- Disclaimer:- I am very much aware that so far it is just an alternative means. I am not expecting it to surpasses any current methodology but overall it is good opportunity to use “Big Data” methods in physics experiments (expect these in future experiments.)

Machine Learning in CANDLES?

- How applicable is ML with our data process flow. Event classification, event reconstruction, etc.

- Why CNN -> a good “feature extractor” we can exploit this trait as an alternative event classifier.

- Scalability, it is much easier to transfer the technique/ software from small to larger experiment. An alternative way to use simulated data to compare with actual data(train a neural network with simulated data and test it on real data, in some way able to verify the physics we are testing and fill in the missing gap, a difficult task using traditional method).

why CNN(Convolution Neural Network)?

- A good feature extractor. Very good at classifying images as demonstrated in Vision recognition field. (ImageNet 2012 challenge).

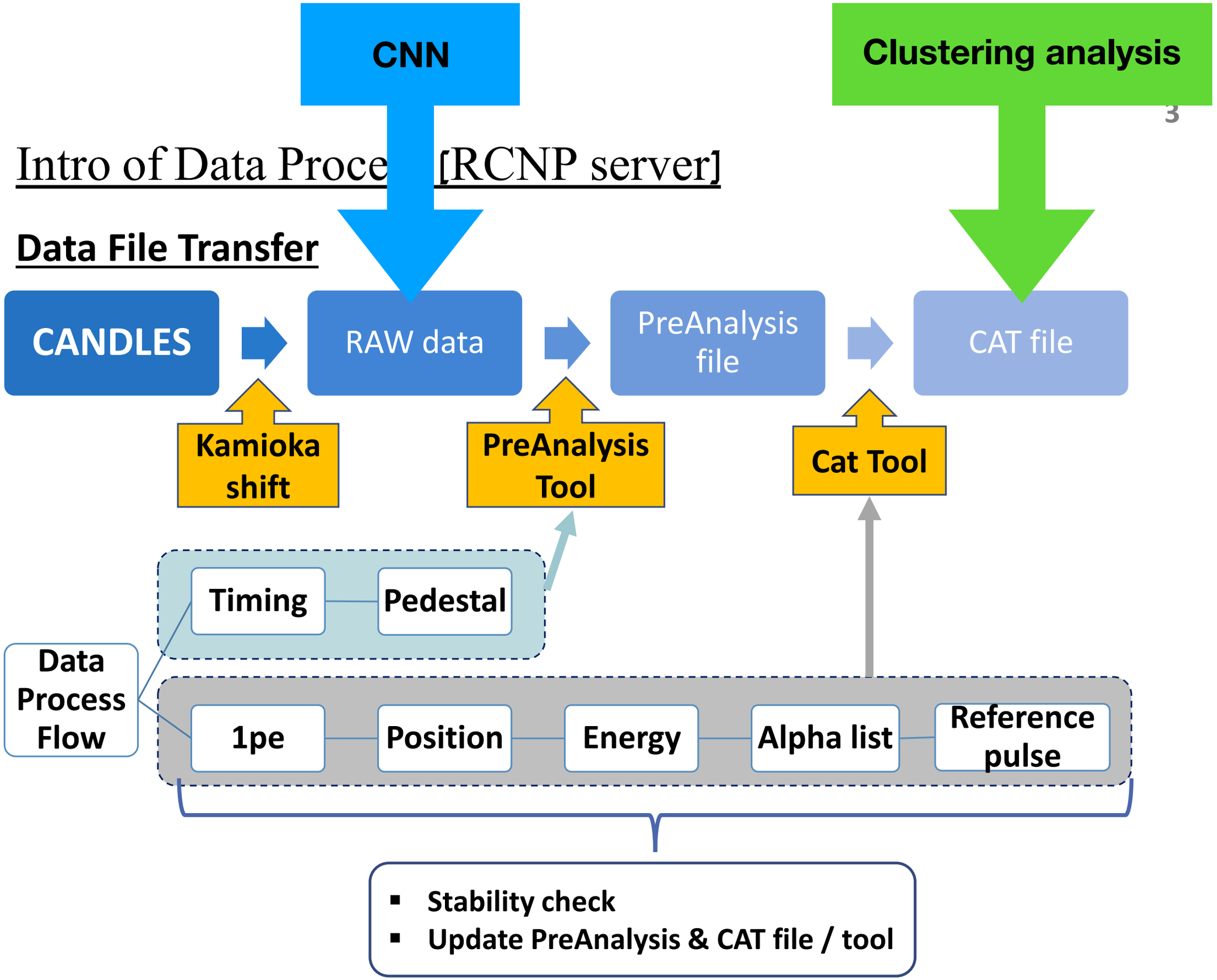
- generalised the data feature needed to CNN.

- Detecting subtle features from data otherwise overlooked.

- HOW about other network type? possible, testing some recurrent type network(I doing this on my 6 years old laptop, bottleneck)

Intro of Data Proce [RCNP server]

Data File Transfer



Chan's slides 2017/08/6

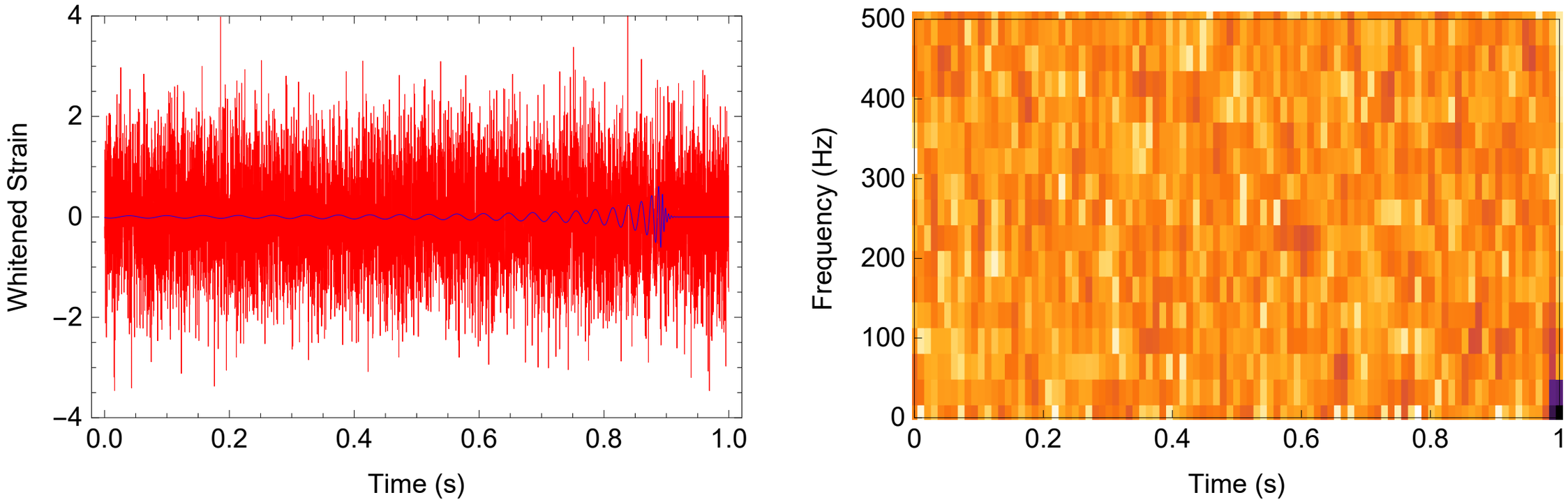


FIG. 2. **Sample of input data**. The red time-series is an example of the input to our DNN algorithm. It contains a BBH GW signal (blue) which was whitened with aLIGO's design sensitivity and superimposed in noisy data with SNR = 7.5 (peak power of this signal is 0.36 times the power of background noise). The component masses of the merging BHs are $57M_\odot$ and $33M_\odot$. The corresponding spectrogram on the right shows that the GW signal on the left is not visible, and thus cannot be detected by an algorithm trained for image recognition. Nevertheless, our DNN detects the presence of this signal directly from the (red) time-series input with over 99% sensitivity and reconstructs the source's parameters with a mean relative error of about 10%.

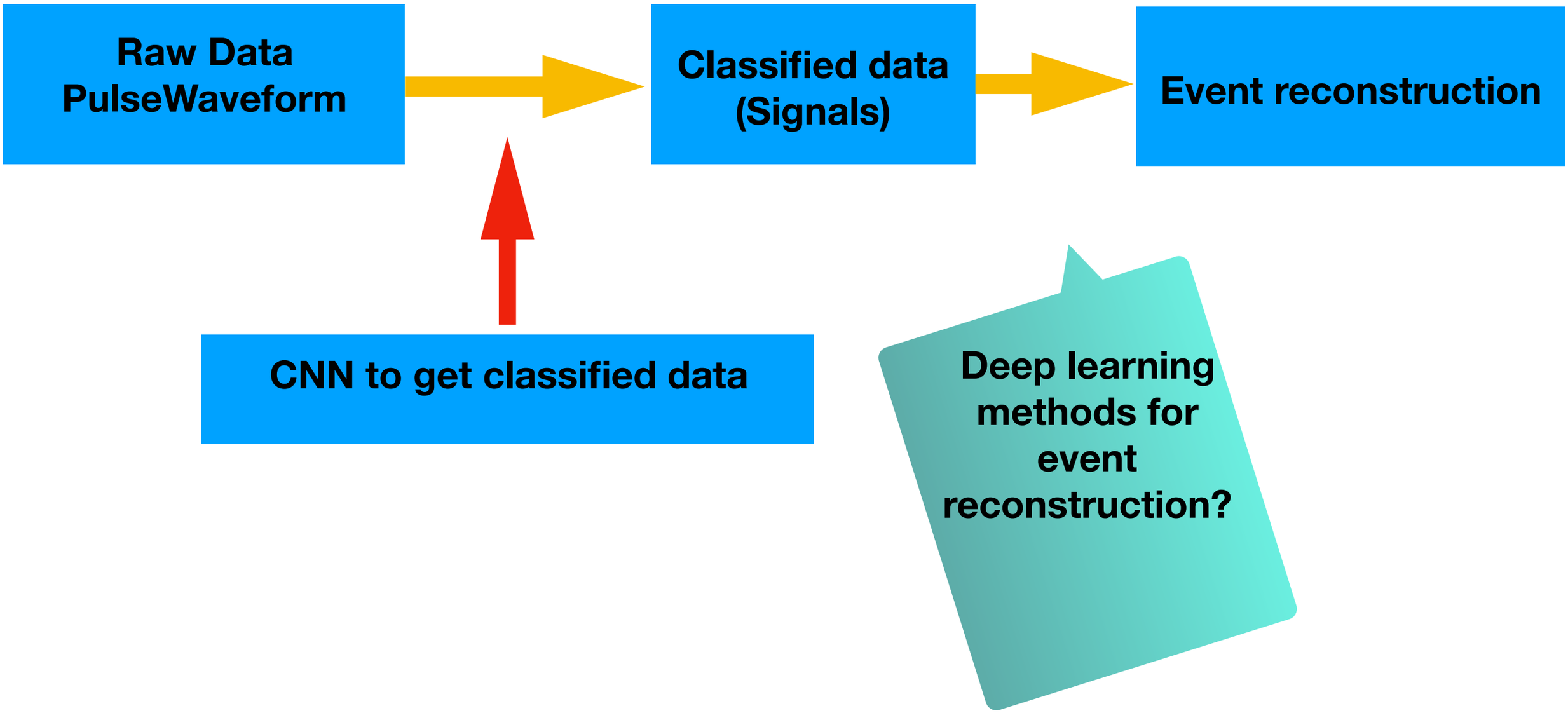
The plot on the right is easily recognisable as signal but not the left plot. However CNN is able to recognise the left plot as signal.(from raw input of the time series data on the left)

This is just one of many example of using CNN for event classification

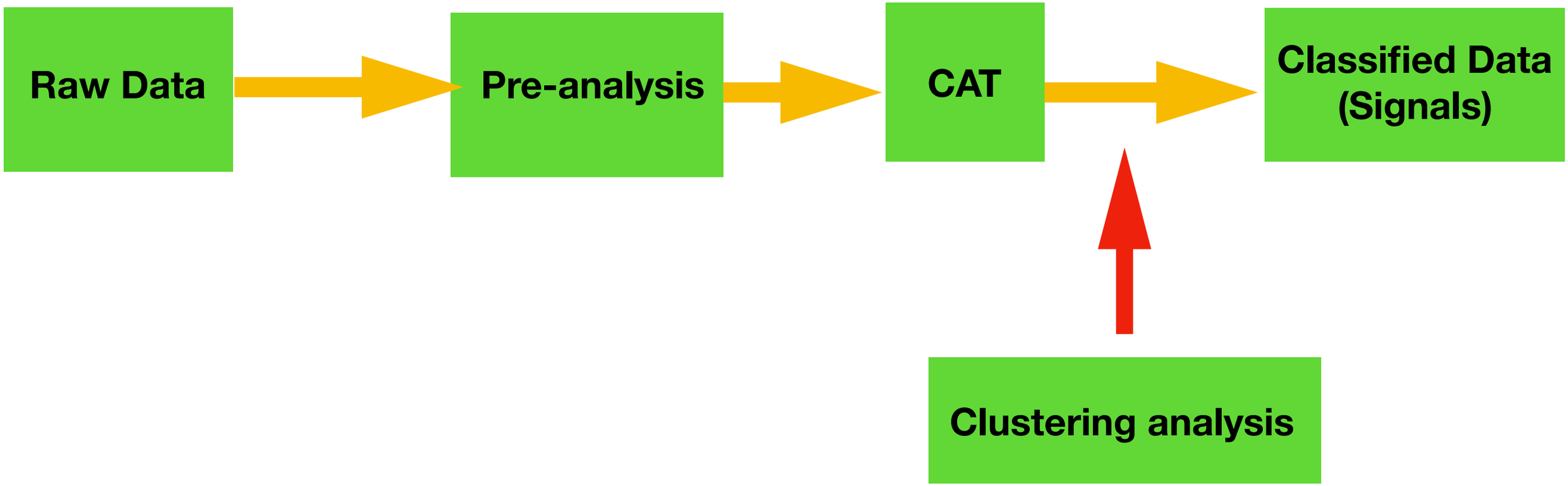
There is two approach we can consider using ML for classification of “Signal” and “background” in Physics experiments

- Low-level raw data -> Near minimal interpretation of the raw data received. Attempt to classify events directly from here. An alternative possibly complementary method with the usual classification methods.
    - ->Deep learning methods

- Reconstructed parameters -> Parameters obtained after event reconstruction, ie:- interpreted raw data that are more understandable physical quantity to work with. reduce the “information” into simpler numbers.
    - ->Unsupervised learning, clustering analysis



Retain all information of the data right up to before event reconstruction

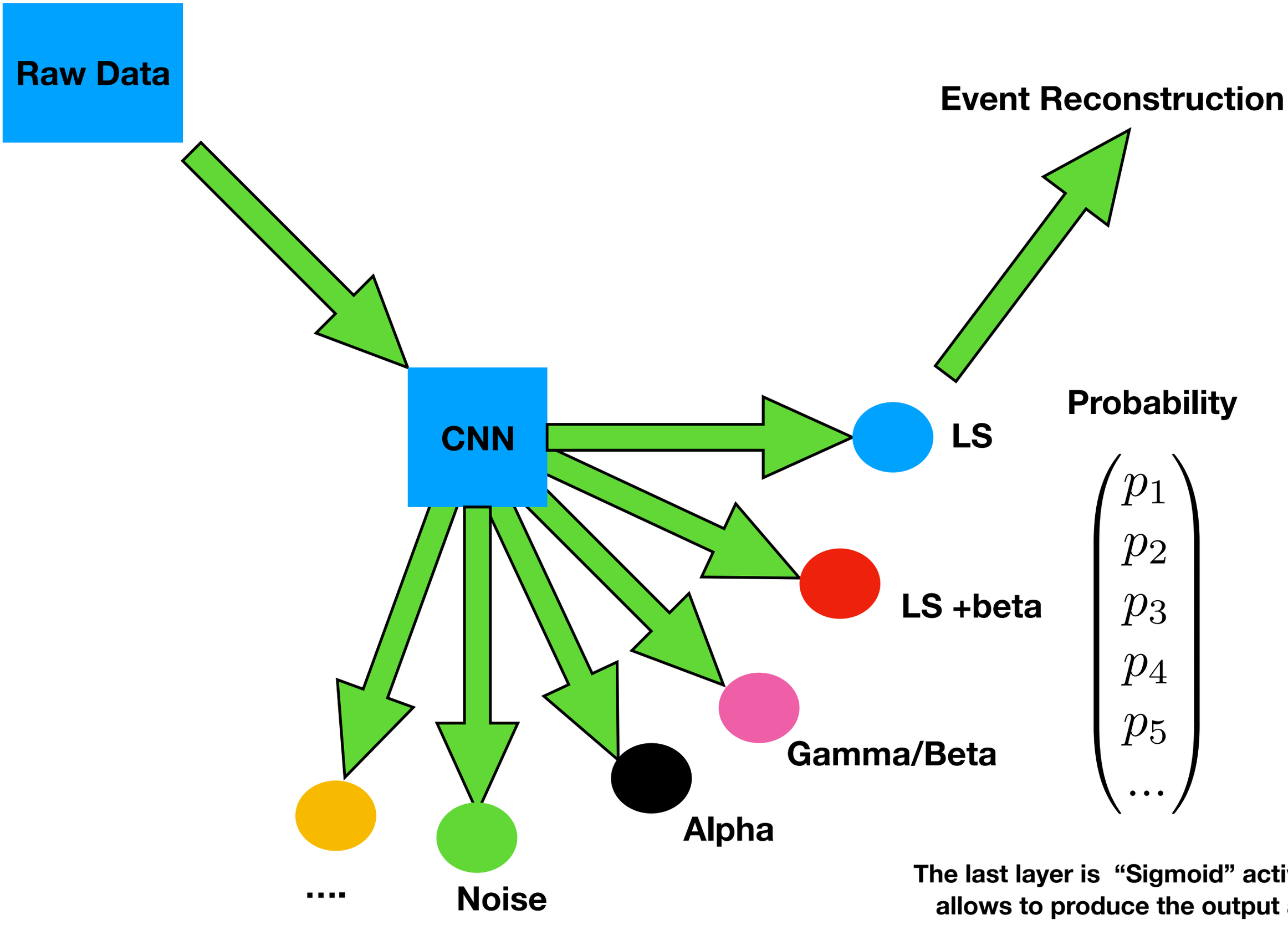


Using reconstructed parameters to “cluster” the data.

1st approach

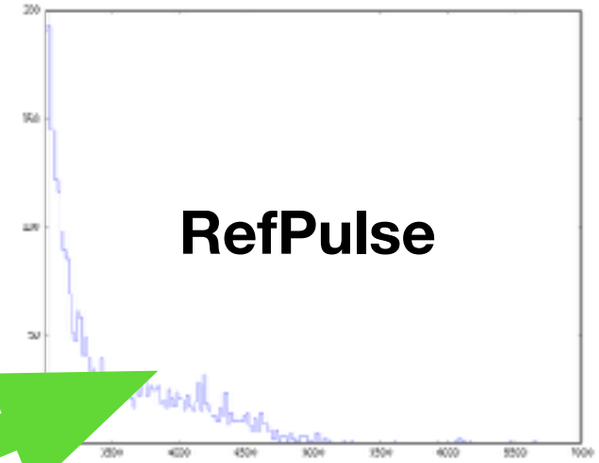
- Using deep learning methods. In particular Convolution Neural network due to their ability to recognise patterns from data.

- This approach requires us before hand to learn the features of the classes we are trying to classify. “signal” and “background”.

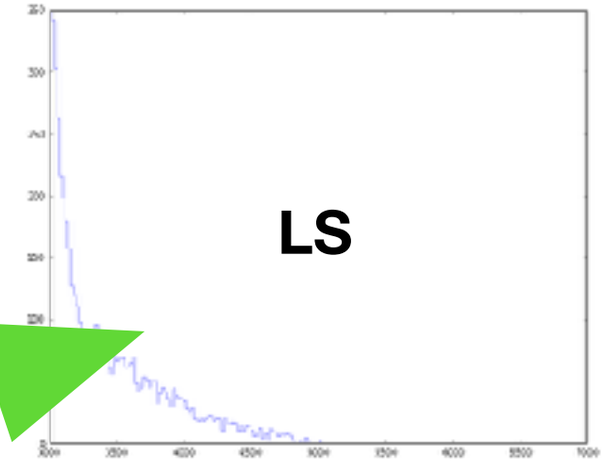


The last layer is “Sigmoid” activated, allows to produce the output as a probability

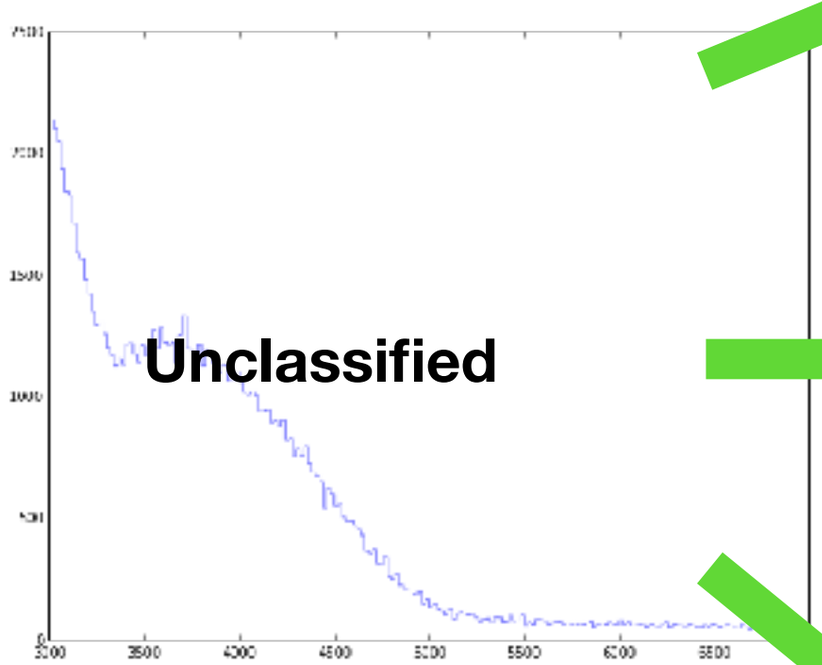
For each entry, we take the highest probability and separate them



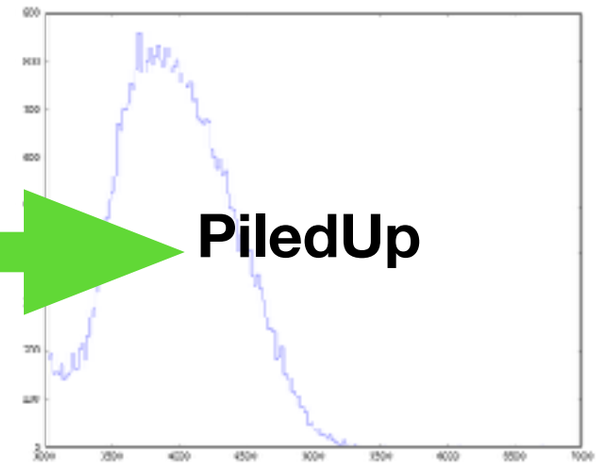
RefPulse



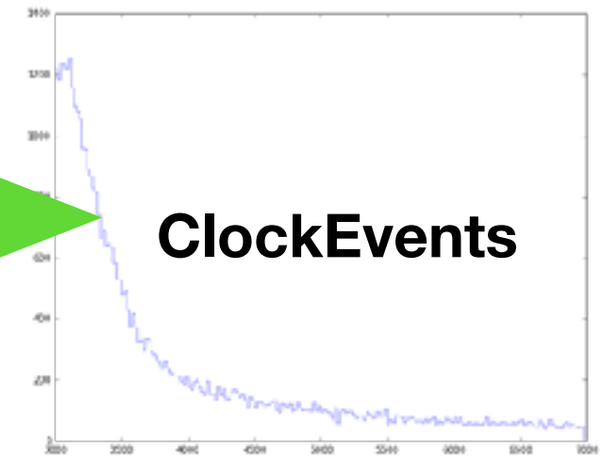
LS



Unclassified

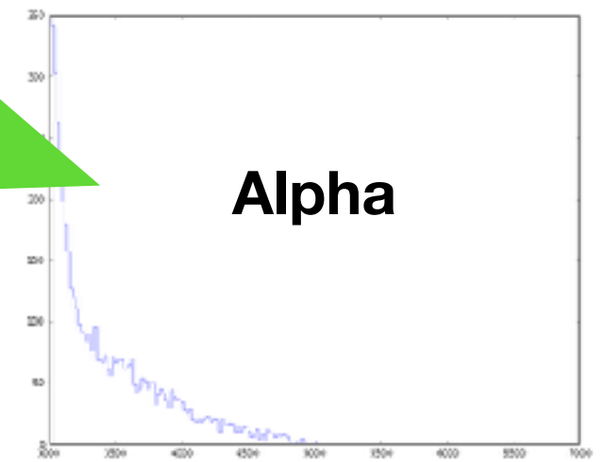


PiledUp



ClockEvents

LS+Beta



Alpha

There are cases where the 2nd highest probability is close to higher.

2nd approach

- Multivariate analysis, clustering analysis approach. (Leaning more on unsupervised techniques)

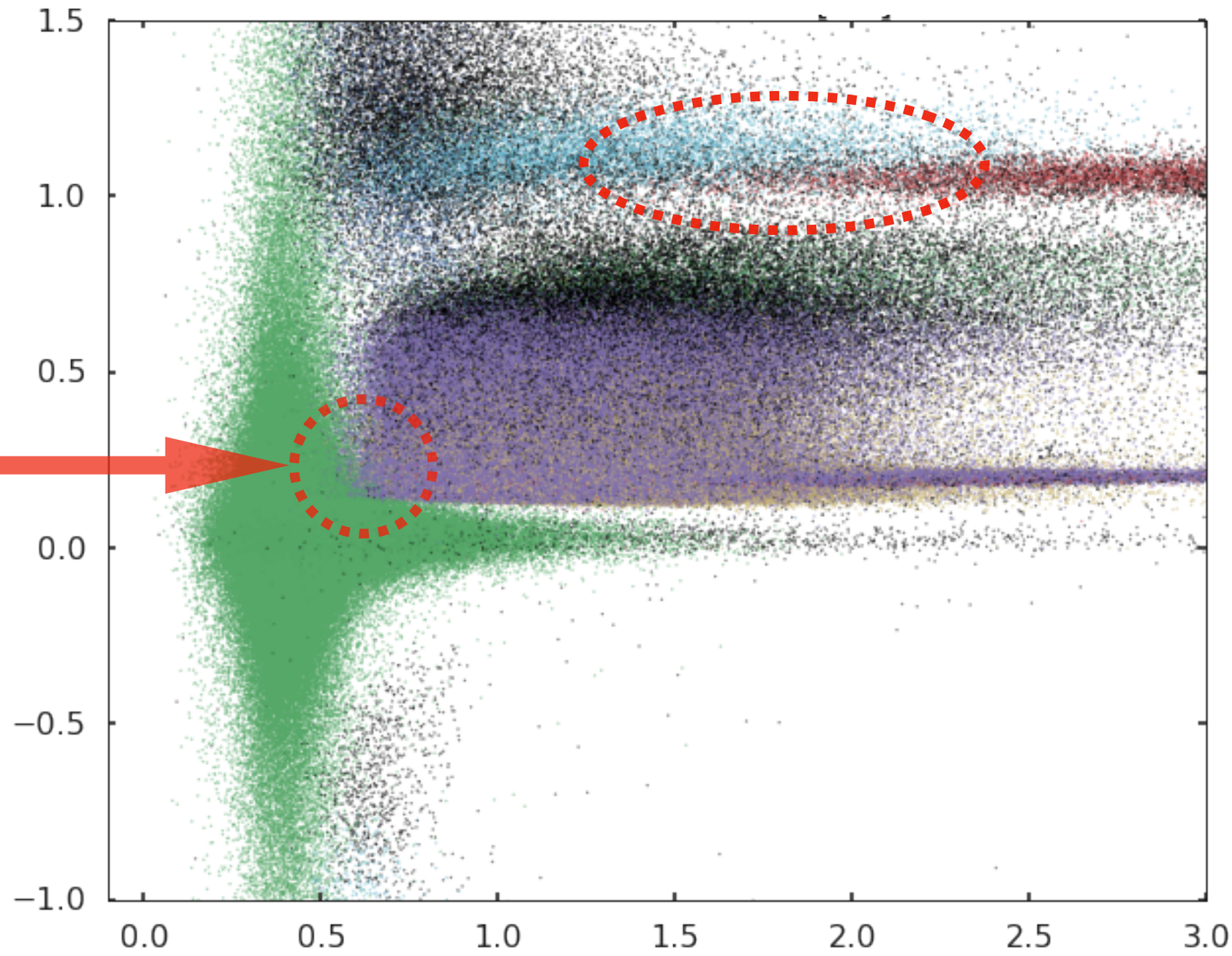
- A multidimensional analysis approach to cluster the data based on the relative “distance” between parameters.

- group data into group that have similar “distance”.

- From there we can individually analyse the group of data.

Something like this.....

In this region, is difficult to give a clear cut condition to separate them



Each colours represents a cluster of similar data. Each group are colour labelled.

Ratio4us against chi-squared of Beta(Ref) plot

How about using features extracted using CNN and do a clustering analysis on it? Looking into it.

The “Cut-based” method

- You define a single cut condition based on the comparison between two parameters. A boundary

- To further improve the selection, additional cut is added with other new parameters. However doing so you lose “information” between the previous 2 parameters. Sometime giving a much stringent cut, thus losing some potential useful data.

    - It is something I noticed when I tried to do separate event based on their shape. a stringent cut, Lose some event just because chi-squared is not favoured.

- Difficulties in doing it properly when consider more and more parameters.

- Clustering analysis, you simultaneous compare all parameters allowing a more cleaner cut.

Clustering Analysis

- There are many algorithm to choose from with each has it pros and cons.
- Conceptually is a bit difficult to implement and understand.
- A great deal of understanding the data itself is important.

Exploratory studies

- Just to demonstrate ML techniques can be implemented into double decay experiments (So far DUNE/CMS are actively implement ML as part of their analysis flow due to complexity for event selections, ML makes things easier/faster to analyse.) Jet physics studies in CMS/2D-3D based image selection.

- Event selection based on reconstructed parameters using ML is actually widely tested in physics itself, giving comparable results with “traditional” methods. Nothing ground-breaking improvement is reported yet so far.

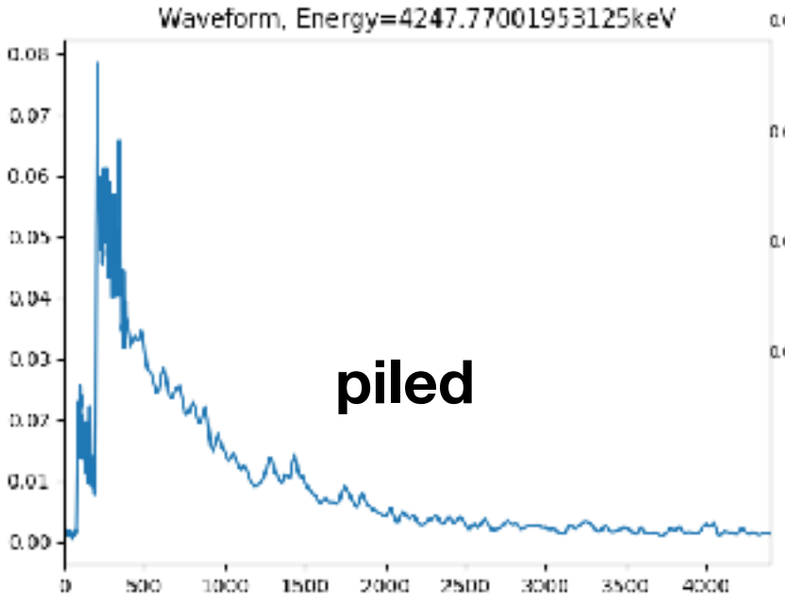
- A simplified implementation and execution over “traditional method”.

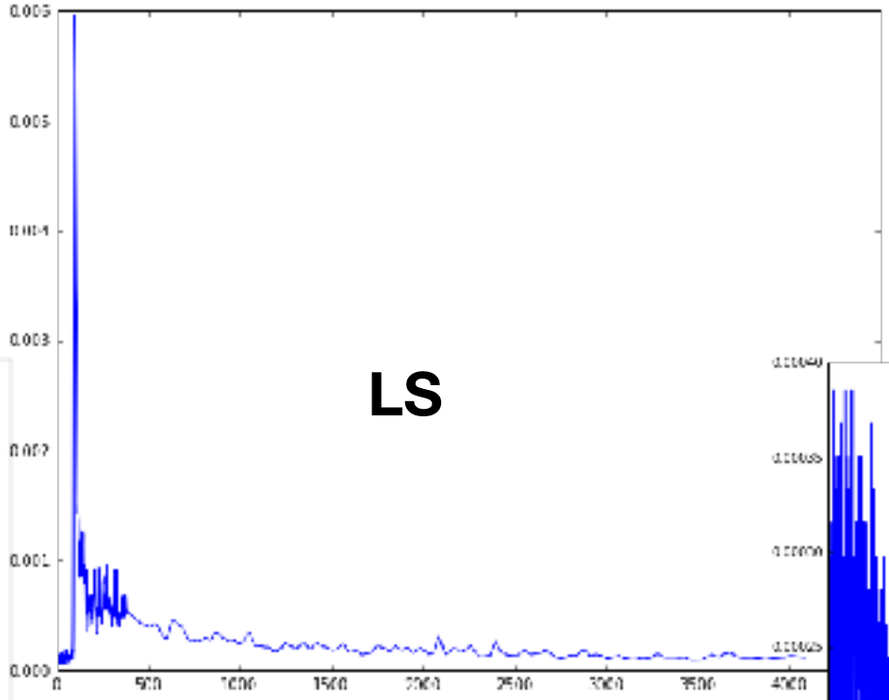
- Using simulated data as standard rather from actual data. A comparison between the perfect information of physics that is available from simulation compared to data.

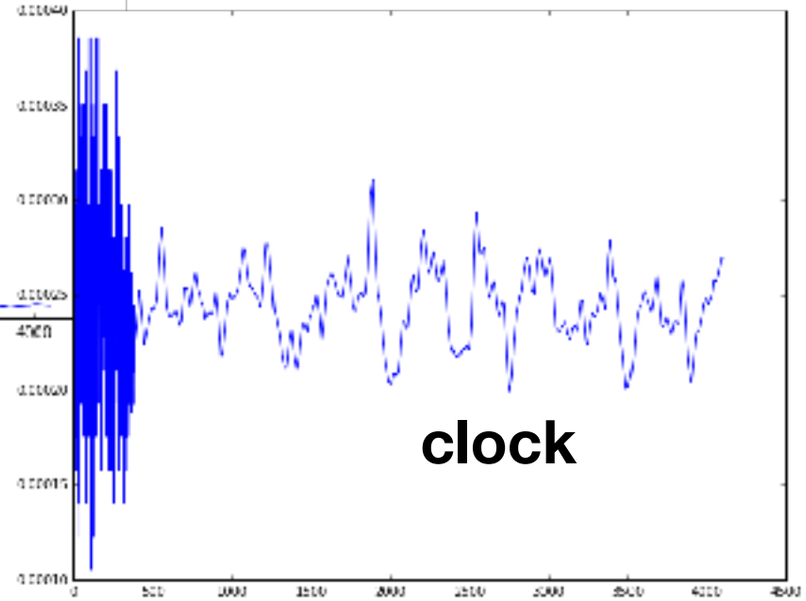
- Difficult but doable.

- Using information obtained through ML, to simulate the physics. ie:- Simulation from experimental data+MC simulation

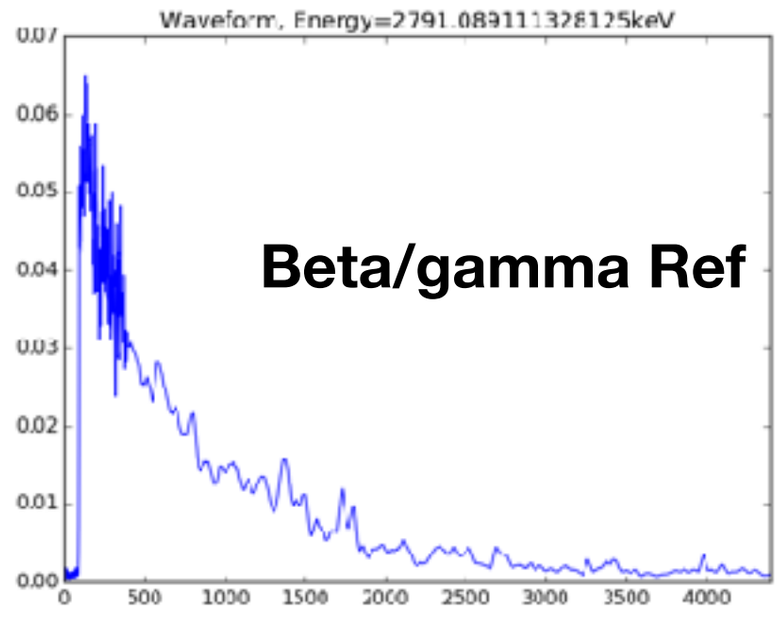
Some preliminary observation/results

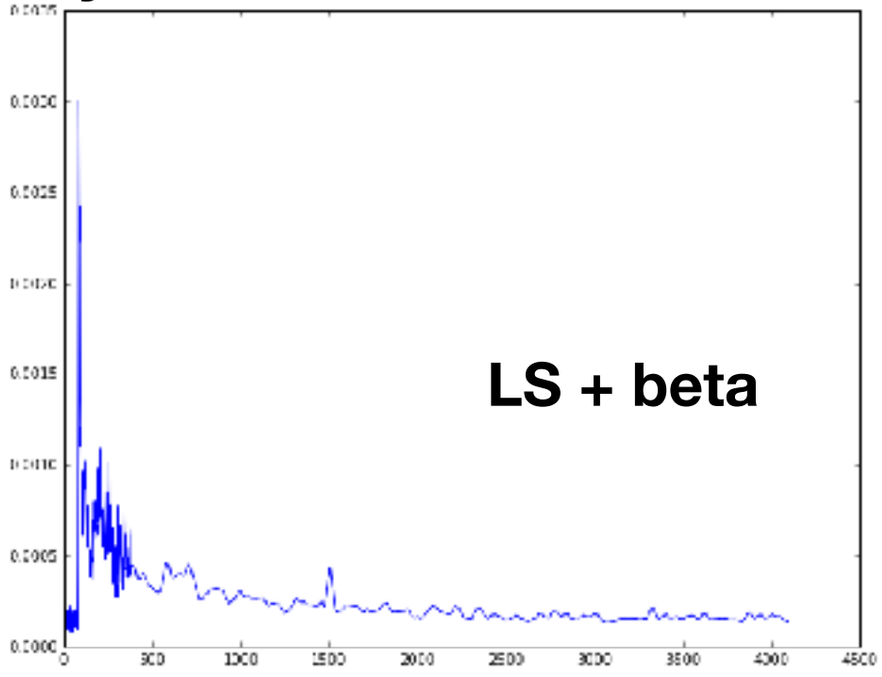




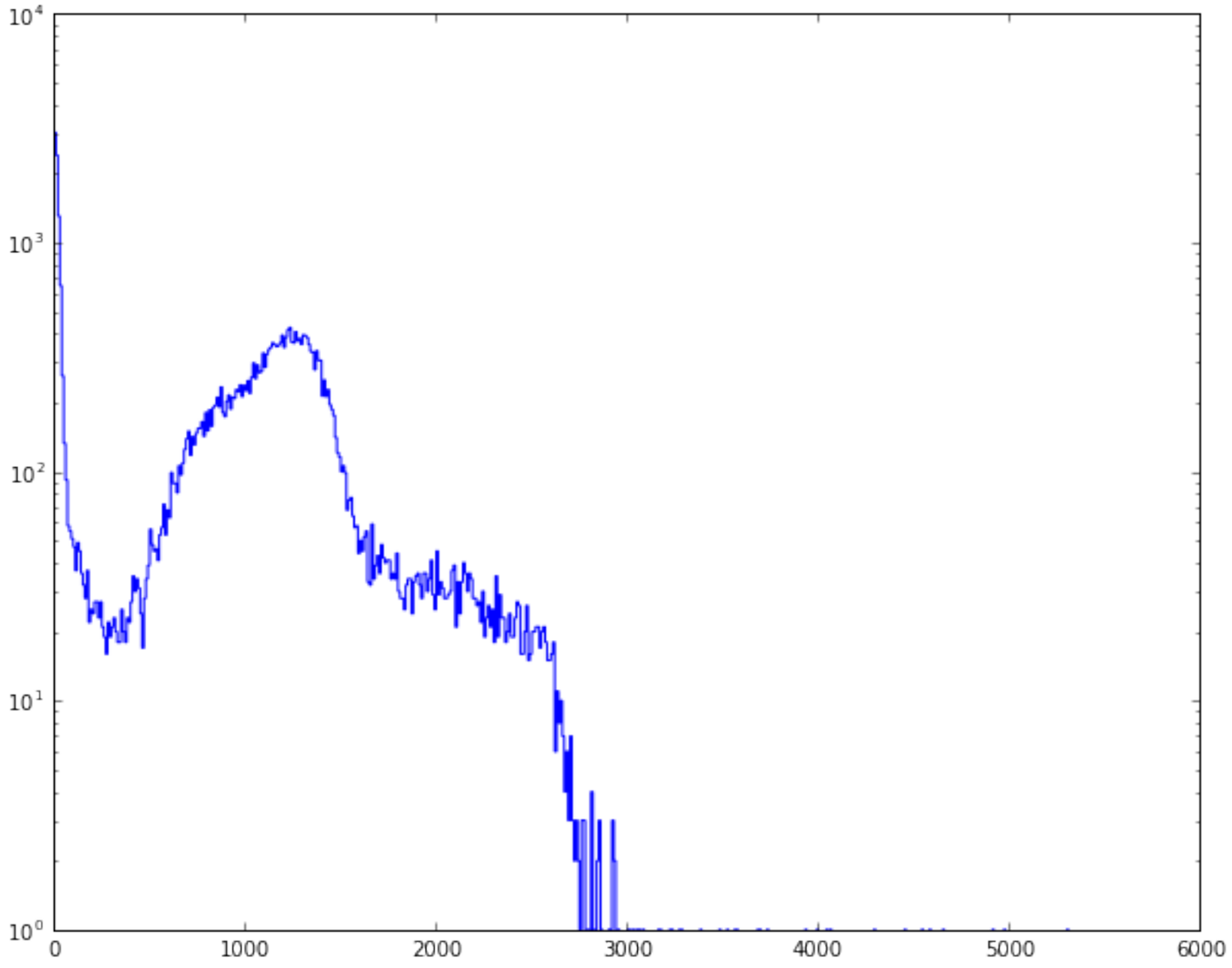


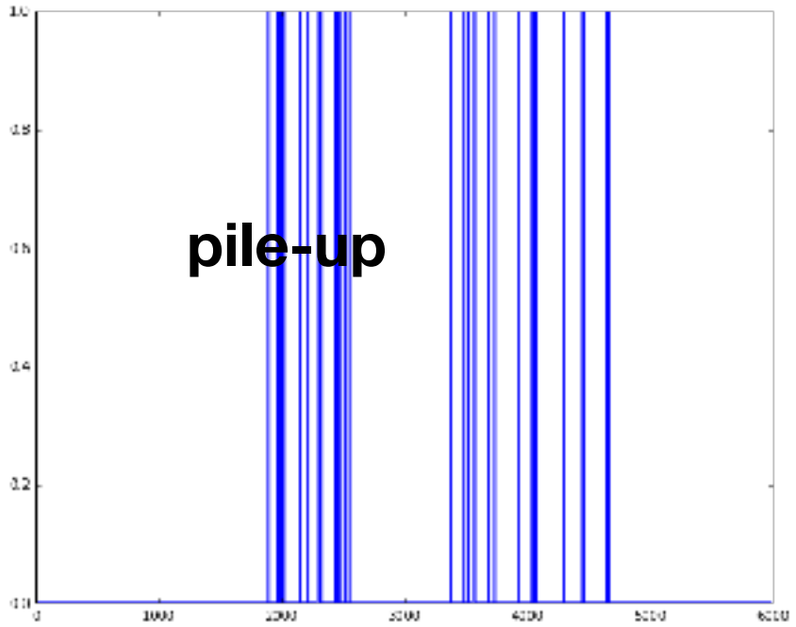
The “groups of waveform” I am trying to identify



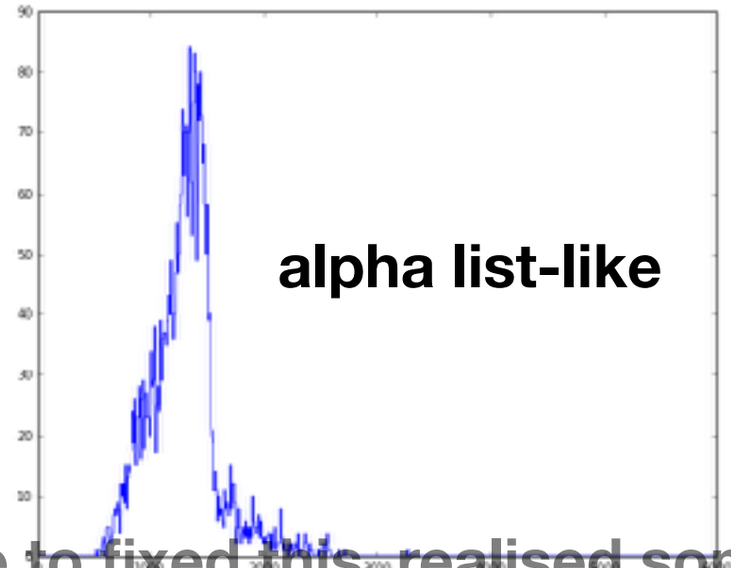


UnCut, all Events



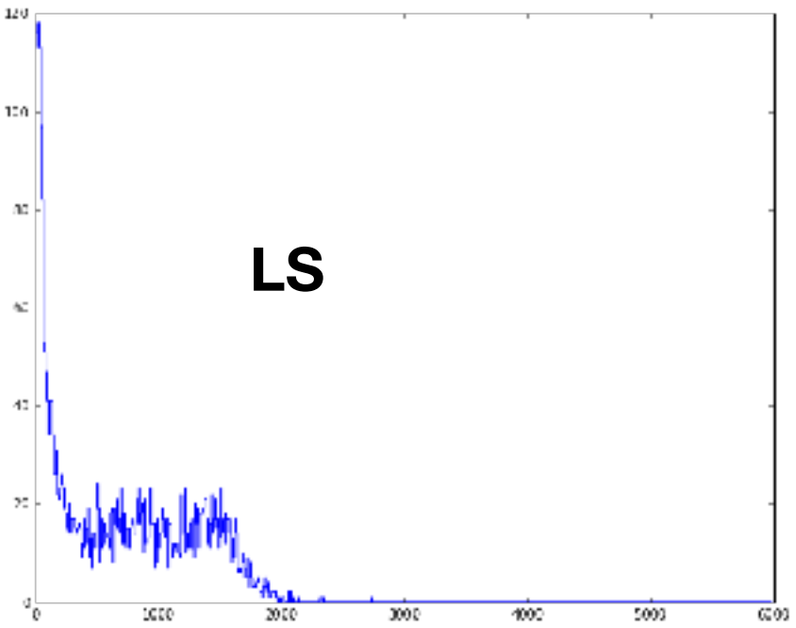


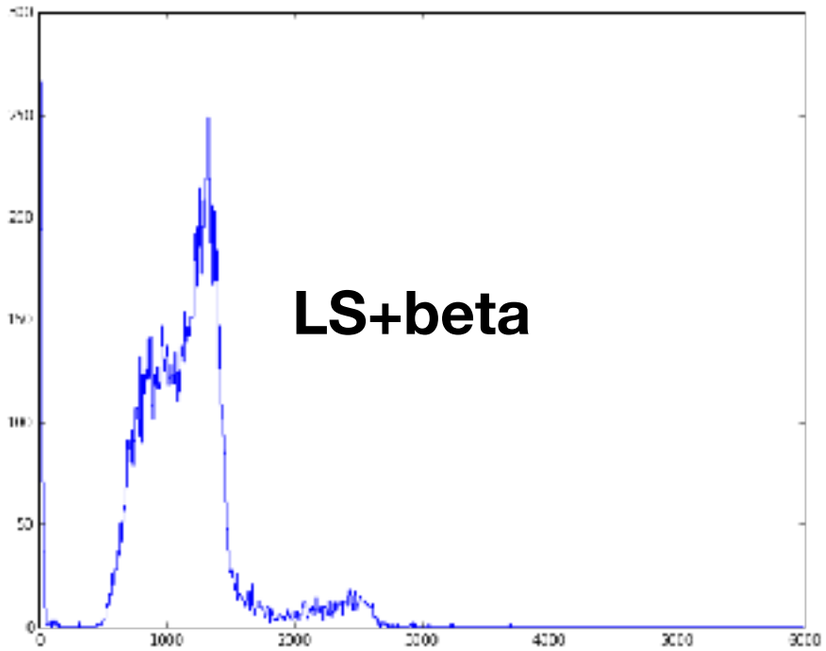
This is probably wrong

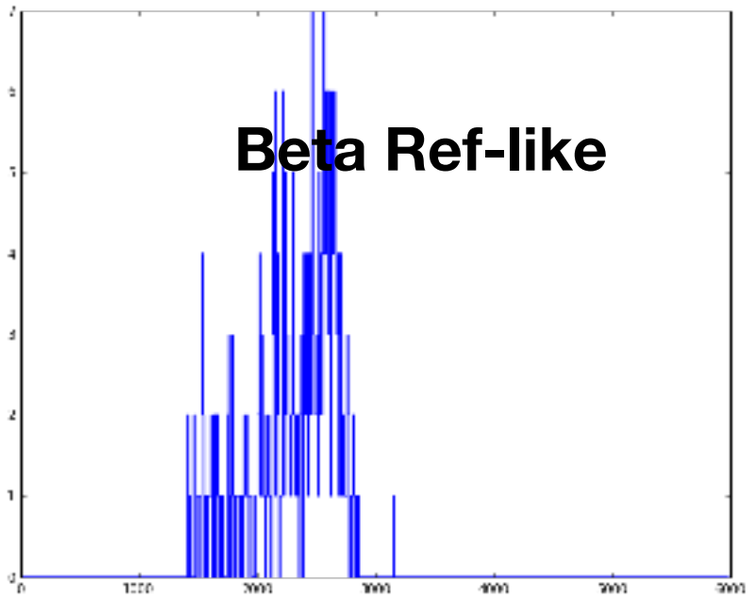


I have to fixed this, realised some mistake in making training data

This is done with a very simple CNN, a single convolution/pooling layers. can be improved with a much larger network/better optimisation to training data.



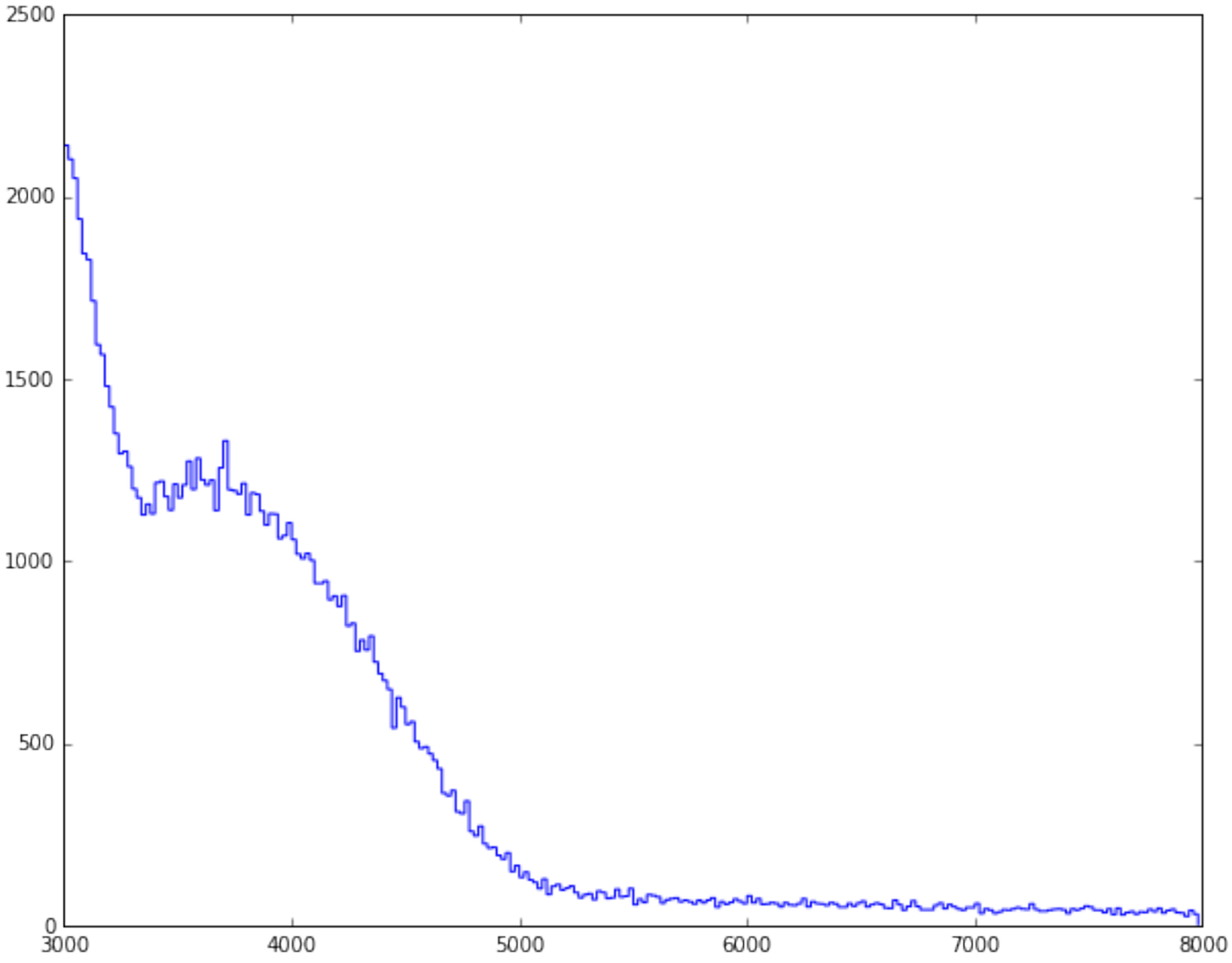


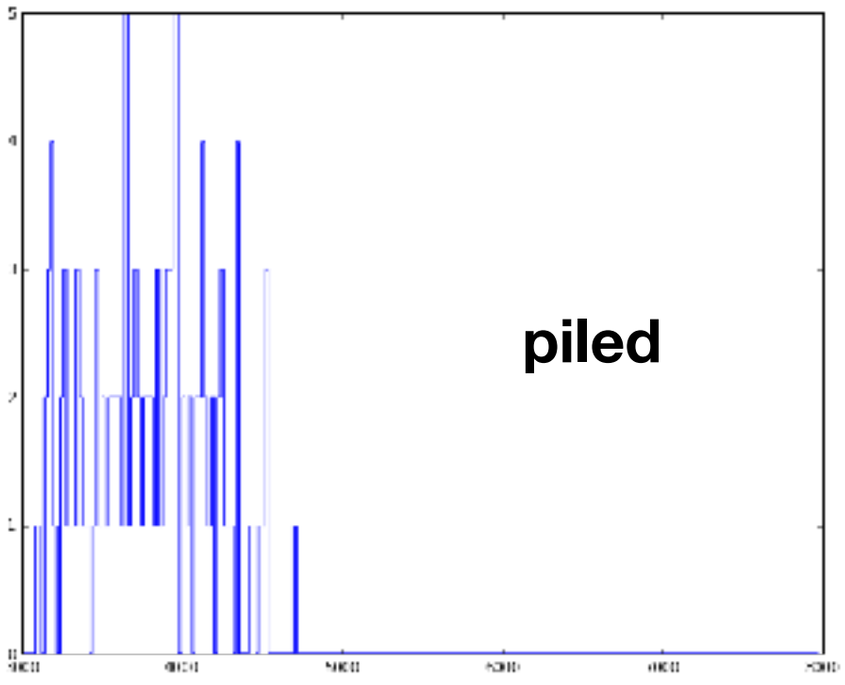


The training data are made by comparing visually, not totally accurate, but something to get started, still some improvement needed

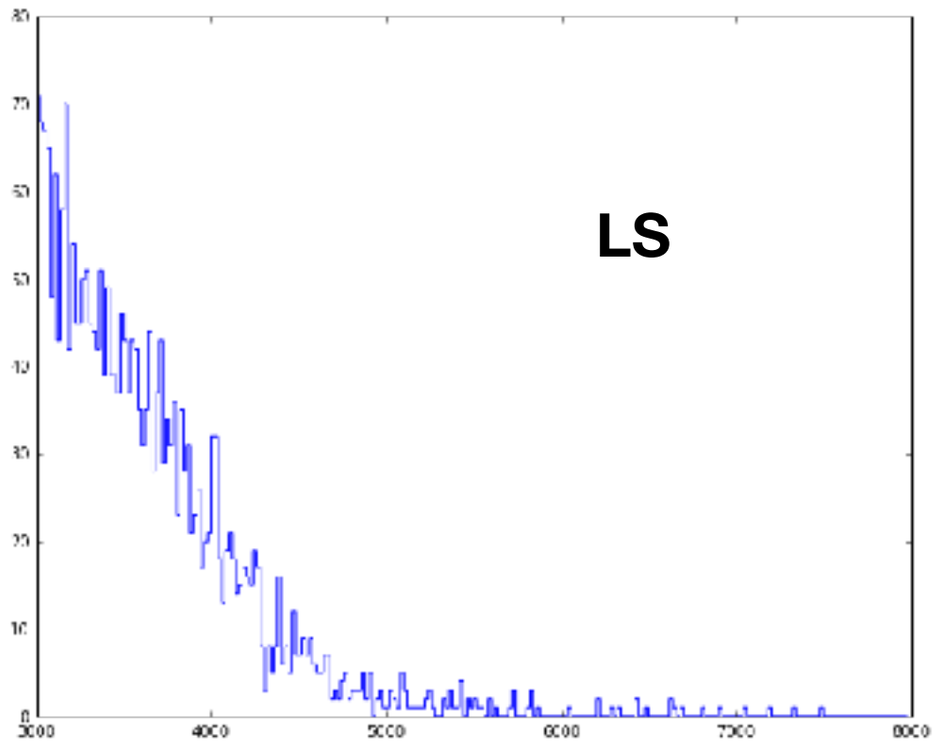
Test it on another set of data, above 3000keV

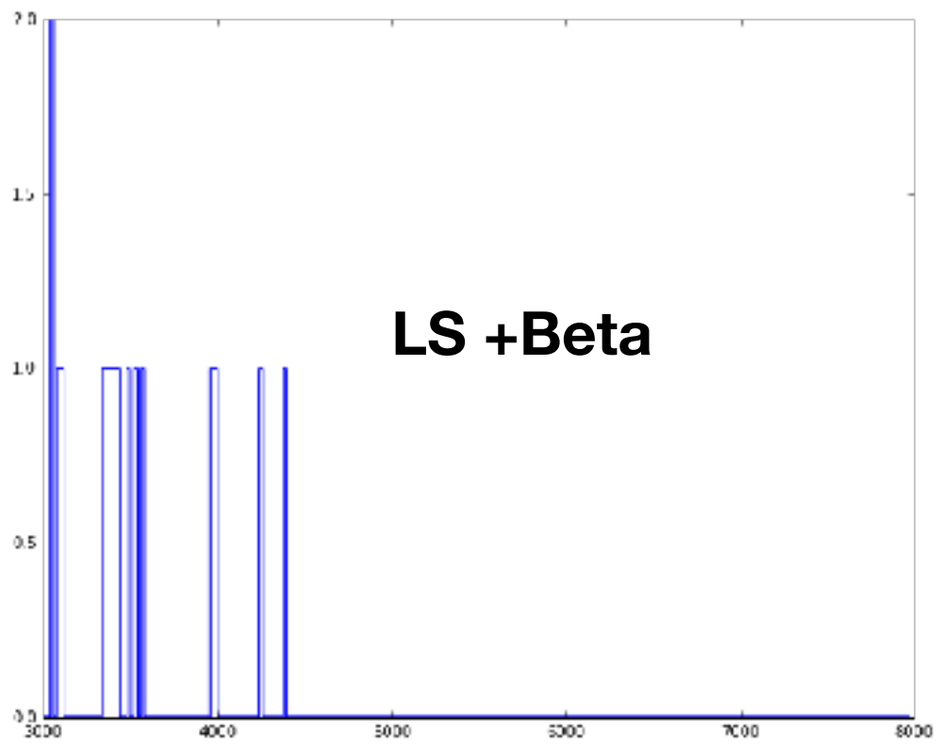
>3MeV





>3MeV

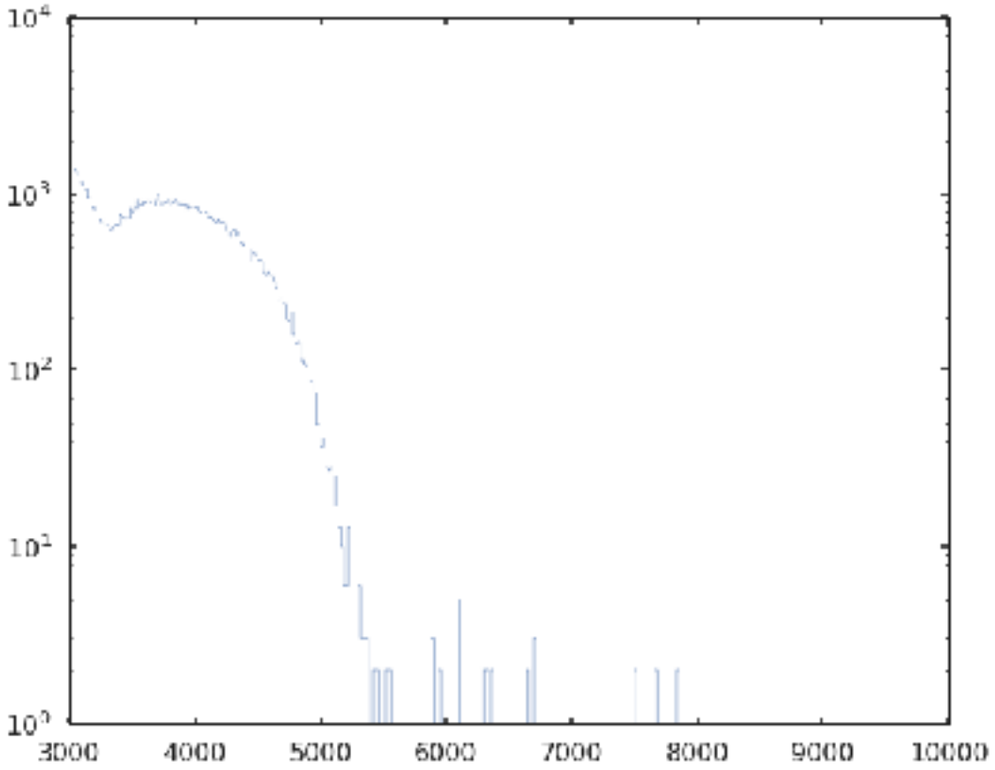


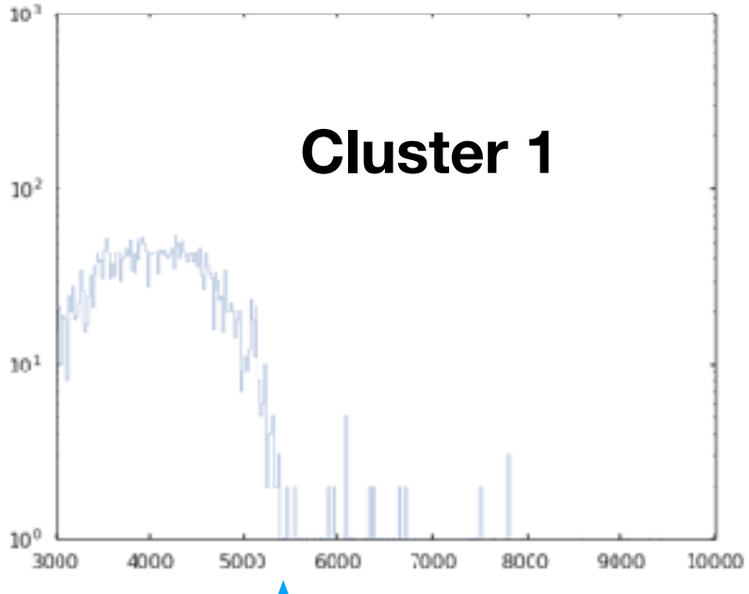


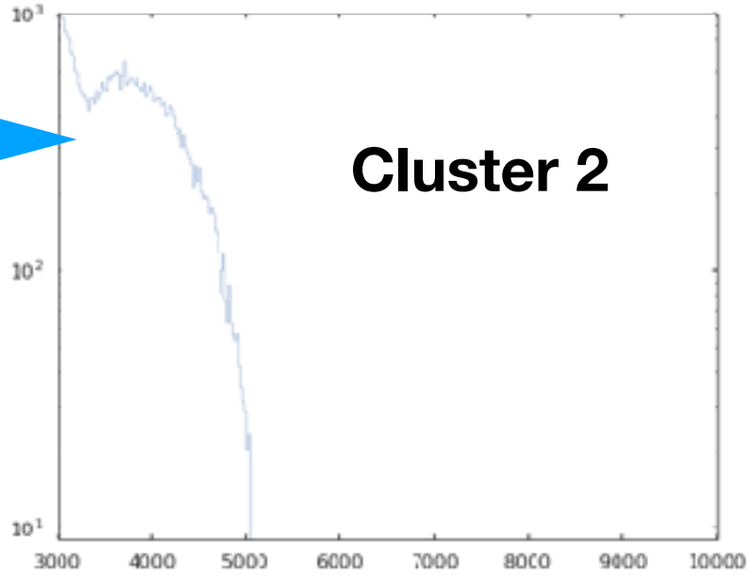
Clustering based techniques

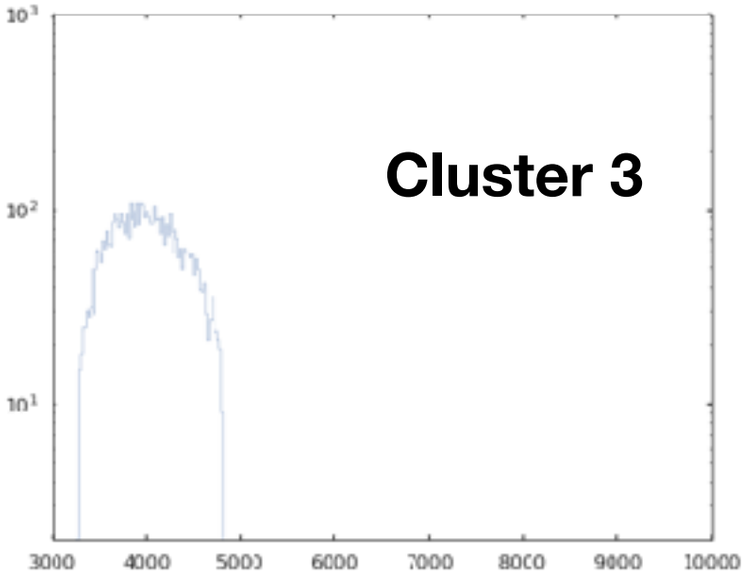
algorithms **Hierarchical Density-Based Spatial Clustering of Applications with Noise**

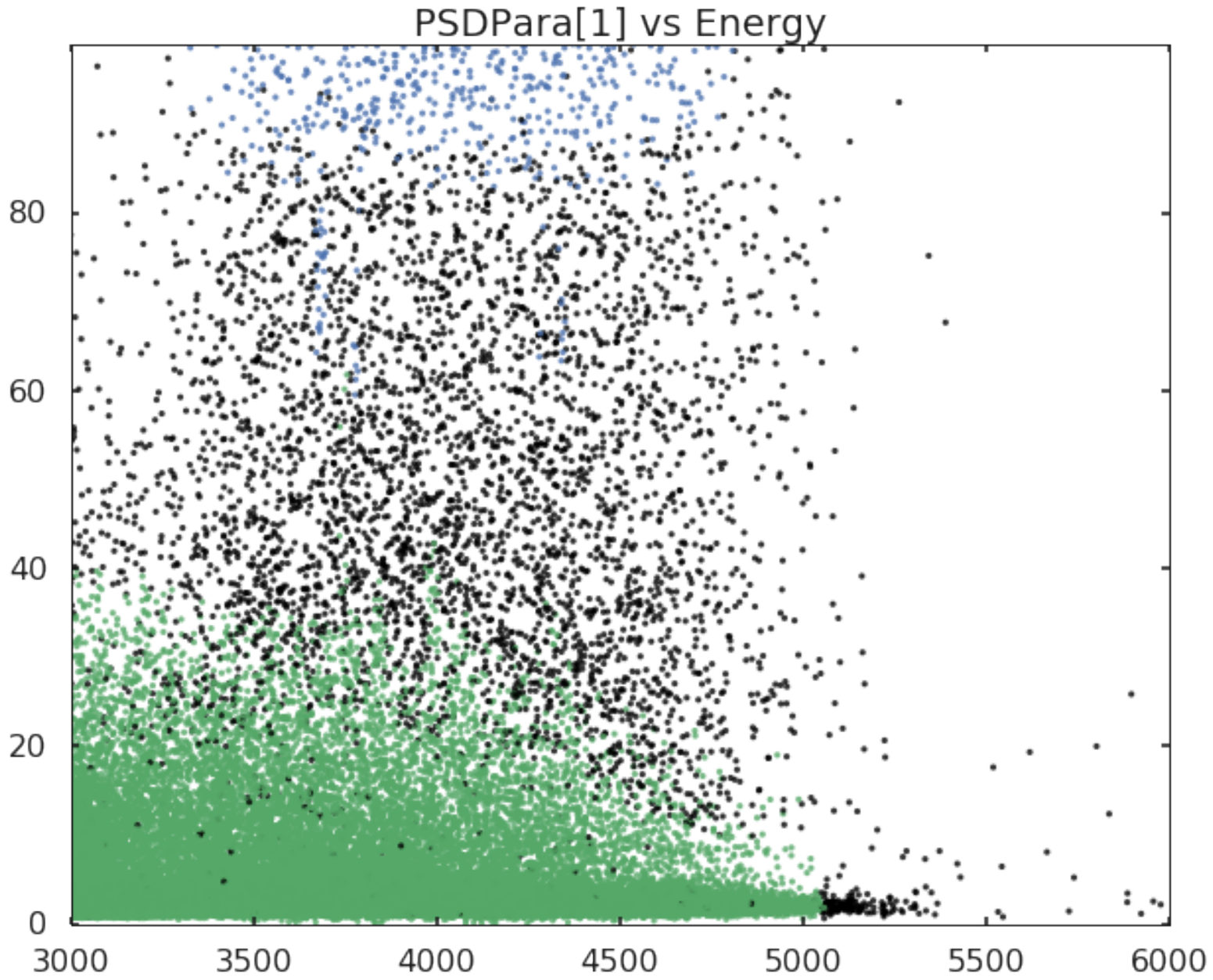
…







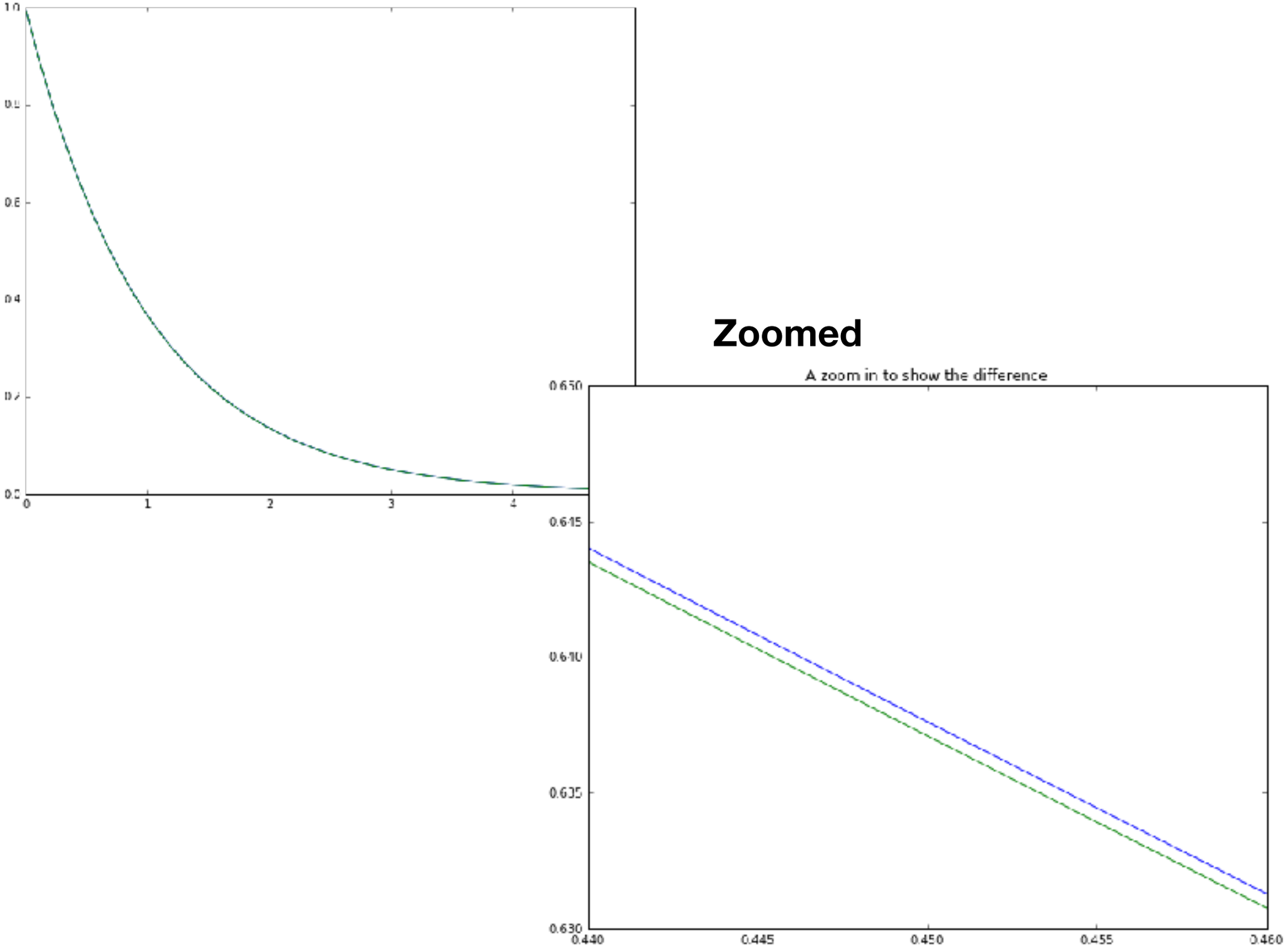



PSDPara[1] vs Energy
80
60
40
20
0
3000
3500
4000
4500
5000
5500
6000

Something far fetched, difficult

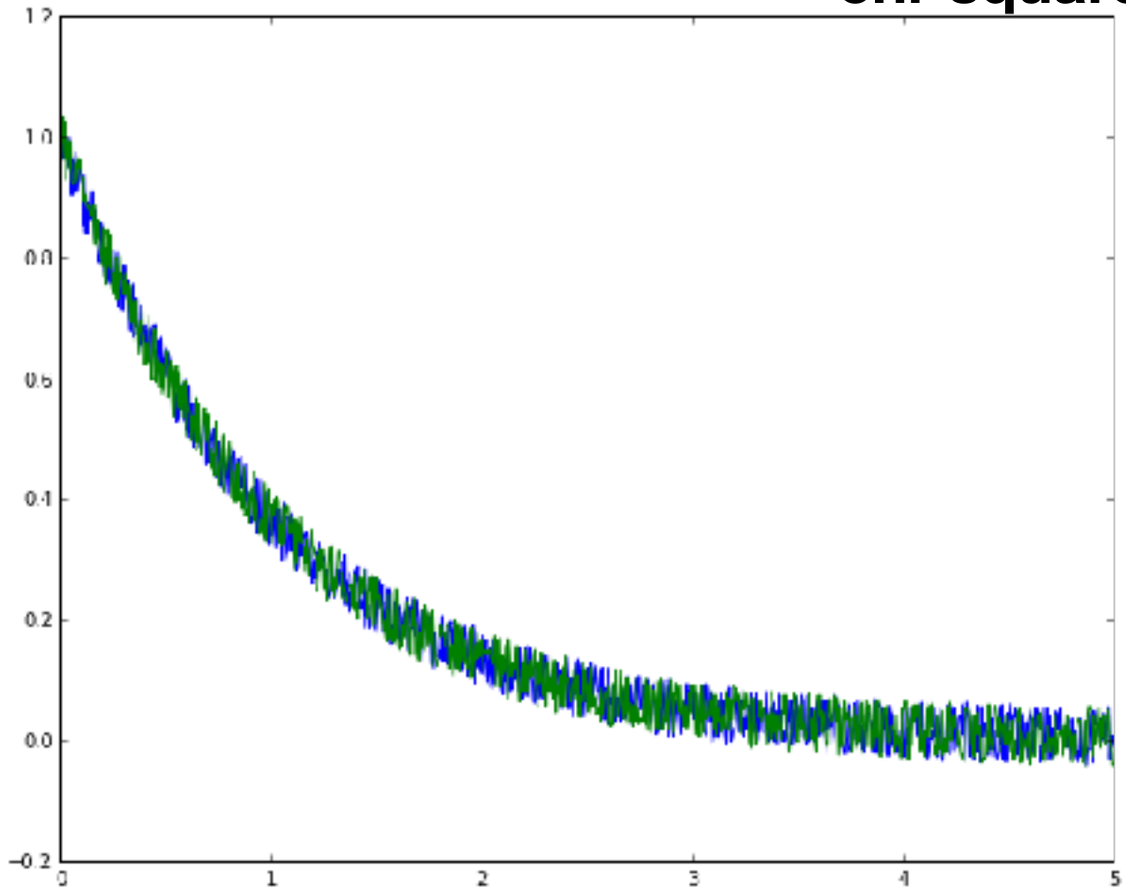
Doublely stacked pulses

- An instance where two separate events happens within the same crystal but can only be treated as single events since they both occur at nearly zero interval making it difficultly to distinguish it.

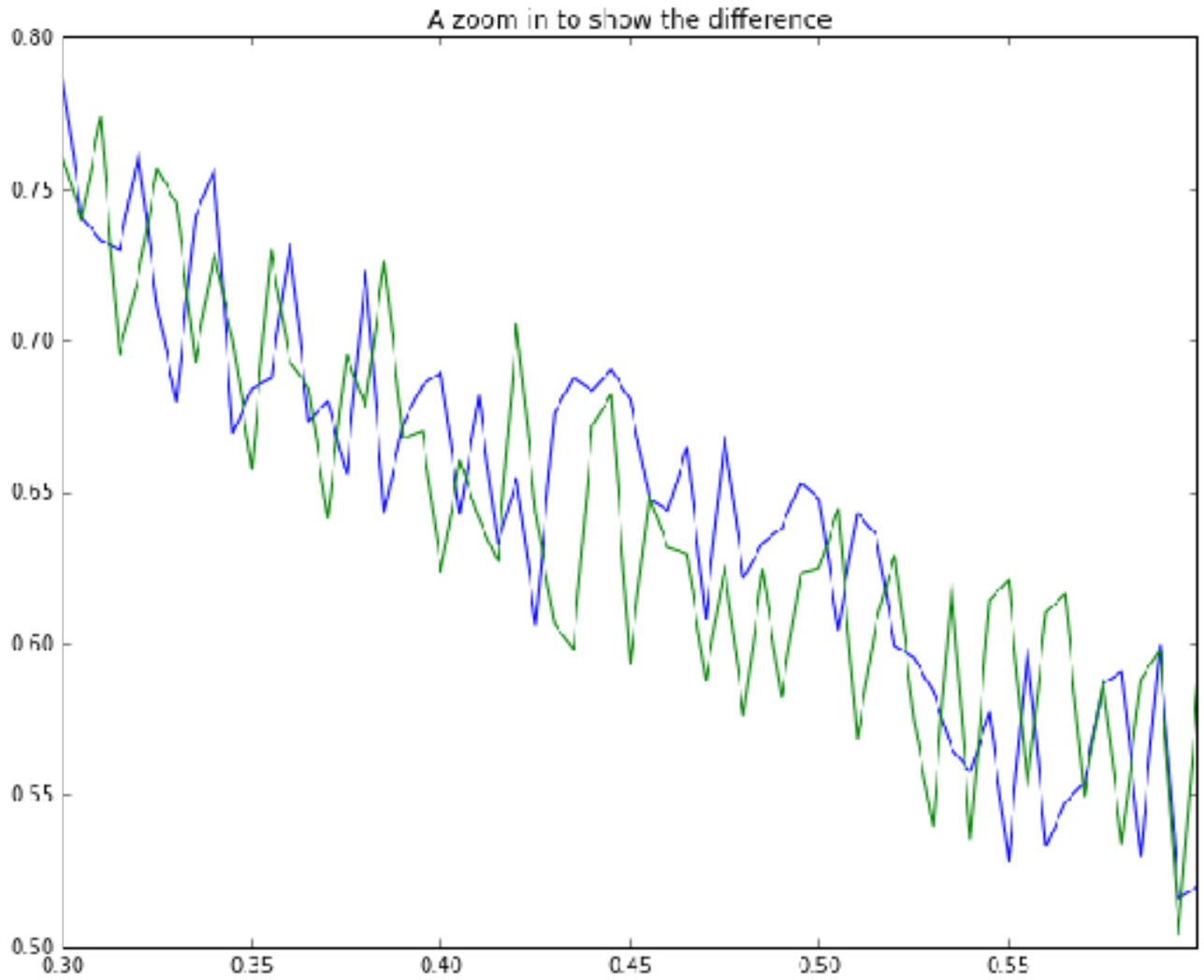

1.0
0.8
0.6
0.4
0.2
0.0
0
1
2
3
4
A zoom in to show the difference
0.650
0.645
0.640
0.635
0.630
0.440
0.445
0.450
0.455
0.460

Zoomed

chi-squared would also have worked, testing the waters



Zoomed



Tried different way to re-representing in different format

I tested this, but the CNN has a success rate of guessing correctly less than 65%.(2/3 chance of CNN getting correct answer, an untrained CNN will get 1/2 ,2 choices)
actual data not yet(difficult to obtained the labelled data)
Signal and noise ratio plays a part here.
I have not yet figure out a good way to identify these separately but it looks possible since the CNN learned something. (overfitting(CNN) was a problem also, It was depended on the noise I induced)

- Make training data from simulation? Looking into it but not an easy feat to do(dealing with individual pmts to get the whole pulse).

- Simulation from DL??