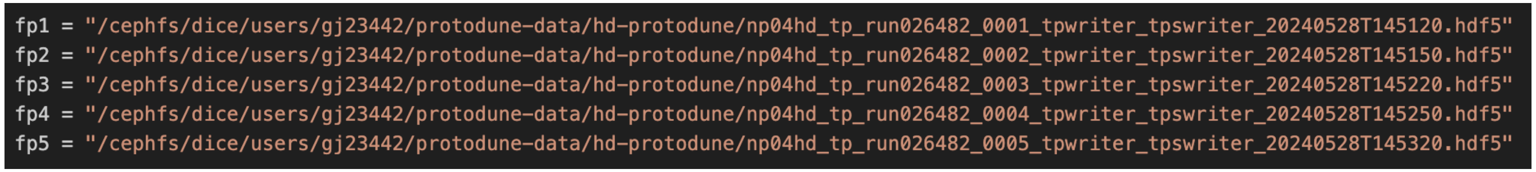
WEEKLY ANALYSIS UPDATE

14 Jan 2025

Samikshya Kar

Updates for the week

1. Five TP stream files for APA 2 channels for run 026482:

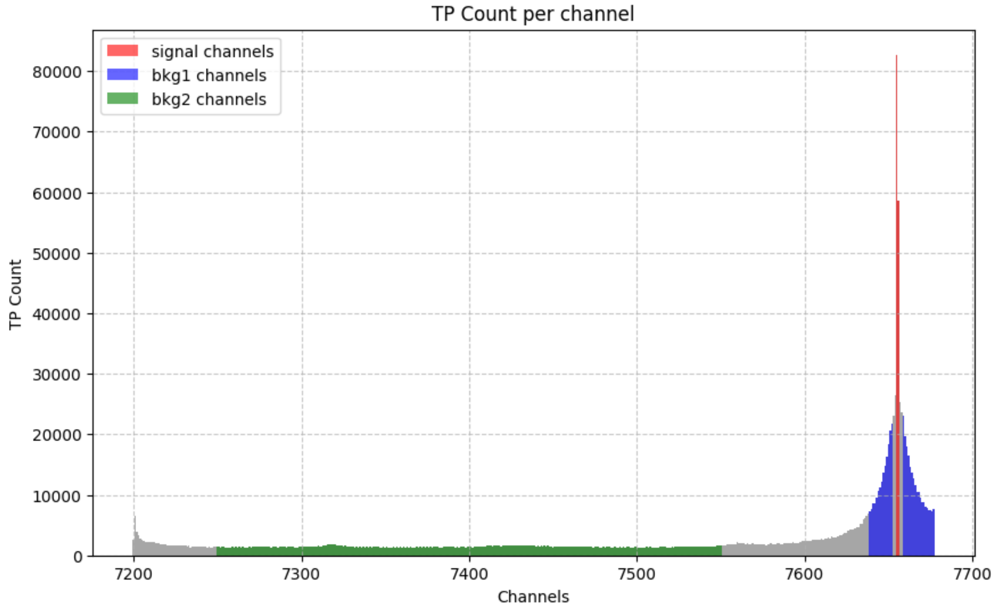


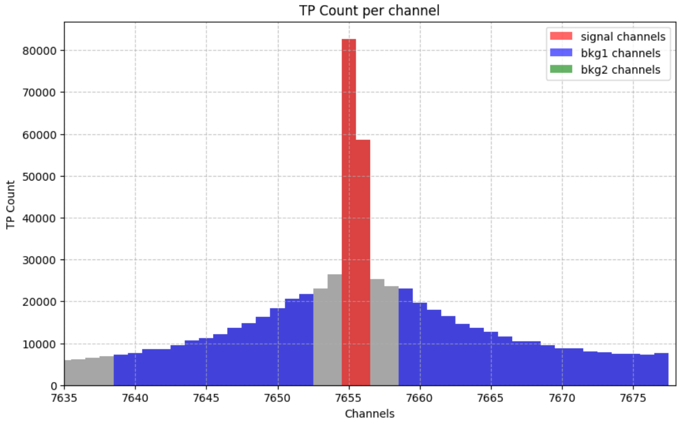
```
fp1 = "/cephfs/dice/users/gj23442/protodune-data/hd-protodune/np04hd_tp_run026482_0001_tpwriter_tpswriter_20240528T145120.hdf5"
fp2 = "/cephfs/dice/users/gj23442/protodune-data/hd-protodune/np04hd_tp_run026482_0002_tpwriter_tpswriter_20240528T145150.hdf5"
fp3 = "/cephfs/dice/users/gj23442/protodune-data/hd-protodune/np04hd_tp_run026482_0003_tpwriter_tpswriter_20240528T145220.hdf5"
fp4 = "/cephfs/dice/users/gj23442/protodune-data/hd-protodune/np04hd_tp_run026482_0004_tpwriter_tpswriter_20240528T145250.hdf5"
fp5 = "/cephfs/dice/users/gj23442/protodune-data/hd-protodune/np04hd_tp_run026482_0005_tpwriter_tpswriter_20240528T145320.hdf5"
```

2. Statistics from TPstream files are significantly high (After cosmic muon removal using DBSCAN Clustering, TP number: 1413261)
3. Visualisation of TP count per channel to identify Bi-207 activity
4. Identification of signal and background region from the above plot
5. Plotting histograms of different TP properties for signal and background region
6. Estimation of Background in the signal region by normalisation using TP number

TP Count per Channel

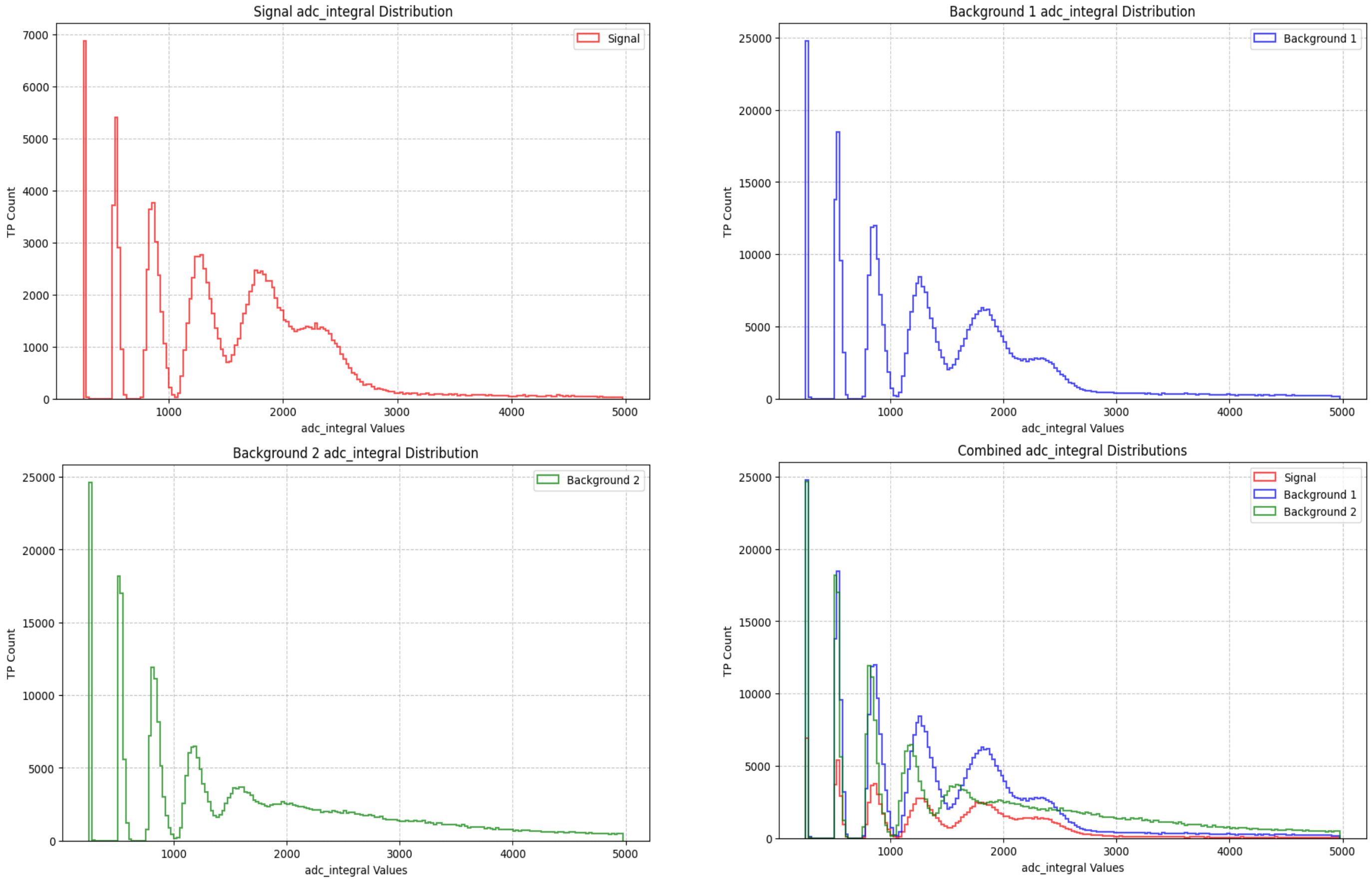
- Datafile: Five tpstream.hdf5 datafiles from run 26482
- Data read using *justintime* package only from APA 2 collection channels
- Collection TPs that are recorded before being passed to Trigger
- Selection:
  - Signal selection: TP count > 30000
  - Background selection:
    - Region 1: TP count > 7000
    - Region 2: Visible flat region





ADC Integral histogram

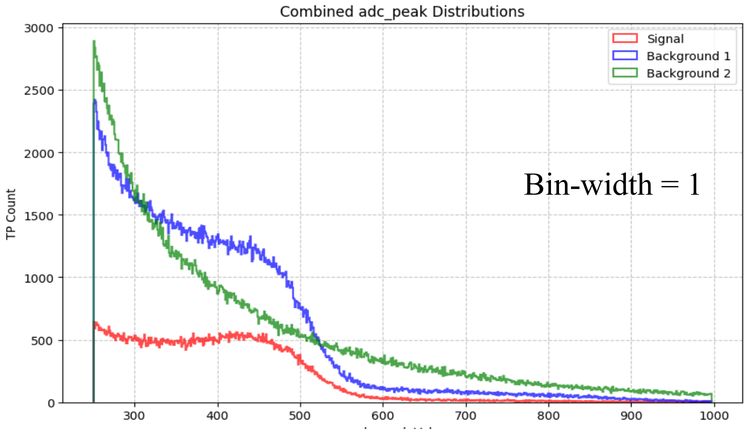
Suggestion: Plot the regions separately for ADC integral histogram

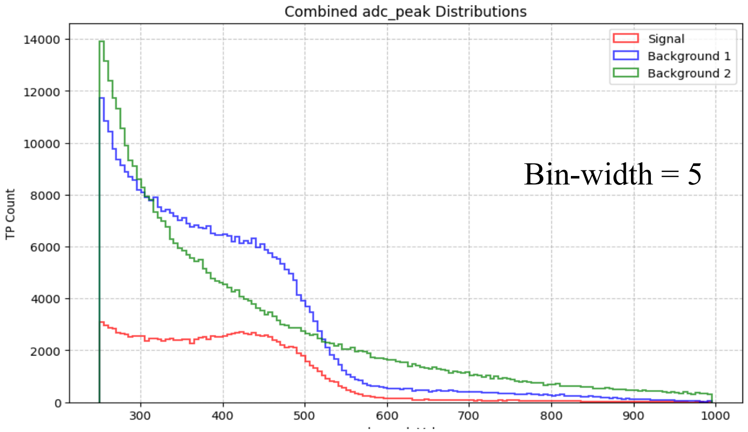


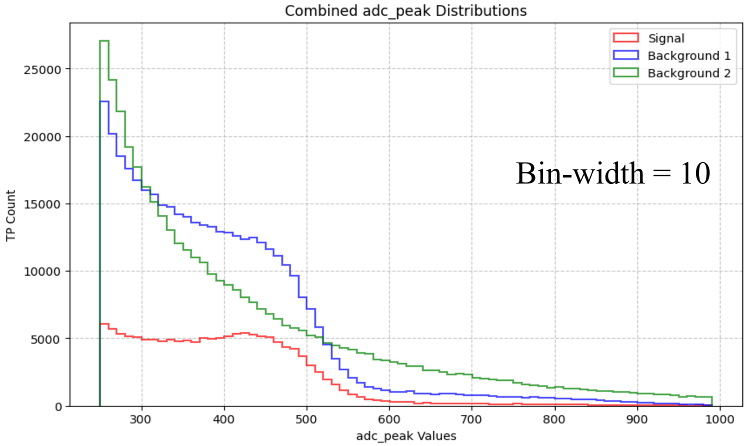
Bin-width = 25

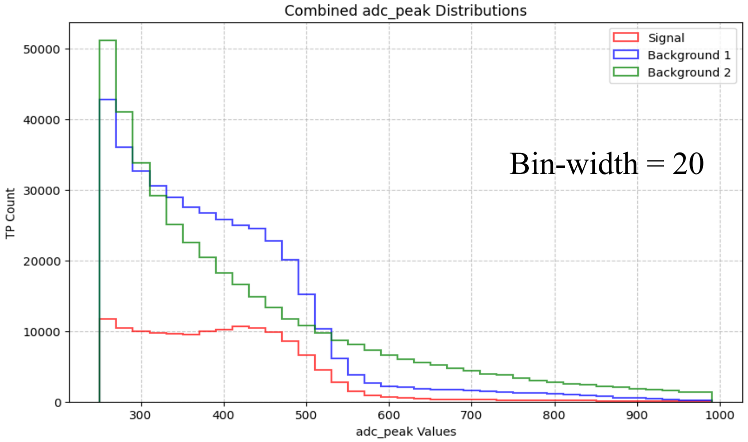
ADC Peak Histogram

Suggestion: Check ADC peak histogram with varying bin width and with unit bin width

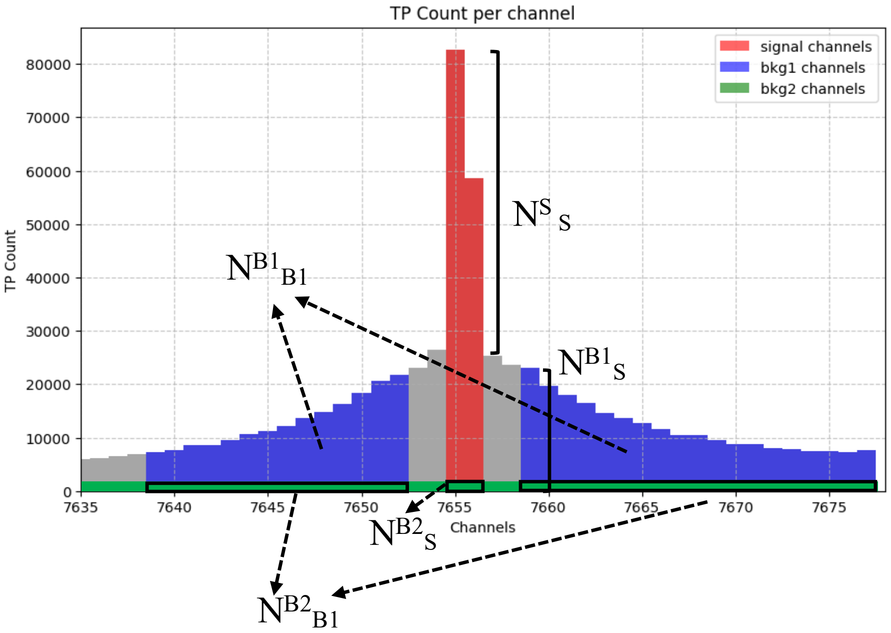








Background estimation Math



- N = Number of TPs
- Q = TP parameter histogram (e.g. : ADC integral)
- C = Number of channels
- Notation: $N^A{}_B$
  - A = Estimated quantity
  - B = Region

- To determine: $\boldsymbol{Q^S{}_S}$
- Assumption: Background 2 is relatively uniform in terms of TP number per channel as compared to the other two regions

Background 2 in Signal region:

$$Q_S^{B2} = \frac{Q_{B2}}{\cancel{N_{B2}}} \times \frac{\cancel{N_{B2}}}{C_{B2}} \times C_S$$

Background 2 in Background 1:

$$Q_{B1}^{B2} = \frac{Q_{B2}}{\cancel{N_{B2}}} \times \frac{\cancel{N_{B2}}}{C_{B2}} \times C_{B1}$$

Background 1 (after Background 2 removal) estimation in Signal region:

$$Q_{B1}^{B1} = Q_{B1} - Q_{B1}^{B2}$$

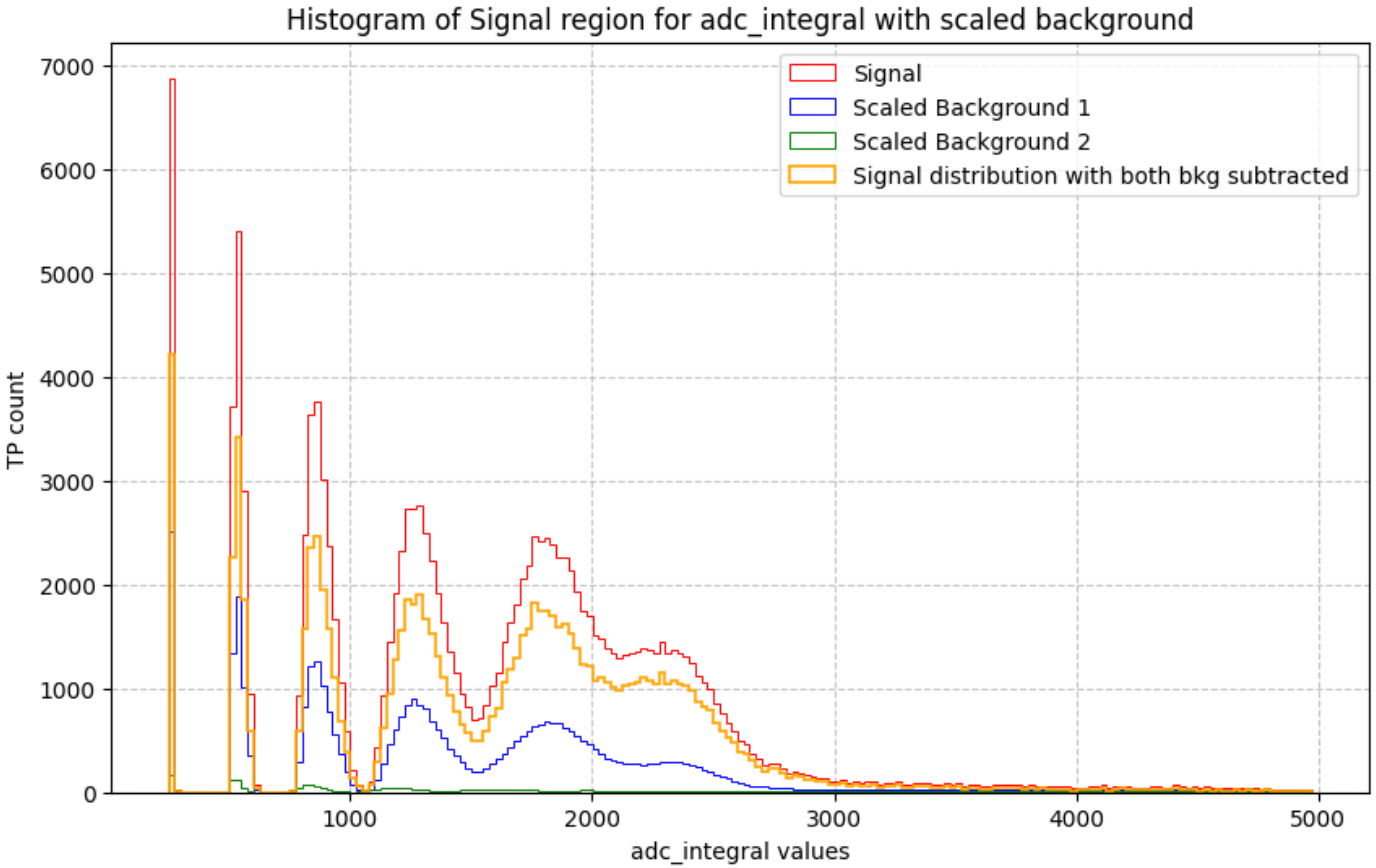
$$Q_S^{B1} = C_S \times (N_S^{B1} - N_S^{B2}) \times \frac{Q_{B1}^{B1}}{N_{B1} - (N_{B2} \times \frac{C_{B1}}{C_{B2}})}$$

Final equation:

$$Q_S^S = Q_S - Q_S^{B1} - Q_S^{B2} \quad (N_S^S = N_S - (N_S^{B1} - N_S^{B2}))$$

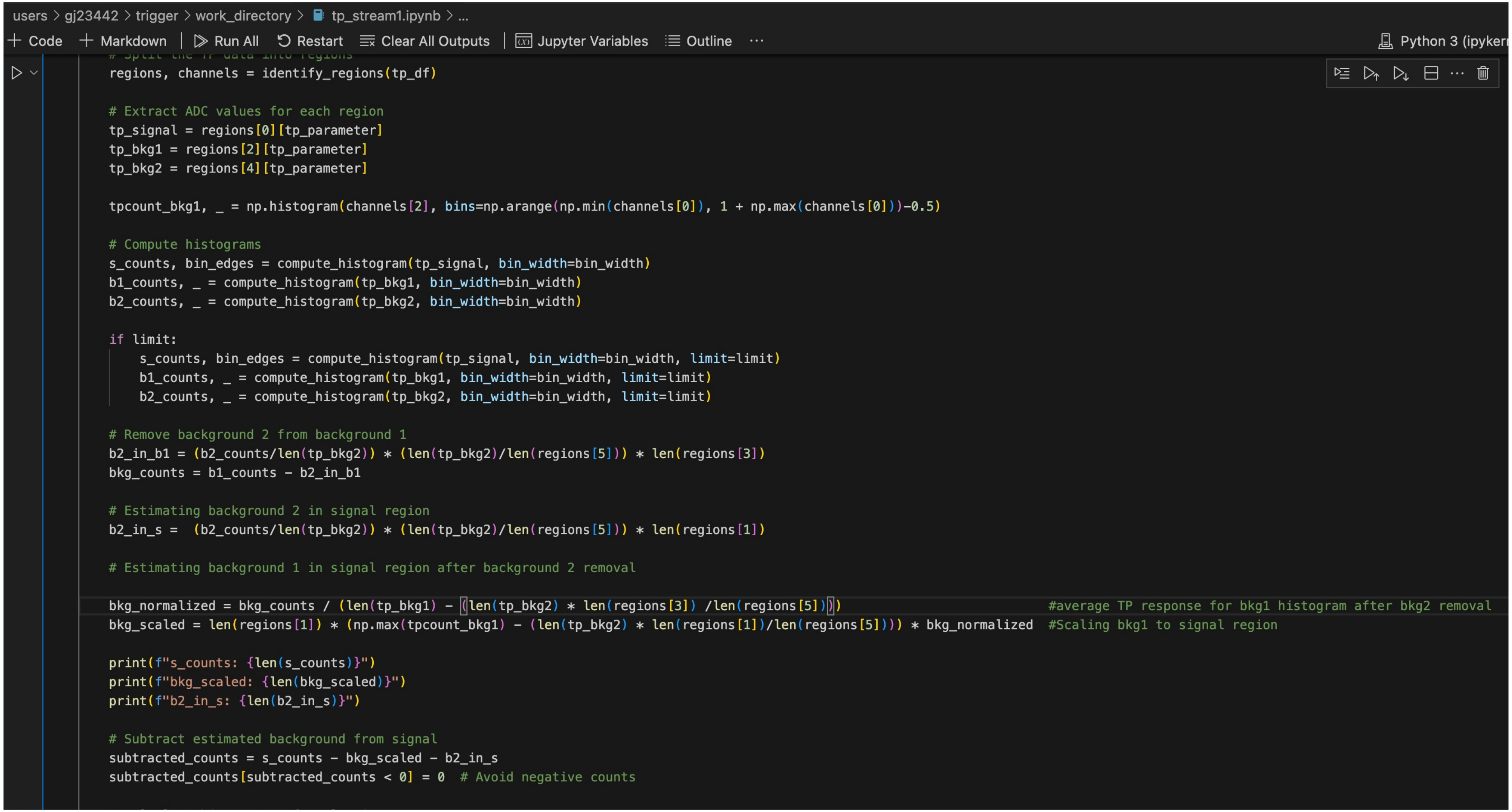
Apologies for the messy diagram, hopefully it clears confusion instead of increasing it

ADC Integral for Signal region



THANK YOU!

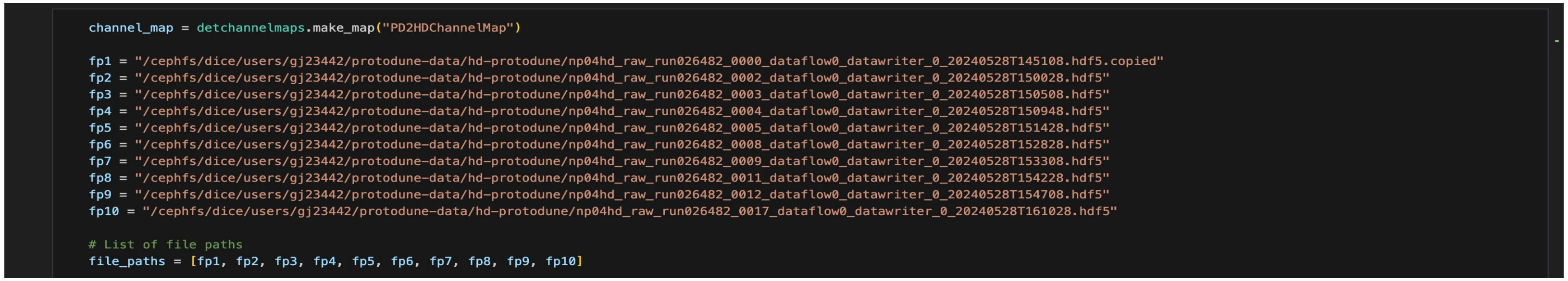
Background estimation implementation



```python
# Split the TP data into regions
regions, channels = identify_regions(tp_df)

# Extract ADC values for each region
tp_signal = regions[0][tp_parameter]
tp_bkg1 = regions[2][tp_parameter]
tp_bkg2 = regions[4][tp_parameter]

tpcount_bkg1, _ = np.histogram(channels[2], bins=np.arange(np.min(channels[0]), 1 + np.max(channels[0]))-0.5)

# Compute histograms
s_counts, bin_edges = compute_histogram(tp_signal, bin_width=bin_width)
b1_counts, _ = compute_histogram(tp_bkg1, bin_width=bin_width)
b2_counts, _ = compute_histogram(tp_bkg2, bin_width=bin_width)

if limit:
    s_counts, bin_edges = compute_histogram(tp_signal, bin_width=bin_width, limit=limit)
    b1_counts, _ = compute_histogram(tp_bkg1, bin_width=bin_width, limit=limit)
    b2_counts, _ = compute_histogram(tp_bkg2, bin_width=bin_width, limit=limit)

# Remove background 2 from background 1
b2_in_b1 = (b2_counts/len(tp_bkg2)) * (len(tp_bkg2)/len(regions[5])) * len(regions[3])
bkg_counts = b1_counts - b2_in_b1

# Estimating background 2 in signal region
b2_in_s =  (b2_counts/len(tp_bkg2)) * (len(tp_bkg2)/len(regions[5])) * len(regions[1])

# Estimating background 1 in signal region after background 2 removal

bkg_normalized = bkg_counts / (len(tp_bkg1) - (len(tp_bkg2) * len(regions[3]) /len(regions[5])))           #average TP response for bkg1 histogram after bkg2 removal
bkg_scaled = len(regions[1]) * (np.max(tpcount_bkg1) - (len(tp_bkg2) * len(regions[1])/len(regions[5]))) * bkg_normalized  #Scaling bkg1 to signal region

print(f"s_counts: {len(s_counts)}")
print(f"bkg_scaled: {len(bkg_scaled)}")
print(f"b2_in_s: {len(b2_in_s)}")

# Subtract estimated background from signal
subtracted_counts = s_counts - bkg_scaled - b2_in_s
subtracted_counts[subtracted_counts < 0] = 0  # Avoid negative counts
```

Outline of Bismuth TP Analysis

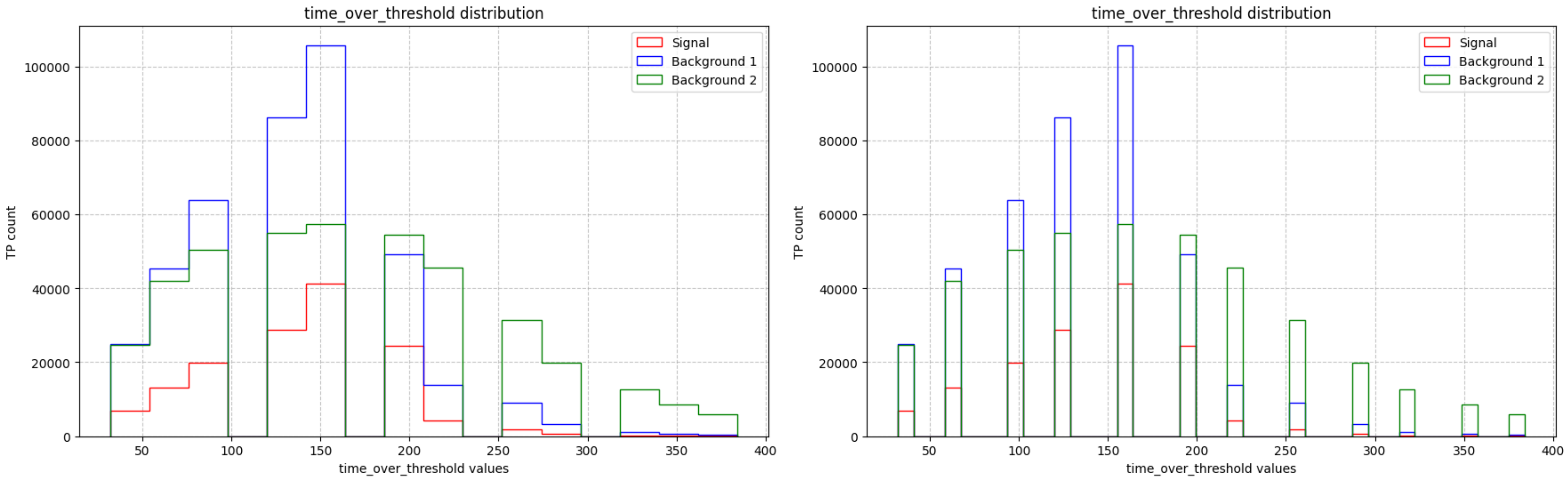
1. Raw data for APA 2 channels for run 026482 TR datafile
2. Analysis with more data:



```
channel_map = detchannelmaps.make_map("PD2HDChannelMap")

fp1 = "/cephfs/dice/users/gj23442/protodune-data/hd-protodune/np04hd_raw_run026482_0000_dataflow0_datawriter_0_20240528T145108.hdf5.copied"
fp2 = "/cephfs/dice/users/gj23442/protodune-data/hd-protodune/np04hd_raw_run026482_0002_dataflow0_datawriter_0_20240528T150028.hdf5"
fp3 = "/cephfs/dice/users/gj23442/protodune-data/hd-protodune/np04hd_raw_run026482_0003_dataflow0_datawriter_0_20240528T150508.hdf5"
fp4 = "/cephfs/dice/users/gj23442/protodune-data/hd-protodune/np04hd_raw_run026482_0004_dataflow0_datawriter_0_20240528T150948.hdf5"
fp5 = "/cephfs/dice/users/gj23442/protodune-data/hd-protodune/np04hd_raw_run026482_0005_dataflow0_datawriter_0_20240528T151428.hdf5"
fp6 = "/cephfs/dice/users/gj23442/protodune-data/hd-protodune/np04hd_raw_run026482_0008_dataflow0_datawriter_0_20240528T152828.hdf5"
fp7 = "/cephfs/dice/users/gj23442/protodune-data/hd-protodune/np04hd_raw_run026482_0009_dataflow0_datawriter_0_20240528T153308.hdf5"
fp8 = "/cephfs/dice/users/gj23442/protodune-data/hd-protodune/np04hd_raw_run026482_0011_dataflow0_datawriter_0_20240528T154228.hdf5"
fp9 = "/cephfs/dice/users/gj23442/protodune-data/hd-protodune/np04hd_raw_run026482_0012_dataflow0_datawriter_0_20240528T154708.hdf5"
fp10 = "/cephfs/dice/users/gj23442/protodune-data/hd-protodune/np04hd_raw_run026482_0017_dataflow0_datawriter_0_20240528T161028.hdf5"

# List of file paths
file_paths = [fp1, fp2, fp3, fp4, fp5, fp6, fp7, fp8, fp9, fp10]
```

3. Set of Background TPs:
    1. Clustering to remove data from cosmics
    2. Zoom in on Bismuth active region
4. Plotting ADC value histogram for signal (Bismuth) and background region
5. To get only Bismuth TPs: Subtraction of the expected background TPs in the signal region to estimate only Bismuth TPs

Time Over Threshold Histogram



- Bin-width = 25 (left) and 10 (right)

Steps for accessing ProtoDUNE data in DICE: Rucio

Use metacat to find the files and rucio to locate where they are stored and also download
Requirements: FNAL Username and Password (both Services and Kerberos)
Get list of datafiles from here: https://metacat.fnal.gov:9443/dune_meta_prod/app/gui/datasets
List of commands:

```
•   . /cvmfs/larsoft.opensciencegrid.org/spack-packages/setup-env.sh
•   spack load r-m-dd-config experiment=dune
•   spack load kx509
•   kinit <FNAL_username>@FNAL.GOV
•   kx509
•   export ROLE=Analysis
•   voms-proxy-init -rfc -noregen -voms=dune:/dune/Role=$ROLE -valid 120:00
•   export RUCIO_ACCOUNT=<fnal_username>
•   rucio whoami
•   rucio list-file-replicas <detector-source>:<file-name>
•   rucio download <detector-source>:<file-name>
```

* Example: hd-protodune:np04hd_raw_run026482_0002_dataflow0_datawriter_0_20240528T150028.hdf5

Steps for accessing ProtoDUNE data in DICE

In Alma9 (done in sc01):

Requirements: Fermilab username and Kerberos password

Get list of datafiles from here: https://metacat.fnal.gov:9443/dune_meta_prod/app/gui/datasets

List of commands:

```
•  . /cvmfs/larsoft.opensciencegrid.org/spack-packages/setup-env.sh
•  spack load metacat
•  spack load kx509
•  kinit <username>@FNAL.GOV
•  kx509
•  export ROLE=Analysis
•  voms-proxy-init -rfc -noregen -voms=dune:/dune/Role=$ROLE -valid 120:00
•  export METACAT_AUTH_SERVER_URL=https://metacat.fnal.gov:8143/auth/dune
•  export METACAT_SERVER_URL=https://metacat.fnal.gov:9443/dune_meta_prod/app
•  metacat auth login -m password <fnal_username>
•  metacat query "files from dune:all where core.file_type=detector \
     and core.run_type=hd-protodune and core.data_tier=raw \
     and core.data_stream=cosmics and core.runs[any]=27296 limit 2”
```

* Further metacat operating instructions: Here