Brandeis UNIVERSITY

Update on Particle ID

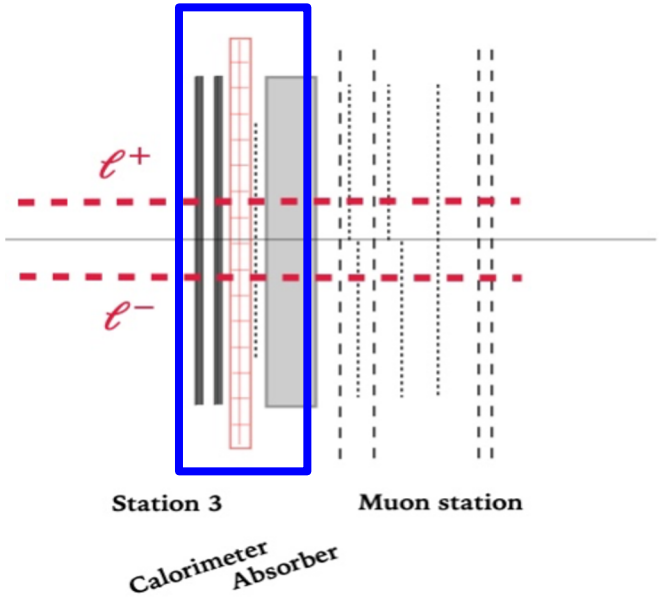
Dowling Wong, A.Apyan, (Brandeis)
W.P.McCormack, P.C.Harris (MIT)

Oct. 2023

INDEX

- Introduction about particle ID
- Current version of ID
- New Clustering algorithm
- Optimized analysis package

- Total # of page: 27
- Estimate time: 20mins

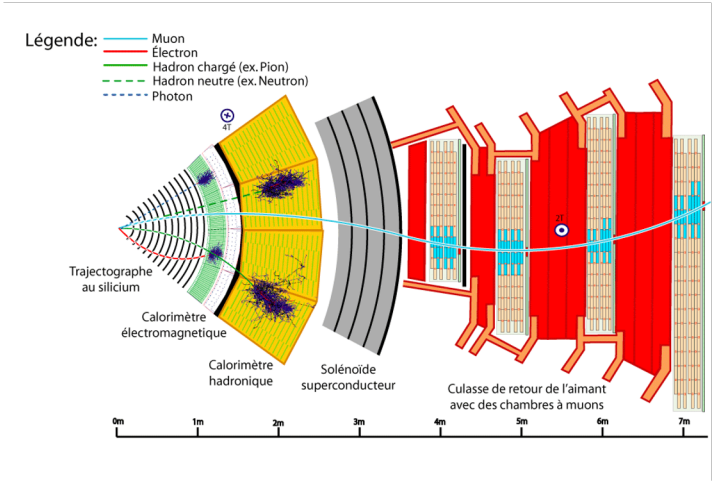


Introduction to ID

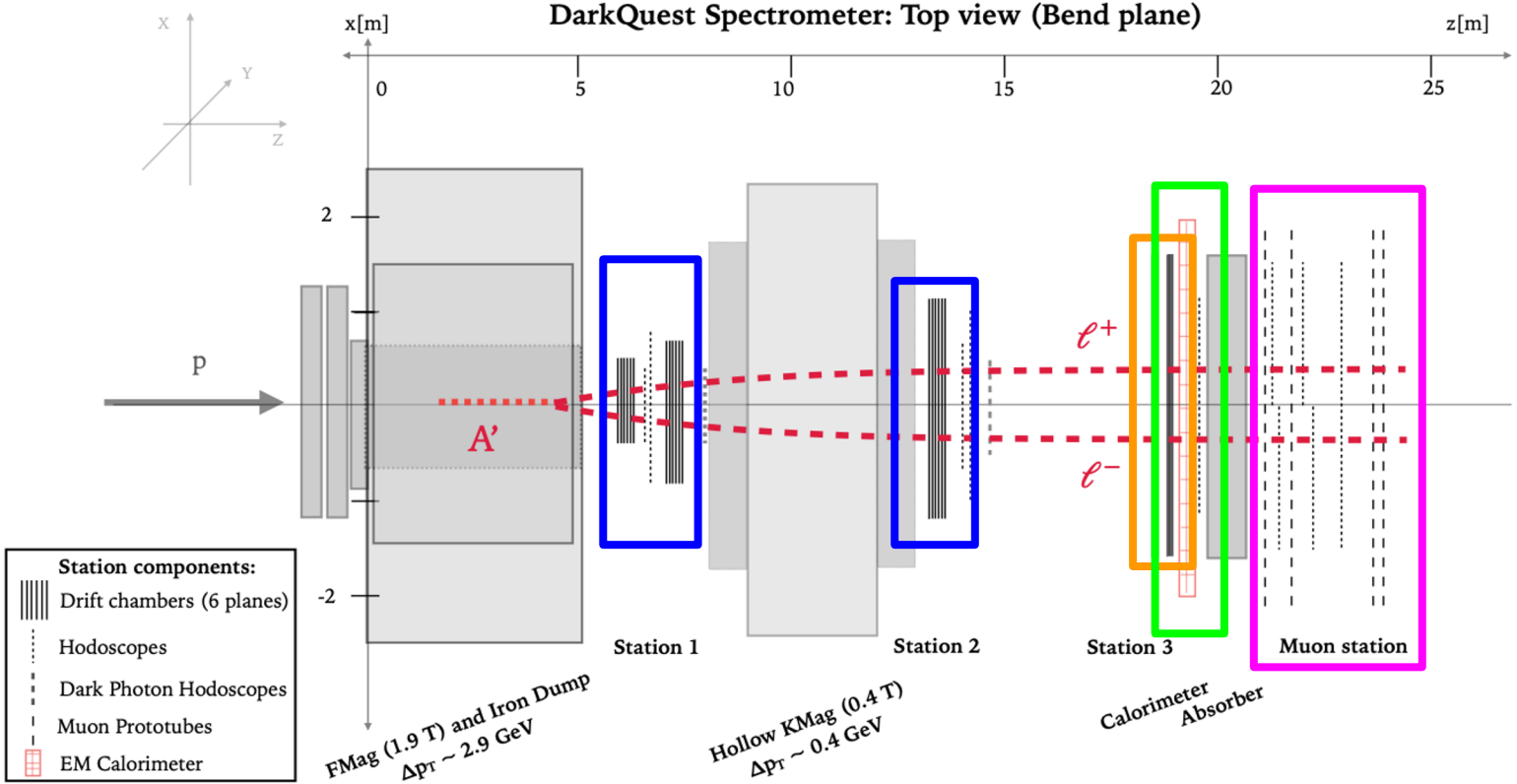
- Short introduction & goal
- Data pattern for different particles
- Historical versions

Particle ID aims to classify different particles in each event, where a particle is reconstructed using multiple detector components. This information helps us to form a complete picture of what happened in events.

The ultimate goal for our work is to build a particle ID like tagging system of particle flows at CMS. That we use tracking chambers, calorimeter, and hodoscope information to reconstruct trajectories and tag particles in each single event.



A considerable part of our work is exploratory, and highly customized for DarkQuest experiment setup.



Full track(st1+st2+st3) [E/p]

st3_stracklet [x, y, z, px, py, pz]

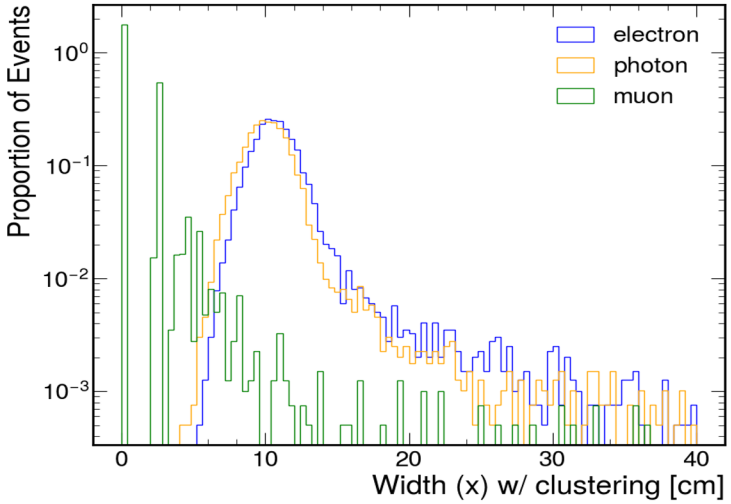
EMCal [widthx, widthy, Wew_x, Wew_y,

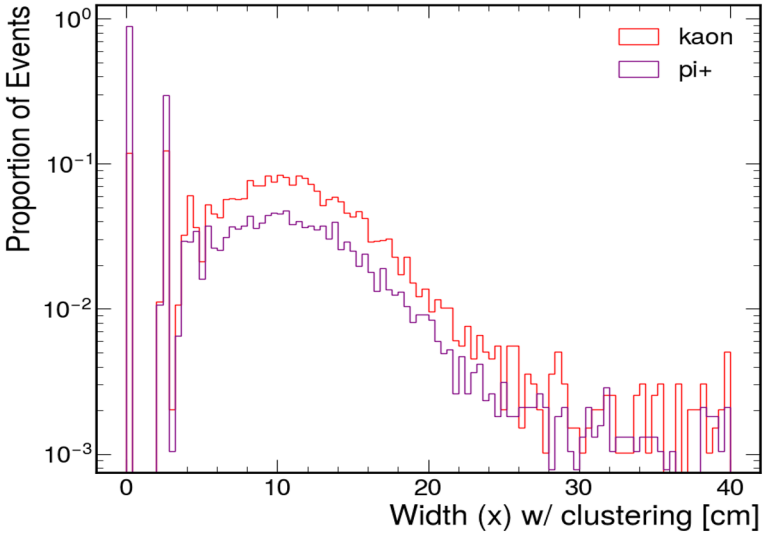
seed_x, seed_y]

Hodoscope [det41_y, det42_y, det43_y,

Quantifying cluster shapes

Full track(st1+st2+st3) , st3_stracklet, Hodoscope. Only has a single value for each particle.

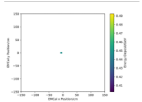
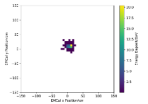
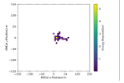
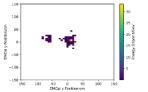
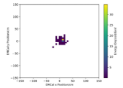
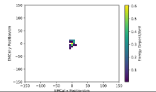




To quantitatively study the shape of cluster on Emcal, we plot the distribution of width(x and y), Energy Weighted width of particles on EMCal. As in DQ

Along with track extrapolation points on EMCal, Energy over momentum, we implemented the first version of ParticleID.

Spreadsheet for particle characteristics

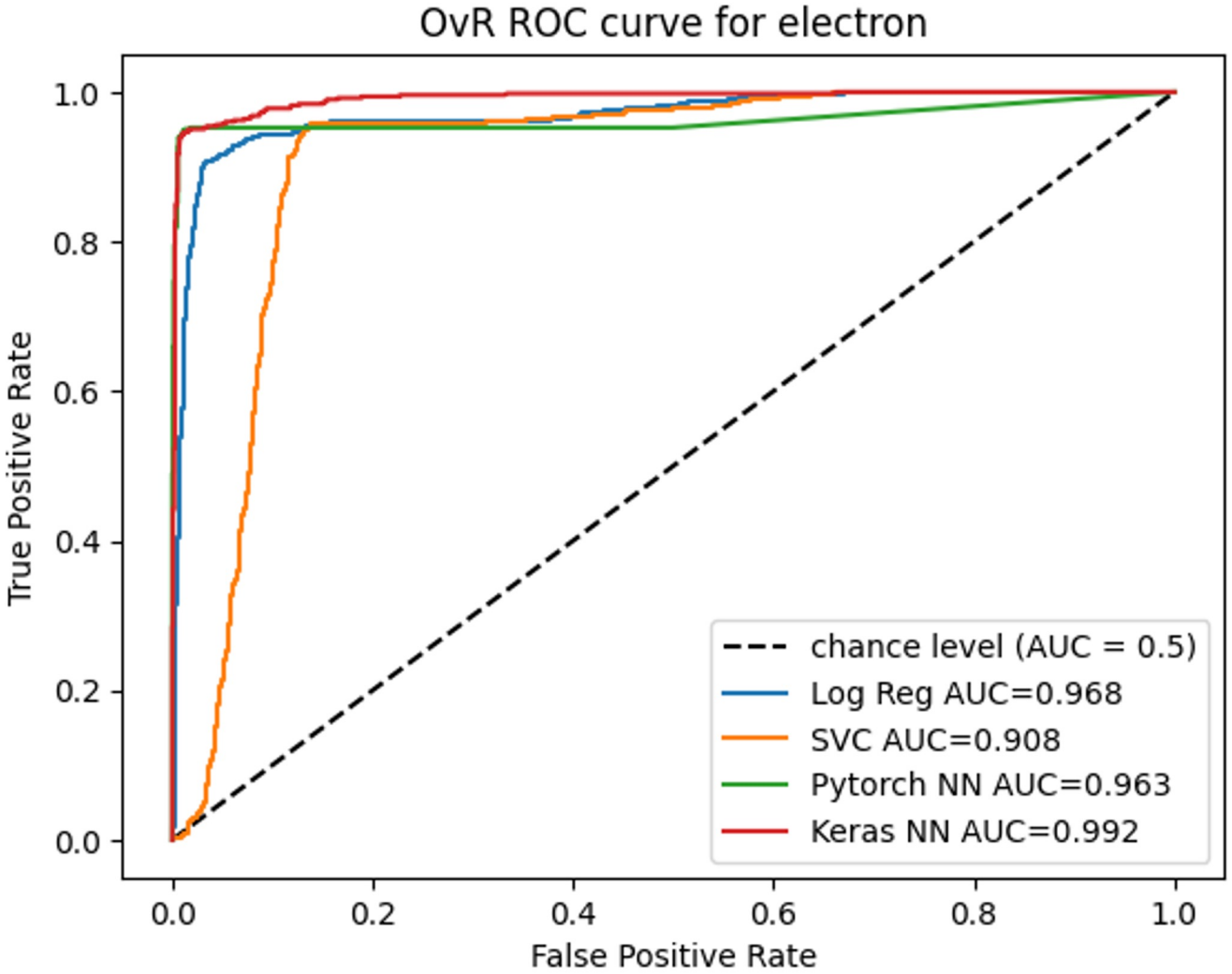
Types	Full track (E/P)	st3_tracklet	EMCal Cluster size	Hodoscope
Muon	Yes, ~0	Yes	Few Points 	Yes, through absorber
Electron/Positron	Yes, ~1	Yes	Large 	No
Pi+/Pi-	Yes, (0,1)	Yes	Middle 	Some
Pi0	No	No	XLarge 	No
Photon	No	No	Large 	No
Klong	No	No	Middle 	No

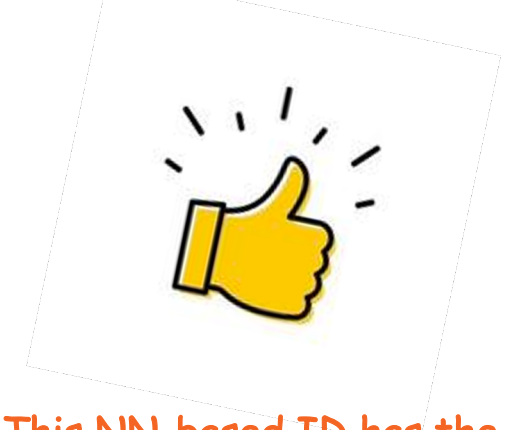
Cut-Based ID for Electron

	Pass Rate
muon	0
electron	59.50%
positron	55.70%
photon	0.40%
pi+	3.40%
pi-	3.80%
klong	0
pi0	0.20%

Inputs:
[widx, widy, Wew_x, Wew_y, Dist_x, Dist_y, E/P]

Well, can we do a DNN based particle ID to improve the efficiency?



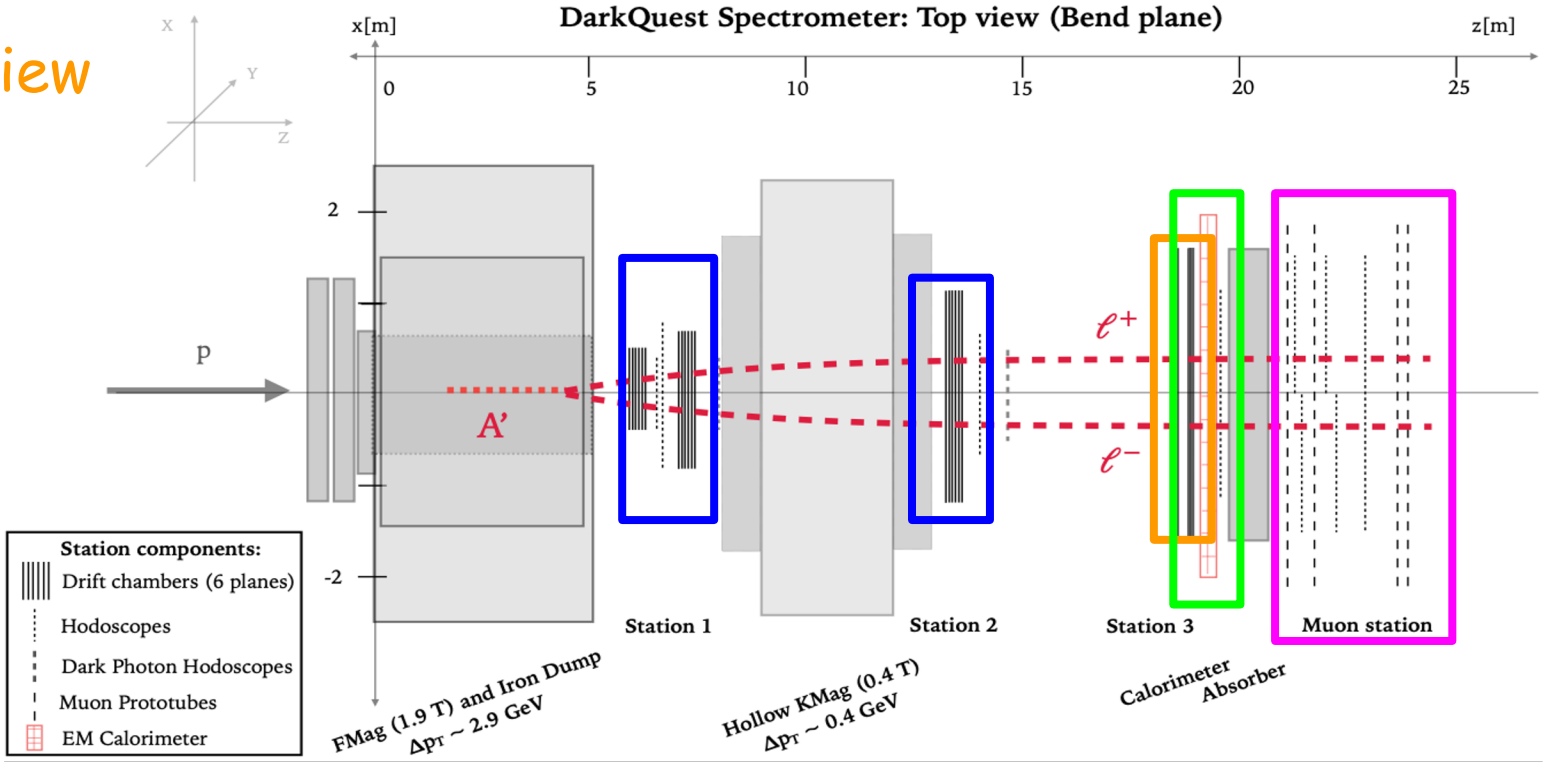


This NN-based ID has the same input as cut-based one, looks good?

The analysis is only on leading cluster of each single particle guns evts.

But we are looking for dark photon into dimuon and dielectron, right?

Review



Full track(st1+st2+st3) and electron

St3_stracklet extrapolate trajectory

EMCal determine how to reconstruct

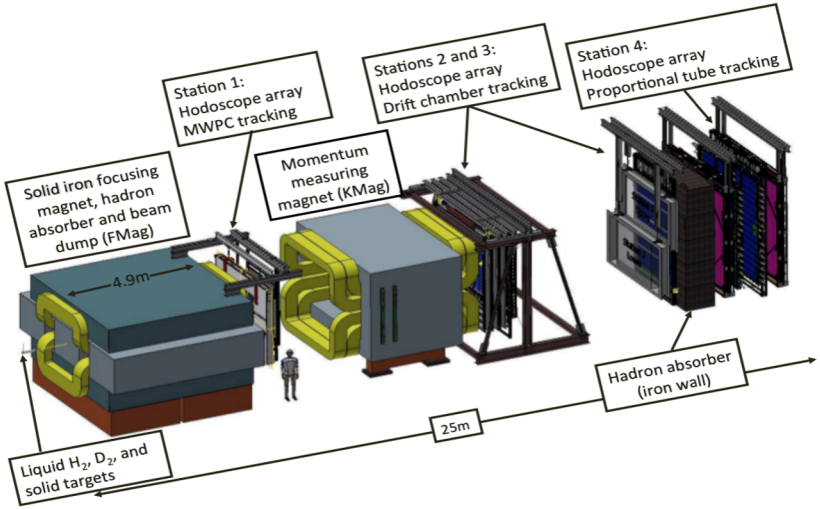
[E/p], effectively distinguish between muon

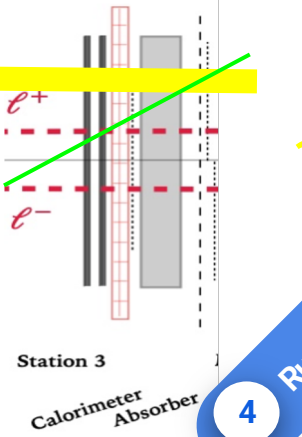
Position and direction to

*Currently use # of seeds to

Reconstruction Logic

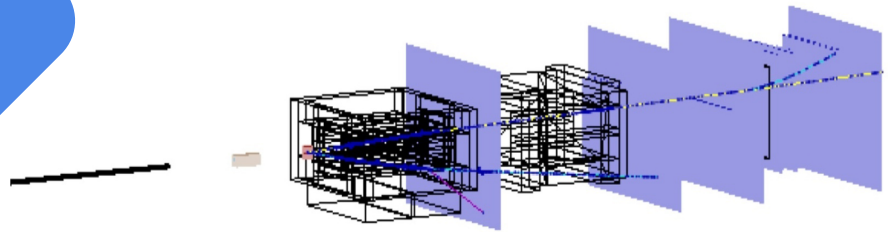
1. Track finding & EMCal clustering
2. Clusters match track & hodoscope hits
3. *Post ana for "missing muon"
4. Run tagger on "particles"

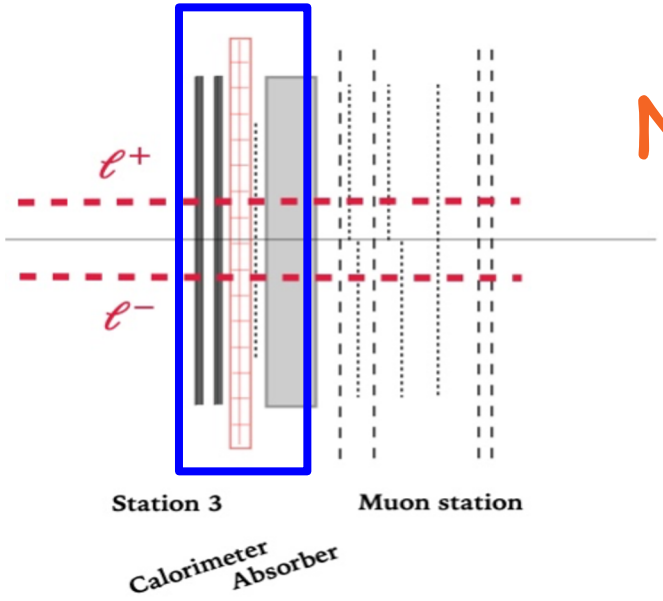




μ+-

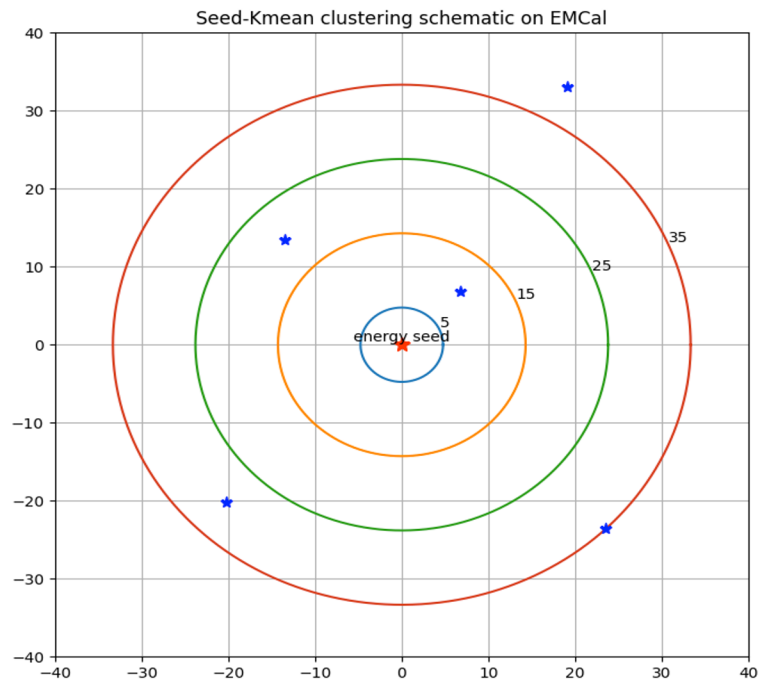
γ





New Clustering & Efficiency

- A′ -> mm
- A′ -> ee
- Single particle guns
- Diphoton samples

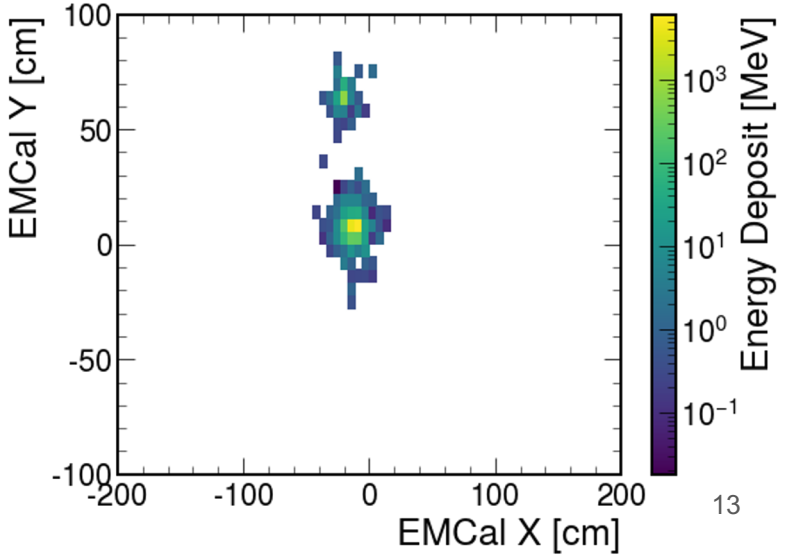


Step1. Set a energy threshold, emin. Collect all hits with eng above this level.

[x_i, y_i, eng_i]

Step2. Loop through the hits,within distance threshold, no hits have higher energy.

*currently set emin=0.2GeV, radius=50cm. Low energy threshold make sure hard to miss a particle.



Step3. Run Kmean once, find if some cluster has been missed(mainly for muon). *exceed tolerance.& (post-ana)

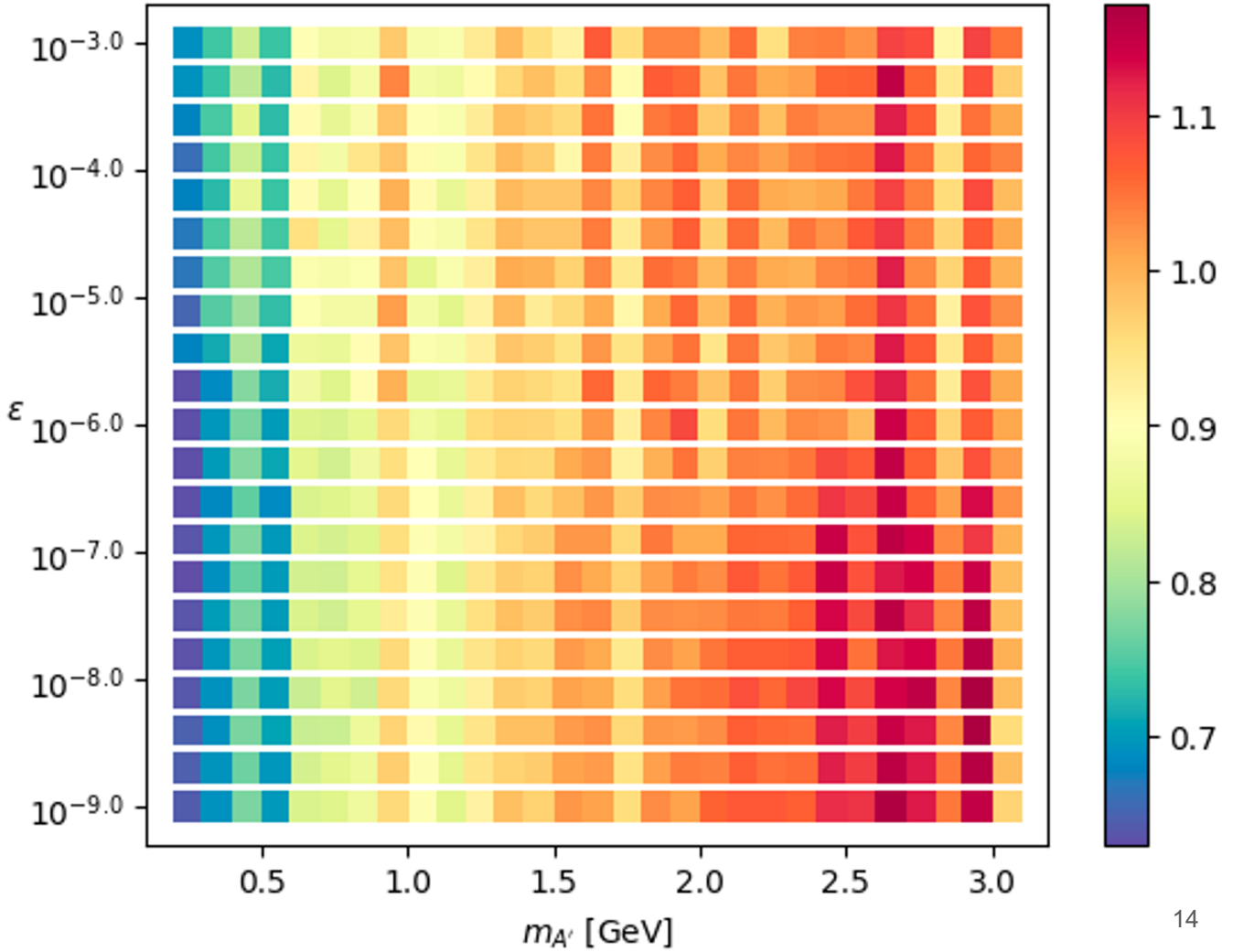
Step4. Once get seeds all set, tagging each hits to clusters.

A'->mm

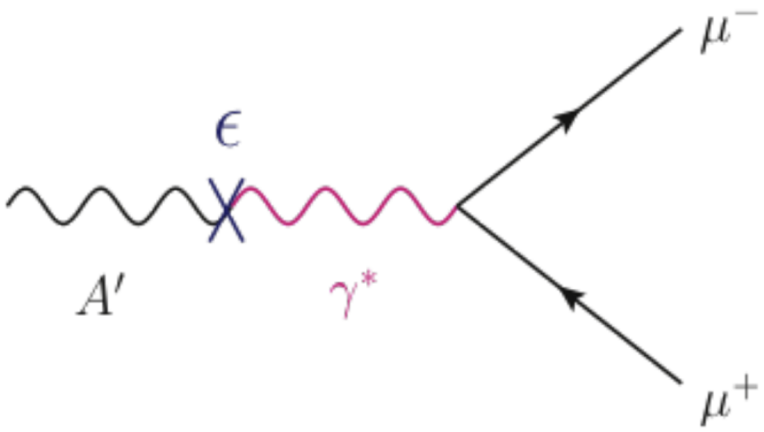
Seed tagging efficiency

All 609 possible mass and coupling combination.

eff_st=(Σ # seeds /Σ # trkls)



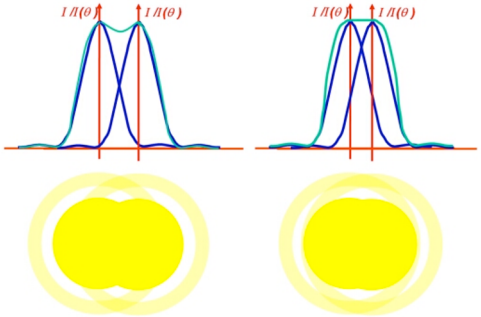
What happened to low mass A'?



*From Ming

EMCal cell

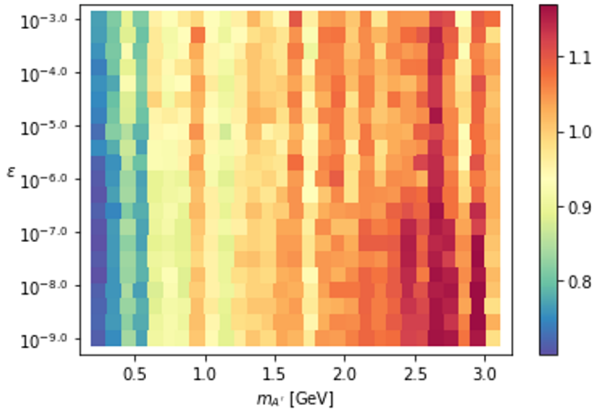
Like Angular resolution



*Diphoton has the same problem

Verification of Guess

- Even lower the energy threshold and radius to 0.1, 10. Still doesn't give visible difference.
- Conclusion after manually checking dark blue samples: there are higher density of events has only one hit on EMCAL

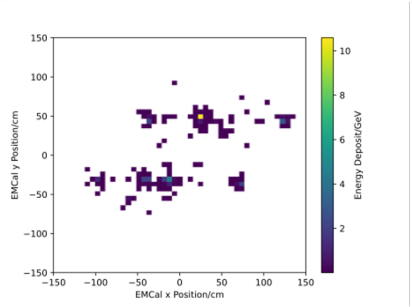


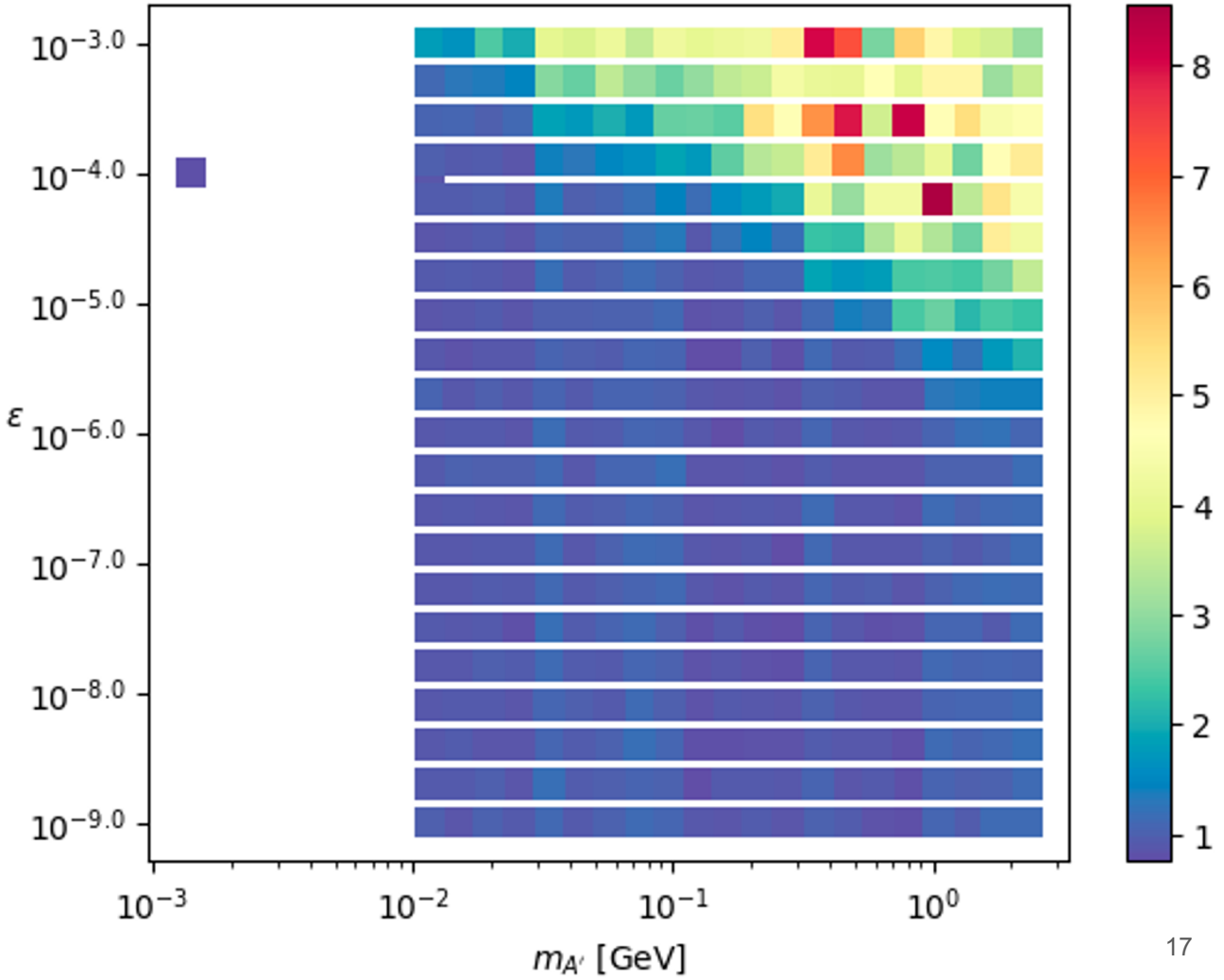
A'->ee

Seed tagging efficiency

All 443 possible mass and coupling combination.

Dark red

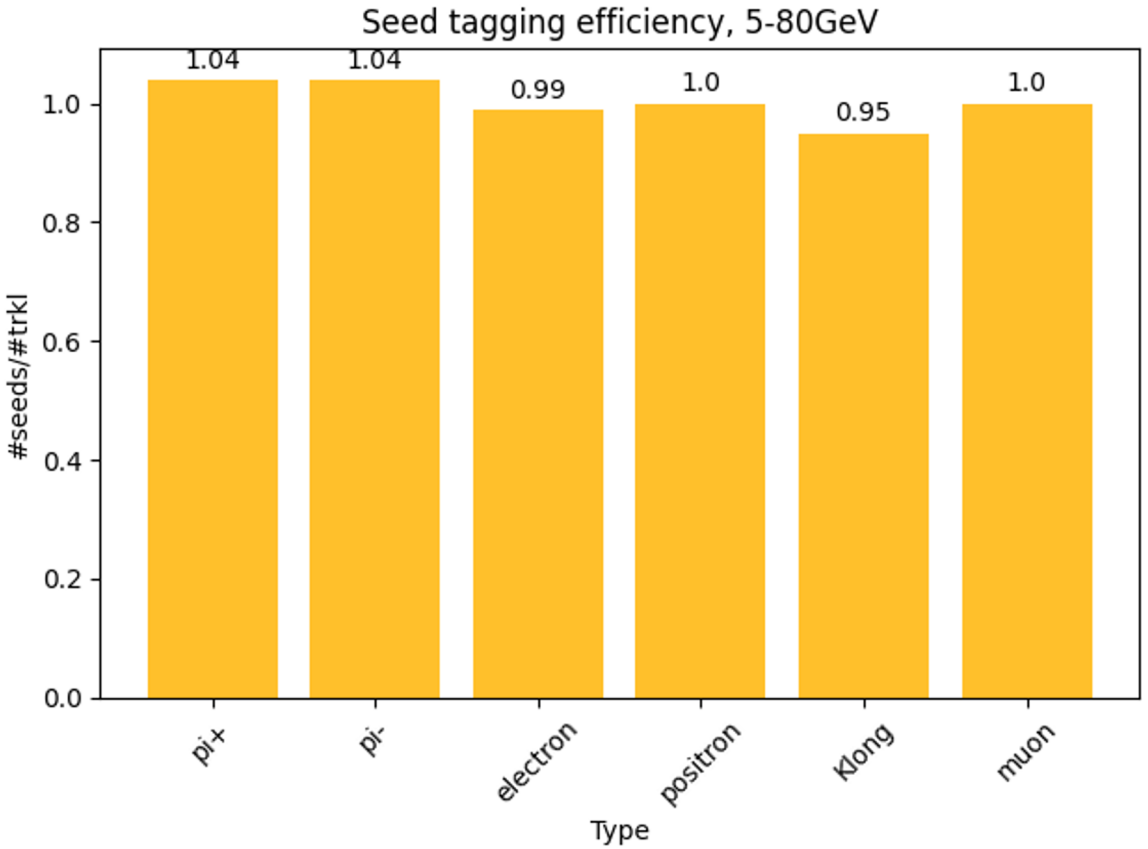




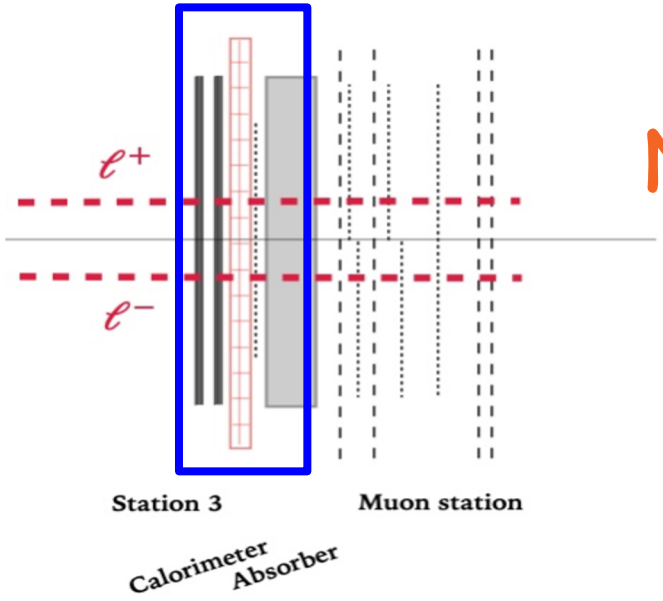
Seed tagging efficiency of single particle gun

1=100%

eff_st=(Σ # seeds /Σ # trkls)



*For Klong, ratio is #evts w/ seed / evts w/ valid EMCal record

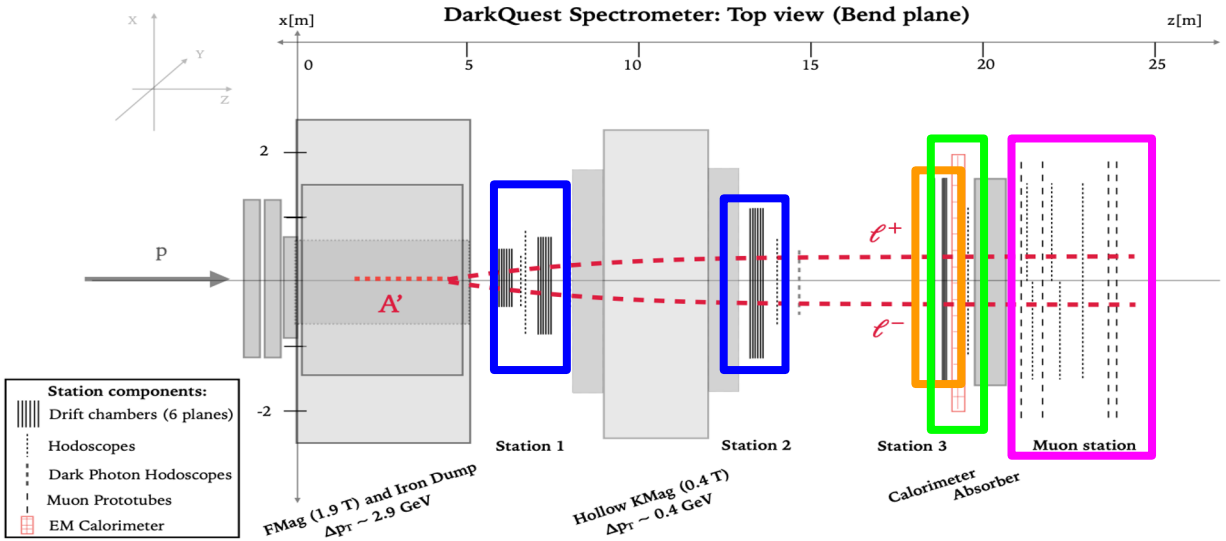


New Particle ID

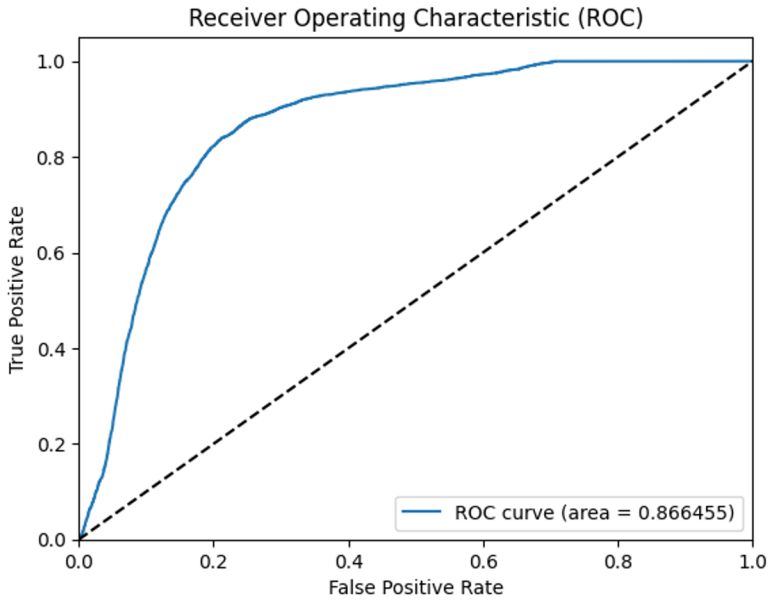
- 2 versions w/ and w/o track
- For both muon and electron
- Tagging efficiency vs Pz

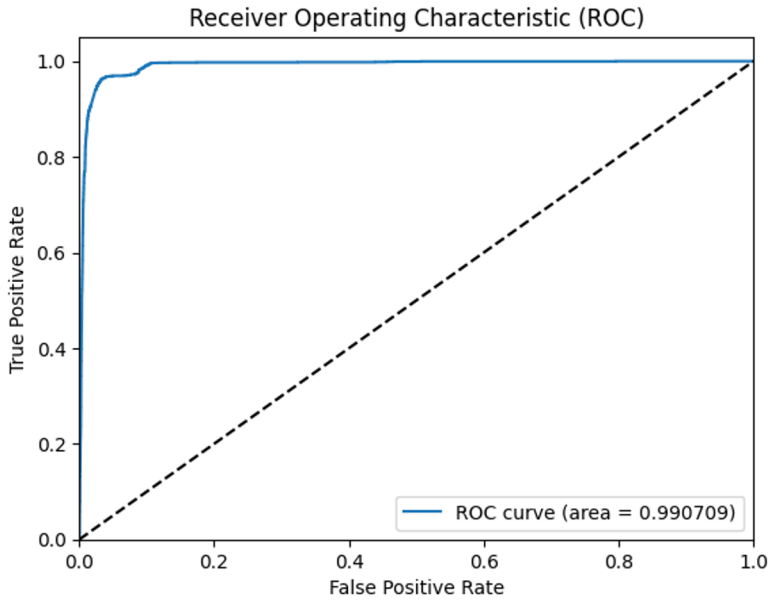
Data preparation

- With the reconstruction logic mentioned above, we can connect hits on each station to a consistent particle trajectory.
- With information array for each particle as NN input. Save into CSV [wid_x, wid_y, wew_x, wew_y, seed_x, seed_y, trkl_x, trkl_y, trkl_z, trkl_px, trkl_py, trkl_pz, E/p, h4_41, h4_42, h4_43, h4_44, h4_45, h4_46]
- Preparing with my package dwong, mention later.



Result: ROC curve for ID w/o full track(E/p)



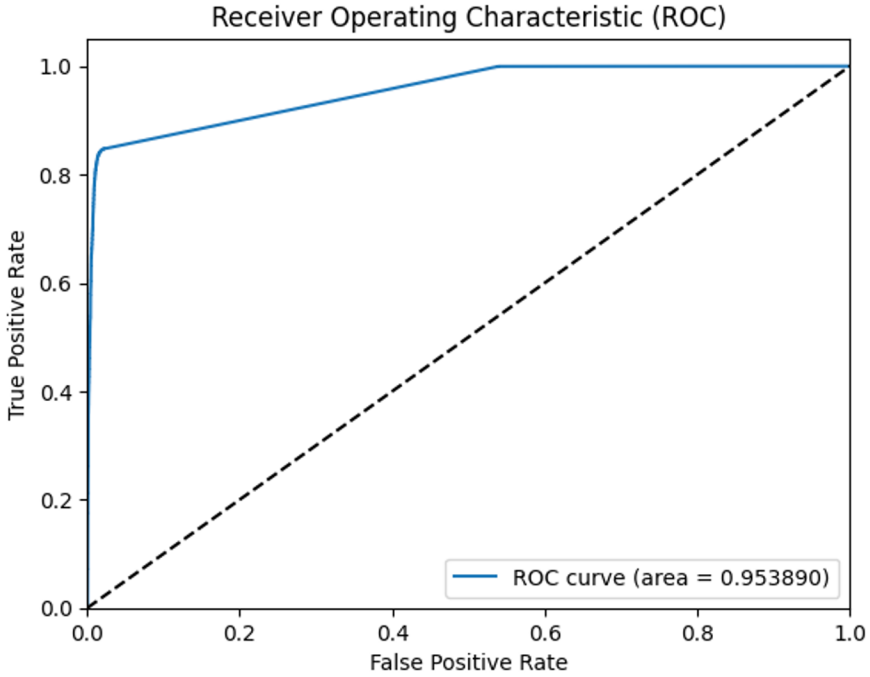


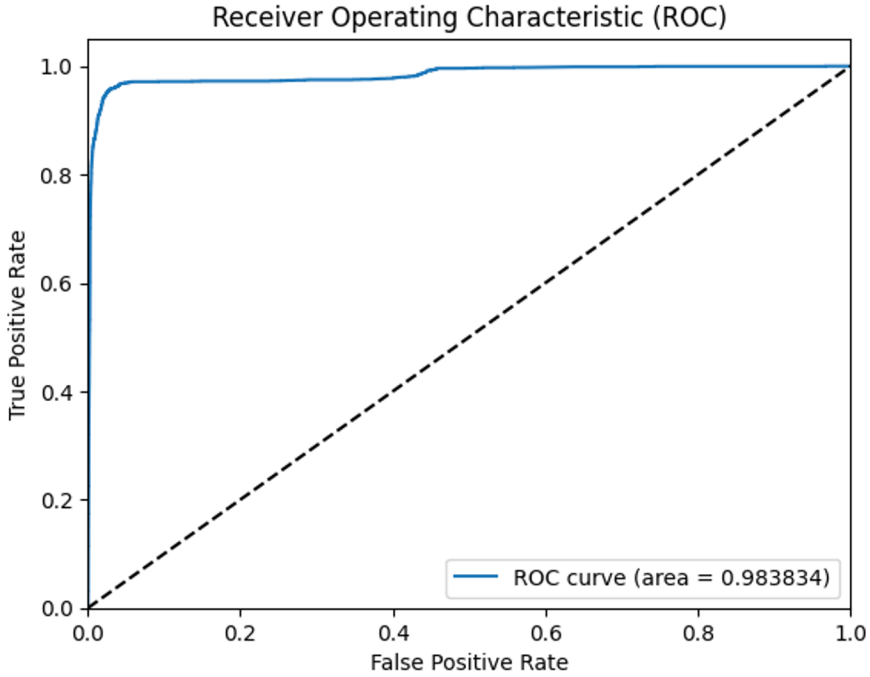
Electron ID

Muon ID

Input: [wid_x, wid_y, wew_x, wew_y, seed_x, seed_y, trkl_x, trkl_y, trkl_z, trkl_px, trkl_py, trkl_pz, h4_41, h4_42, h4_43, h4_44, h4_45, h4_46]

Result: ROC curve for ID w/ full track(E/p)



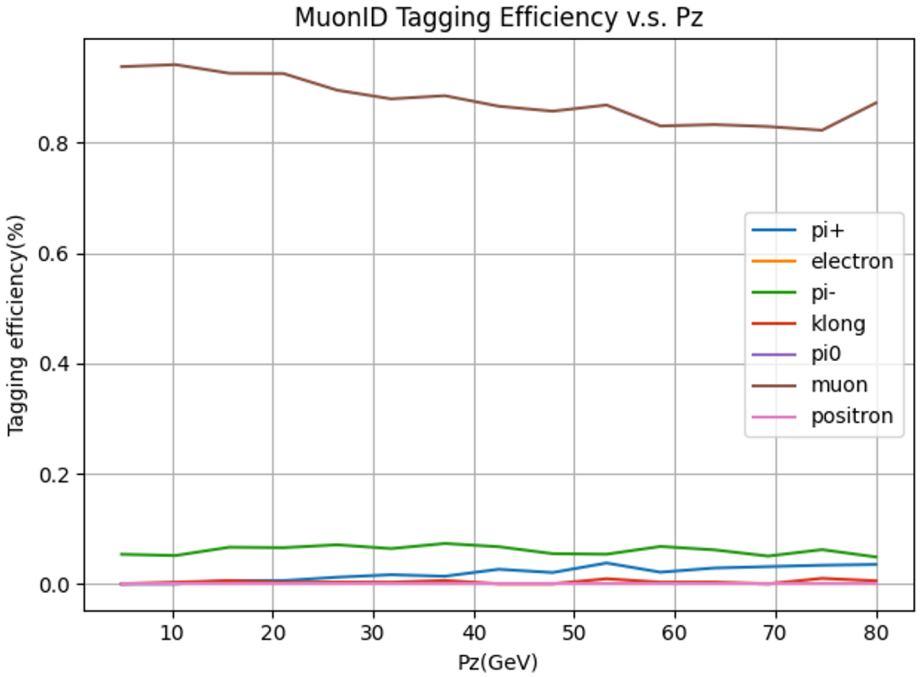


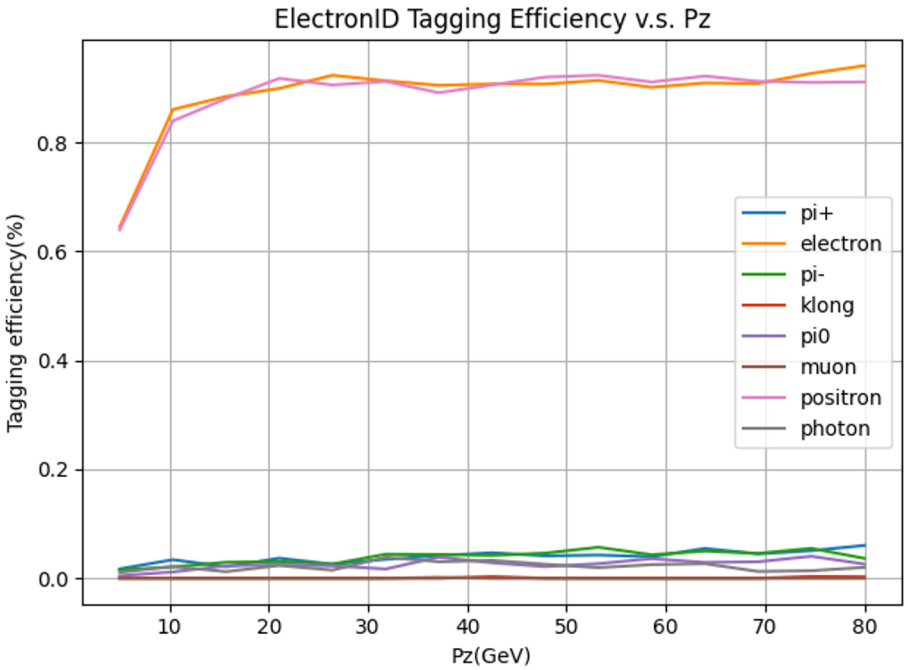
Electron ID

Muon ID

Input: [wid_x, wid_y, wew_x, wew_y, seed_x, seed_y, trkl_x, trkl_y, trkl_z, trkl_px, trkl_py, trkl_pz, E/p, h4_41, h4_42, h4_43, h4_44, h4_45, h4_46]

Result: Tagging efficiency vs Pz





Why range 5-80GeV?

DarkQuest: A dark sector upgrade to SpinQuest at the 120 GeV Fermilab Main Injector

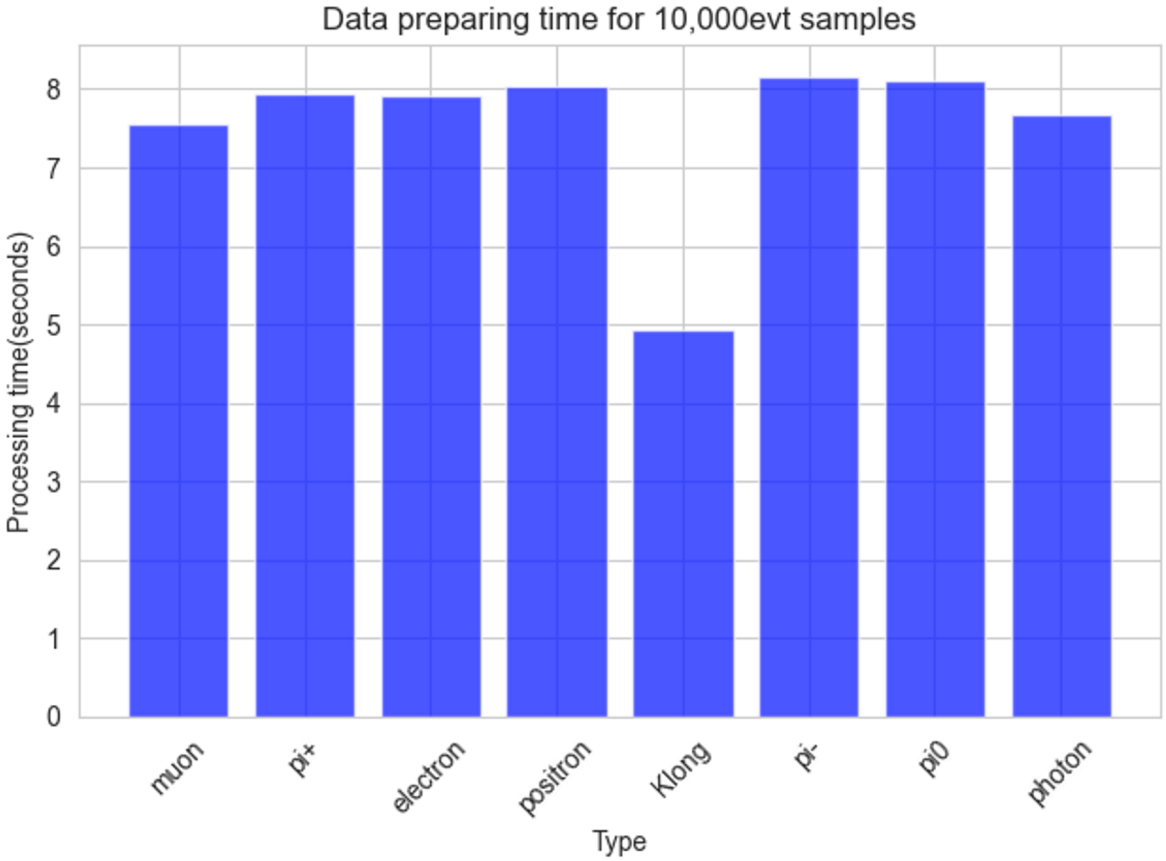
```python
def getData(fname="", procName="Events"):
    file = uproot.open(fname)
    dq_dict = file[procName].arrays(library="np")
    dq_events = {
        "Hits":{
            "detID": dq_dict["hit_detID"],
            "edep": dq_dict["hit_edep"],
            "elmID": dq_dict["hit_elmID"],
            "hit_pos": dq_dict["hit_pos"]
        },
        "track":{
            "x": dq_dict["track_x_CAL"],
            "y": dq_dict["track_y_CAL"],
            "ID": dq_dict["eventID"],
            "pz": dq_dict["track_pz_st1"]
        },
        "st23": {
            "ntrack23": dq_dict["n_st23tracklets"],
            "px":   dq_dict["st23tracklet_px_st3"],
            "py":   dq_dict["st23tracklet_py_st3"],
            "pz":   dq_dict["st23tracklet_pz_st3"],
            "x":   dq_dict["st23tracklet_x_st3"],
            "y":   dq_dict["st23tracklet_y_st3"],
            "z":   dq_dict["st23tracklet_z_st3"],
            "Cal_x": dq_dict["st23tracklet_x_CAL"],
            "Cal_y": dq_dict["st23tracklet_y_CAL"]
        }
    }

    return dq_events
```

Package dwong!!!

- DQ analysis package dwong released on pip. People could play around.
- Include analysis functions, NN-ID model, training frame, csv saving tools, samples and EMCal plot tools.
- Optimized with numpy and cython.

It's all open source!



Repo for analysis code:

Repo for dwong package:

Pypi page of dwong

Search projects

dowling7

dwong 0.1

```
pip install dwong==0.1
```

Latest version

Released: less than 20 seconds ago

Dowling's integrated data analysis for DQ experiemnt setup. Include data analysis functions, csv saving tools, Particle ID model, DNN model frame for ID training and EMCal ploting tools

Manage project

Navigation

- Project description
- Release history
- Download files

Project links

- Homepage

Statistics

GitHub statistics:

- Stars: 0
- Forks: 0

Project description

dwong, a package for DarkQuest.

dwong is a comprehensive Python package, created by student Dowling Wong, tailored for data analysis and neural network-based particle identification in the DarkQuest experiment. The aim of this project is to streamline DarkQuest's data analysis process by providing exemplary data-processing functions.

The package mainly contains four modules: dwong, dplot, dcsv and dkeras.

- dwong
  - emcal_bytuple
  - multi_clusters
  - h4_bytuple
  - prepare_data_bytuple
- dplot
  - emcal_evt(x, y, eng)
  - emcal_pdf(ntuple_name, filename, path)
- dkeras

Thank You!
Questions?