ML4NP: Meeting 14.07.2020

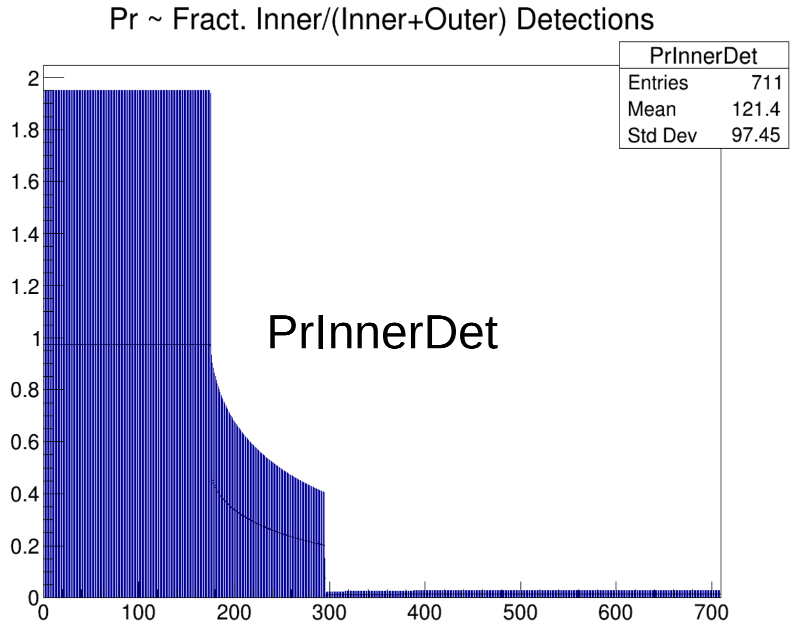
Summary

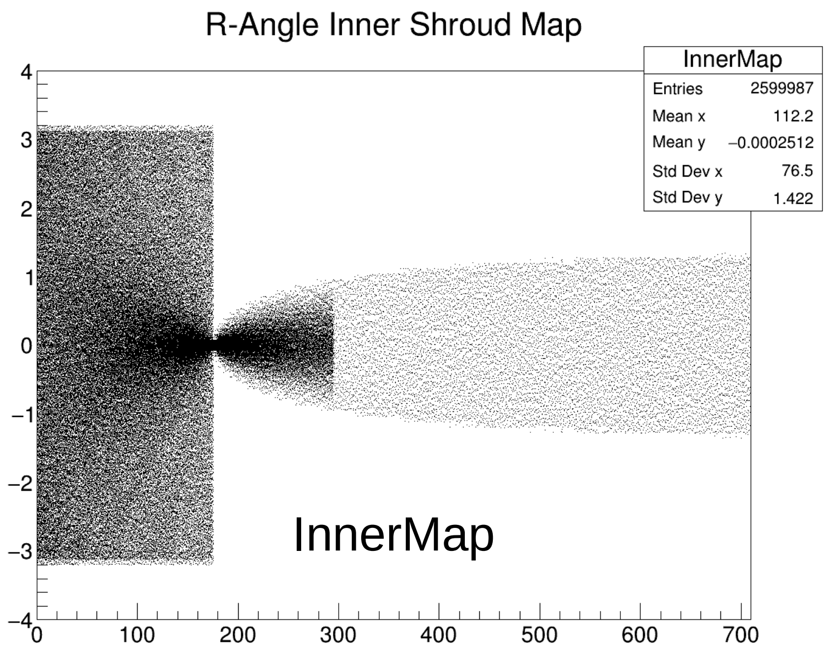
- Spatial Map:
- ML Part I: Rejection of single Ar39 decay
- ML Part II: Classification Muons vs Ar39 Pileups

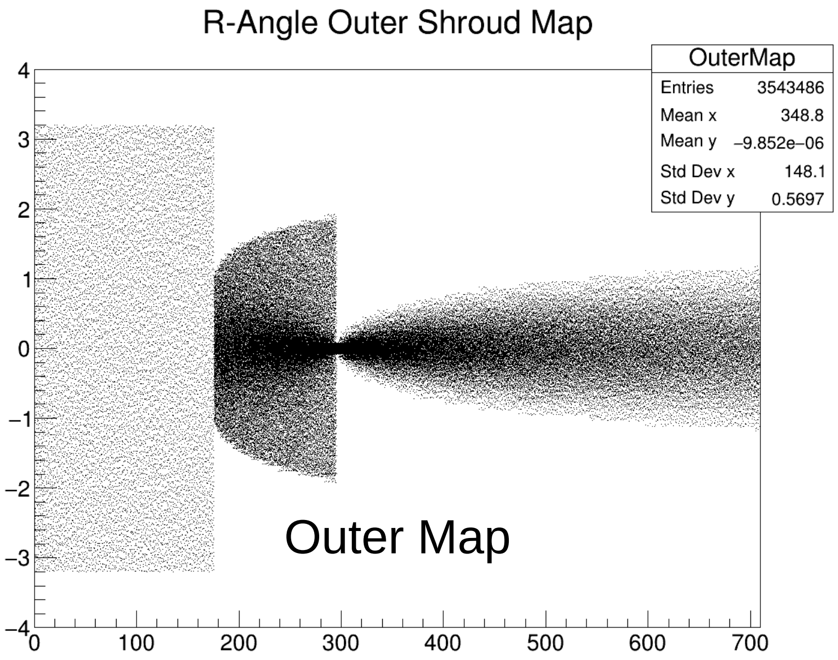
Spatial Map (last week)

- Production of 3 maps:

1) PrInnerDet: given *NPE*, it computes *NPE_in* that hit the InnerShroud and *NPE_out* that hit the OuterShroud
2) Spread NPE_in in InnerShroud wt InnerMap
3) Spread NPE_out in OuterShroud wt OuterMap

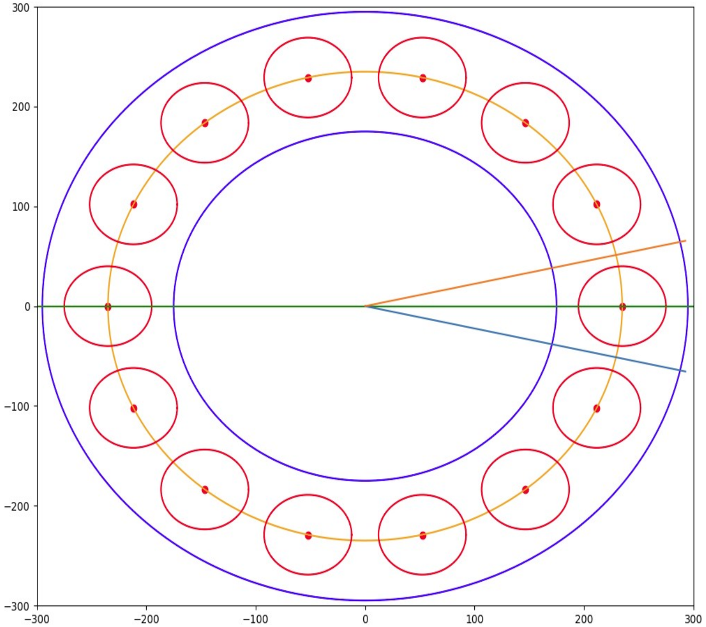


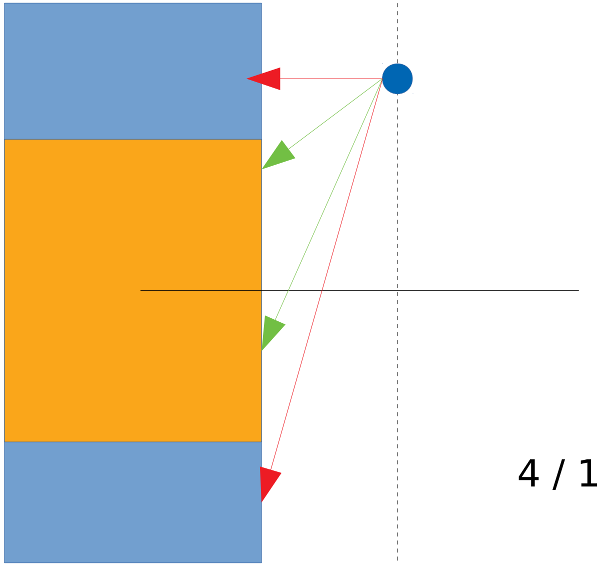




Spatial Map (updates)

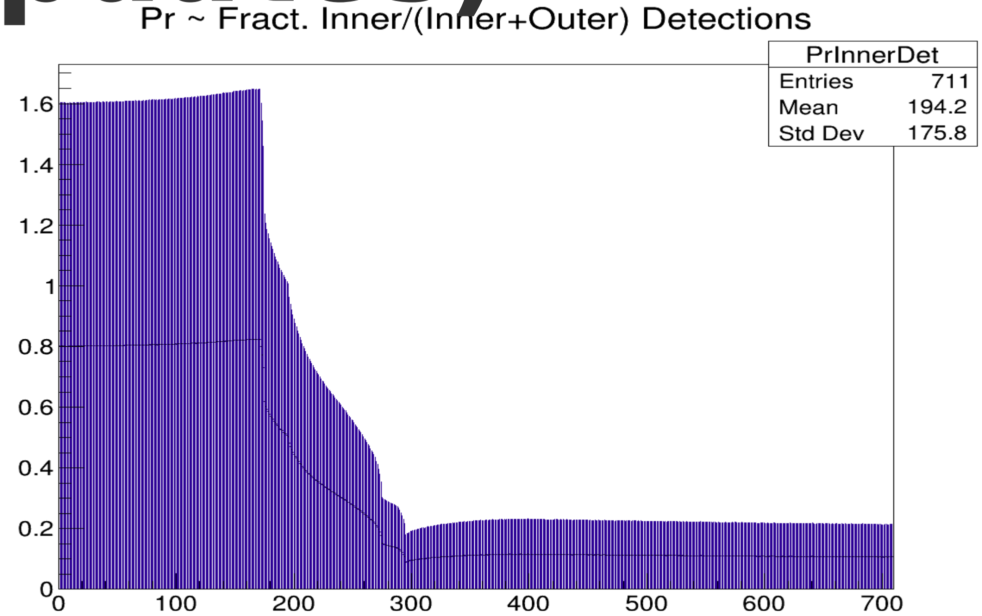
- Inclusion of Ge crystals
  (14 strings in geometry = 14 circles in 2D)
- We always sampled in point along a line, extend with sampling in a range +-0.22 rad.
- The chance to hit Ge crystals depends on:
  - Starting z
  - OP emission angle
- By sampling z, OP phi and OP theta, we compute where the OP trajectory cross the Ge surface (cylinder 23.5+40mm).
  - If the crossing point has z in +-445mm, enable Ge (*green arrows*).
  - Otherwise, disable Ge (*red arrows*).
- The OP trajectories ends when they go outside the roi. We cut a 2D trajectory based on where the 3D trajectory cross +-845mm.

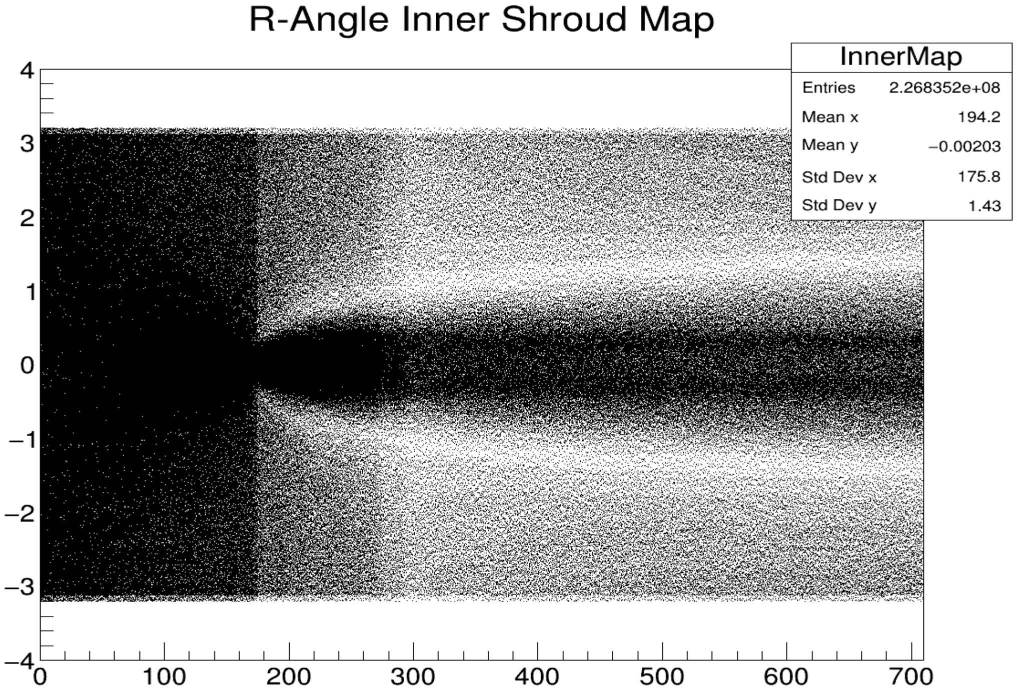


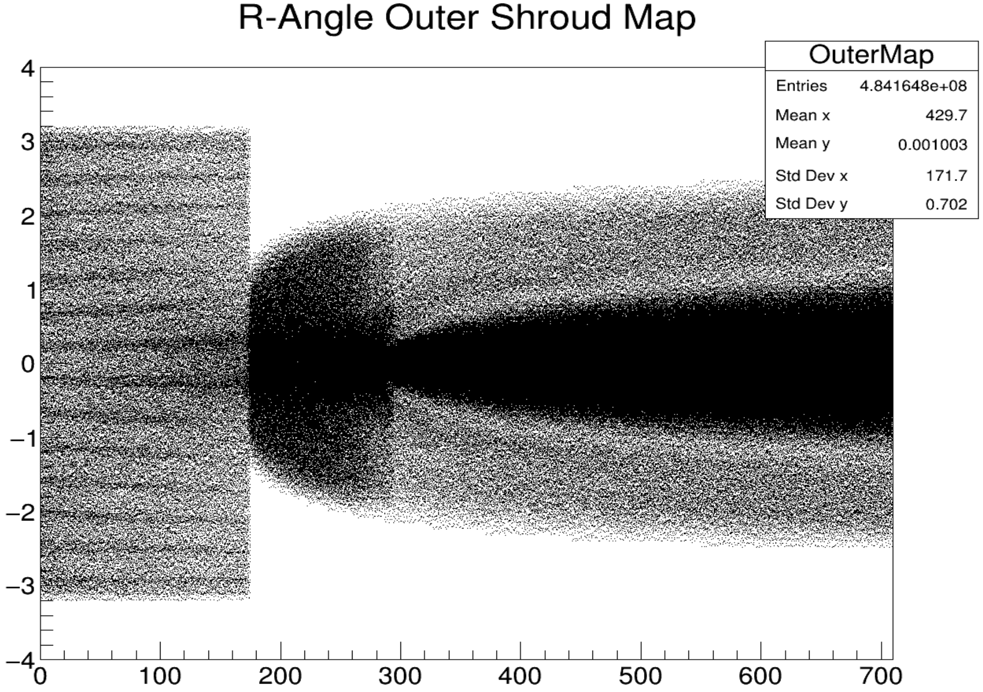


Spatial Map (updates)

- The new maps include:
  - sampling of shifting angle and z,
  - sampling of OP orientations,
  - enabling/disabling Ge crystals,
  - early cutting of OP trajectories.







ML Part 1: Rejection of Ar39

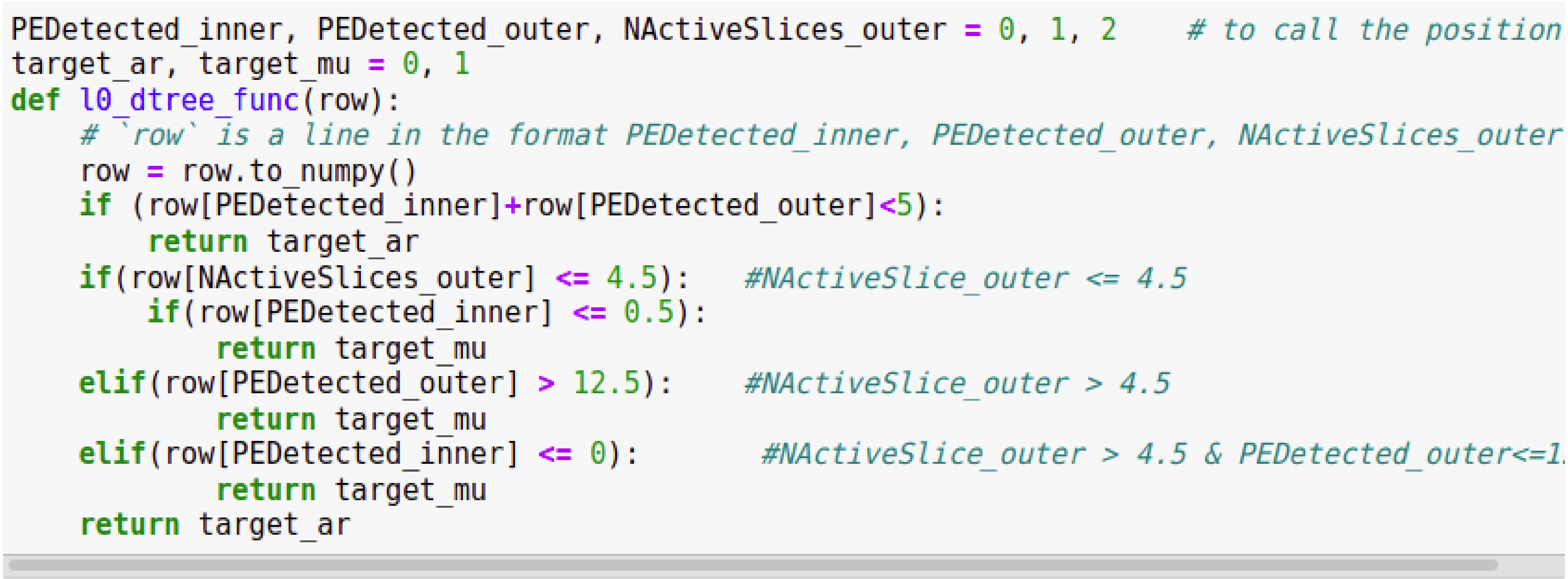
- **Context:**
  - Very low-energy events: detections up to 60 PE
  - Level 0 Tagging: specialized in 1 Ar39 to suppress its high-activity
- **Requirements**:
  1) Reject single Ar39 decays as much as possible (high TNR),
  2) Save very low-energy muons (PE<=60), if possible.

ML Part 1: Rejection of Ar39

- **Available Data**: 5 834 261 Ar39, 26 881 Mu (*old data + new 1M muons by CJ*)
- **Preliminary cut at 5 PE** to discard most of the single Ar39 decays (*avg 3 PE*):
  - 5 110 Muons, 1 474 688 Ar39
- **Training Data** (*balanced*): 4 500 Muons, 4672 Ar39s (undersampling)
- **Test Data** (*unbalanced*): 610 Muons, 1M Ar39s

- Trained model:
  - Evaluation on Test Data:
    - Accuracy: 0.880, Purity: 0.004, Efficiency: 0.852, F1: 0.009
  - Conf. Matrix:
    TN: 879742, FP: 120258,
    FN: 90, TP: 520
  - **TPR: 85.25%, FPR: 12.03%** (*without considering preliminary cut*)

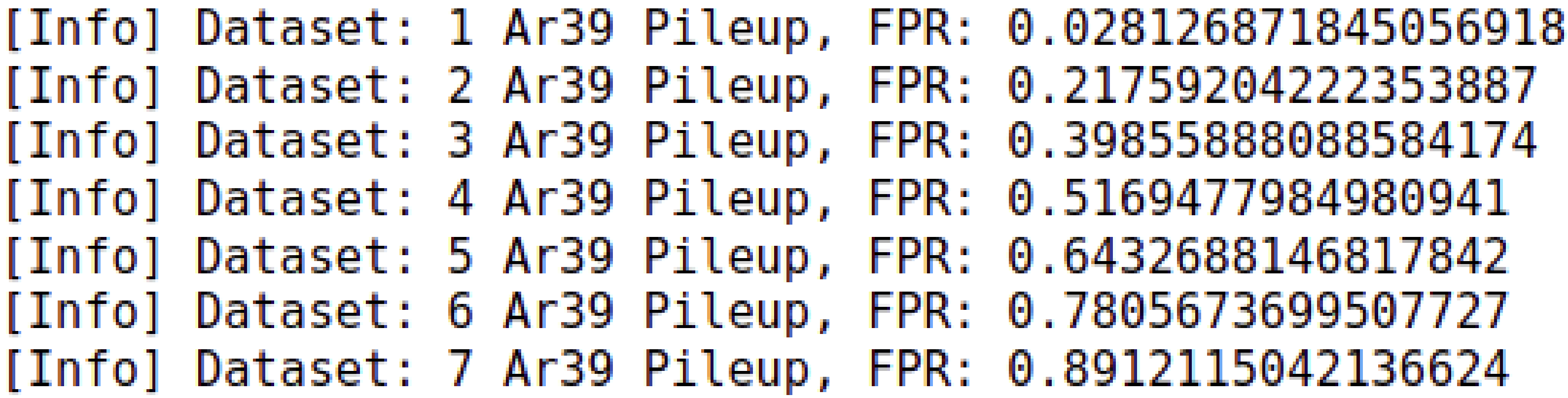
From model to code...



```python
PEDetected_inner, PEDetected_outer, NActiveSlices_outer = 0, 1, 2    # to call the position
target_ar, target_mu = 0, 1
def l0_dtree_func(row):
    # `row` is a line in the format PEDetected_inner, PEDetected_outer, NActiveSlices_outer
    row = row.to_numpy()
    if (row[PEDetected_inner]+row[PEDetected_outer]<5):
        return target_ar
    if(row[NActiveSlices_outer] <= 4.5):   #NActiveSlice_outer <= 4.5
        if(row[PEDetected_inner] <= 0.5):
            return target_mu
    elif(row[PEDetected_outer] > 12.5):     #NActiveSlice_outer > 4.5
            return target_mu
    elif(row[PEDetected_inner] <= 0):       #NActiveSlice_outer > 4.5 & PEDetected_outer<=1
            return target_mu
    return target_ar
```

ML Part 1: Rejection of Ar39

- Passing the various Ar39 classes through this selection strategy.



```
[Info] Dataset: 1 Ar39 Pileup, FPR: 0.028126871845056918
[Info] Dataset: 2 Ar39 Pileup, FPR: 0.21759204222353887
[Info] Dataset: 3 Ar39 Pileup, FPR: 0.39855888088584174
[Info] Dataset: 4 Ar39 Pileup, FPR: 0.5169477984980941
[Info] Dataset: 5 Ar39 Pileup, FPR: 0.6432688146817842
[Info] Dataset: 6 Ar39 Pileup, FPR: 0.7805673699507727
[Info] Dataset: 7 Ar39 Pileup, FPR: 0.8912115042136624
```

ML Part 2: Mu vs Ar39 Pileups

- **Context:**
  - Low-energy events: detections up to 115 PE
  - Level 1 Tagging: wide classification 1, 2, …, 7 Ar39 decays vs Muons
- **Requirements**:
  - Prompt tagging of muons, suppression of Ar39 (single or pileup)
  - Good metrics: Efficiency/Purity

- First attempt:
  - **CNN model**: slower inference but no cost for feature creation (*at least on my pc, e.g. 701000 instances in 16 seconds*)
  - The previous Dtree has faster inference (*e.g. 901235 instances in 8 seconds*) but requires feature creation