batega	Txarango • Som Riu	4:34
Compta amb Mi		3:08

El Meu Poble
Txarango

1:56 4:24

8

Spotify

Accueil

Rechercher

Bibliothèque

ÉCOUTÉ RÉCEMMENT

Valencià.
PLAYLIST

Valencia
ALBUM

Calm Vibes

El Meu Poble Txarango • Som Riu	4:24
Governant Txarango • Som Riu	4:02
Corazón Viajero Txarango • Som Riu	3:50
Camp de Batalla Txarango • Som Riu	3:42
Com Dues Gotes d'Aigua Txarango • Som Riu	3:59
Tant de Be	3:13

Autoplay

Autoplay similar songs when your music ends.

SHOW ADVANCED SETTINGS

LOG OUT

About Spotify

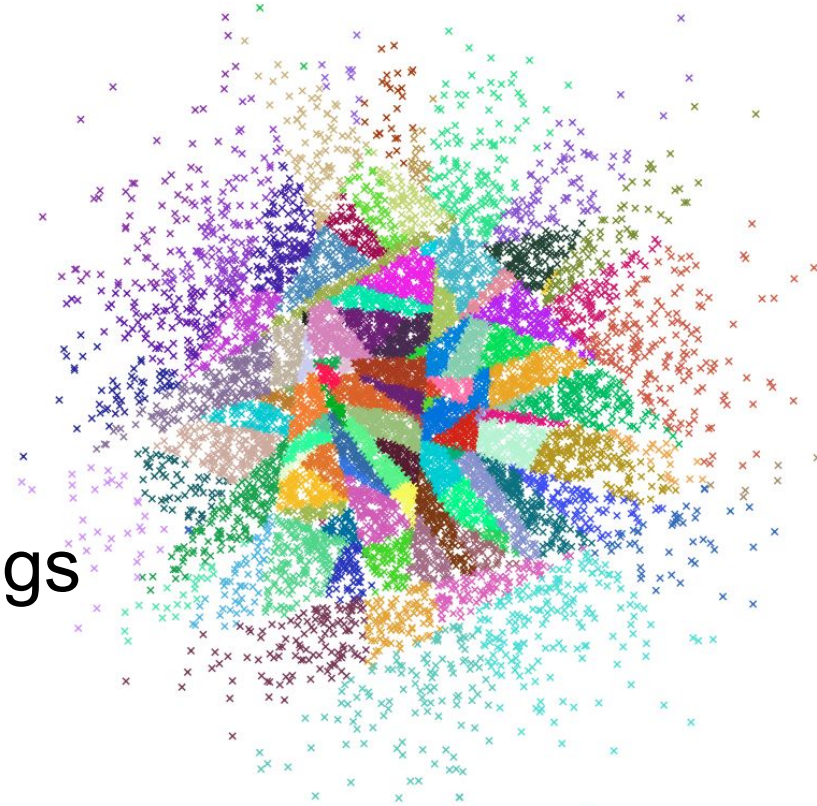
greysab

El Meu Poble
Txarango

1:56 4:24

9

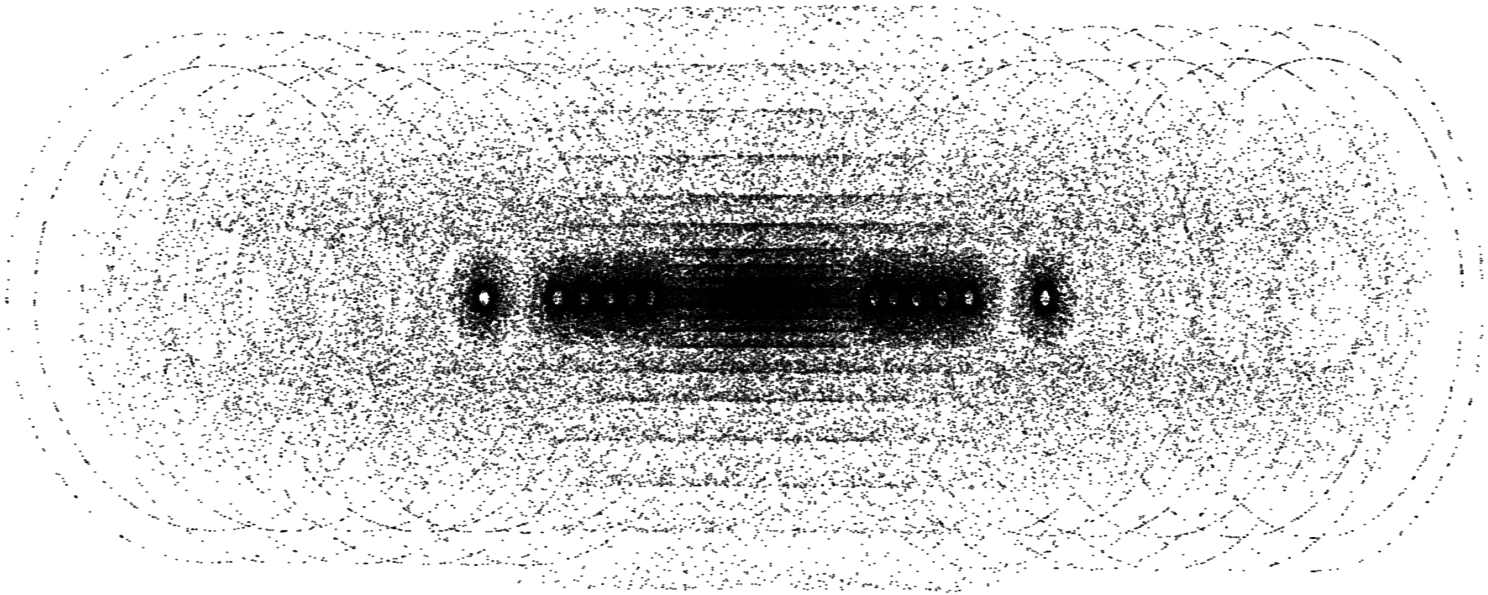
Spotify / **annoy**



- “Many millions of songs”
- **< 0.1ms** to get **n** similar songs

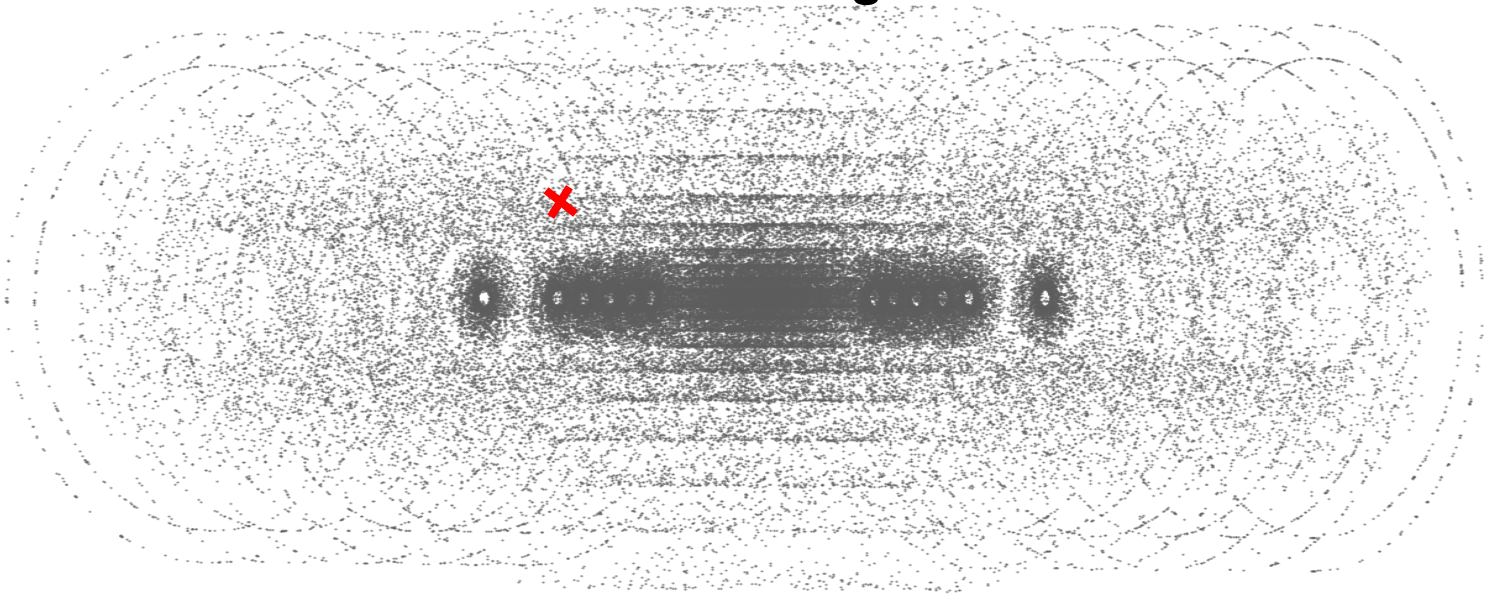
  [high-dimensional space]
- Unsupervised

Approximate Nearest Neighbors (ANN)



A **bucket** of neighbors

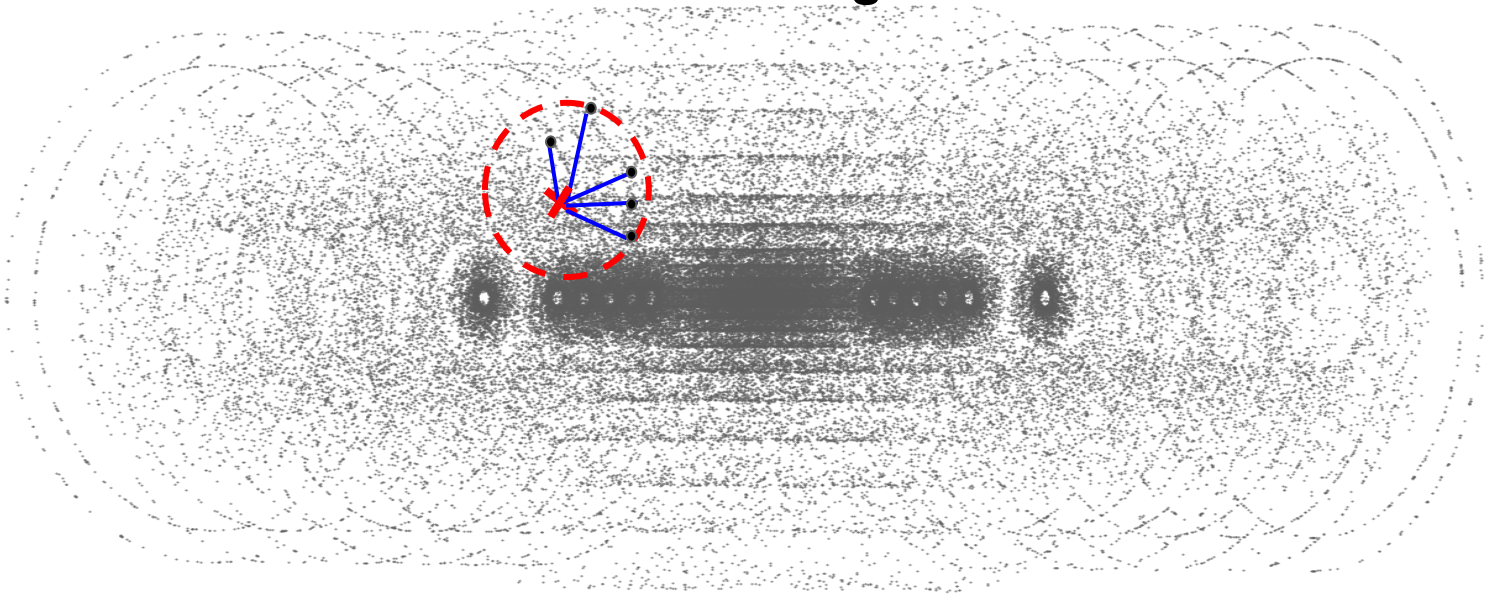
5 nearest neighbors in N dimensions



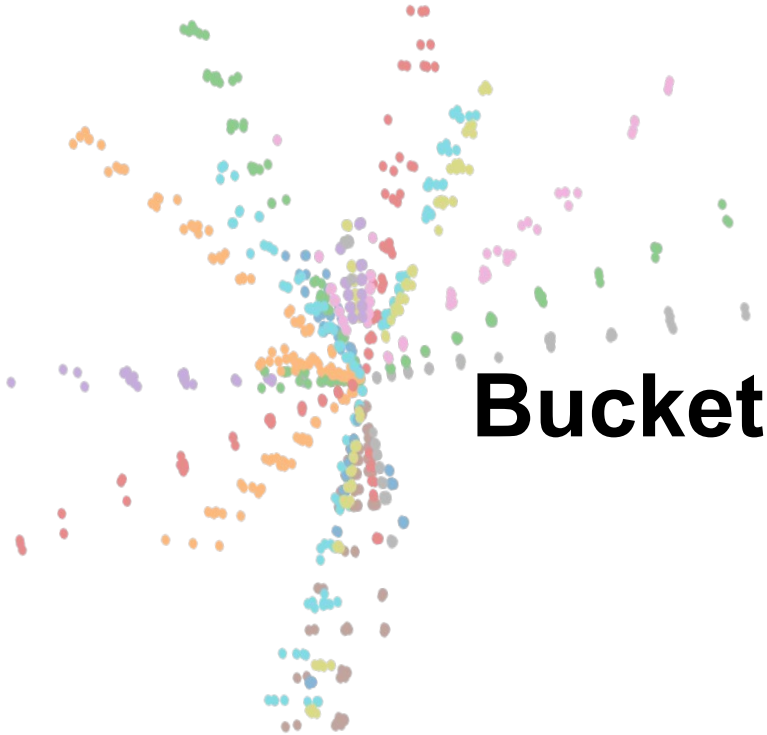
Traditionally, we do an exhaustive search - brute force

A **bucket** of neighbors

5 nearest neighbors in N dimensions

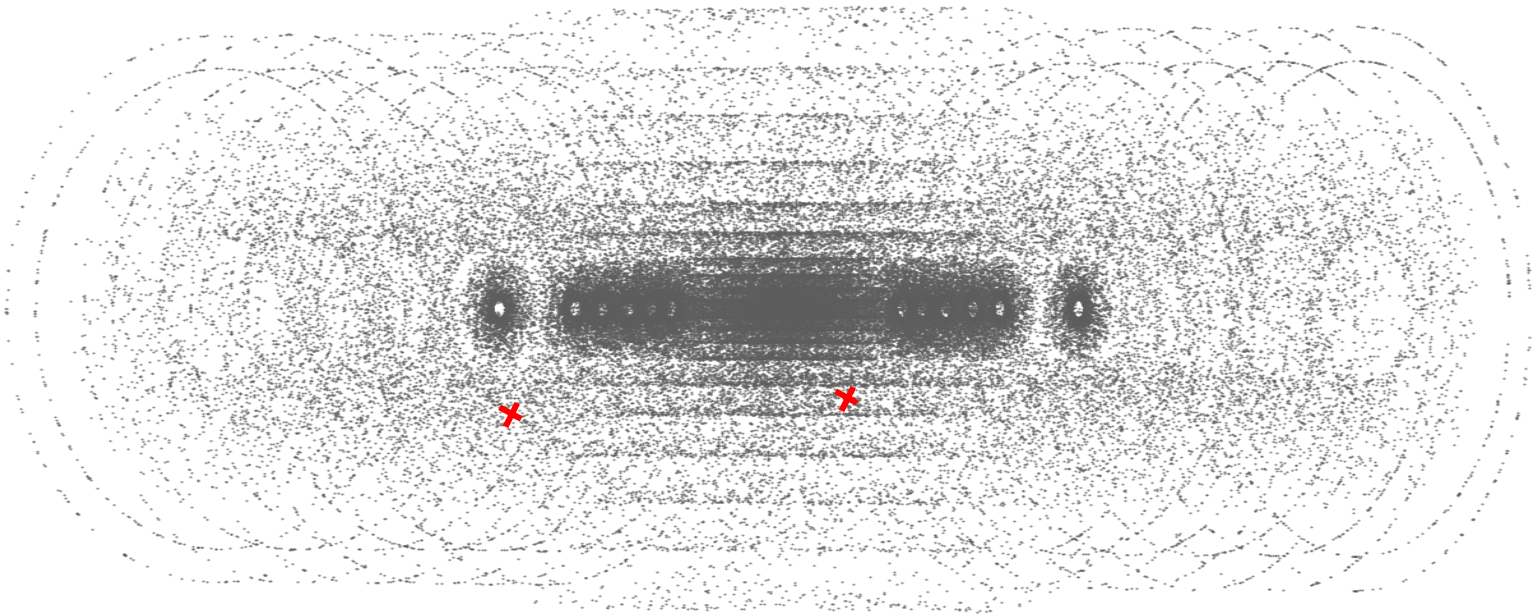


Traditionally, we do an exhaustive search - brute force

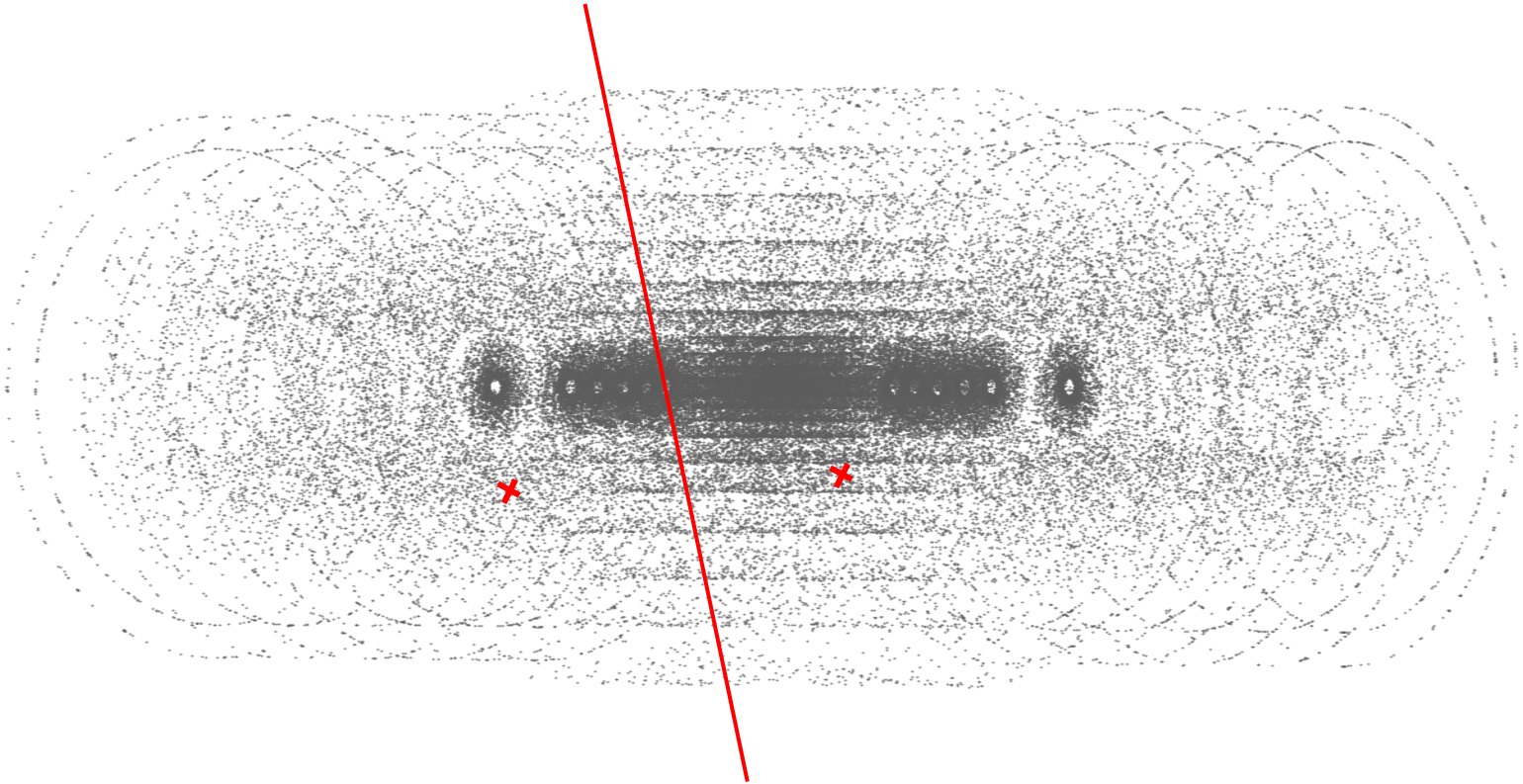


Bucketing mechanism

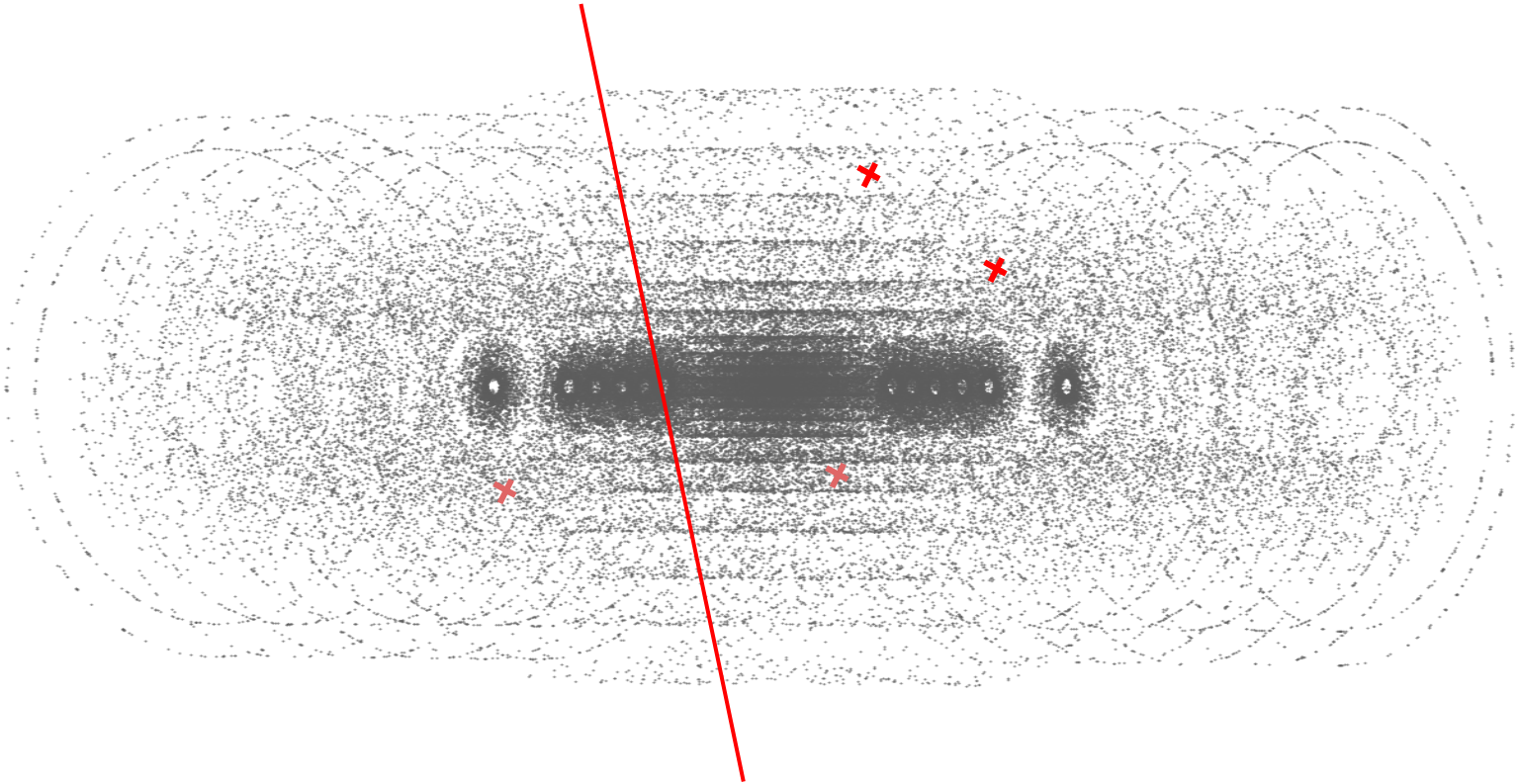
A **bucket** of neighbors



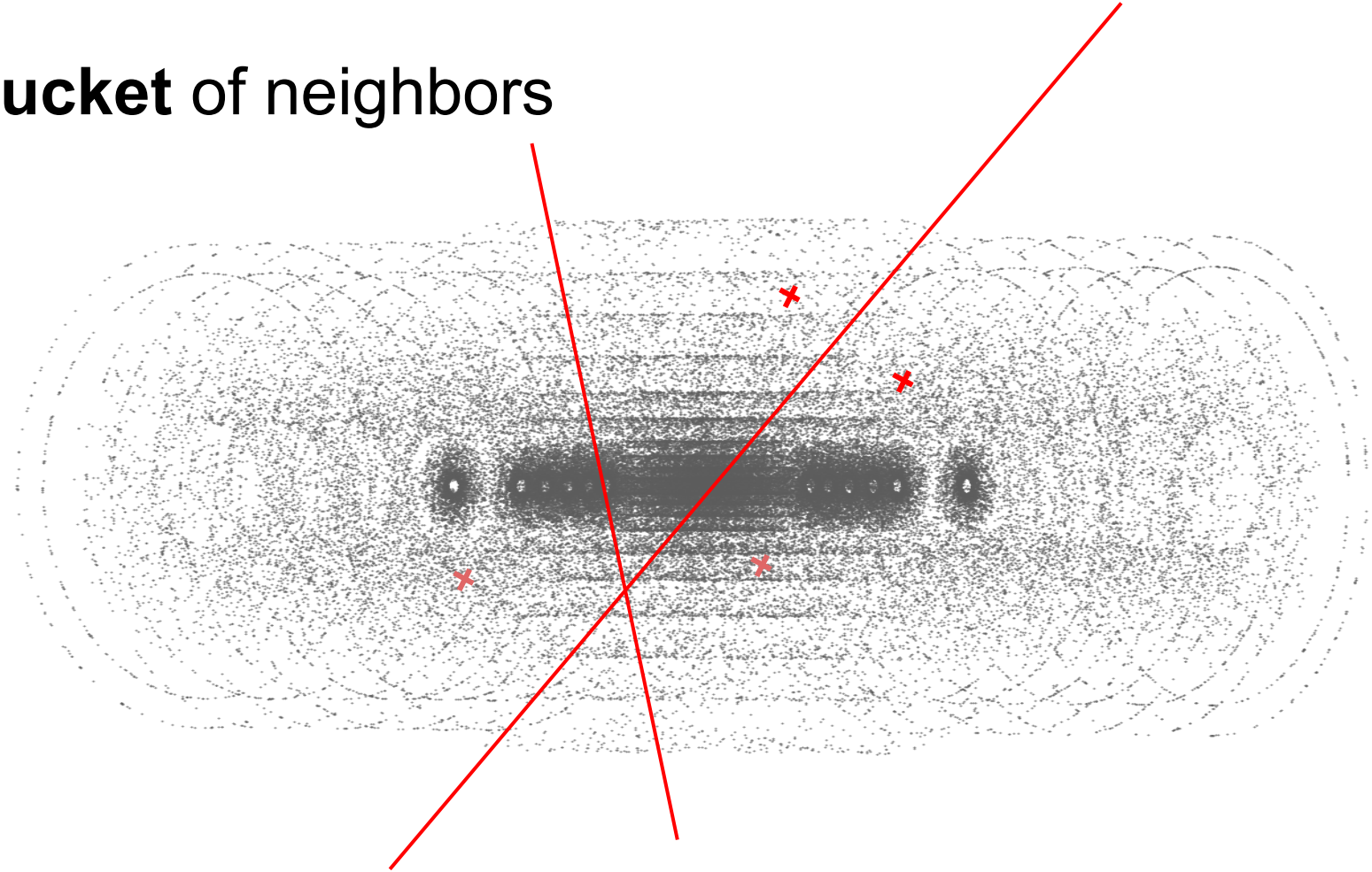
A **bucket** of neighbors



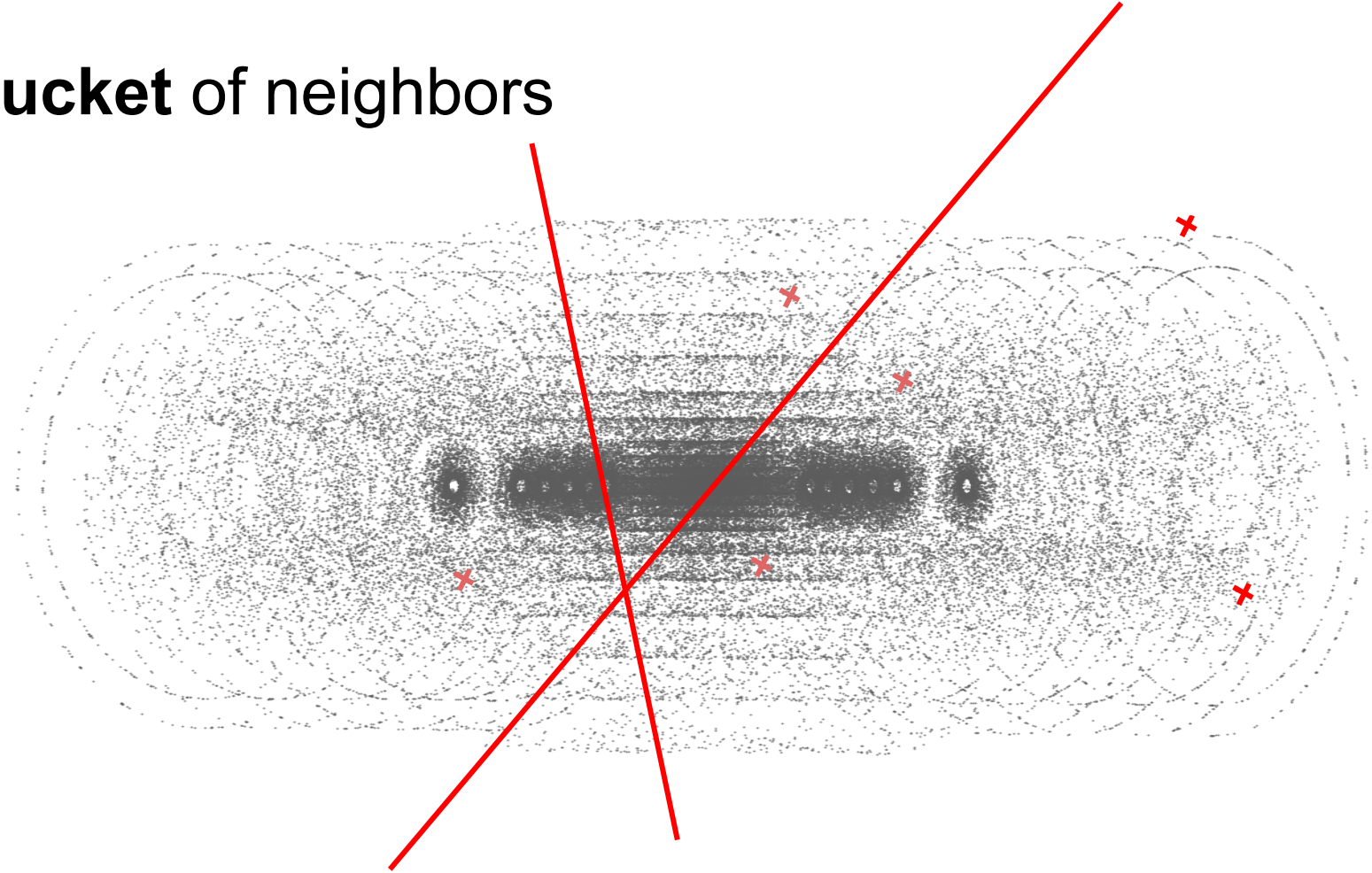
A **bucket** of neighbors



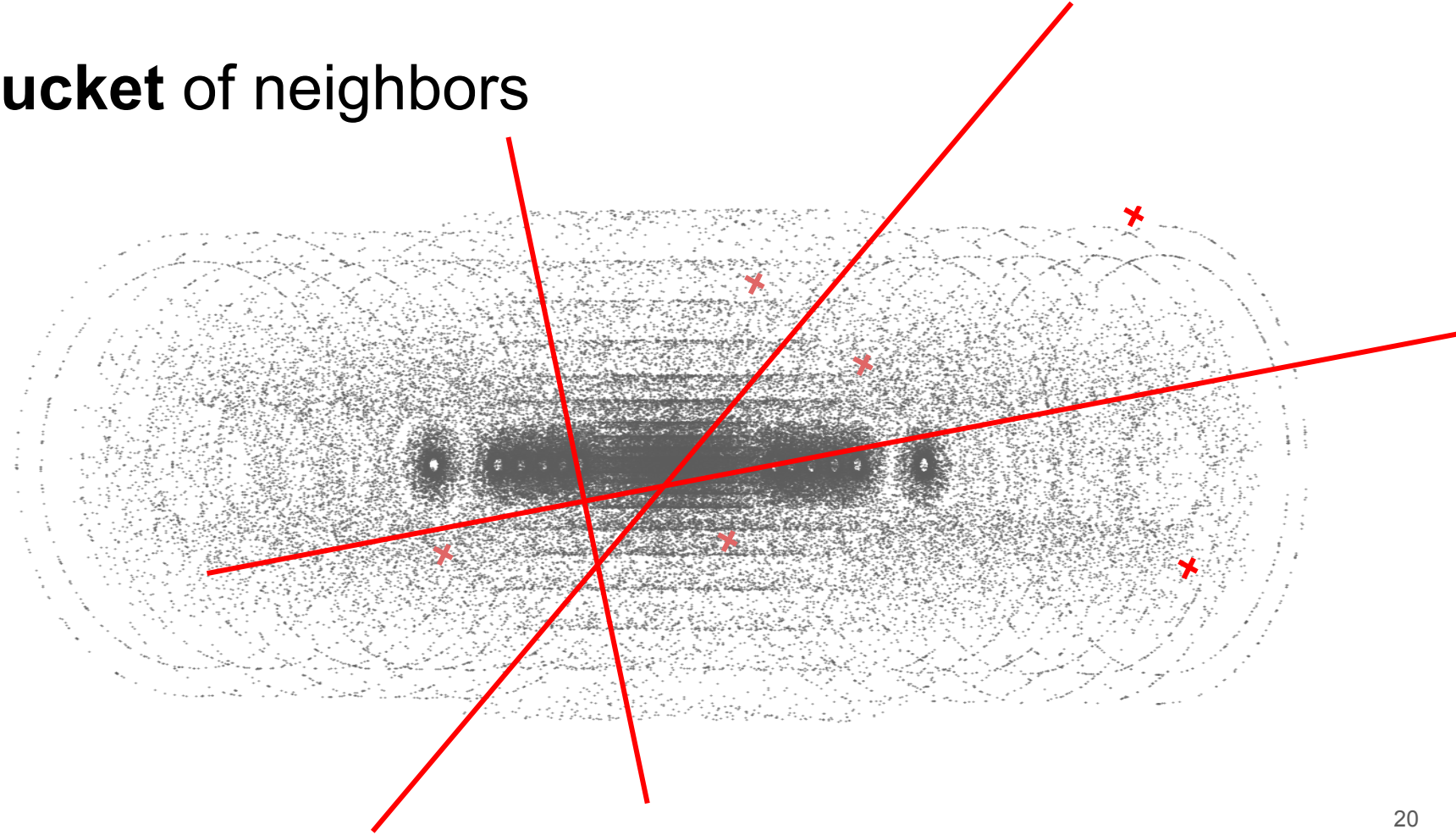
A **bucket** of neighbors



A **bucket** of neighbors

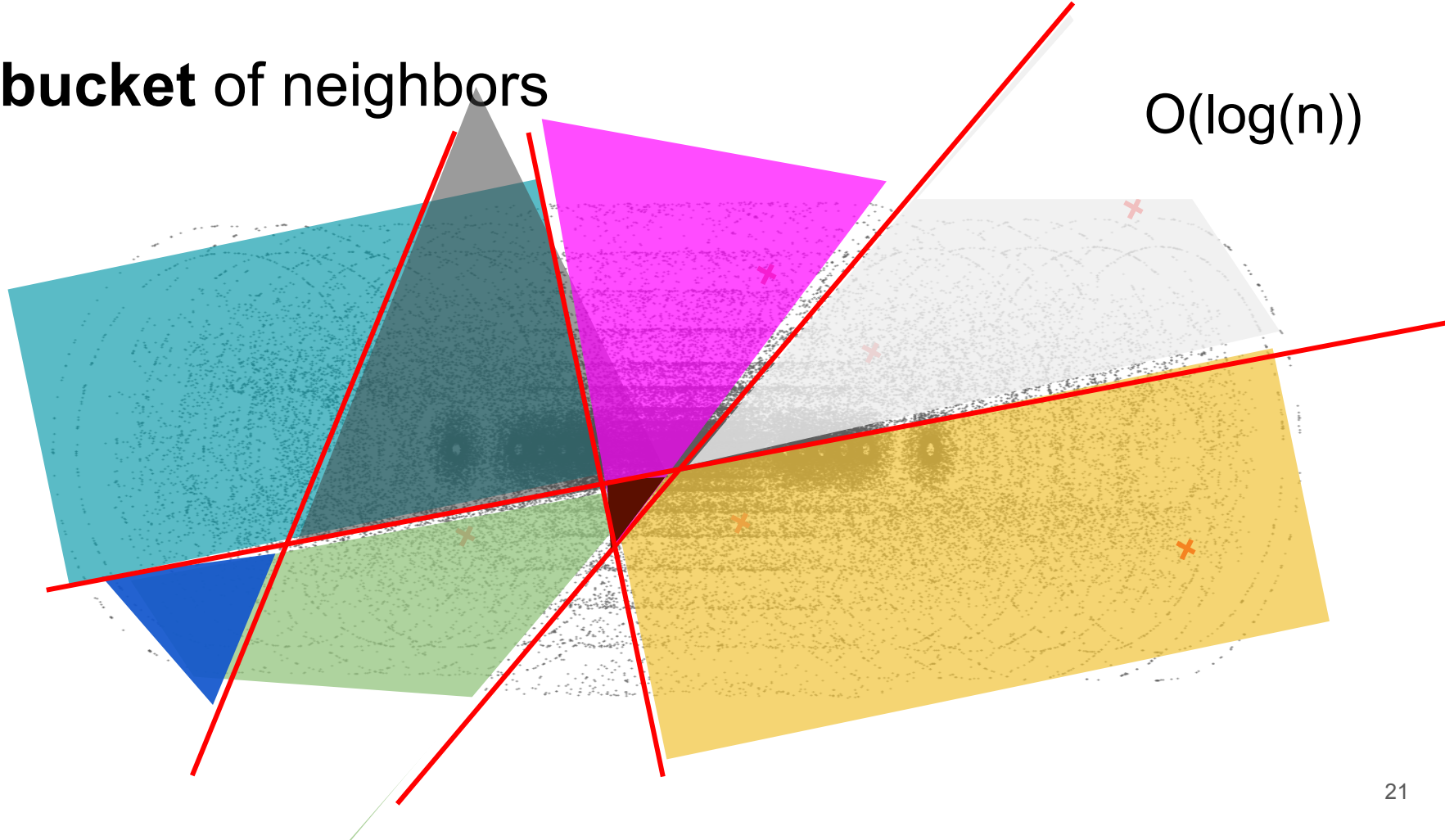


A **bucket** of neighbors

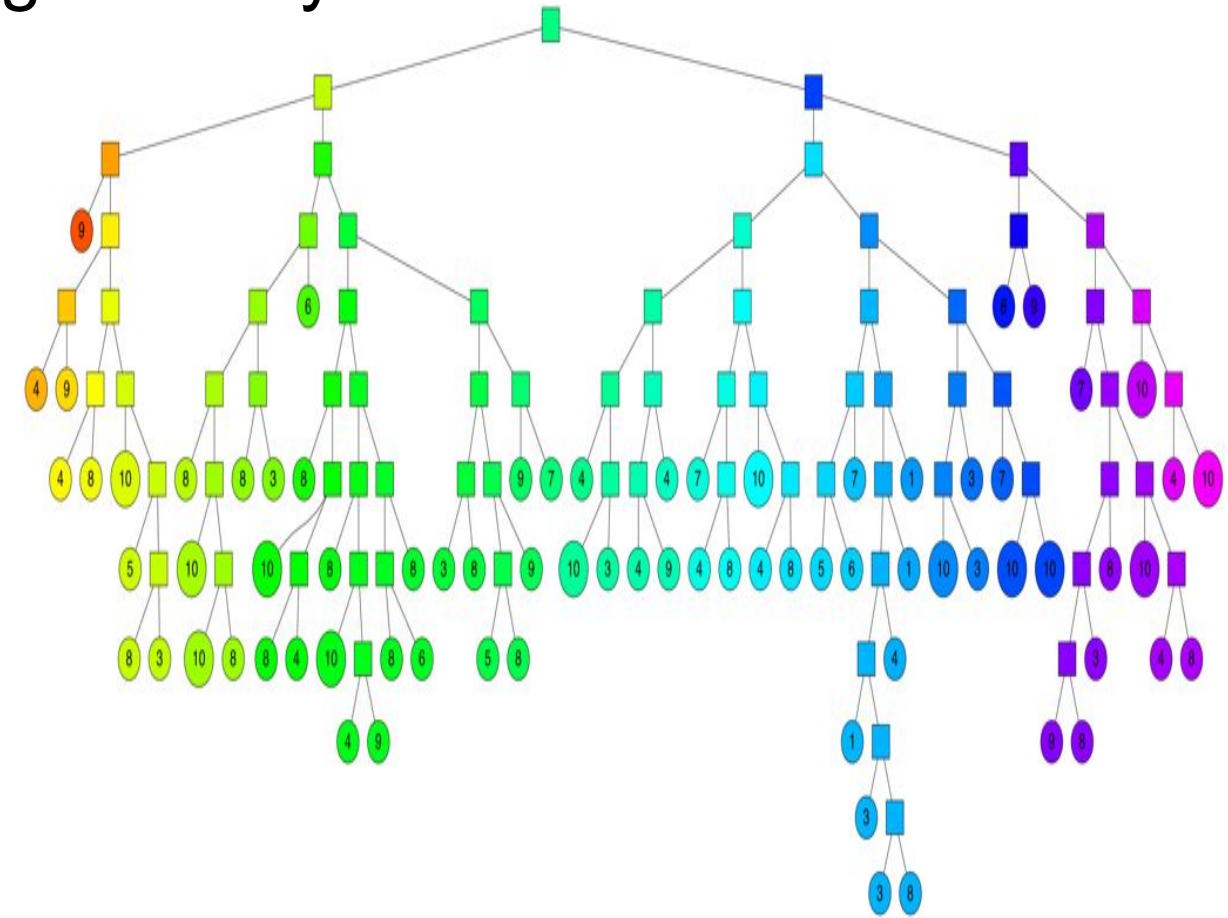


A **bucket** of neighbors

O(log(n))



Building a Binary Tree



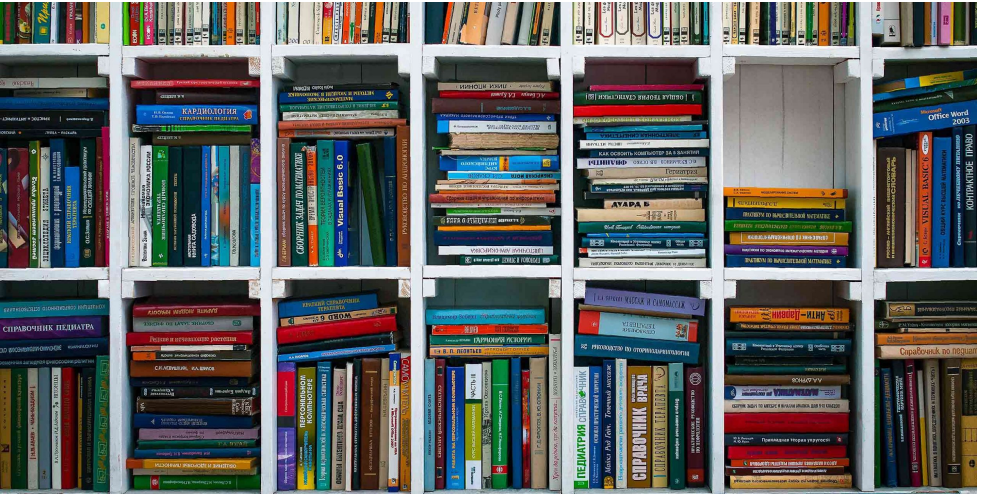
Data structures and algorithms

Data structure and Algorithm

- Some data structures associate with algorithm : trees, heaps, graphs...
- Some structures represent some concepts better than others
- The data structure can significantly reduce complexity

> "Algorithms are like **verbs** and data structures are like **nouns**. An algorithm is just a method for doing something on a computer; a data structure is a <u>layout for memory that represents some sort of data.</u>"
>
> Quora, **What's the difference between data structures and algorithms?**

Search complexity



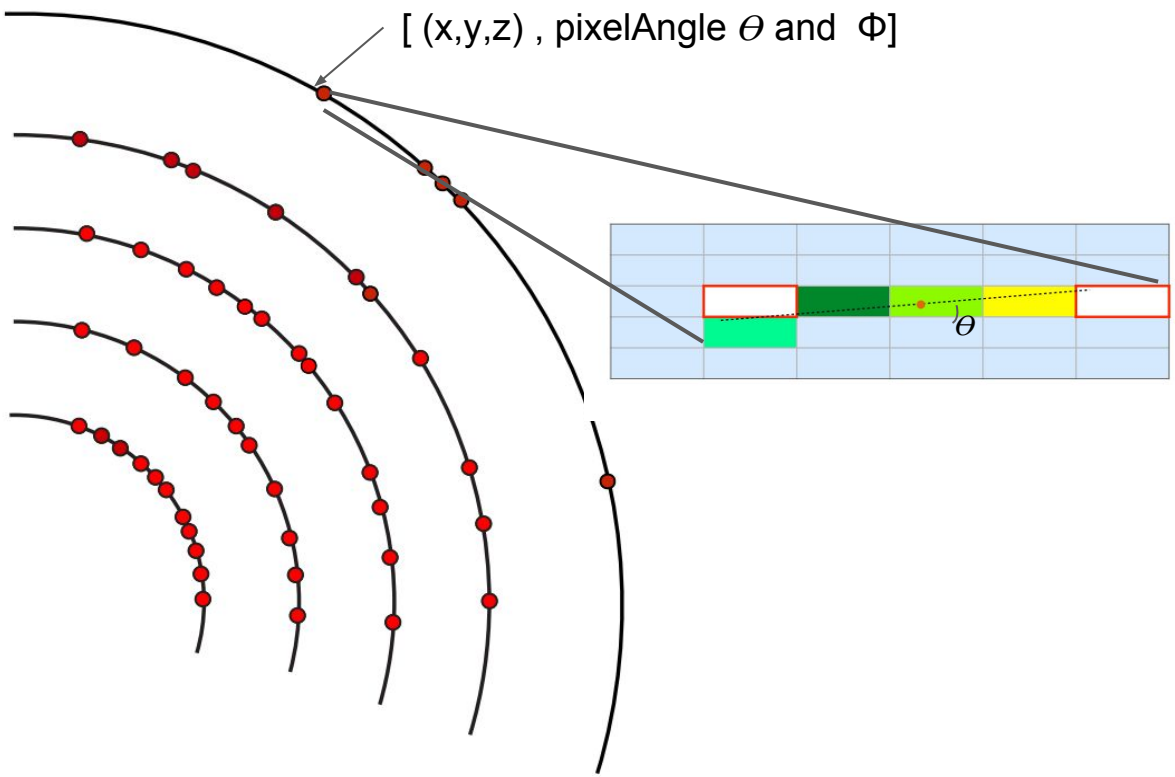
vs

Dataset

- 10K particles, 100K points
- 5 features : global x, y, z,

  local Θ, Φ
- Hashing library used :

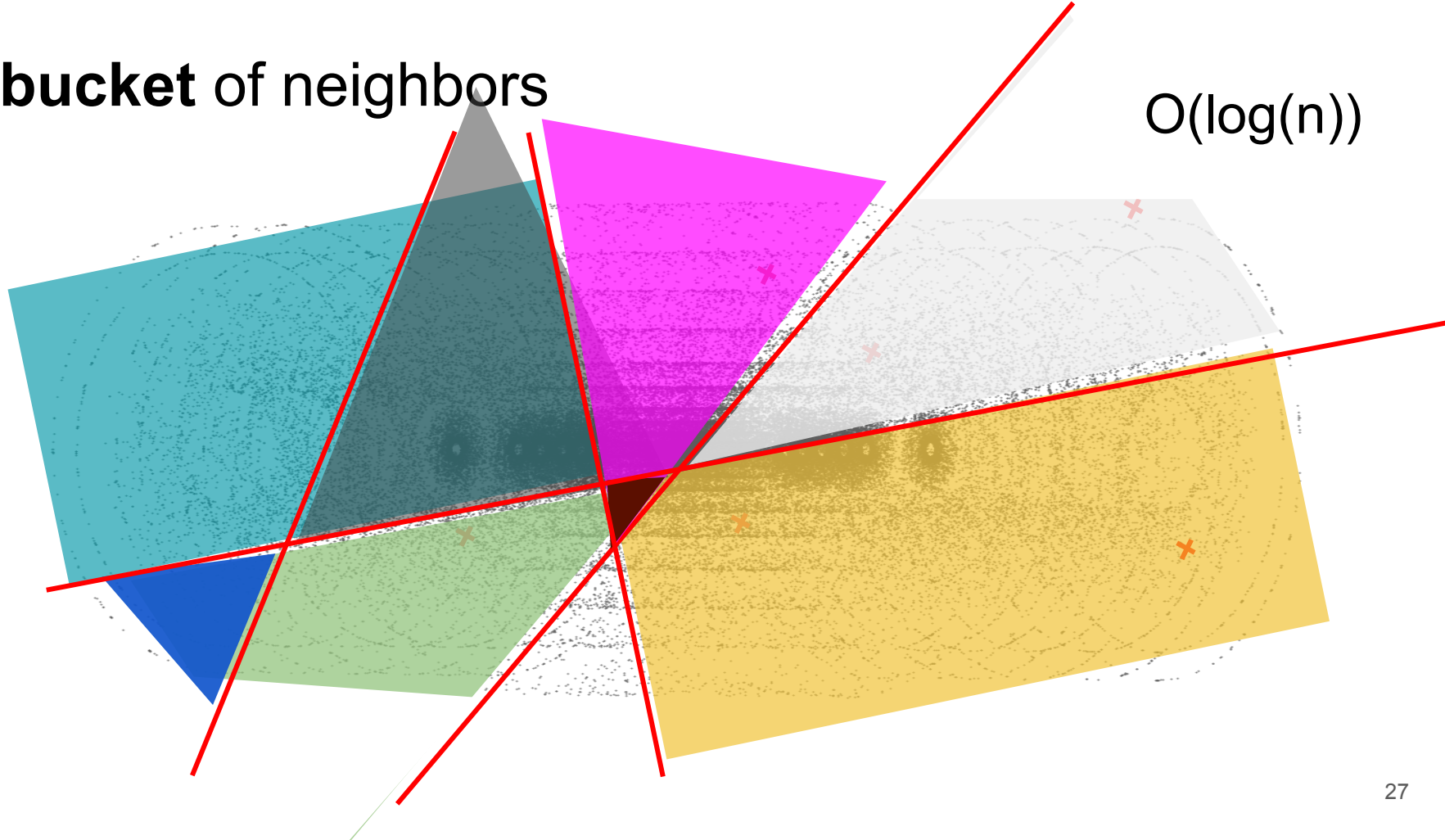
  HNSW, Annoy (spotify)
- Buckets parameters
  - bucket size
  - nb trees (graphs)



Simulated with ACTS, Ttbar event, mu 200

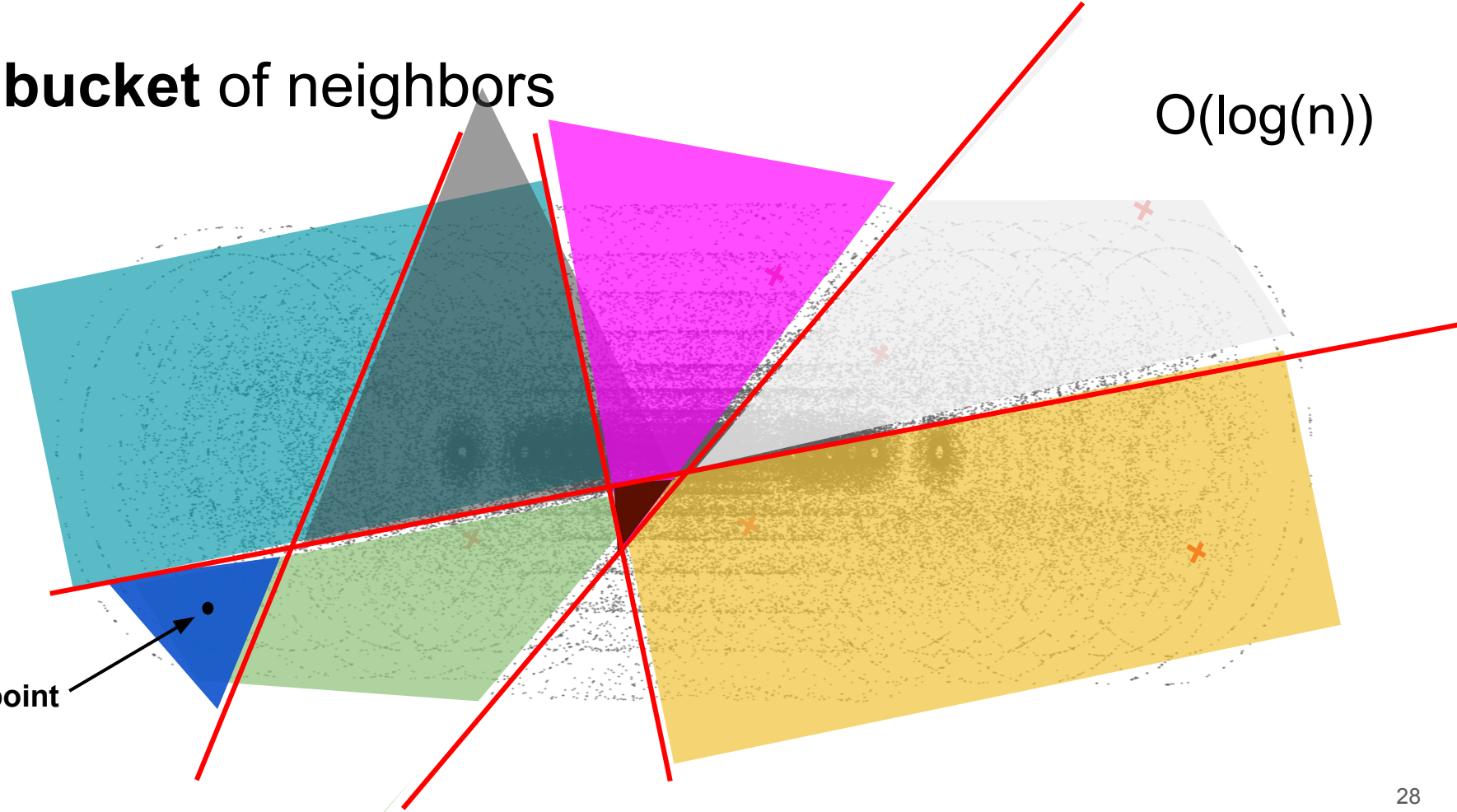
A **bucket** of neighbors

O(log(n))



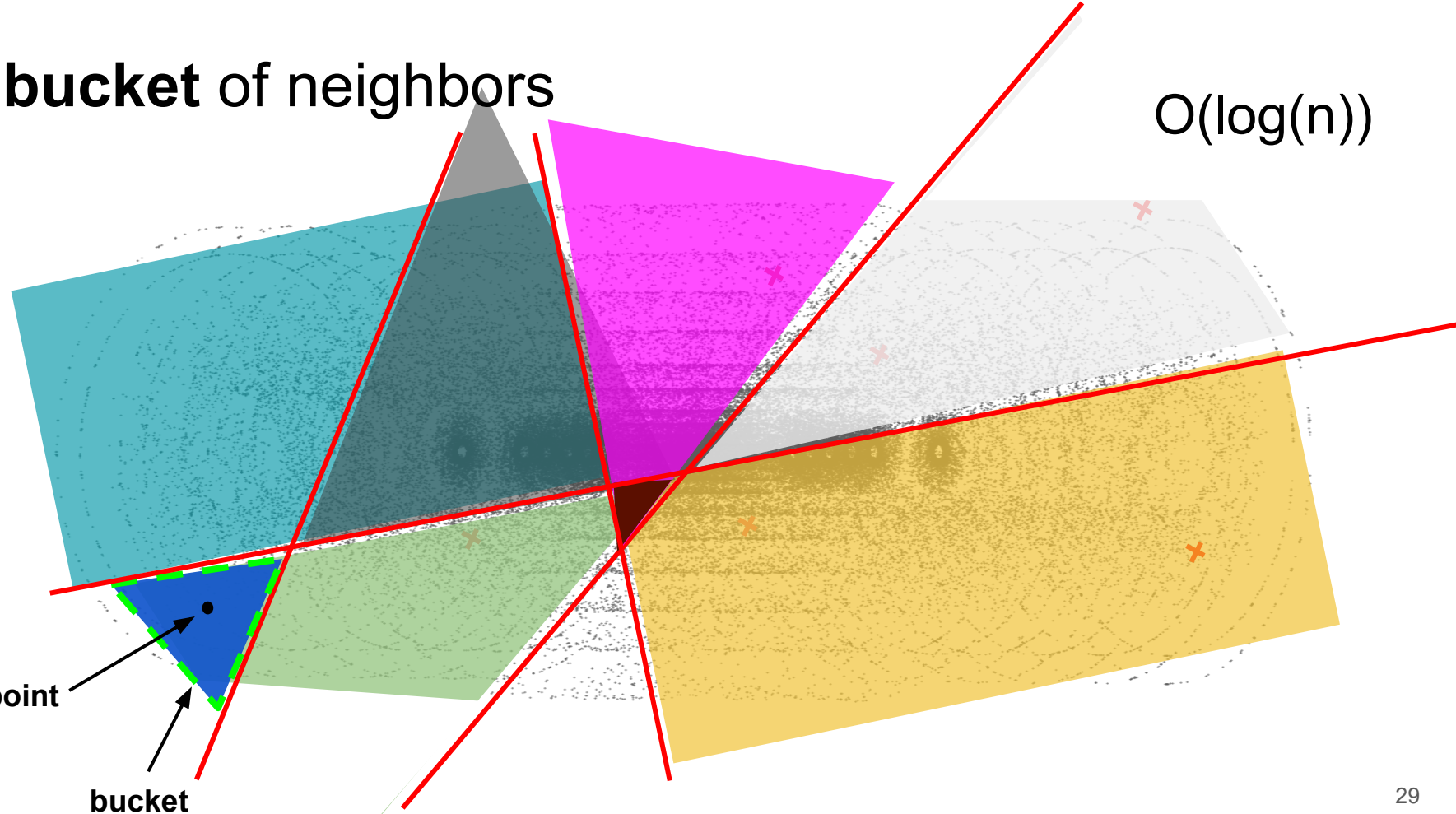
A **bucket** of neighbors

O(log(n))



A **bucket** of neighbors

O(log(n))

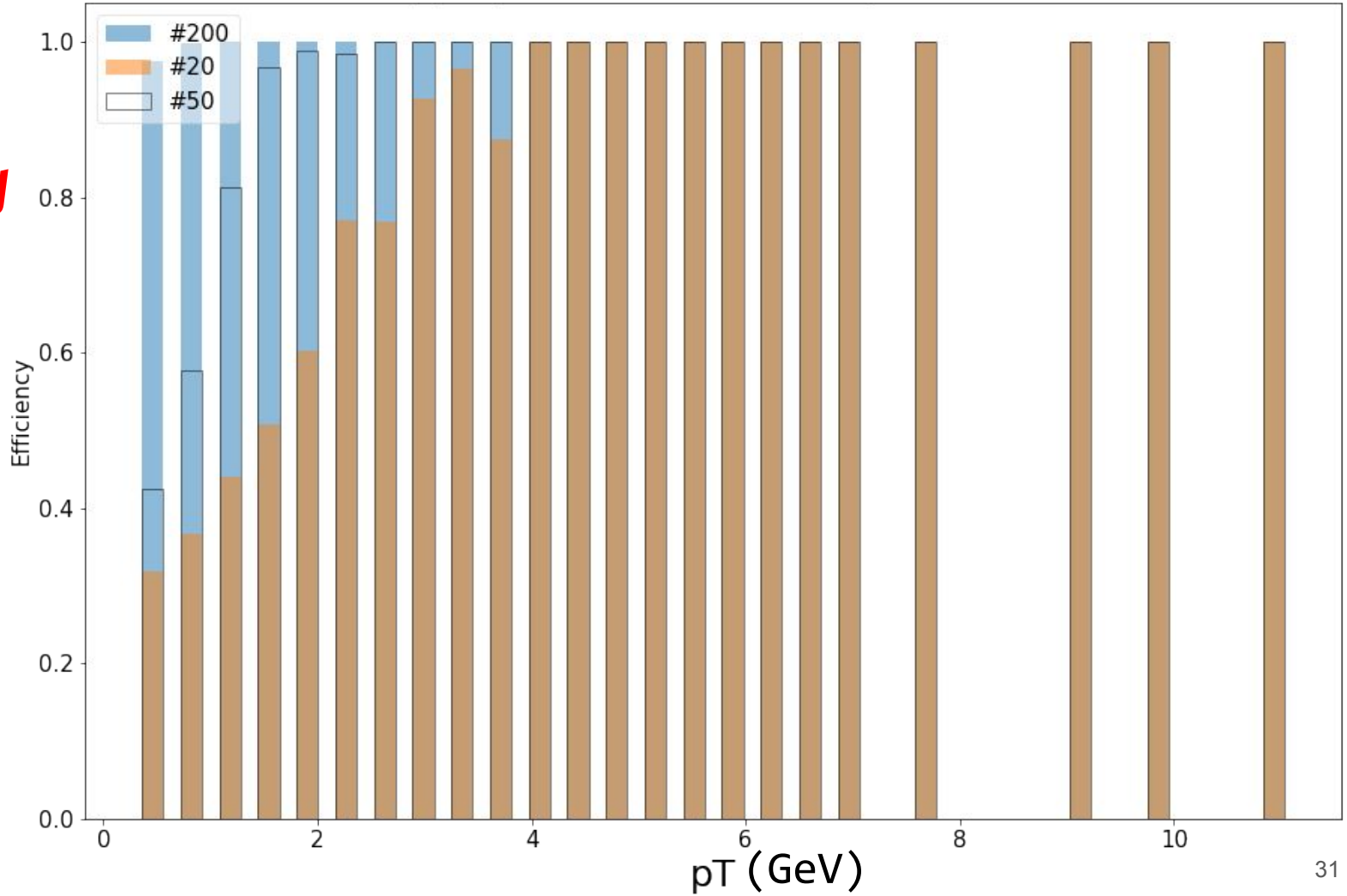


Efficiency

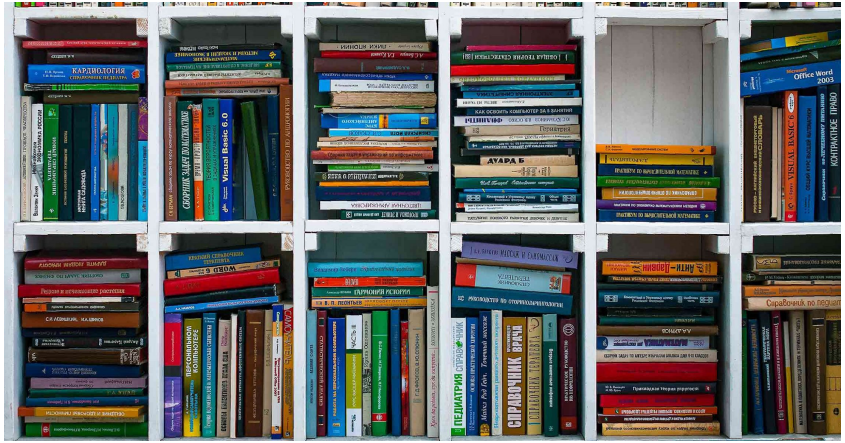
- Using truth information :
  - Count number of trackable particles found in a bucket
  - “Trackable particles” = 8 points (avg 11)
  - Efficiency as :

    nb_trackable_particles_found/ total_trackable_particles (~9K)

0.07 ms per query



Data structure … again

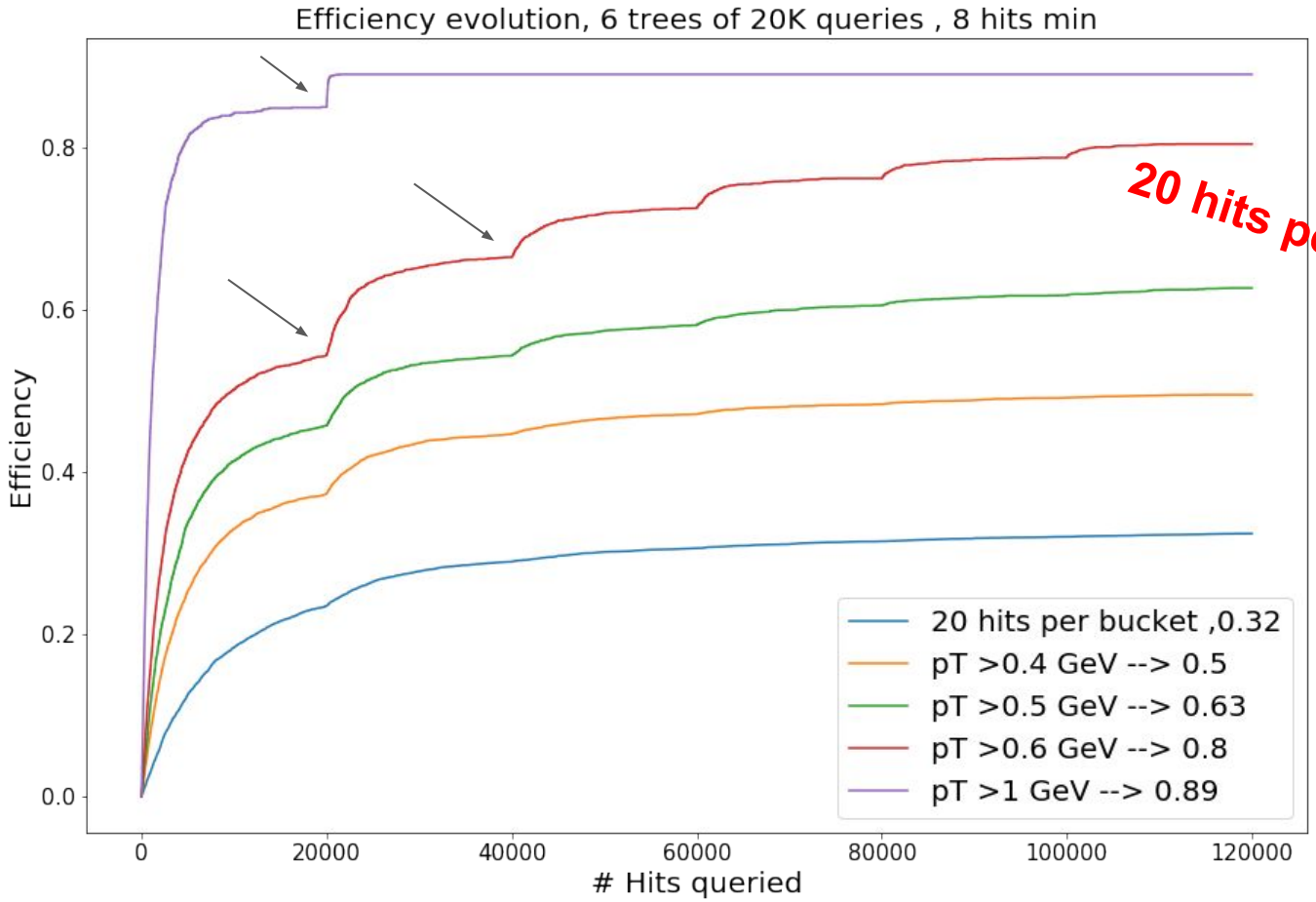


Finding & **removing** particles

Rebuilding data structure

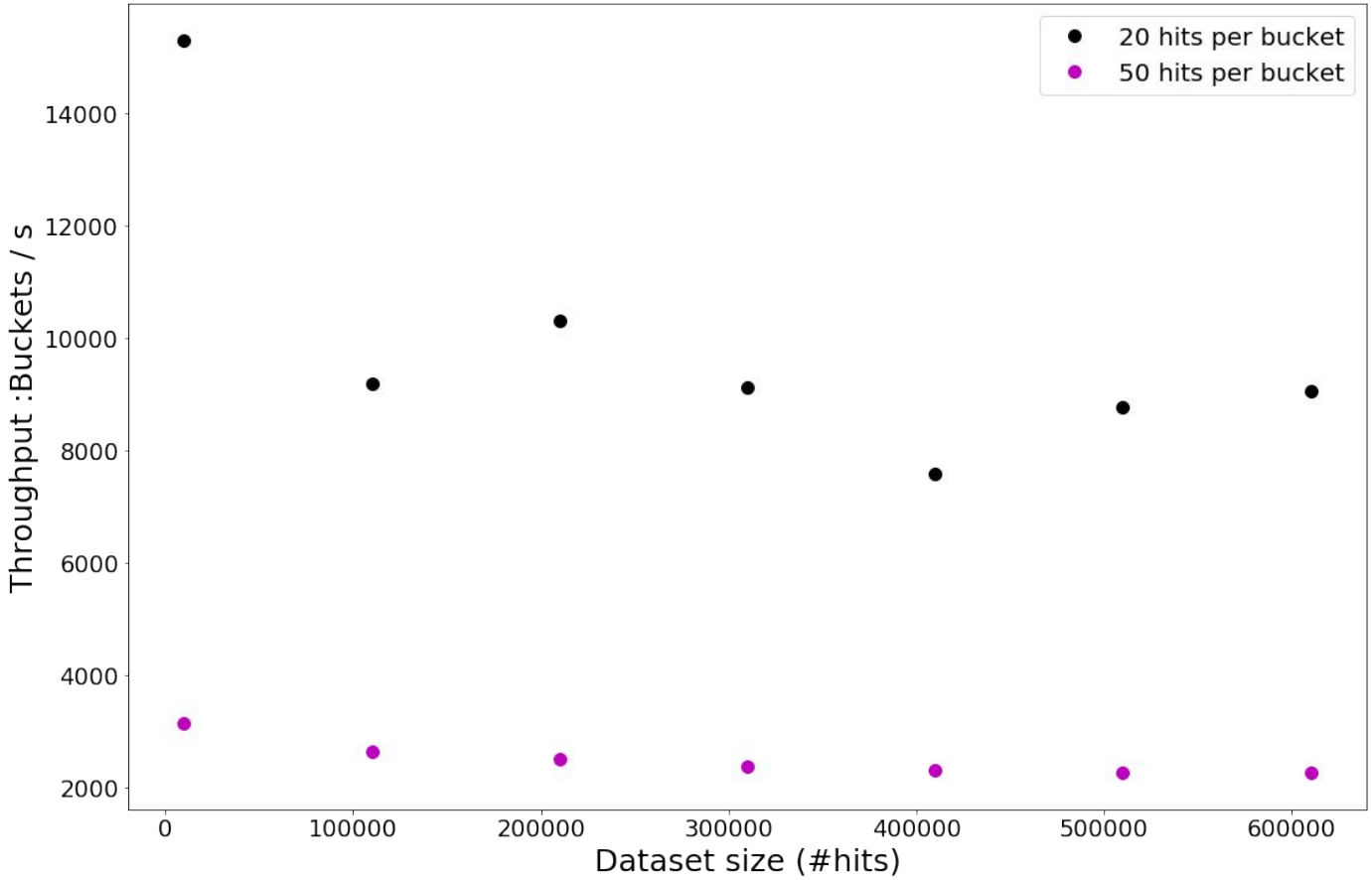
Rebuilding the data structure

0.07 ms per query



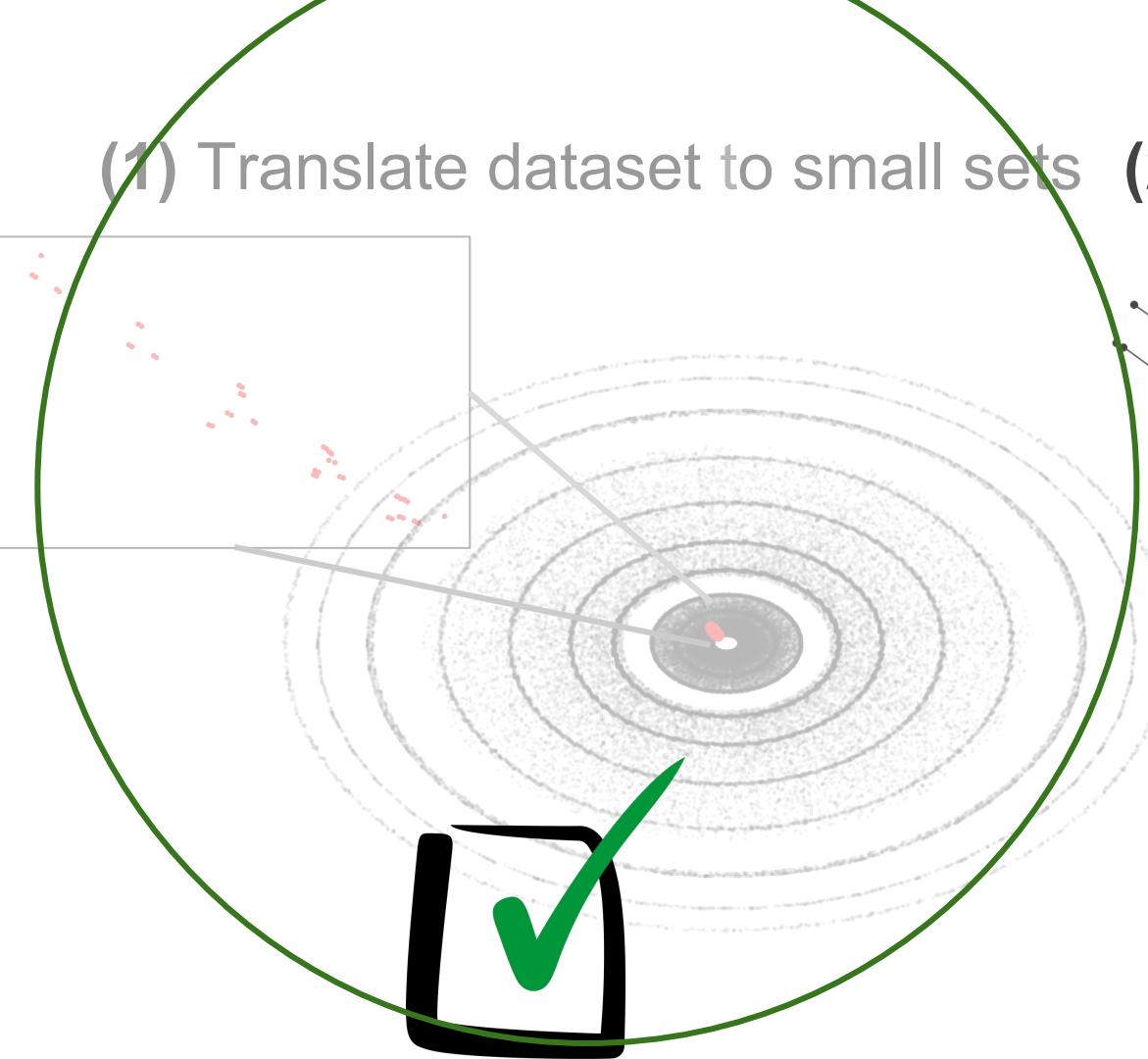
20 hits per bucket

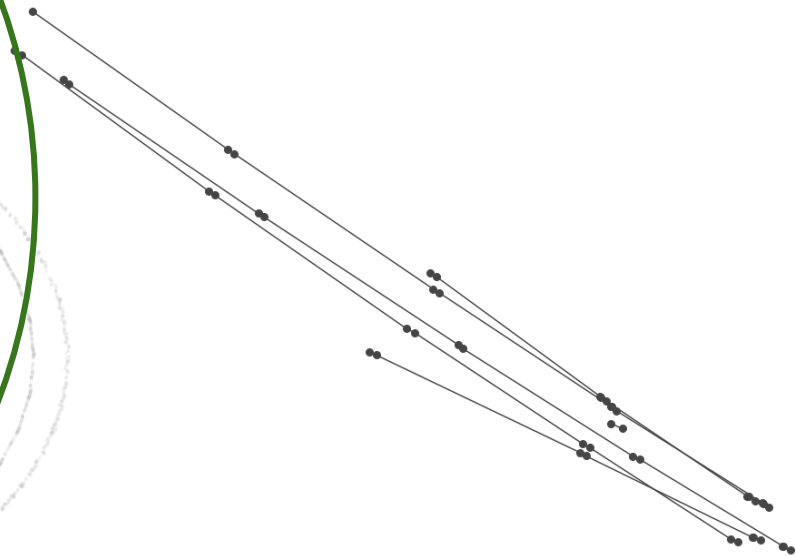
Scaling of bucketing



(1) Translate dataset to small sets

(2) Tracking in small set





Clustering inside buckets

Clustering inside buckets

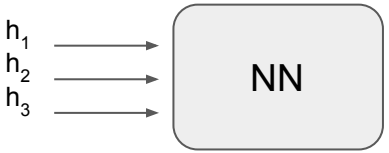
- Complexity low enough for (any) technique to work **fast**...**but**
- Data driven approach has **- many -** advantages
  - A single technique to work hand in hand <u>with data structure</u>
  - Independance towards physics
    - expected to work as well for displaced tracks
    - Low pT, impact parameter…etc
  - Towards non-parametric approaches

Proposed approach

Supervised model trained on **buckets**

evaluating if **N** hits belong together

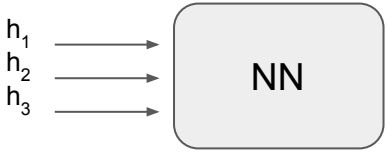
Supervised learning

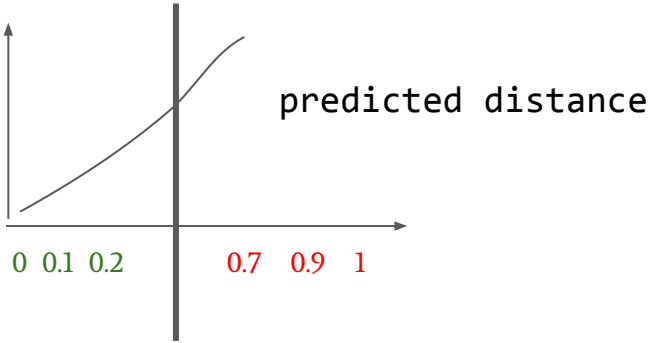


0 same particle

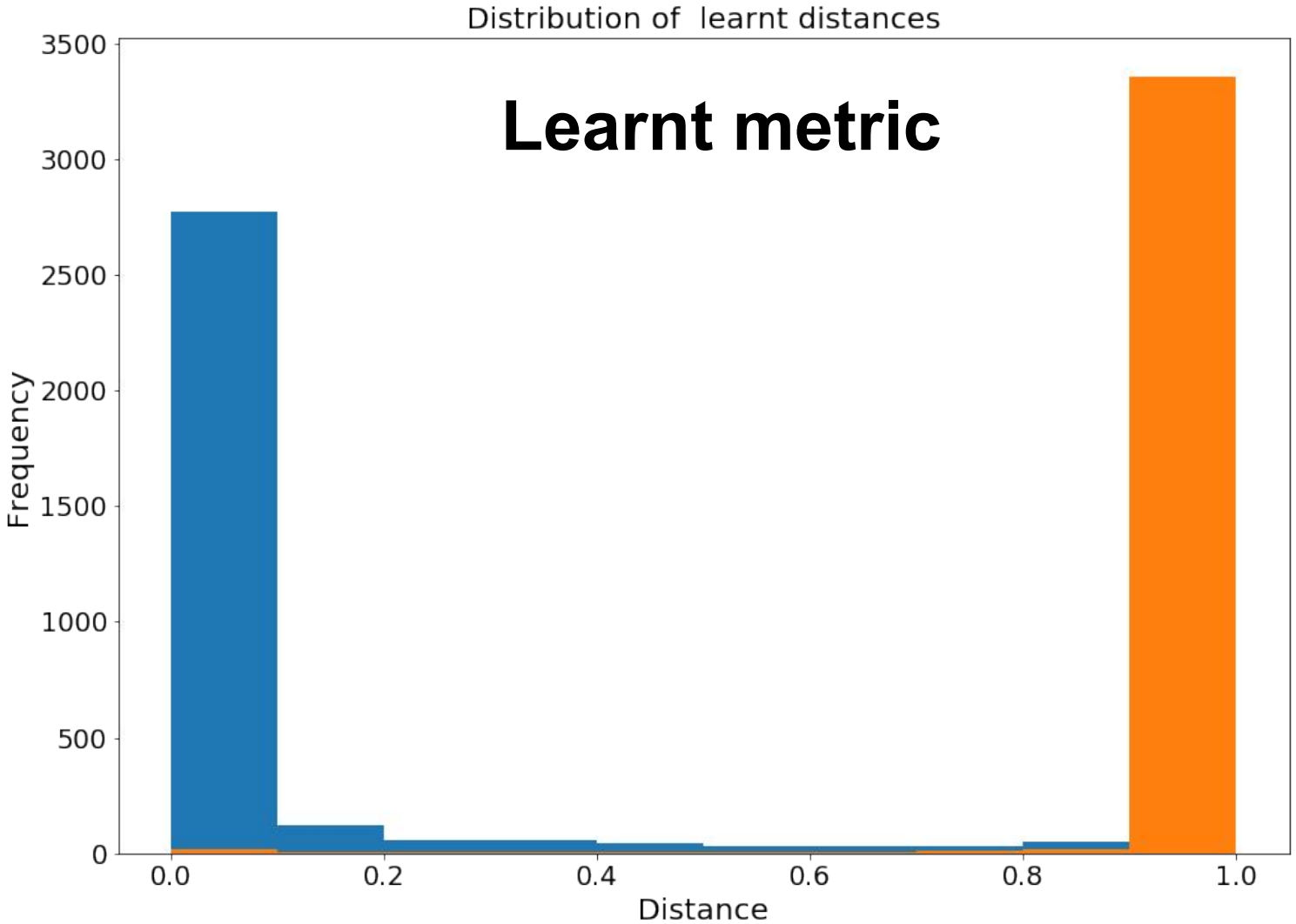
1 different particle

Training

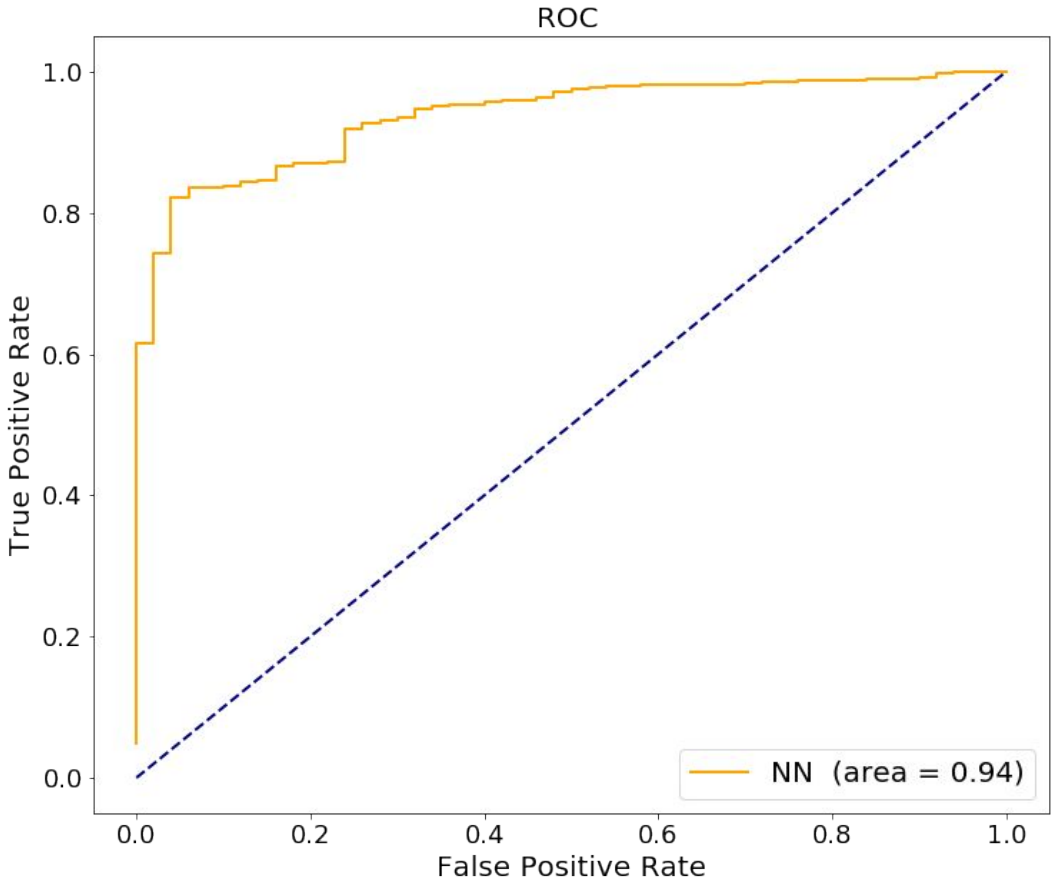




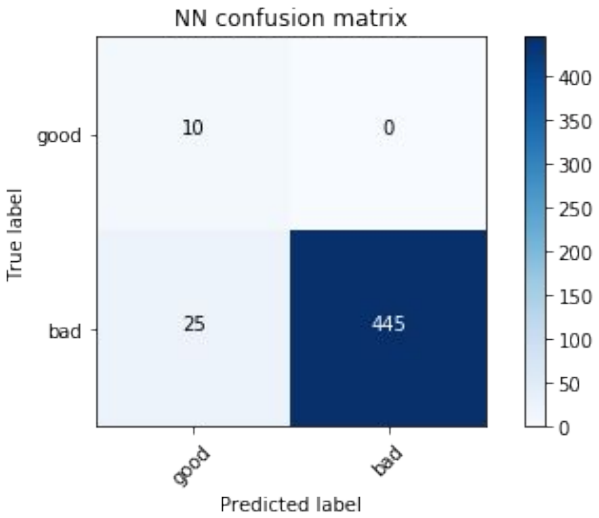
Test (new event)


Distribution of learnt distances
Learnt metric
Frequency
Distance
0
500
1000
1500
2000
2500
3000
3500
0.0
0.2
0.4
0.6
0.8
1.0

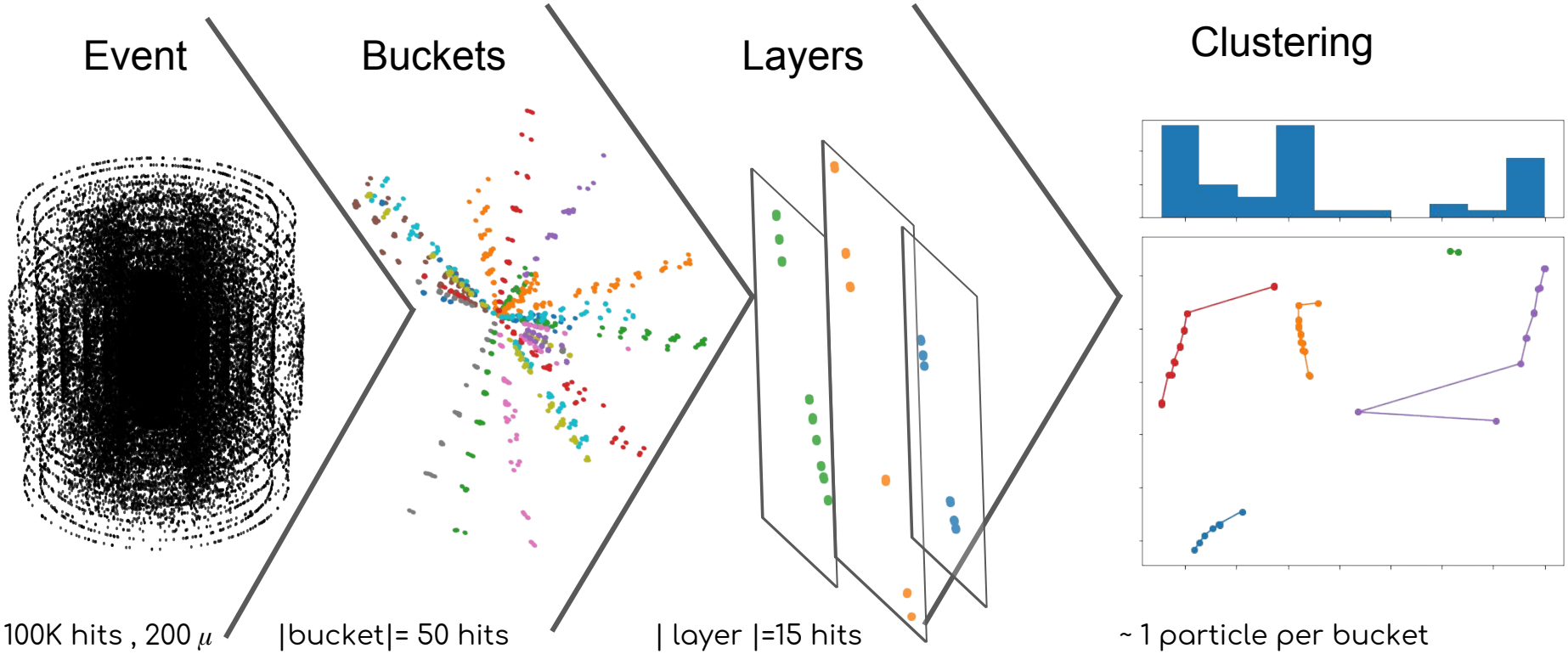
Preliminary results



5 Buckets - N=30 hits
3K segments/predictions



Buckets and data structure



Conclusion and next steps

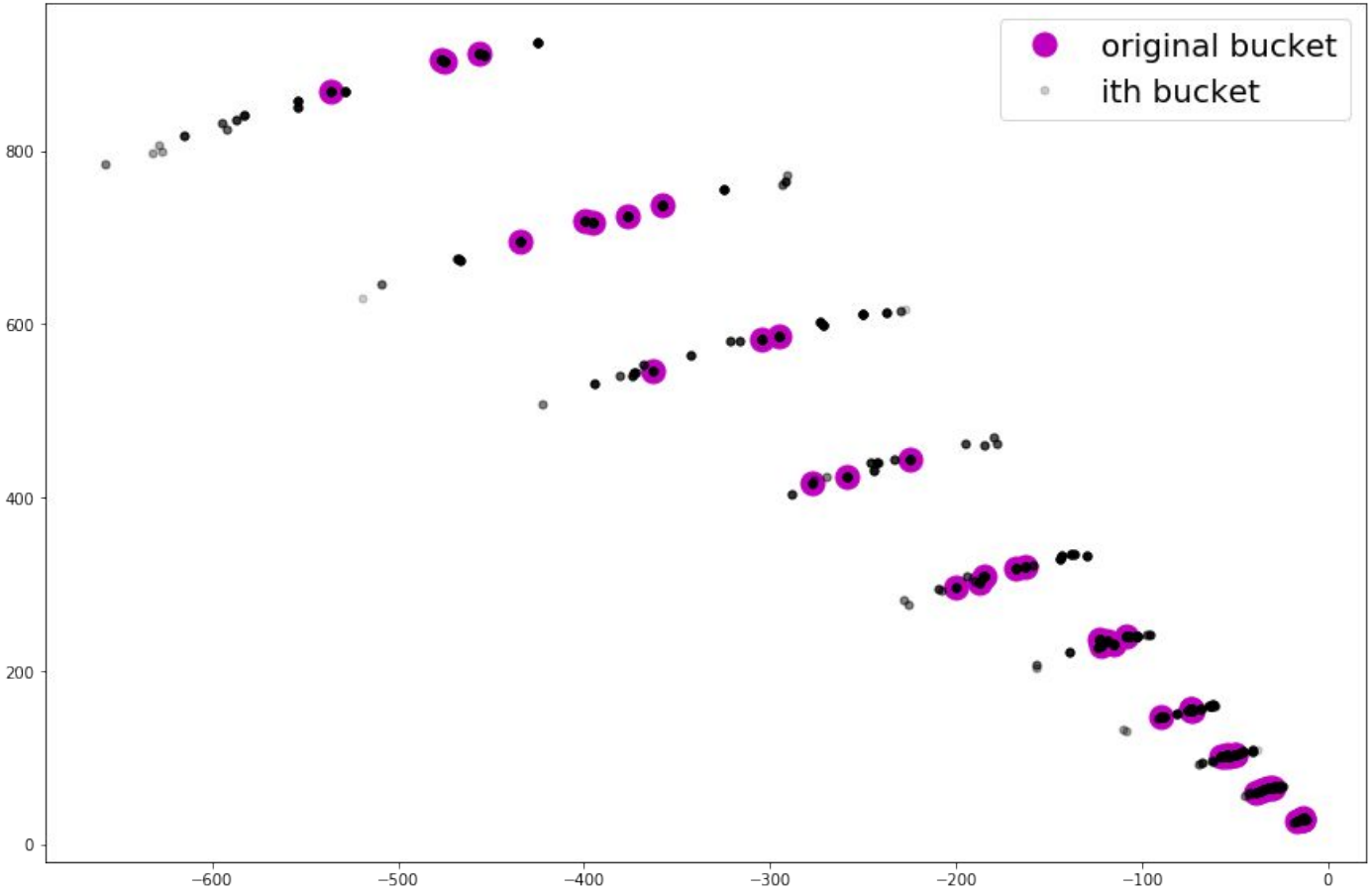
- Faster and efficient data structures for tracking : ~reading tracks
- Open source hashing /ANN libraries :

github.com/nmslib/hnswlib and github.com/spotify/annoy

Next

- Robust against physics parameters change : displaced tracks, Trigger.
- Implementation into Athena

Backup


original bucket
ith bucket
800
600
400
200
0
−600
−500
−400
−300
−200
−100
0