Unsupervised learning for low energy spectroscopy's waveform

Ken

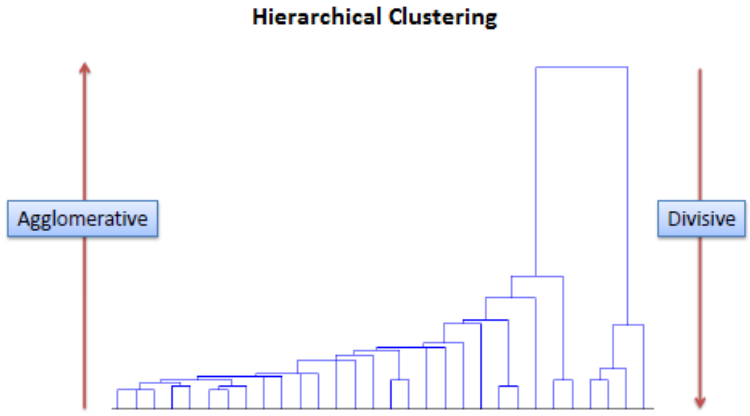
June 15, 2018

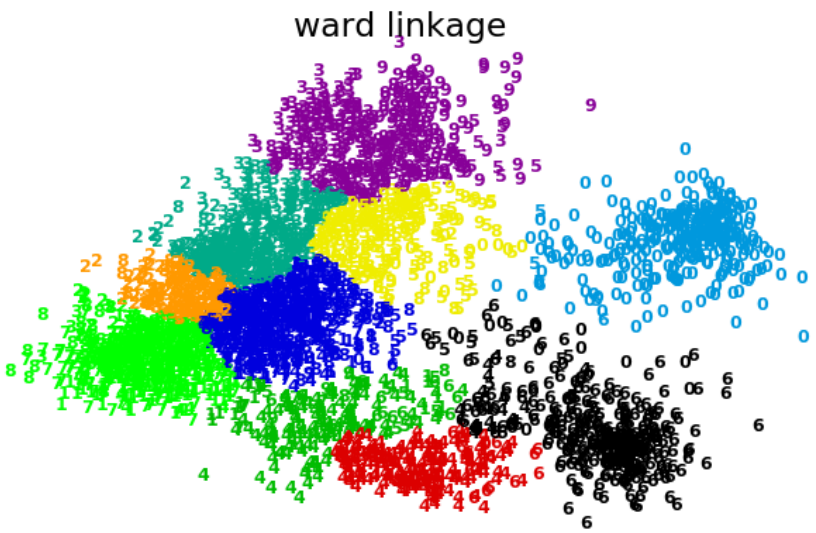
~~CNN for waveform classification~~
Unsupervised learning for low energy spectroscopy's waveform

Lets talk about the algorithm/model/code(whatever suitable) itself, Title not finalised
the 2 main concepts

- Hierarchical Clustering
- Auto-encoder

Hierarchical Clustering



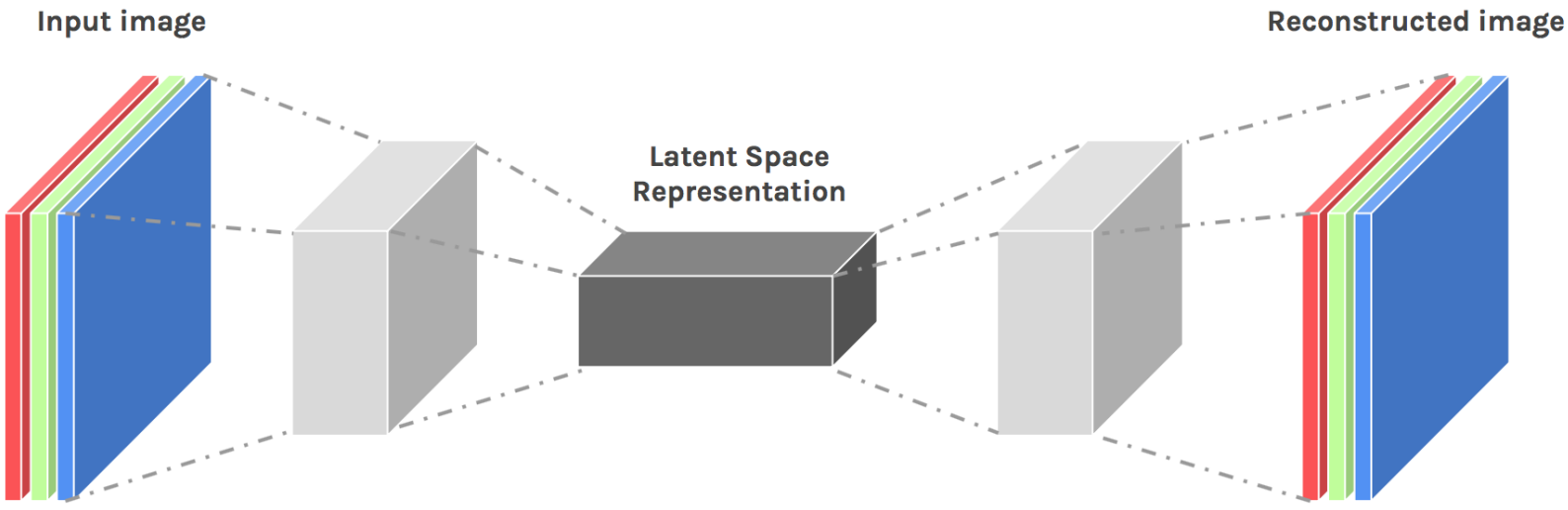


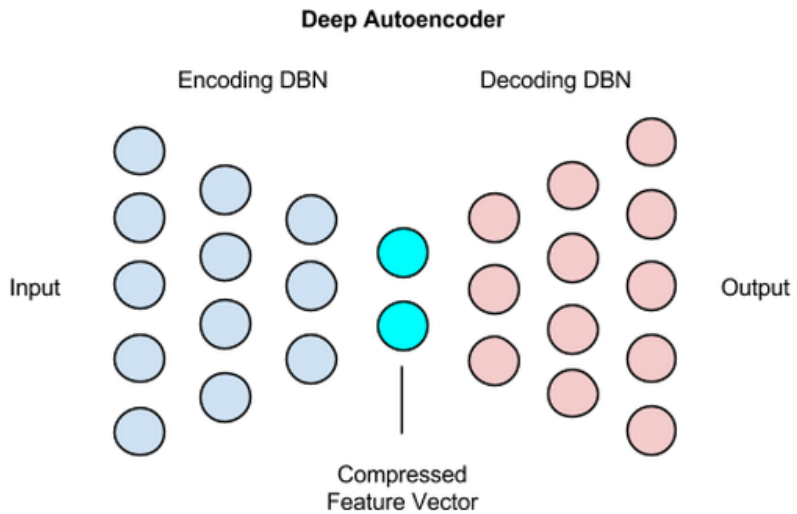
How?

The "distance" between each data is calculated. This "distance" indicates how closely the data are related to one and another. The data can be "arrange" in such a way that where the distance between neighbours indicates how closely related they are. χ^2 is a type of "distance"

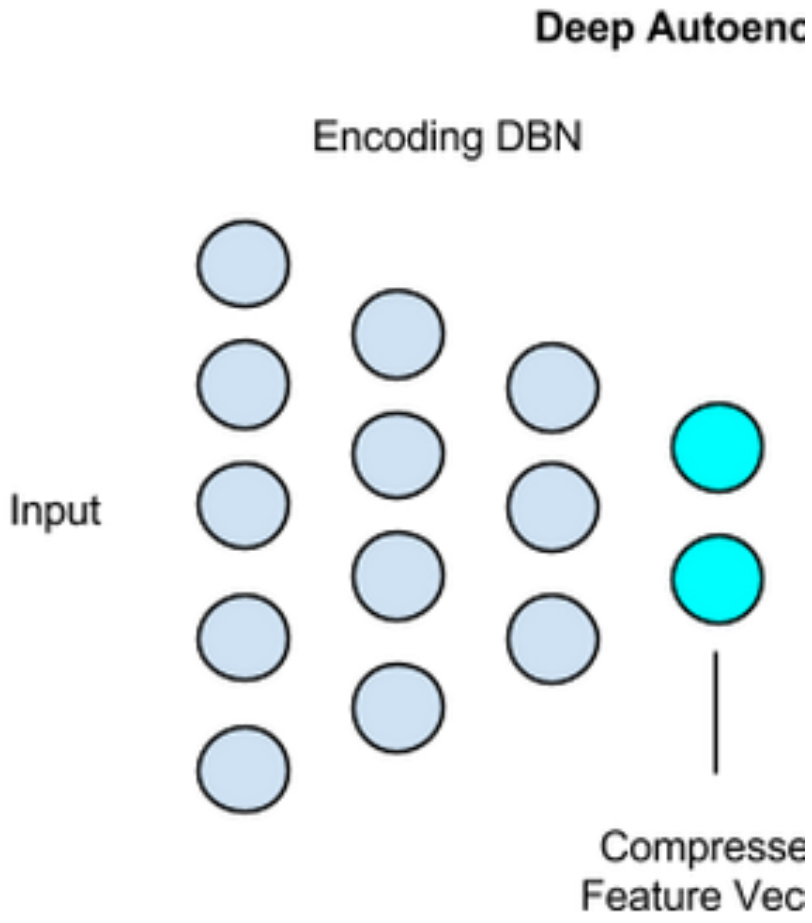
tldr. data that have similar features will tend to cluster together.

Auto-encoders





Taking the **encoder** of the auto encoder and do hierarchical clustering on it



The features of waveform(low level parameter) is reduced to few parameters (high level parameters/engineered features). Of course we can do this with waveform directly, but uneven "χ^2" of NN can highlight the features of waveform more substantially. Another reason is also to reduce the computing workload of the clustering.

Each data can be **tagged** to identify which group they belong

"Distance" for Hierarchical clustering - ward

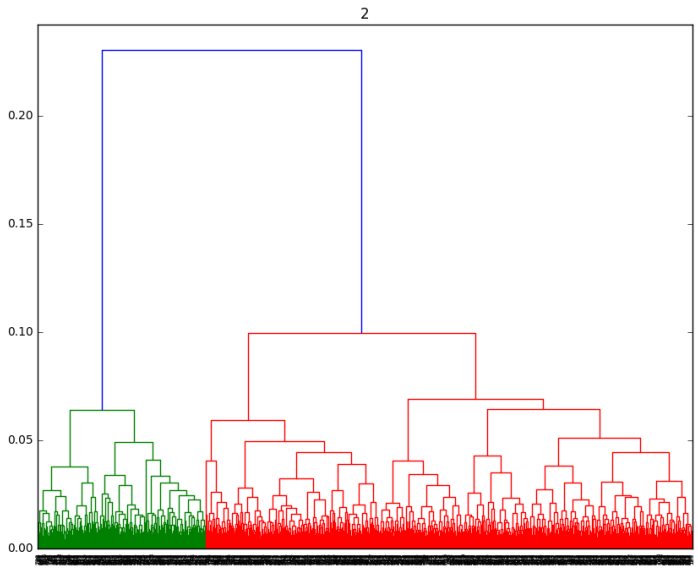
Divide and Conquer Hierarchical clustering of Encoder convoluted features

Training data?

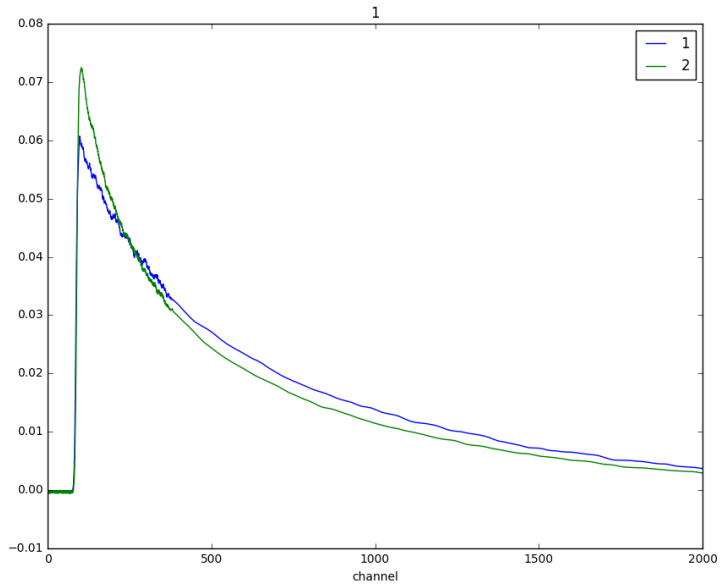
- The data we have itself is the training data.(Just need to clean up through normalisation and eliminate the pedestal)
- To make sure every node in the auto-encoder is rigorously "stretch", The data is flipped and shuffled.
- (Part of this reason is the decoder output, which can be use for de-noising as I noticed it when training the auto-encoder)
- *Auto-encoder itself is just comprises of convolution layers due its efficiency to capture features and less parameters of the neural network to train on.

Lets test it - Can it identify α and β/γ by itself?

A randomly shuffle beta and alpha. Visualising clustering in dendrogram.

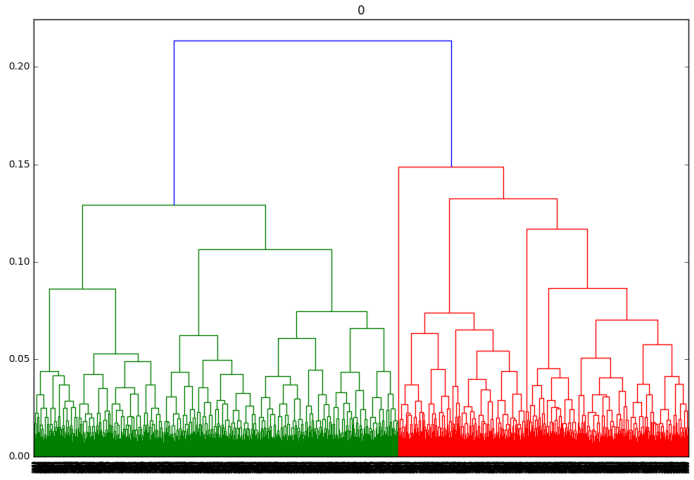


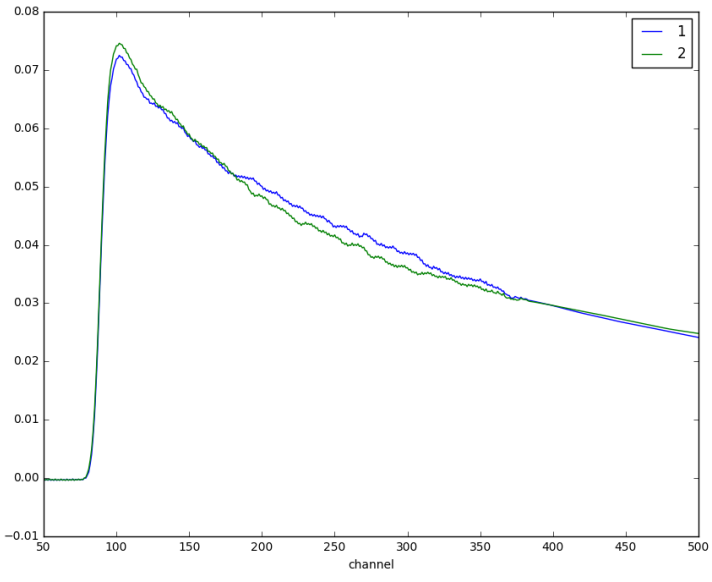
The averaged waveform of the two group

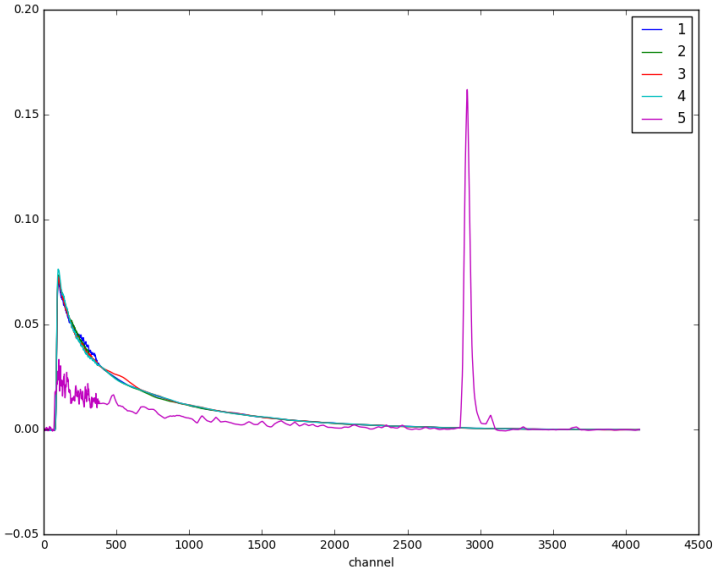


In this example, the beta is particularly noisy since it is a smaller group.

Lets test it - The α list?

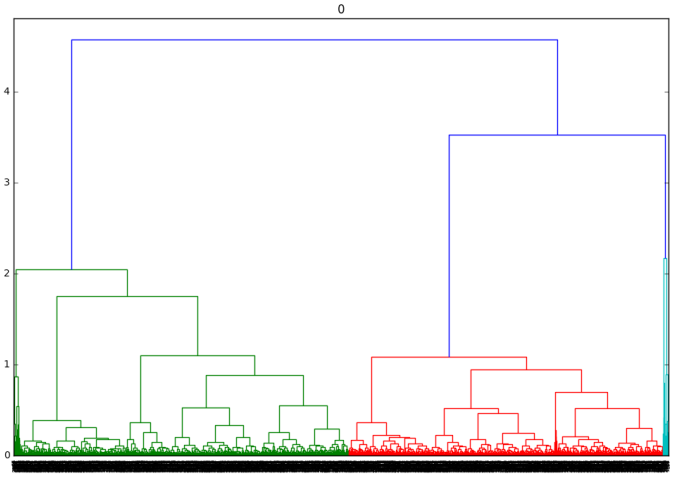




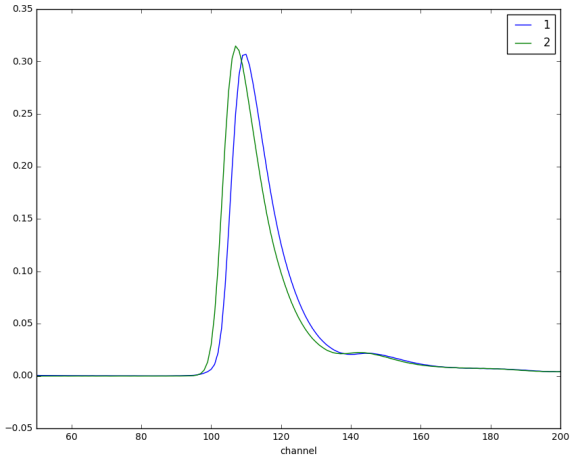


Picking shorter "distance" cut-off of clustering, to check the unique lone branch.

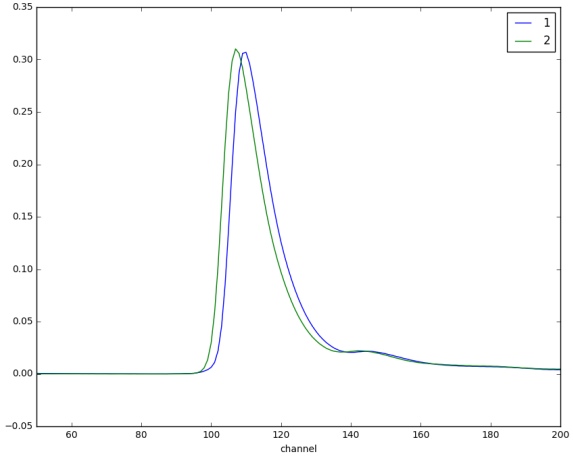
How about LS reference list?



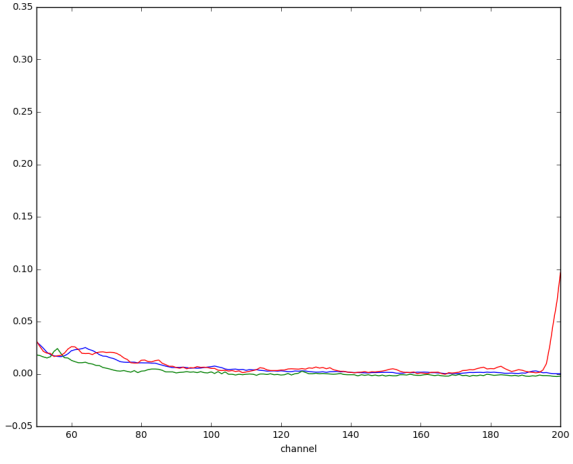
At 3 branch,



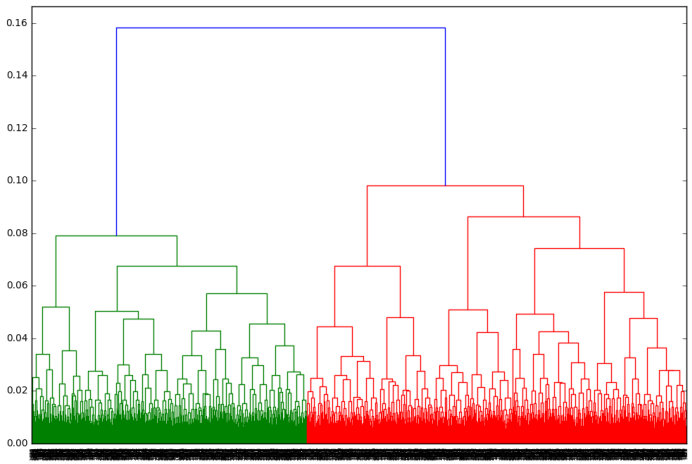
At 2 branch,

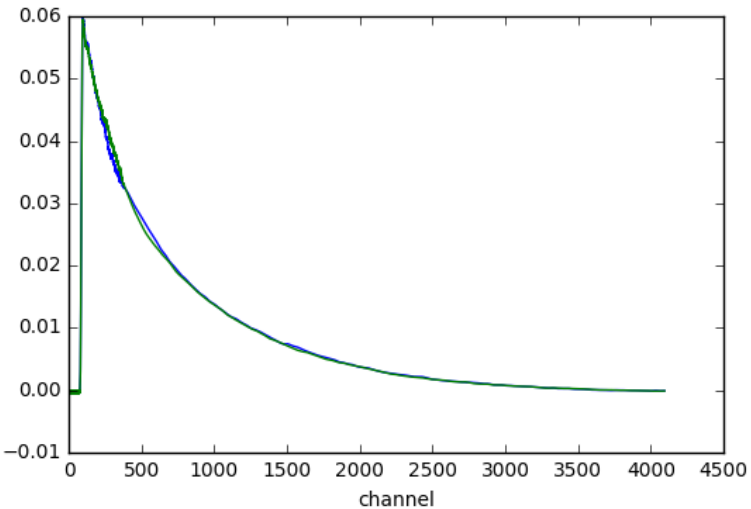


Above first 2 group, below 3rd sample waveforms

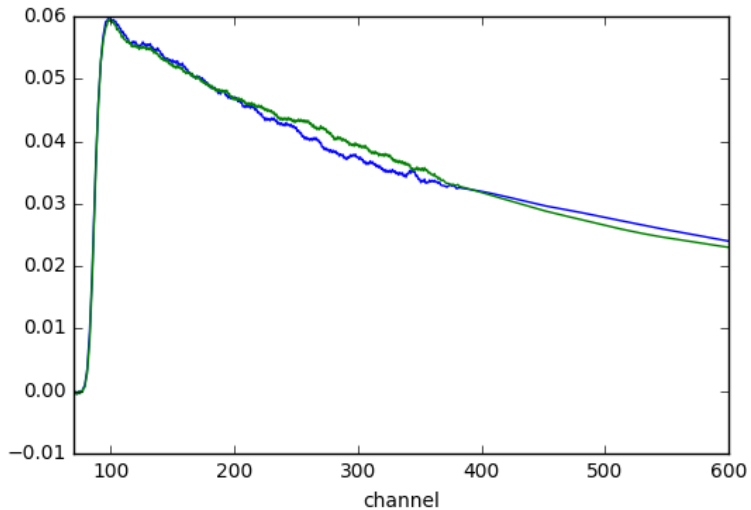


How about the β/γ reference pulse?





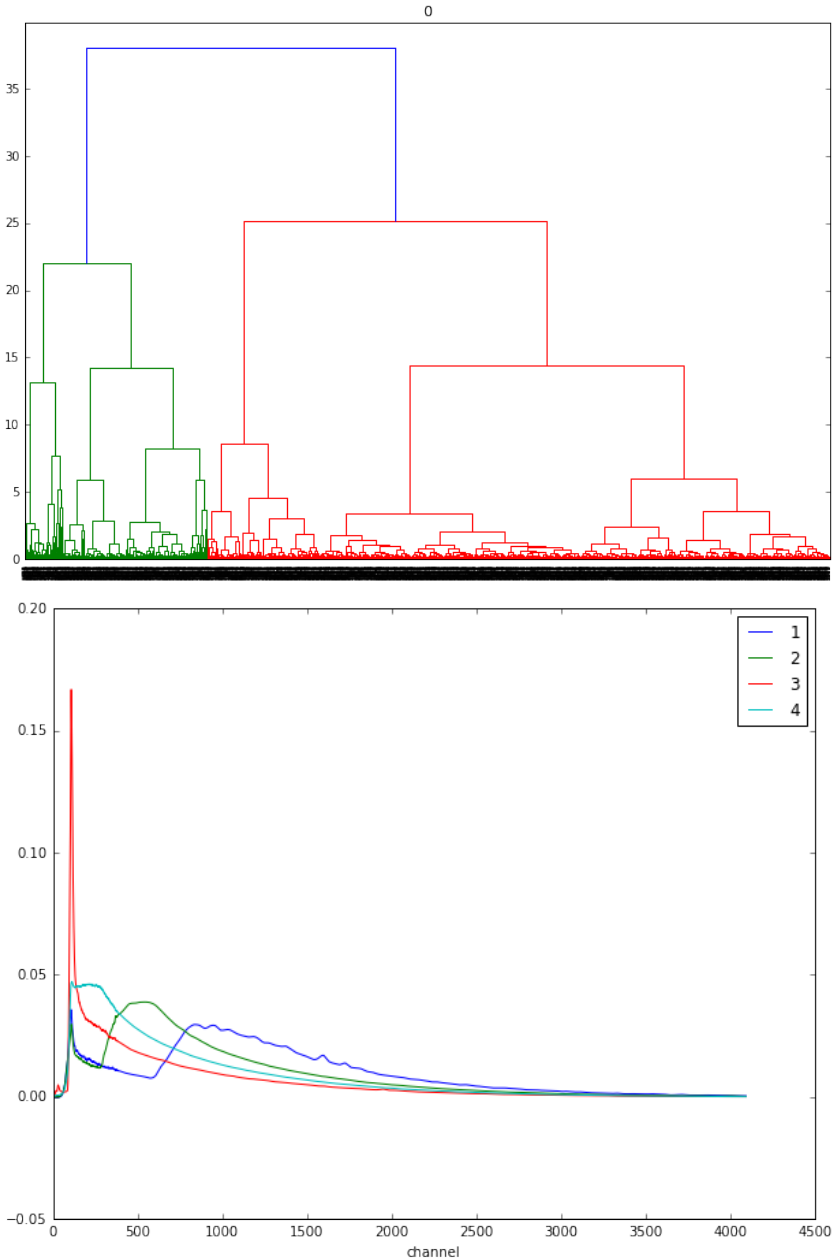
Zoom on the first 600 channels



A possibility of identifying unique waveform characteristic corresponding to the source causing it? - MAYBE? decay source/pmtall reconstruction difference/etc.? *too ambitious i think

Testing on a larger dataset that are more messy

- The dataset tested on: All of Run009 events that are above 3000keV and after GoodQuality cut. aprox$\sim$ 60,000 events.
- It is done in chunks, 3000 events. (limitation of computer).
- The results are verified manually. (ordering of the label - Still figuring out a way to do this elegantly)
- Selecting the wanted shape, reducing the event sample size further. down to $\sim$30,000 events.
- Repeat again until the remaining events are of what we wanted.

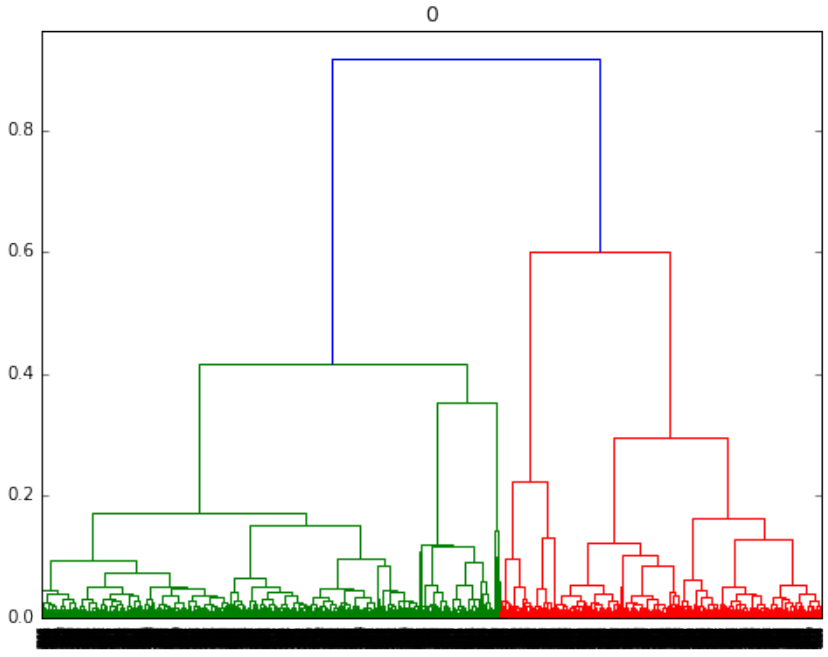


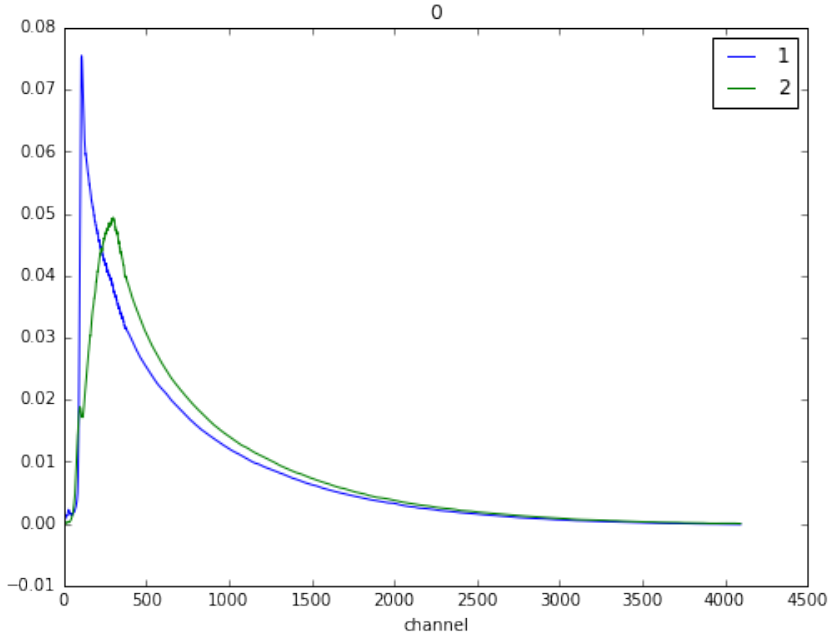
65675 events

Initially, the dataset will contain all kinds of waveforms. Decided on 4 clusters and the dendrogram appear to have same cut-of of 4 clusters. Since each event are tagged, retain the cluster wanted, this case is "4".

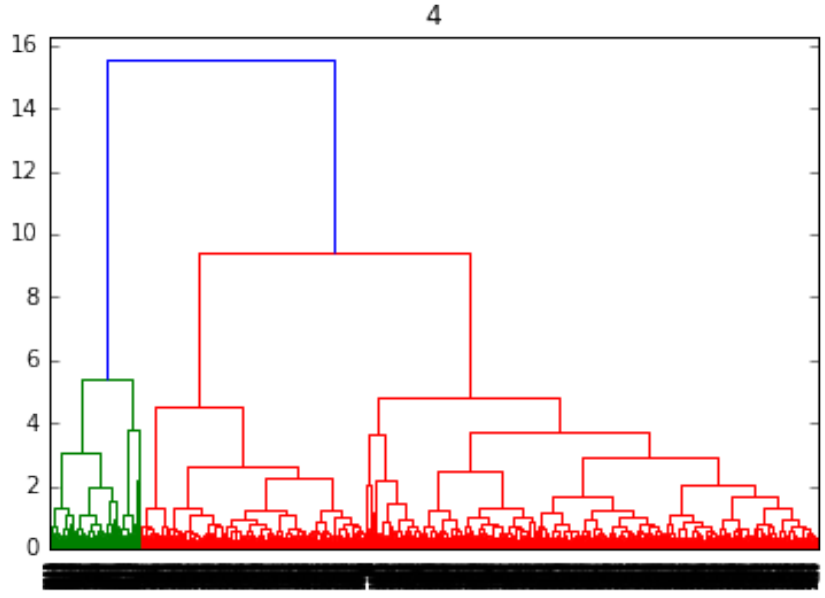
Repeat the clustering procedure again.

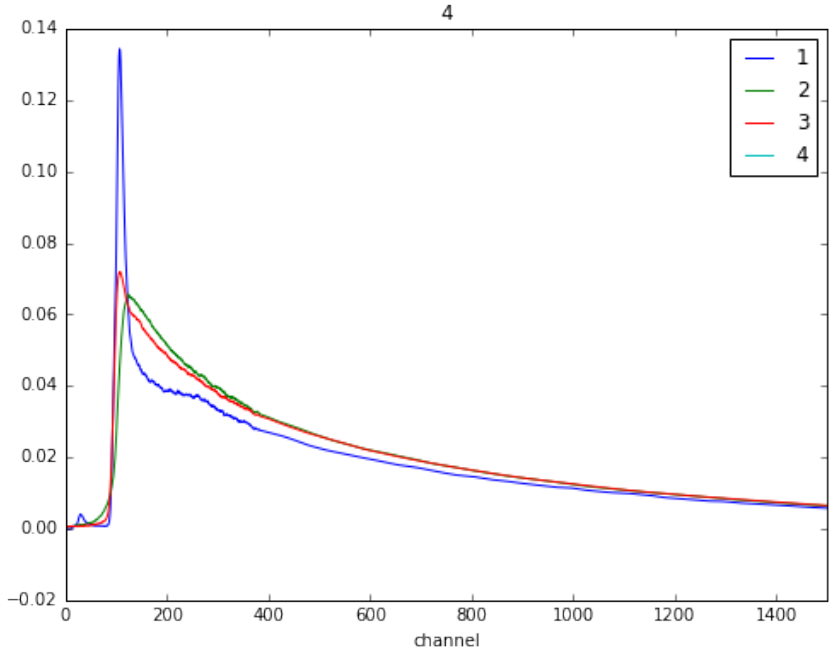
43785 events



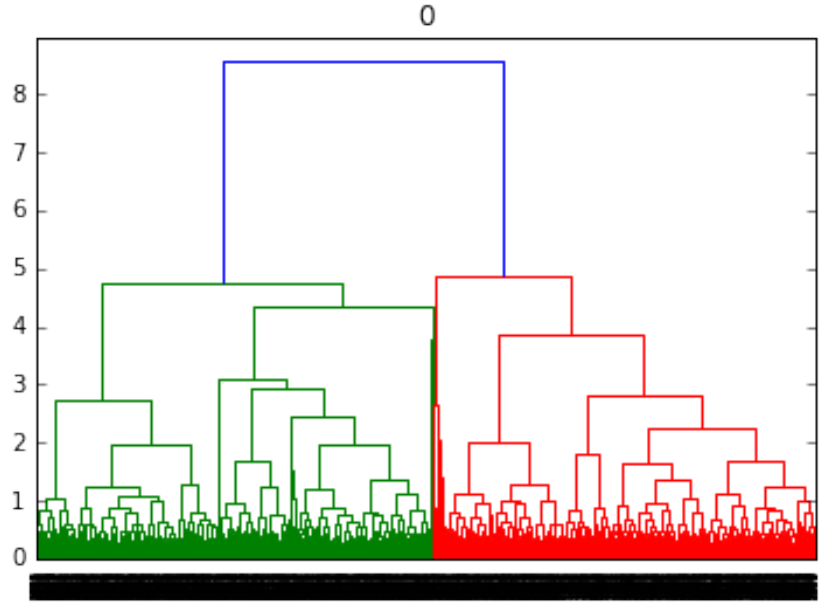


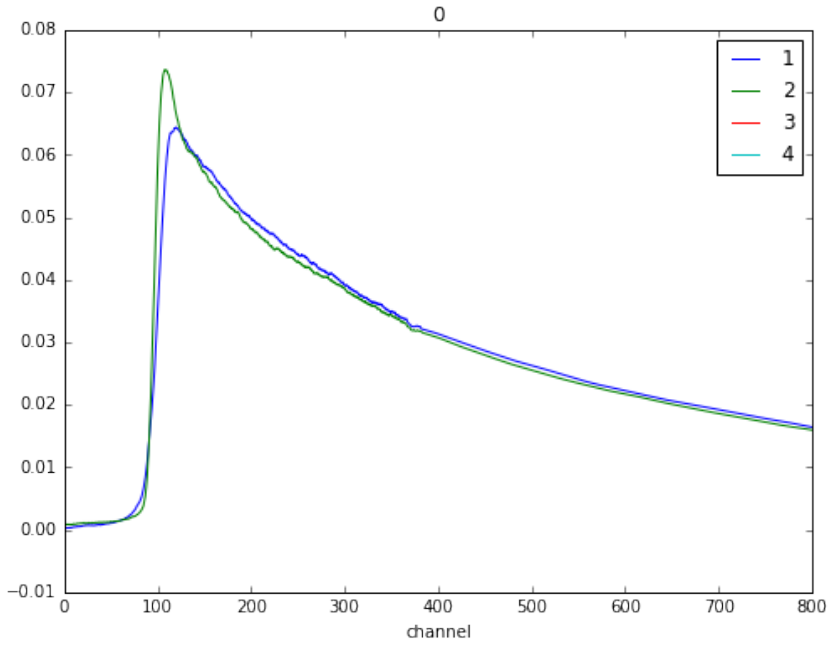
28045 events



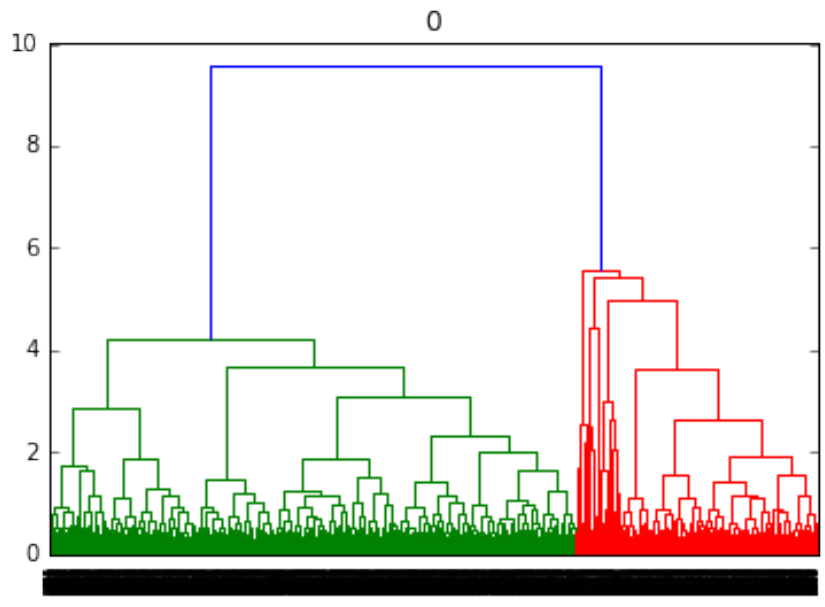


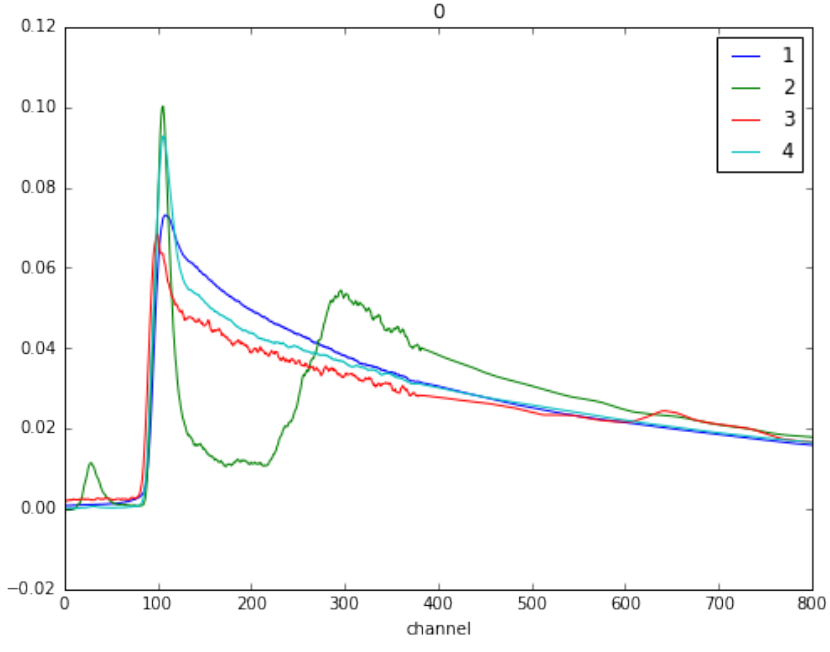
19589 events



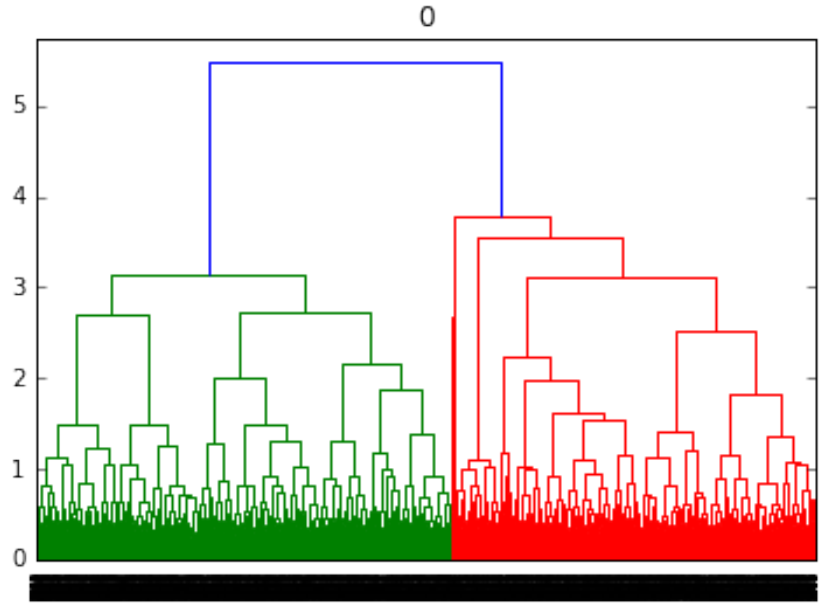


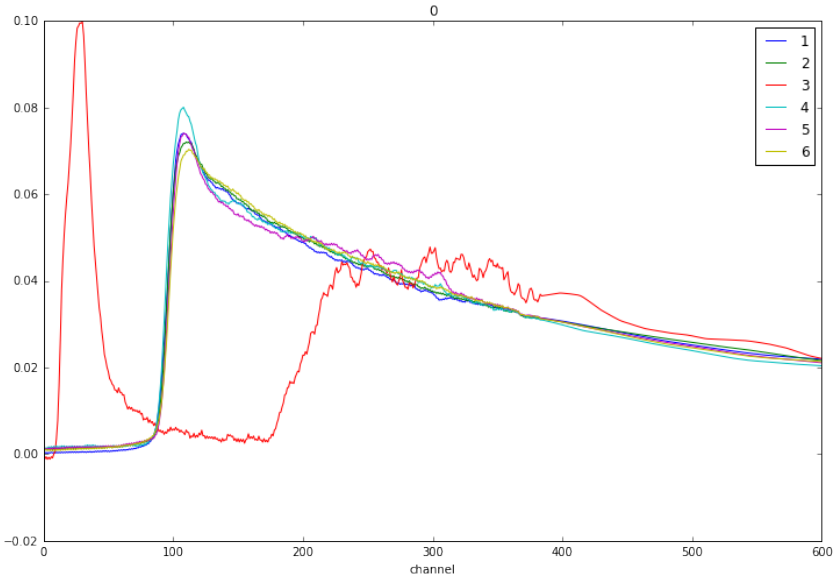
7416 events



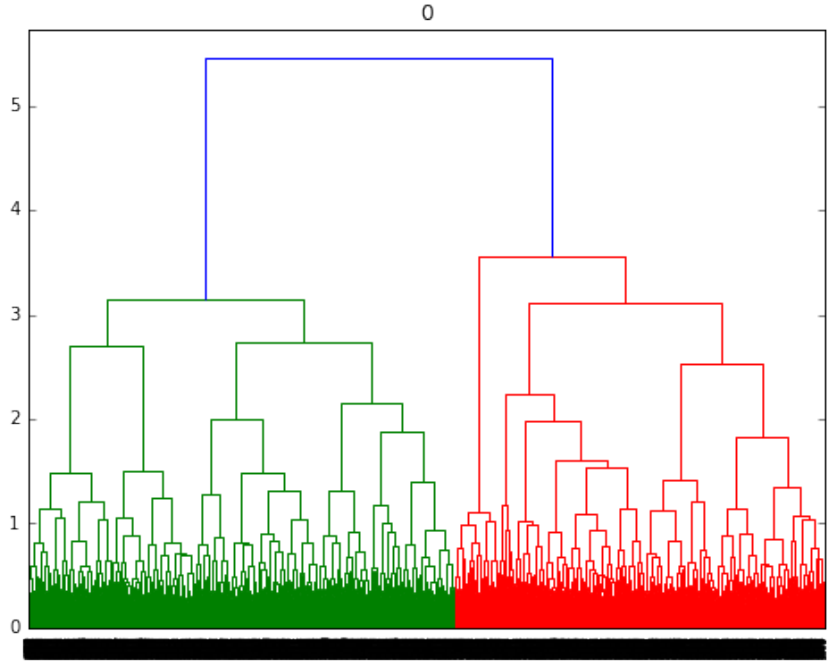


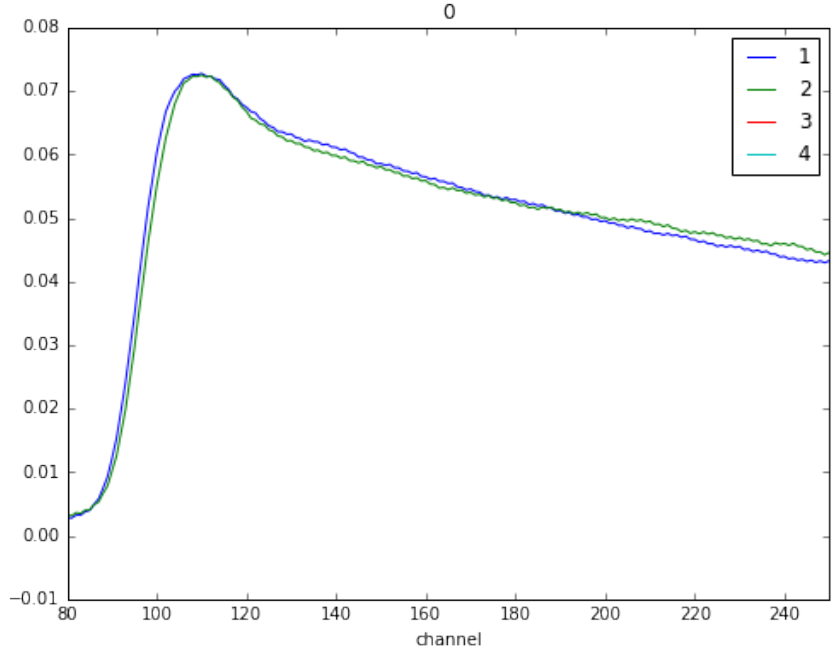
4181 events





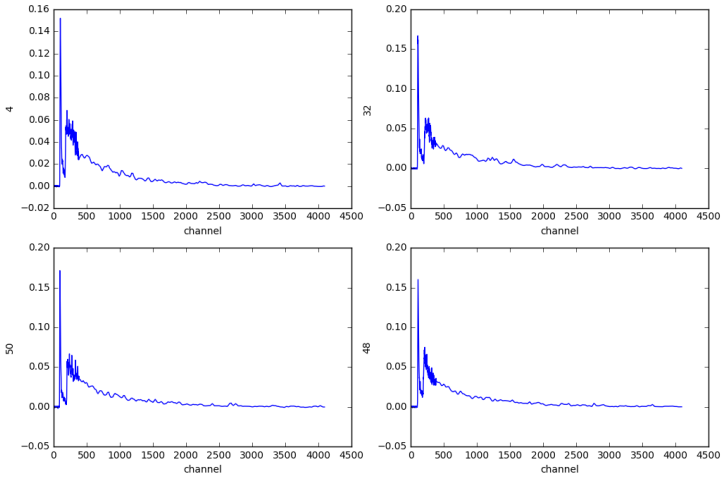
4168 events

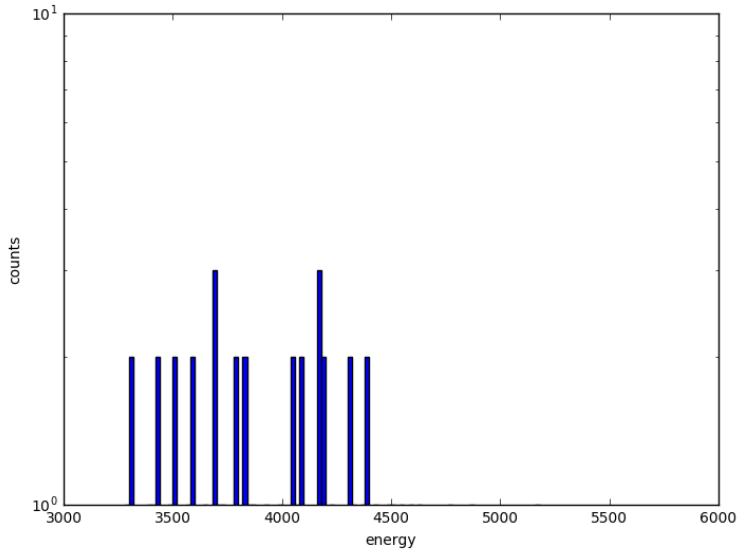


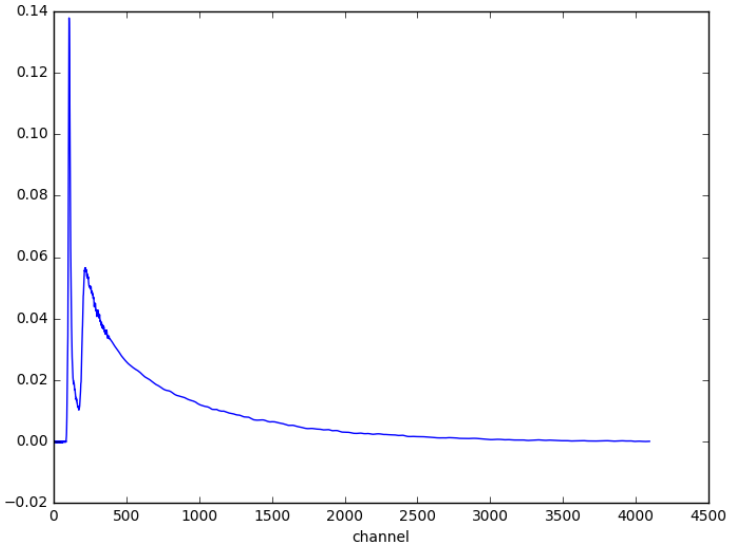


checking the 4168 events, at 8 clusters

1st cluster - 53 events

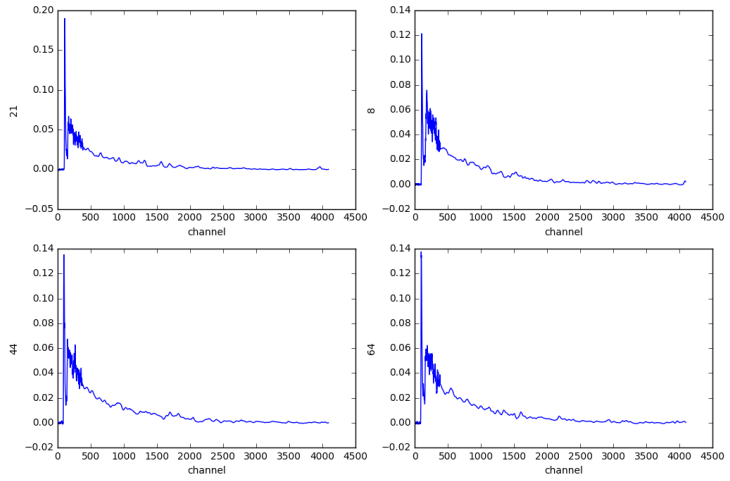


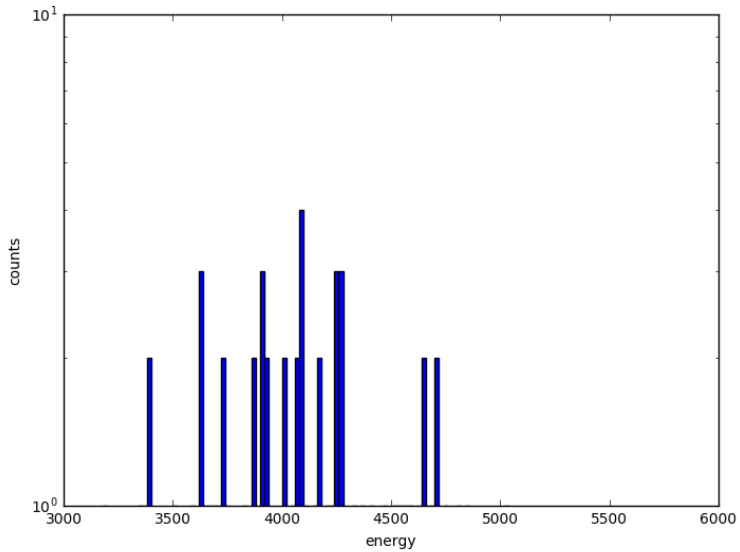


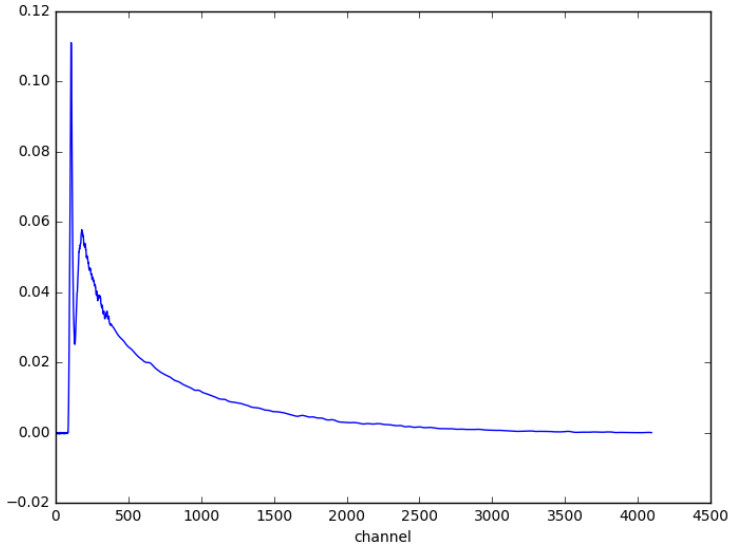


checking the 4168 events, at 8 clusters

2nd cluster - 66 events

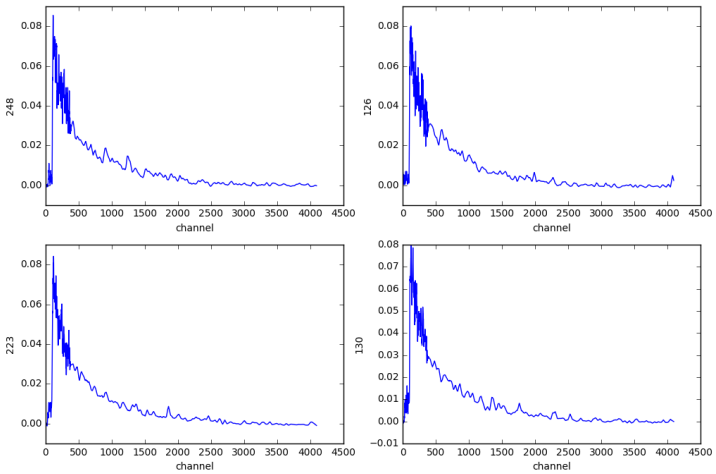


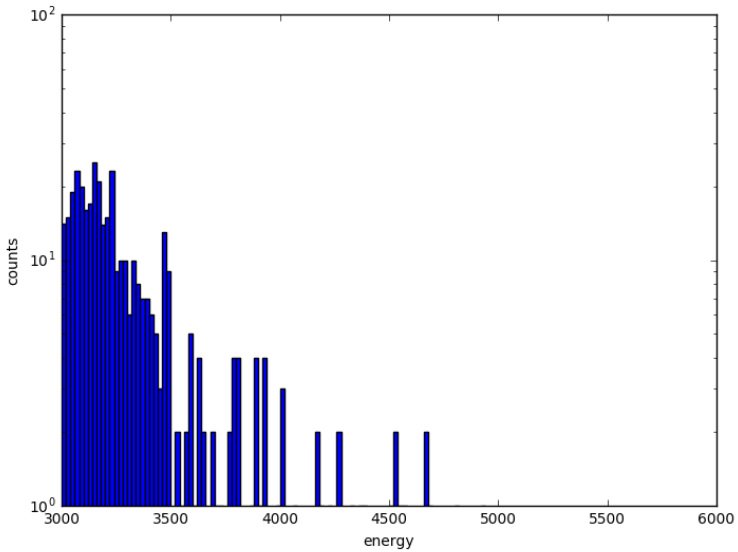


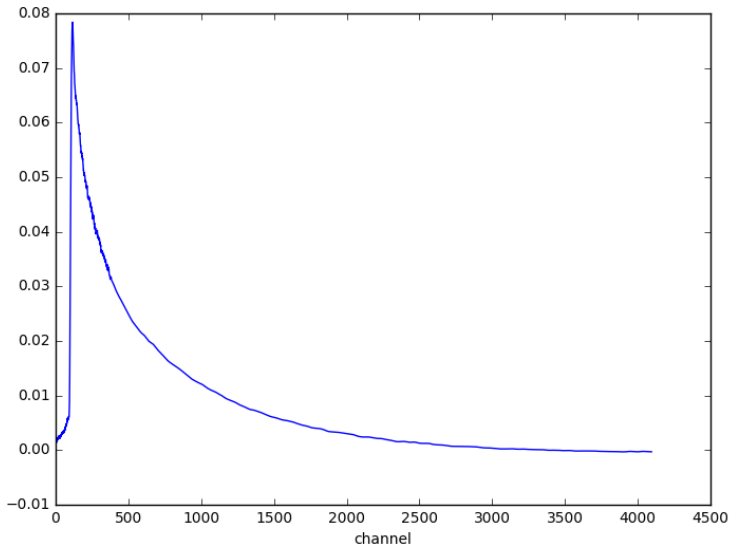


checking the 4168 events, at 8 clusters

3rd cluster - 390 events

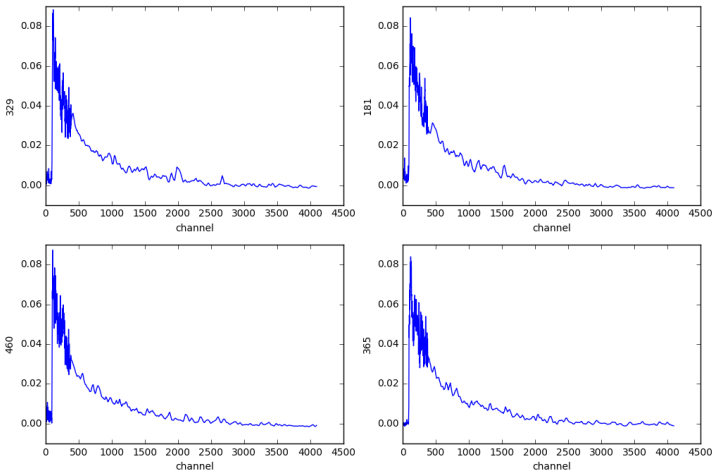


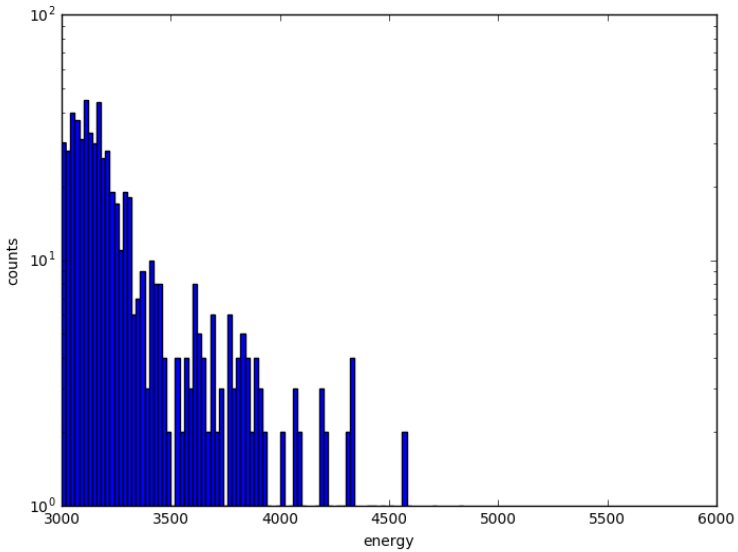


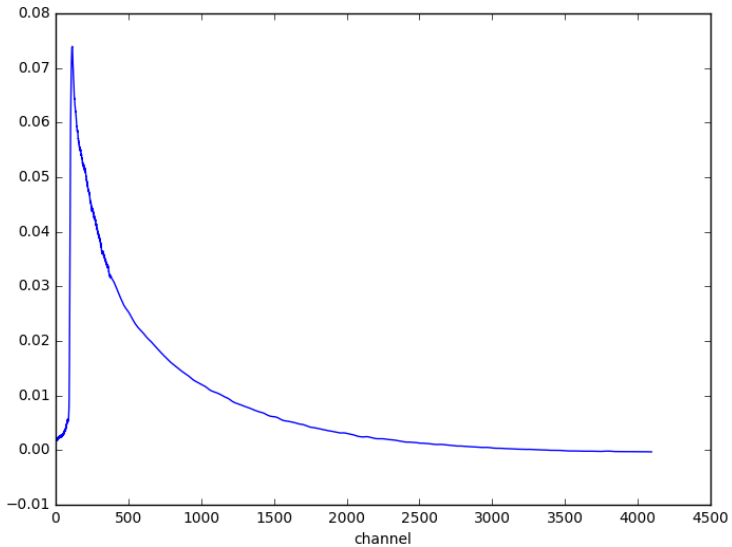


checking the 4168 events, at 8 clusters

4th cluster - 626 events

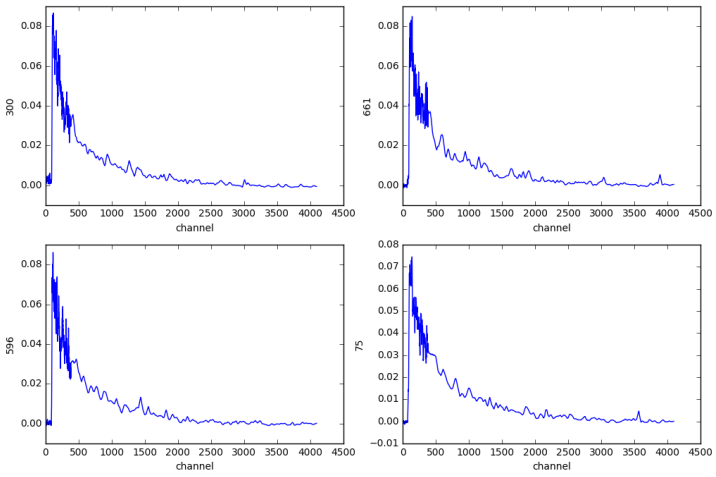


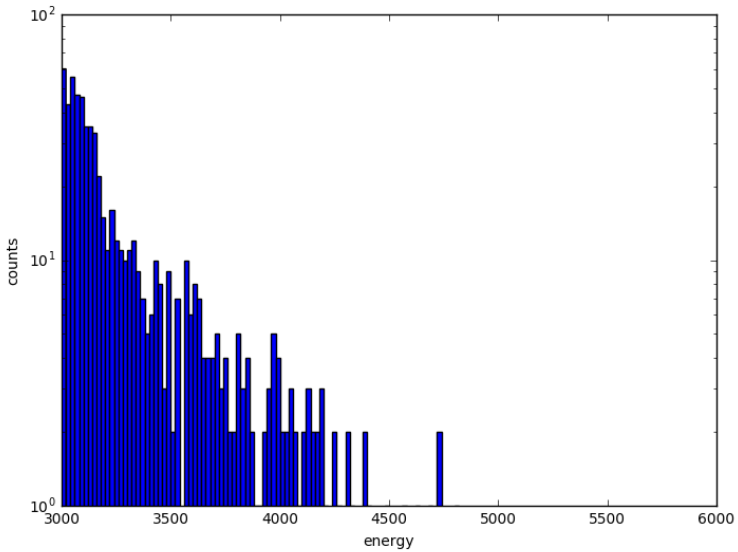


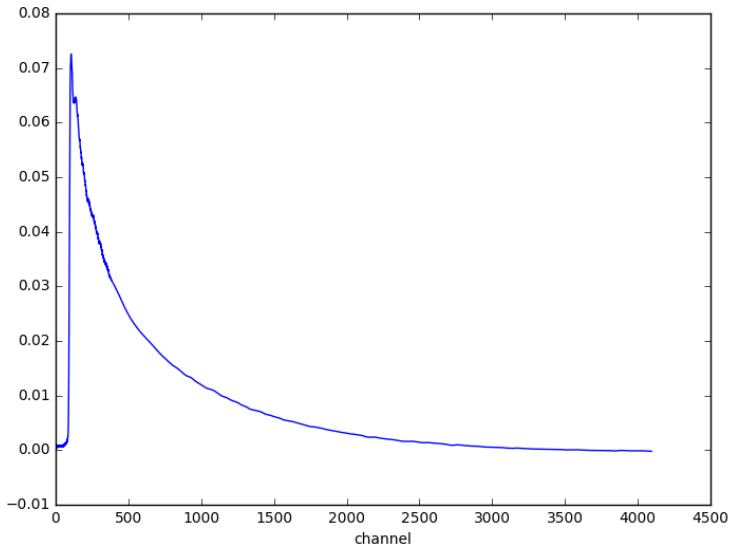


checking the 4168 events, at 8 clusters

5th cluster - 669 events

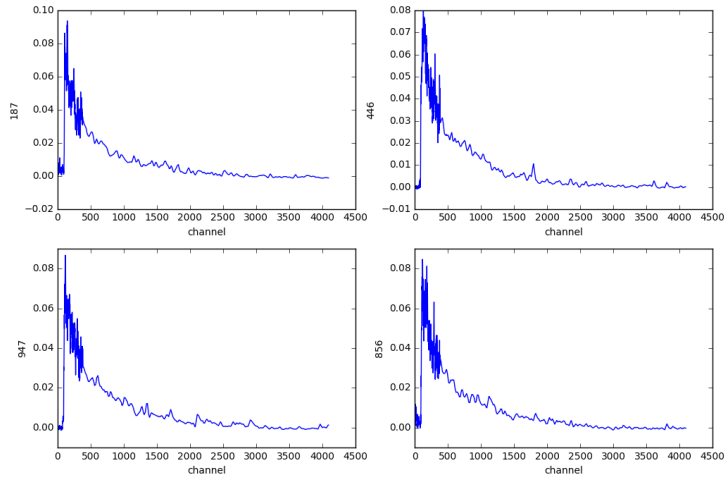


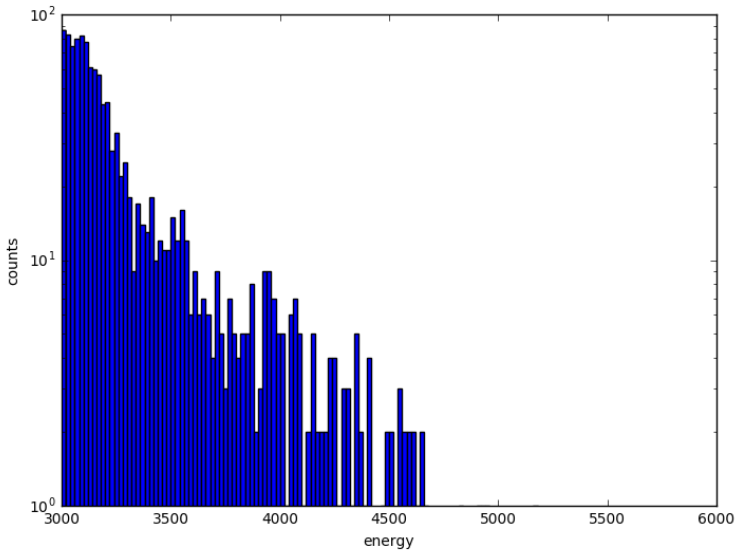


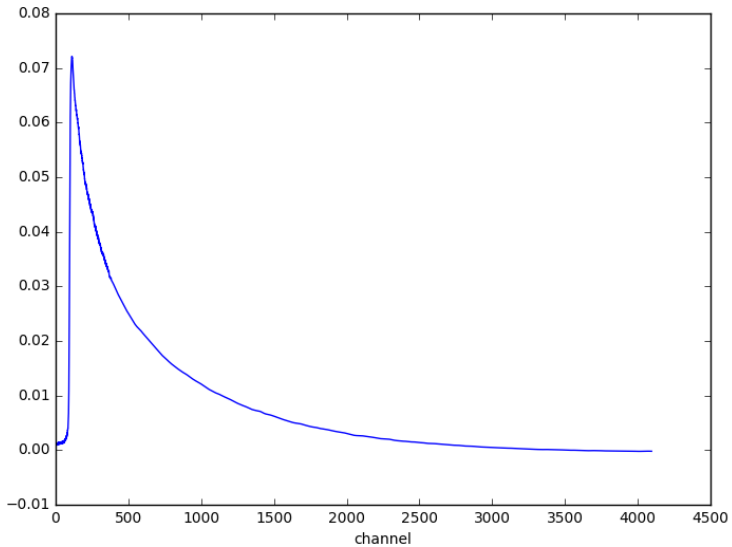


checking the 4168 events, at 8 clusters

6th cluster - 1256 events

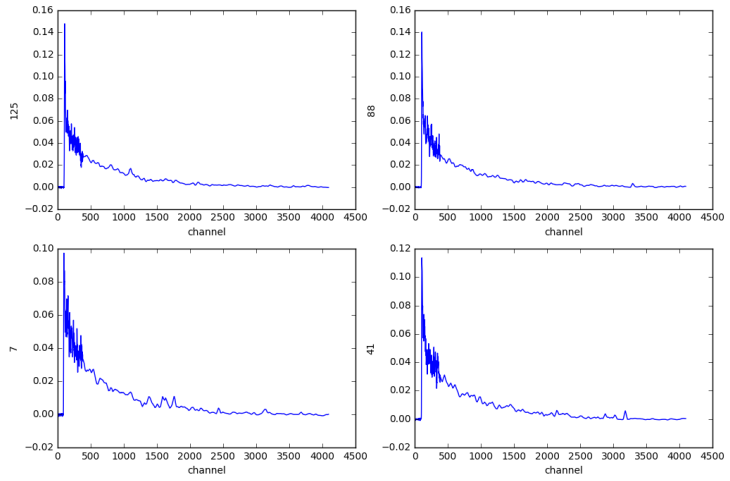


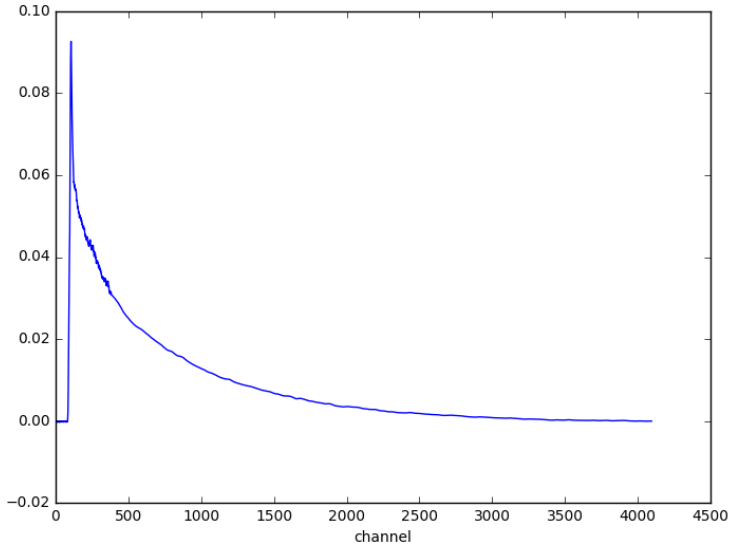


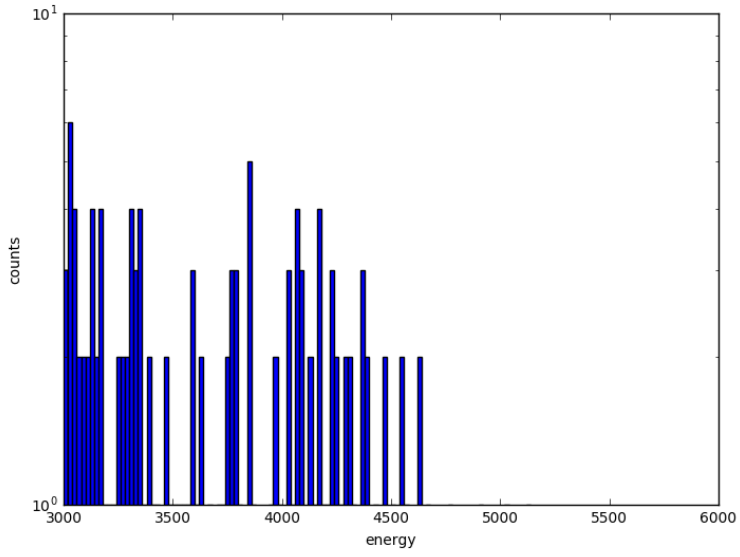


checking the 4168 events, at 8 clusters

7th cluster - 141 events

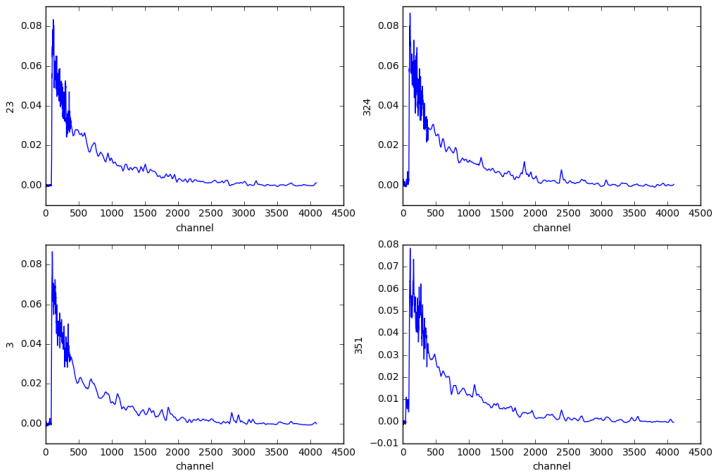


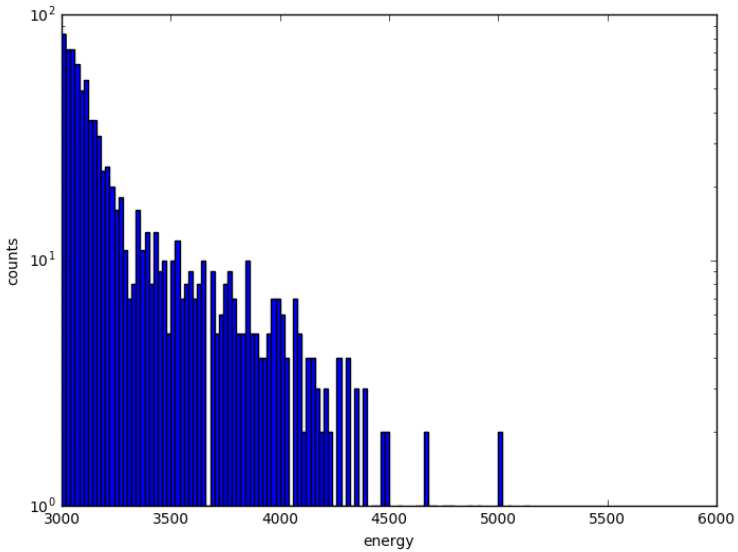


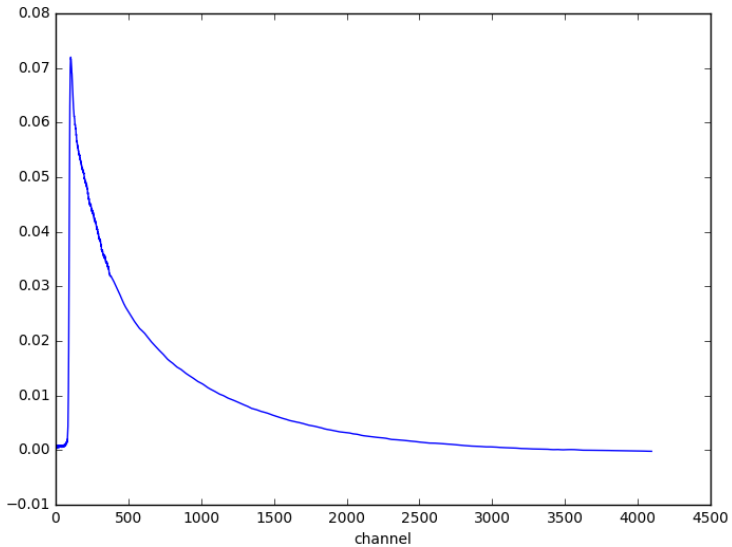


checking the 4168 events, at 8 clusters

8th cluster - 967 events

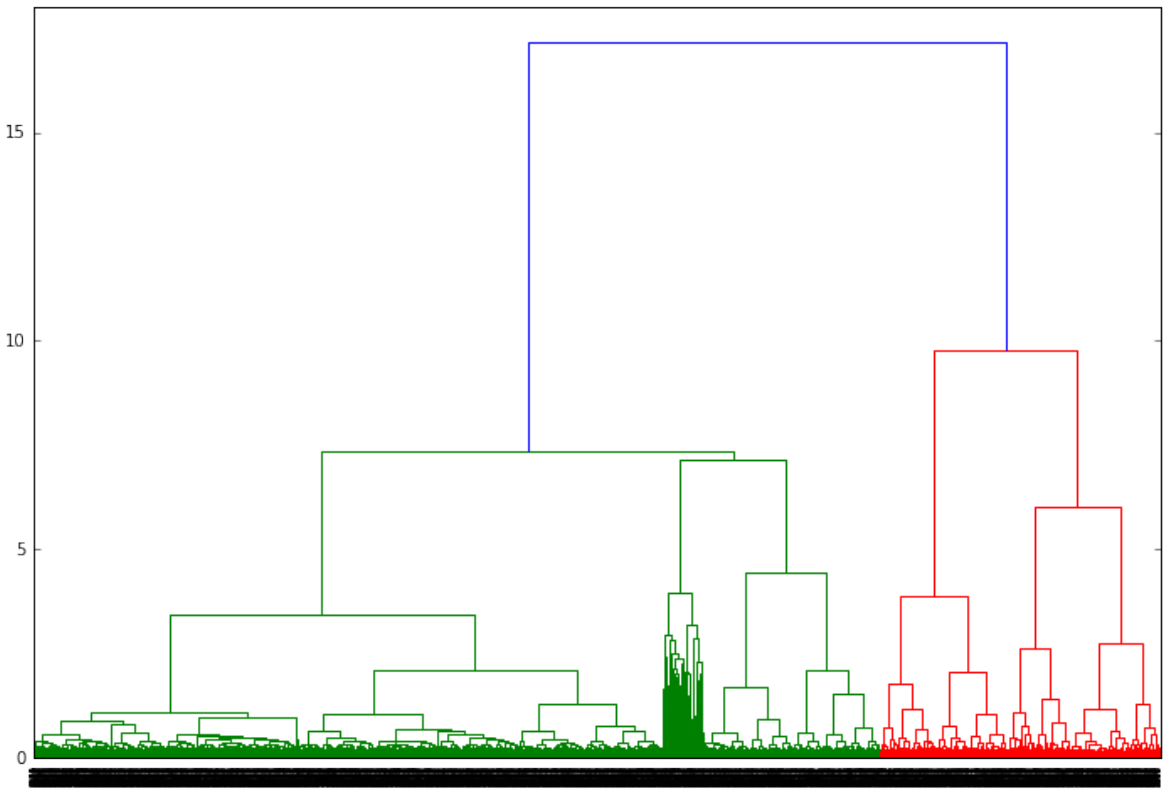






Are χ^2 good enough? χ^2 and PSD can be used as a baseline for us to compare **OR** a more relax cut condition where final sorting is done through this method.

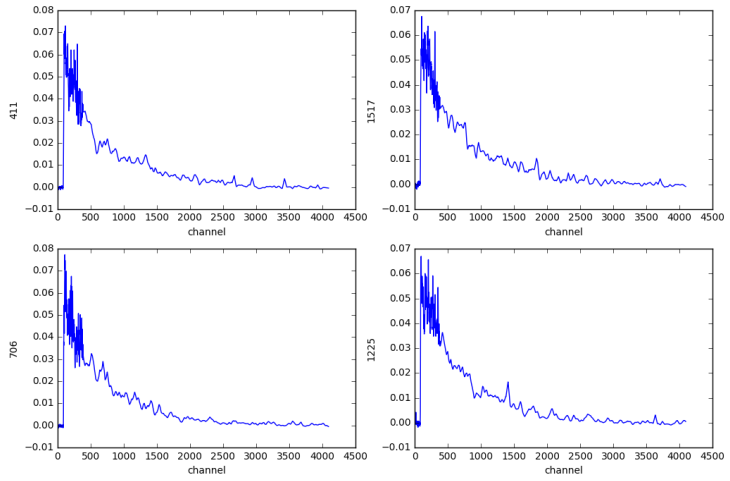
PSDPara$<$ 1.5 cut on the 65675 events, 5145 events

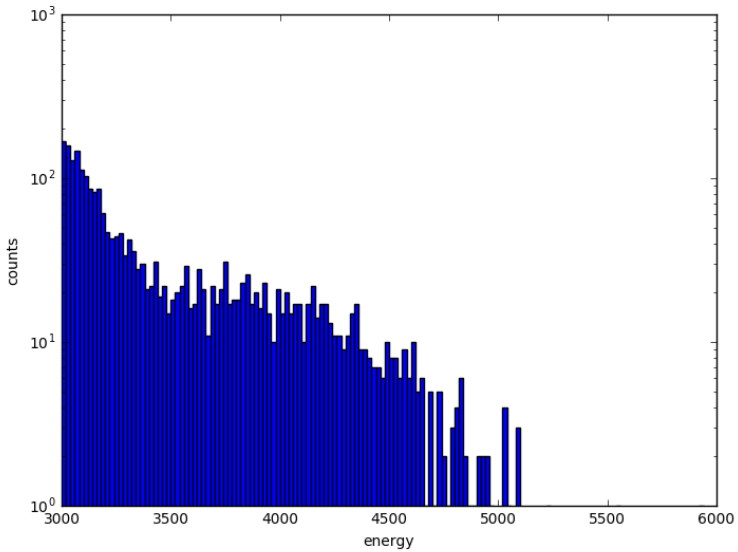


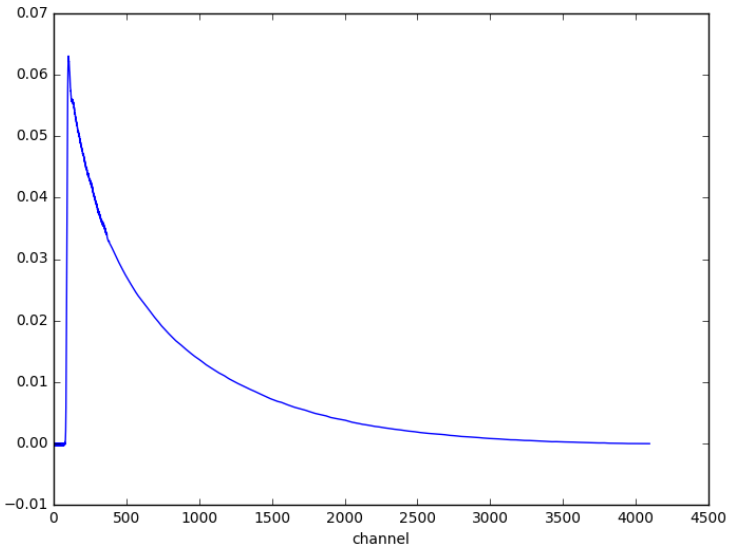
decided to take cut-off of 9 different groups. I was curious here, using the cluster label made and tag to each event and plot the energy spectrum of each cluster.

PSDPara< 1.5

1st cluster - 2552 events

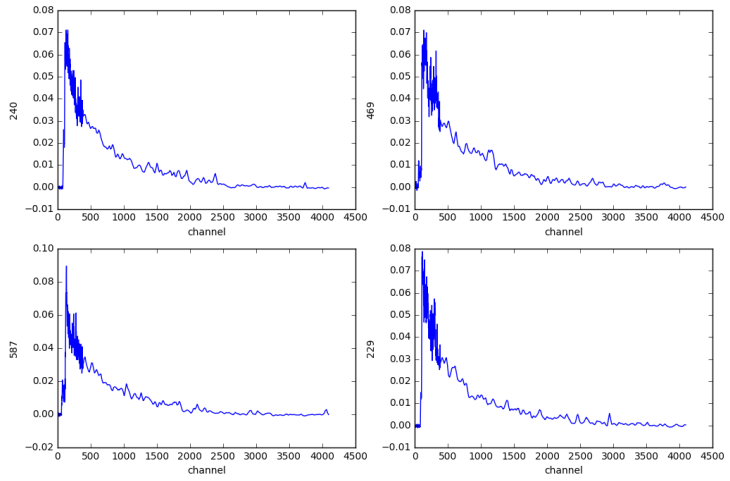


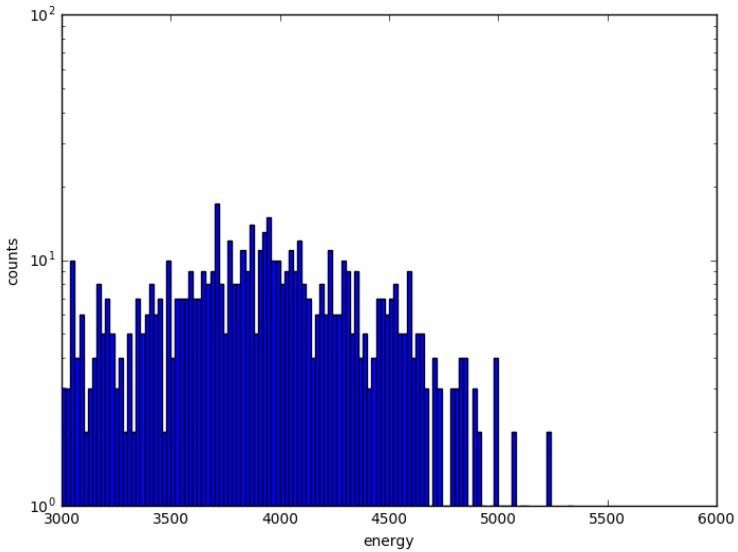


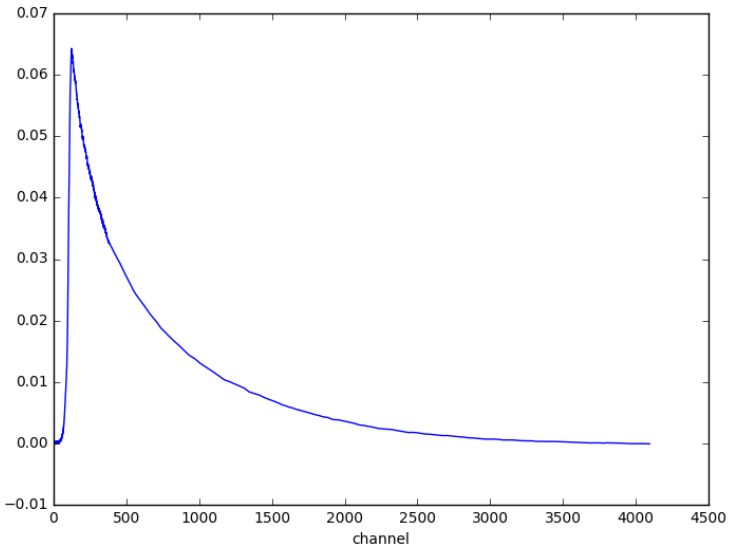


PSDPara< 1.5

2nd cluster - 640 events

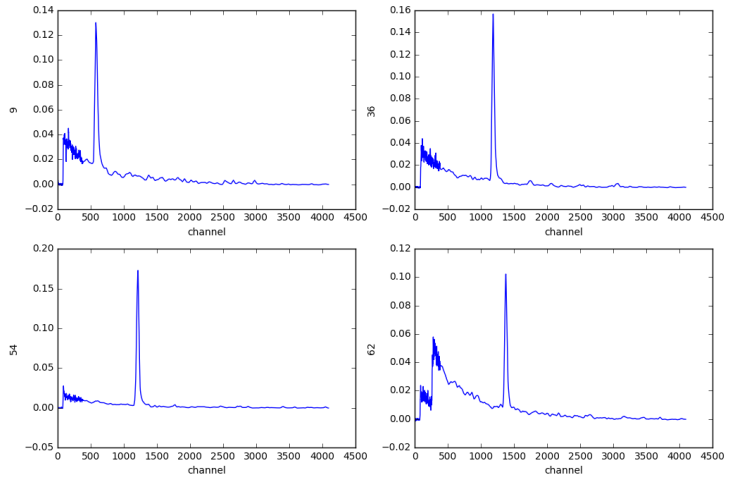


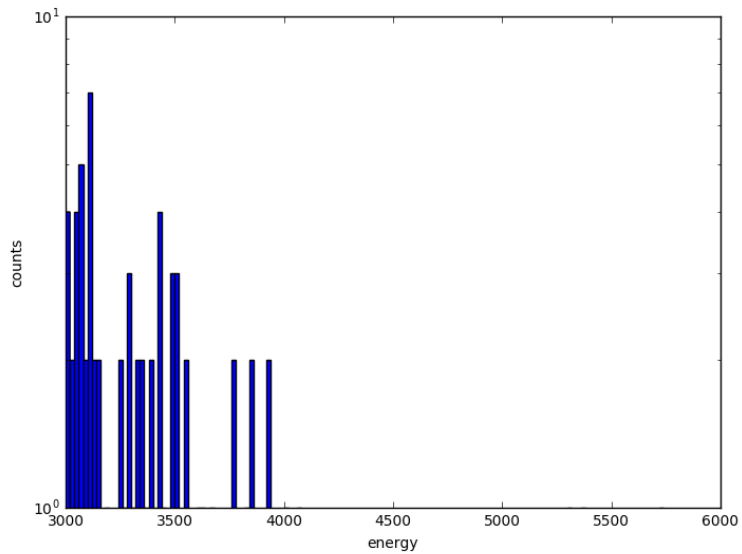


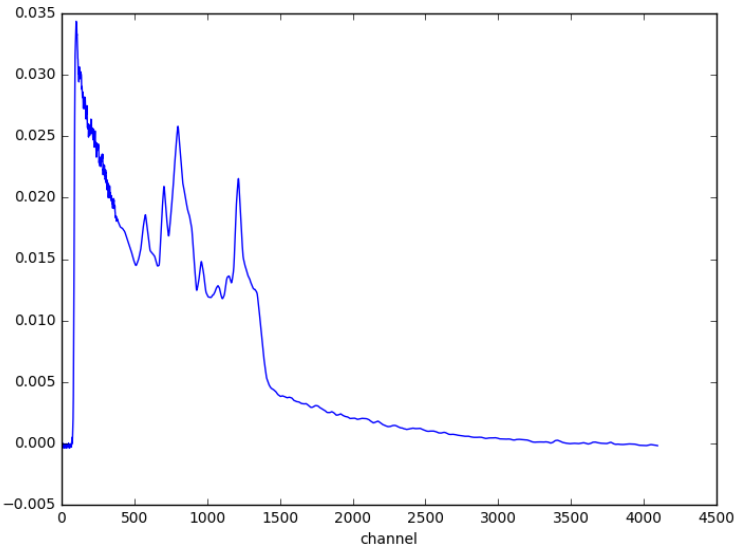


PSDPara< 1.5

3rd cluster - 71 events

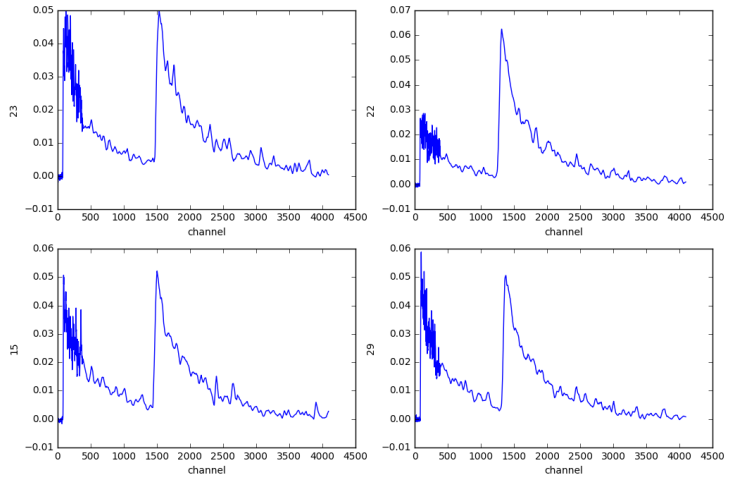


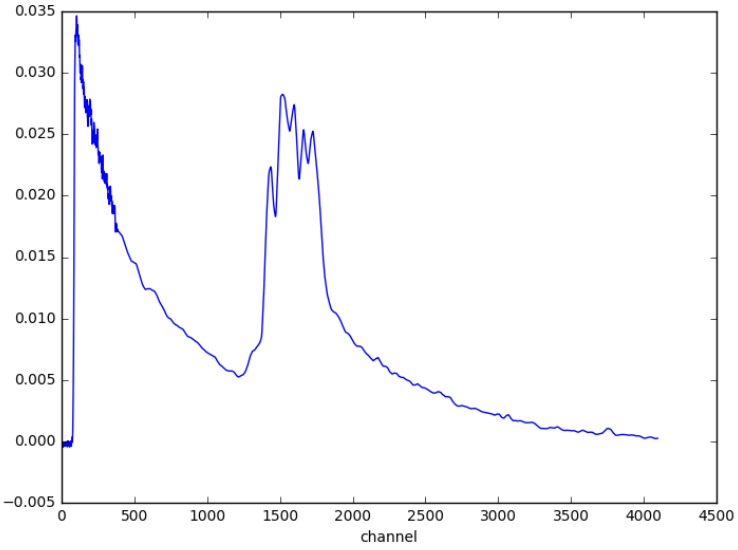


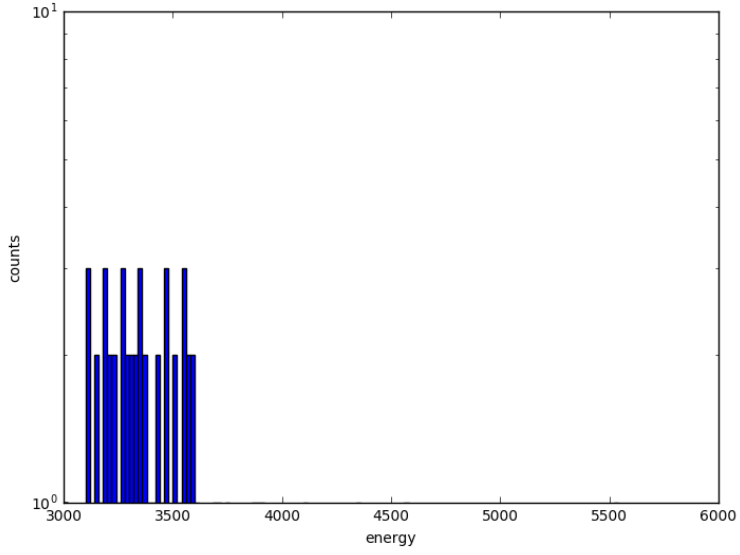


PSDPara< 1.5

4th cluster - 58 events

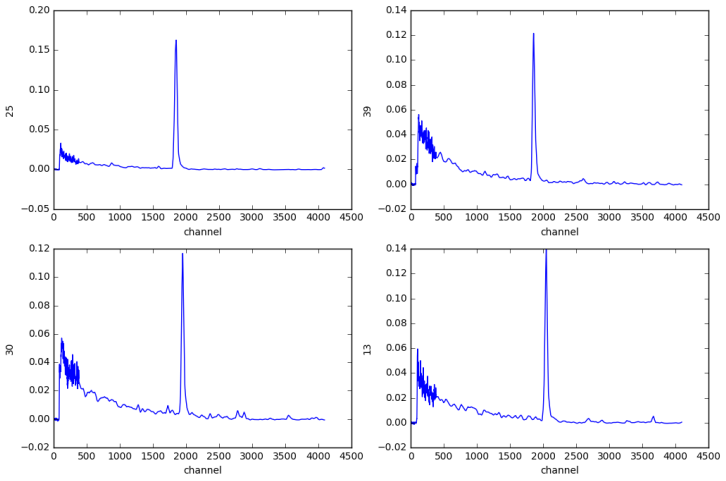


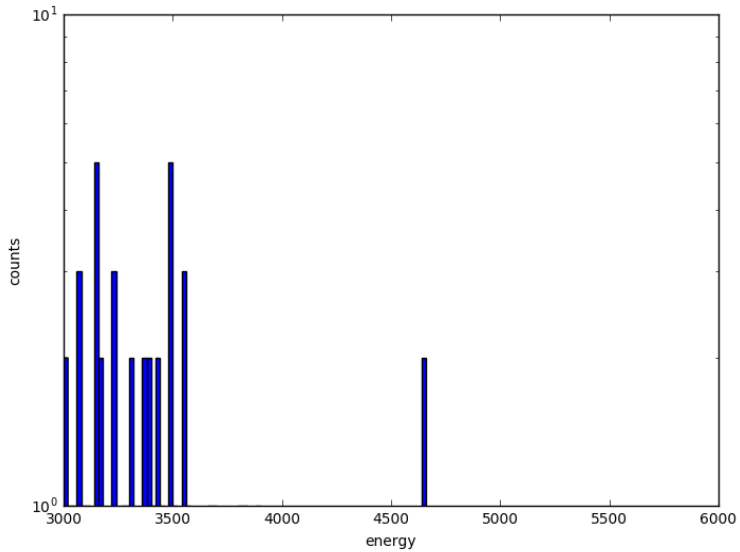


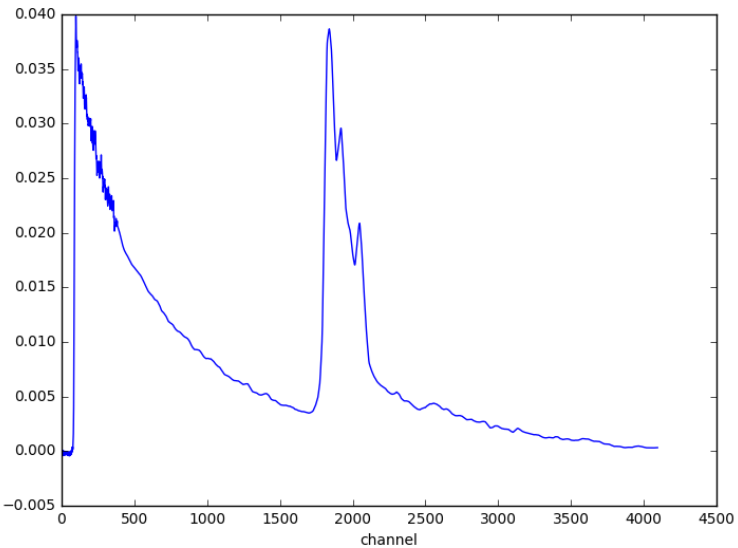


PSDPara< 1.5

5th cluster - 48 events

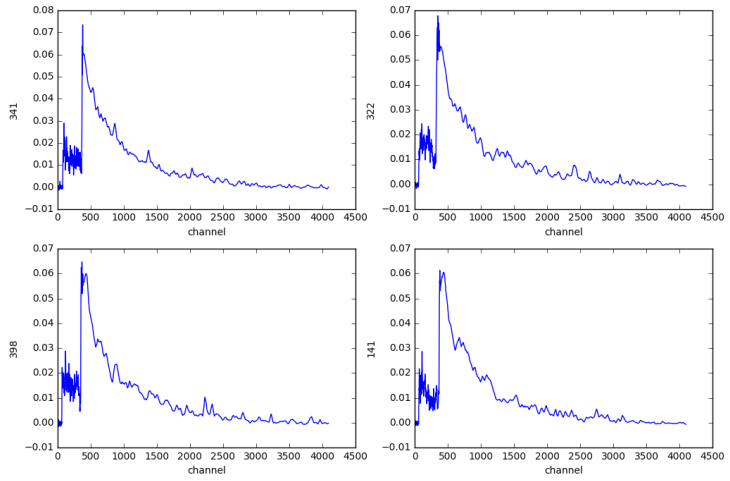


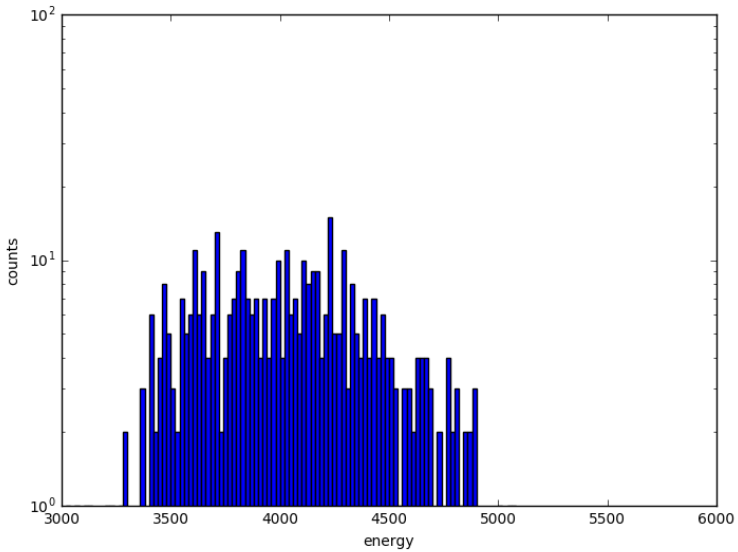


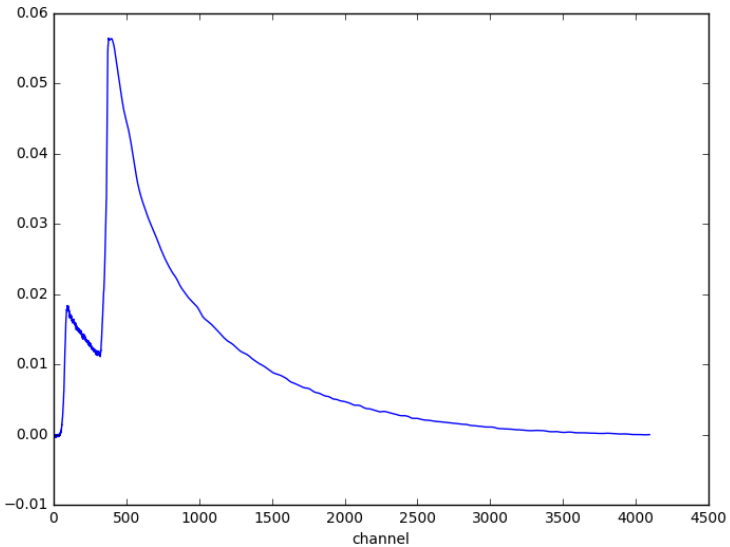


PSDPara< 1.5

6th cluster - 420 events

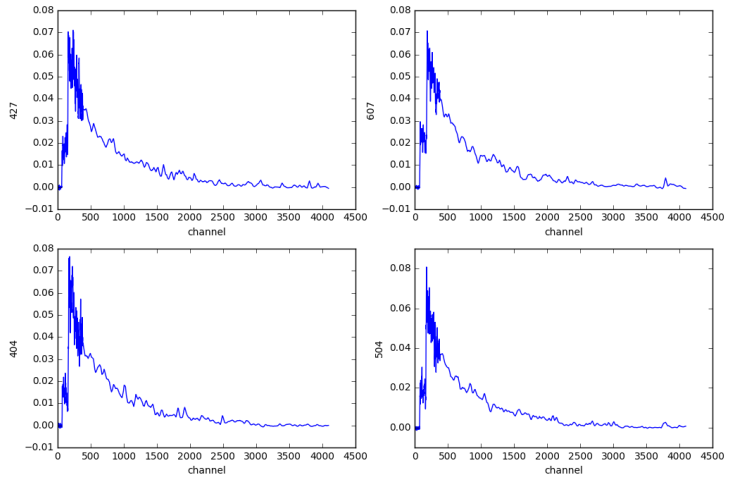


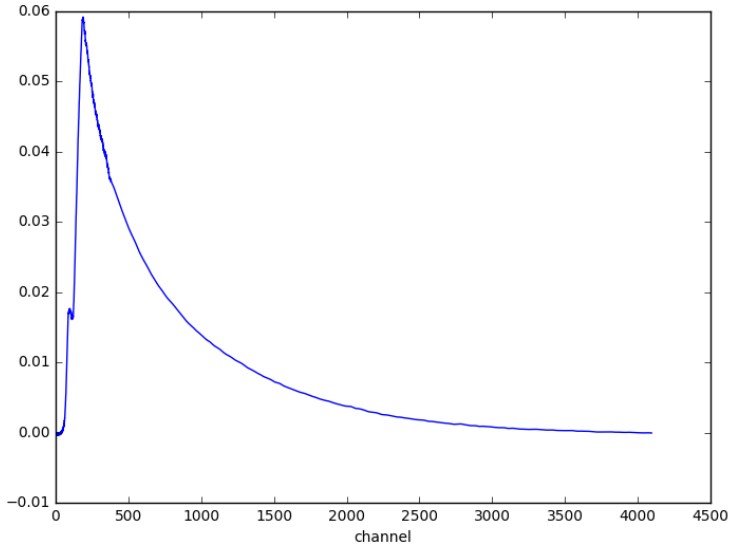


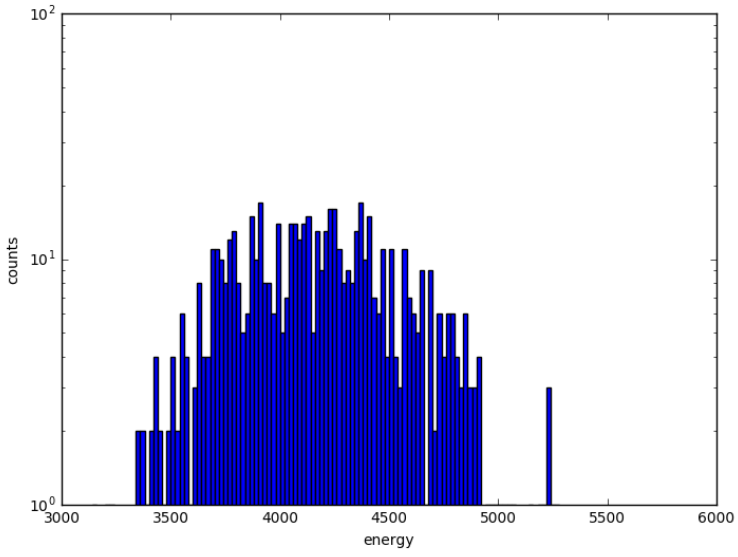


PSDPara< 1.5

7th cluster - 617 events

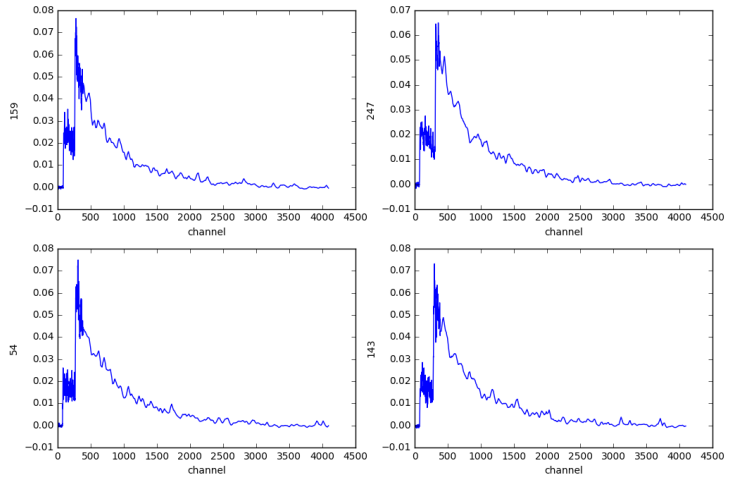


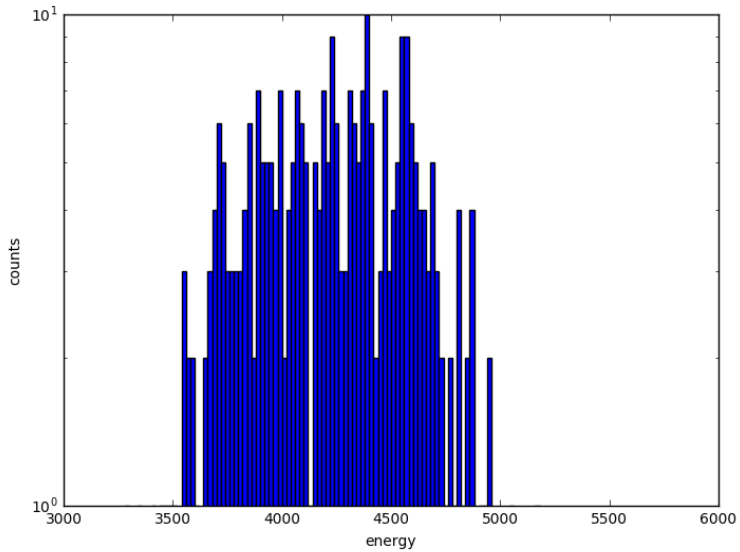


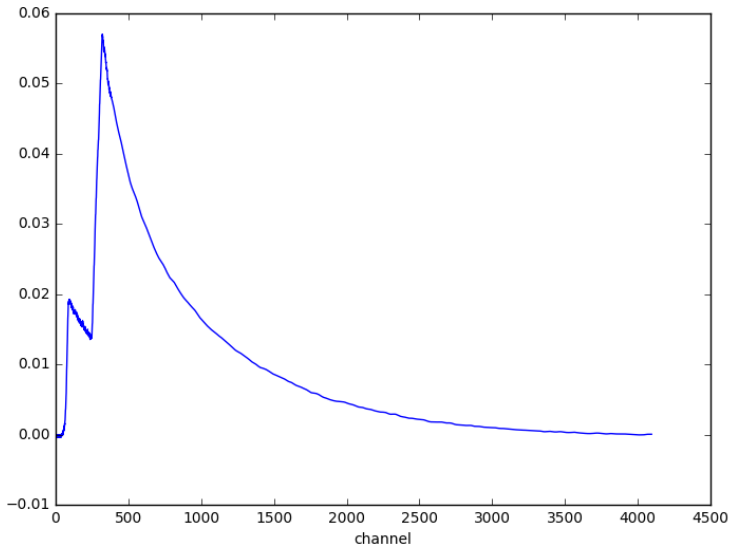


PSDPara< 1.5

8th cluster - 300 events

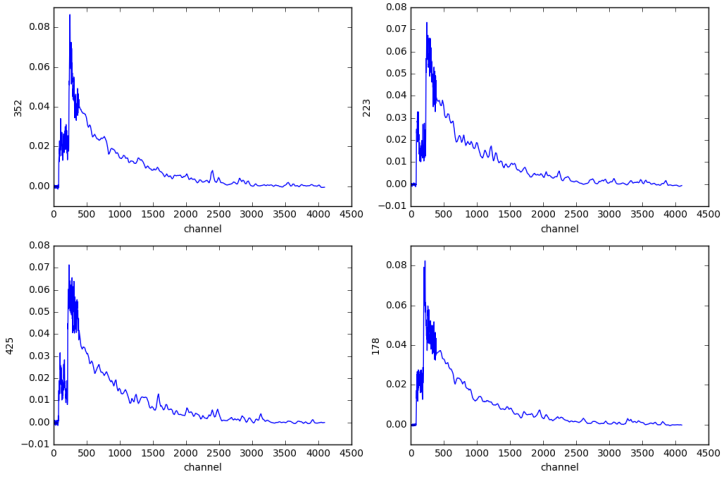


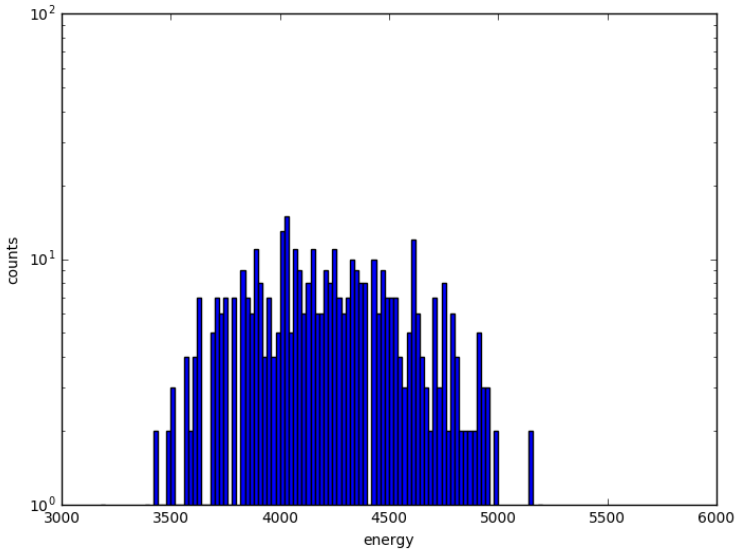


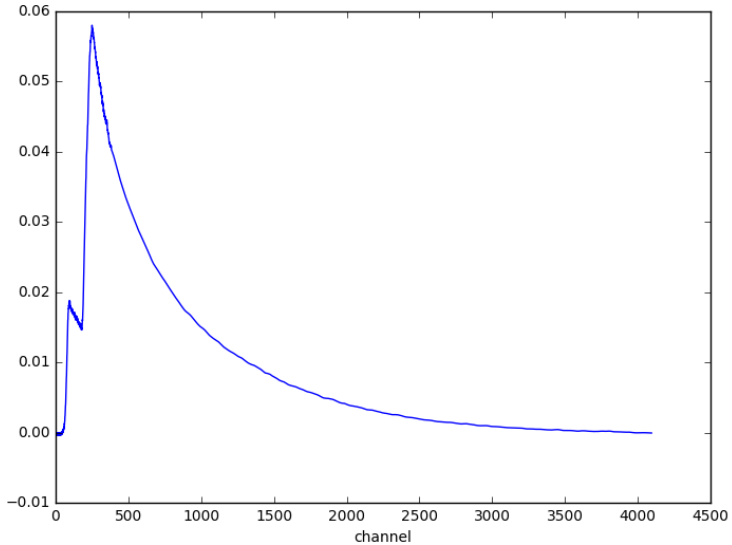


PSDPara< 1.5

9th cluster - 439 events







Pushing piled-up rejection efficiency to >99%

Rejection of pile-up events

- Pile-up events : $^{212}Bi \rightarrow ^{212}Po$ decay
- Radioactive contamination in CaF_2 : Th-chain

Th-chain

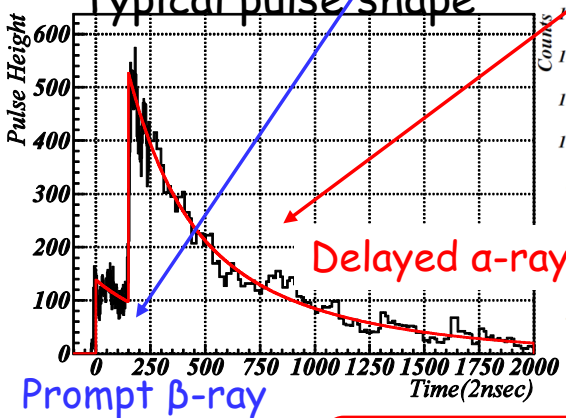
^{232}Th ($T_{1/2}$ =1.1x10^{10}year)

Pile-up ($T_{1/2}$ = 0.3μsec)

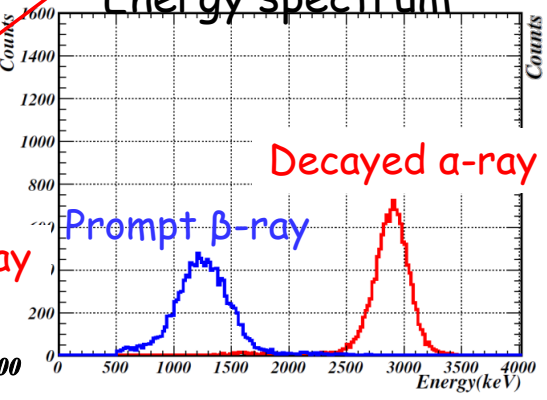
^{212}Bi (Q_β=2.2MeV) prompt —β (64%)→ ^{212}Po (Q_α=7.8MeV) decayed —α→

Maximum energy : 5.2MeV

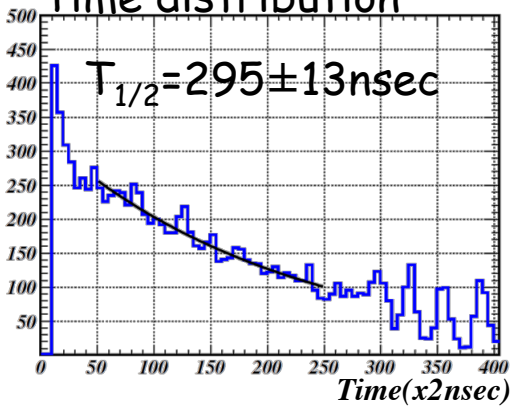
Typical pulse shape



Energy spectrum

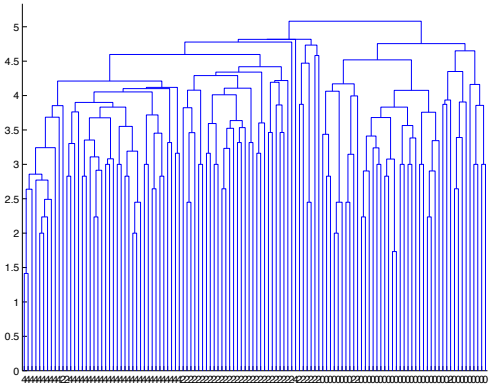


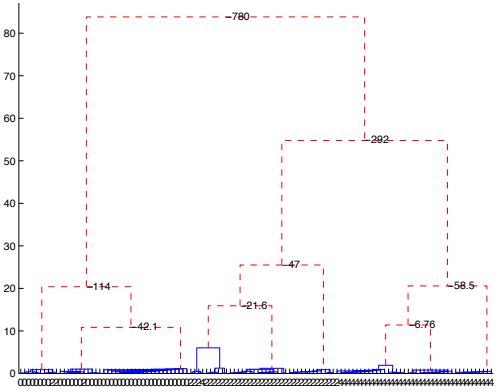
Time distribution



We can identify the pile-up events
current rejection efficiency > 95%

Beyond Encoder+Hierarchical clustering

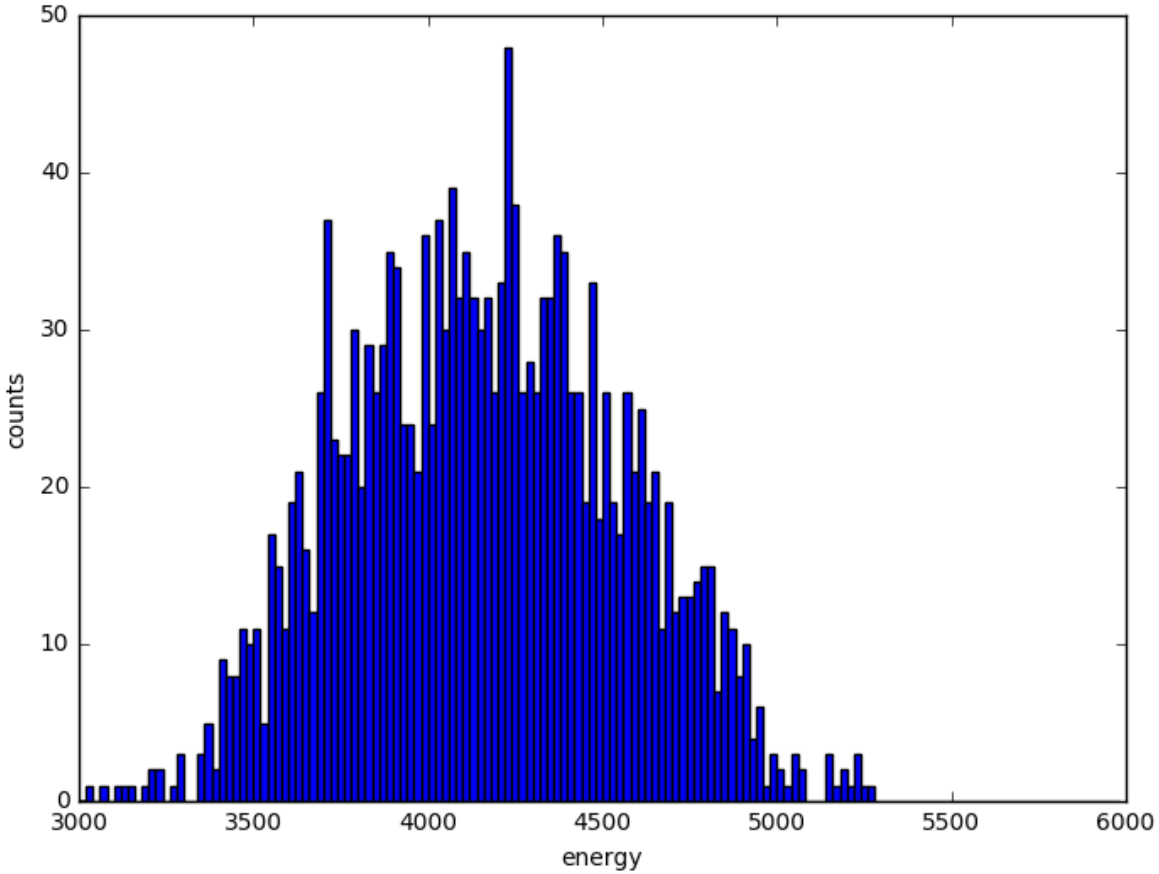




- Bayesian Hierarchical Clustering - more efficient clustering method?
- Hierarchical clustering in larger dataset(ie:- doing $>\sim$5000 sample at once) is not practical - the calculation procedure of the "matrix" get ridiculously huge.
- My personal agenda for bolometer, feeding 2 different photon and phonon waveform(Analogous to 3 colour channels for image classification) simultaneously for efficient clustering of physics events.
- rather than average plot, using heatmap like pulse features to visualise it.

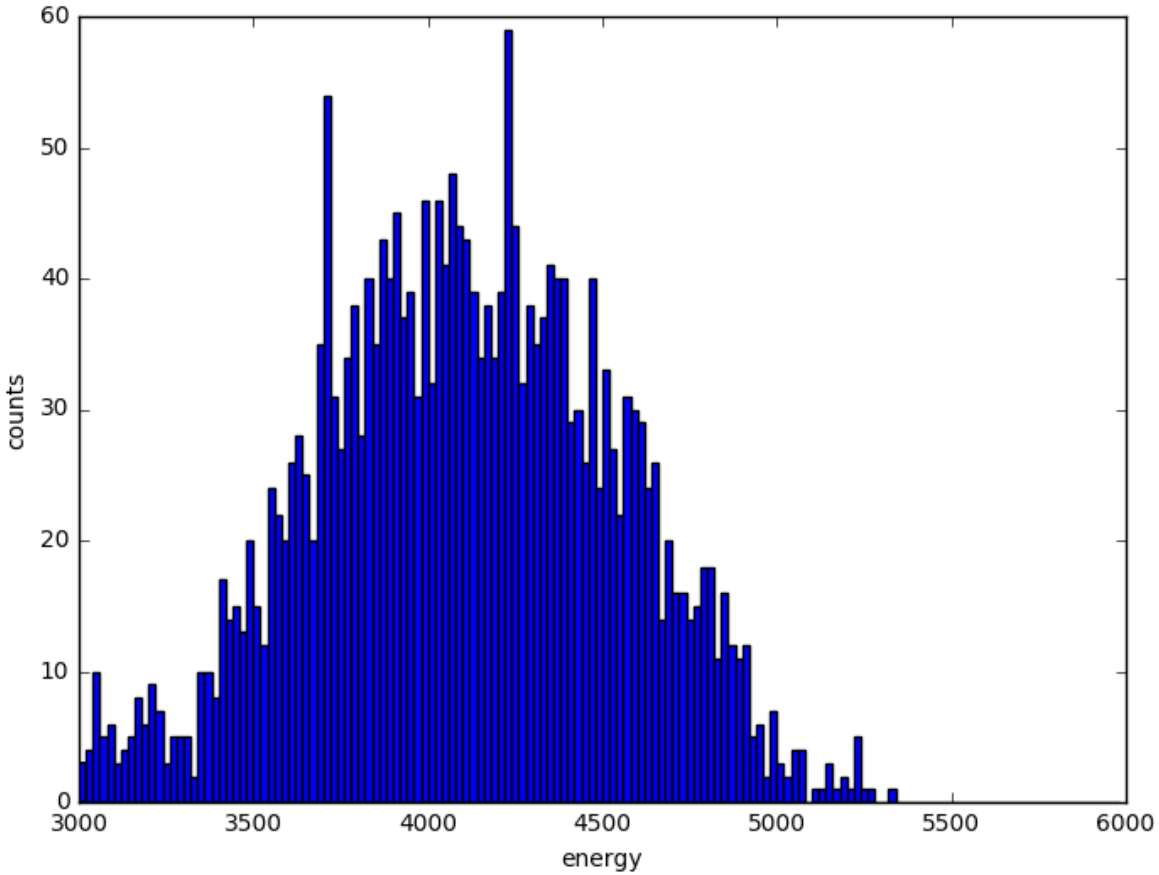
Extra

dataset psdpara$<$ 1.5, energy spectrum of cluster 6,7,8,9 added together, double pulse background



1776 events

+ cluster 2



2416 events