



# Particle Discrimination Using Machine Learning Techniques

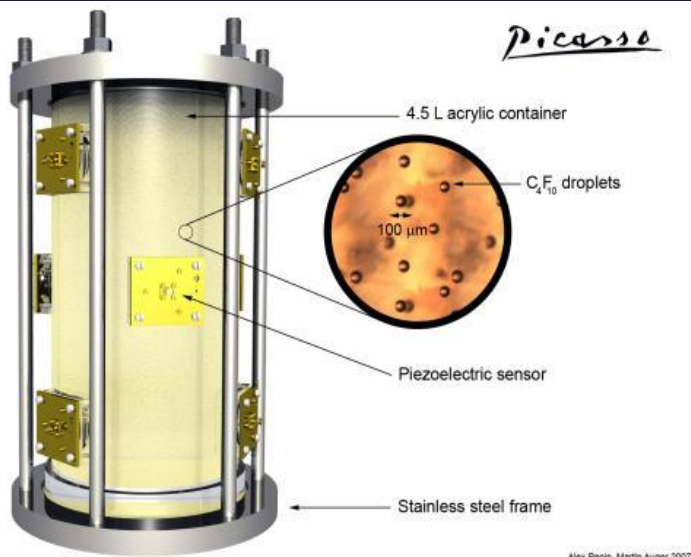
- ❖ Megan McArthur
- ❖ Ubi Wichoski, Caio Licciardi, Kyle Yeates, Alexandre Le Blanc
  
- ❖ CASST 2021
- ❖ August 23rd 2021

# Background

# The Experiments

## PICASSO

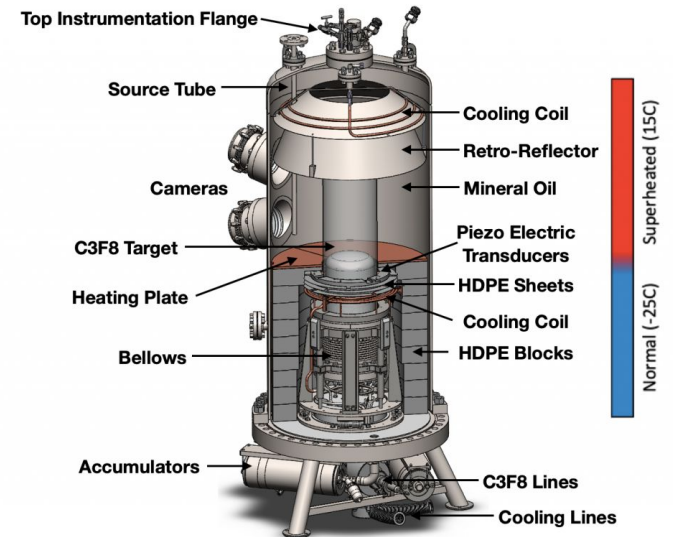
Superheated Droplet Technique



Searching for dark matter through the detection of Weakly Interacting Massive Particles (WIMPs)

## PICO

Bubble Chamber

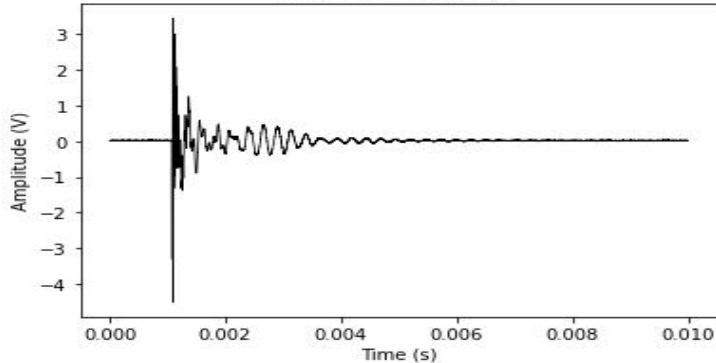


<https://www.picoexperiment.com/pico-40/>

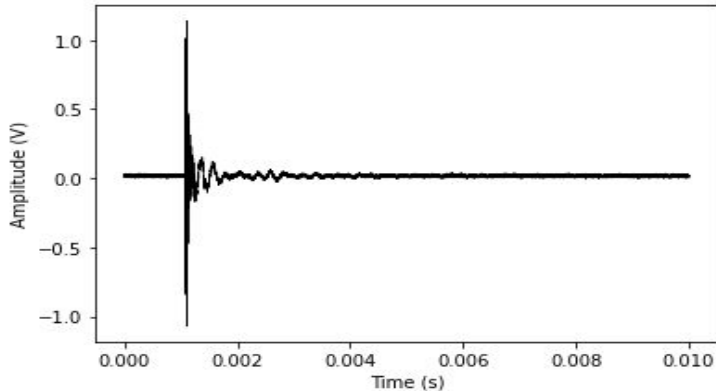
[http://www.picassoexperiment.ca/experiment\\_detector.php](http://www.picassoexperiment.ca/experiment_detector.php)

# Figuring out what's what

Run 105 : Event 002



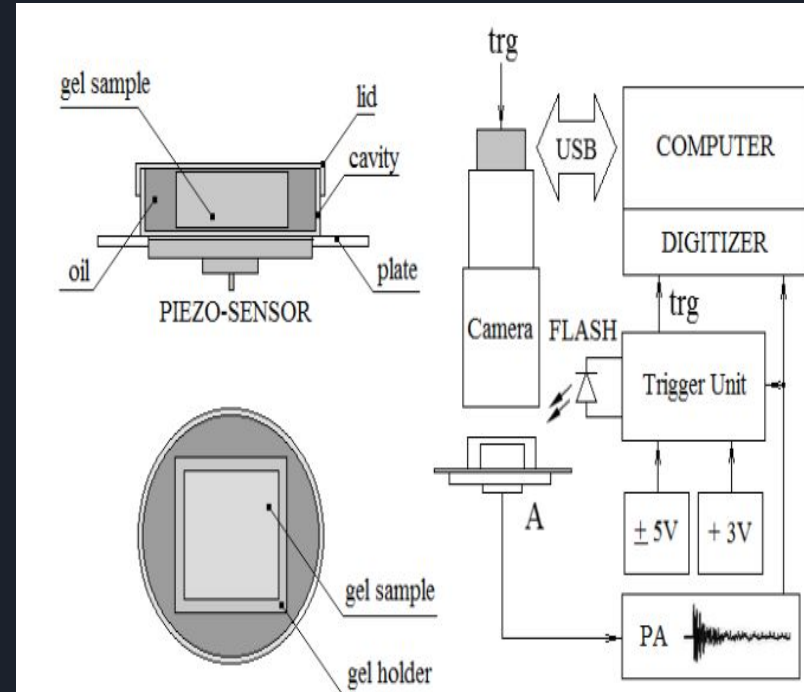
Run 109 : Event 001



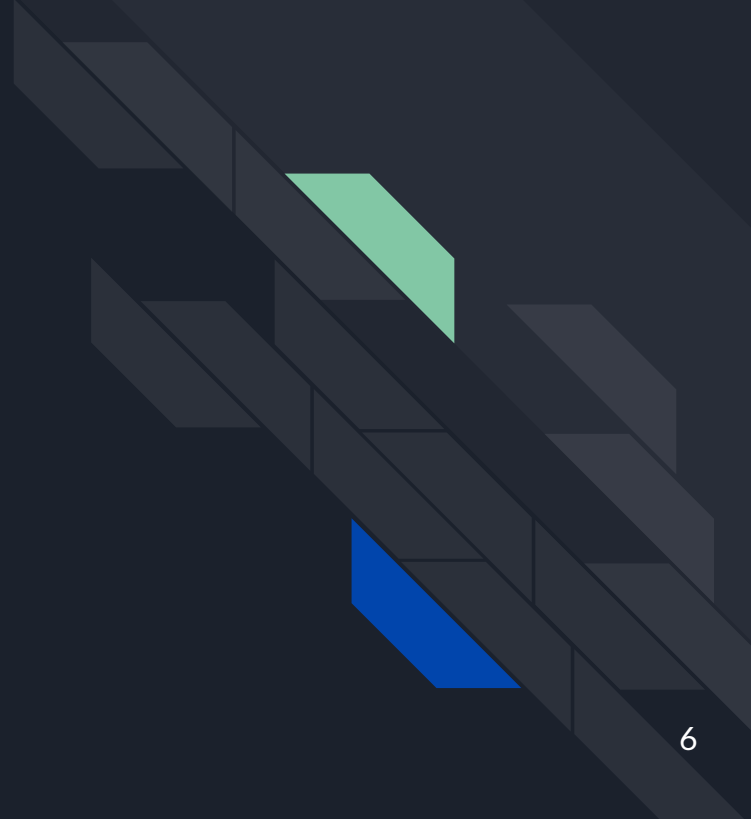
- Signals are generated by a variety of interacting particles, not just WIMPs
- We need a discriminator, something that will tell us which signals belong to which particles
- There have been a few successful tools developed that could make this discrimination for PICO data
  - Deep Learning algorithms using a combination of image and signal processing
  - The Acoustic Parameter (AP)
- Why haven't these tools worked as expected for PICASSO?

# Navaraj Dhungana's Work

- Navaraj took data from an experiment that used the same technique as PICASSO, but was dedicated to finding differences between alpha and neutron induced signals
- He included a camera in his apparatus so we could confirm whether or not an event was actually caused by bubble formation or from some other noise
- This PICASSO-Like data is the data that we are using in our analysis, as we need the information from the images going forward



# The Plan



# Machine Learning

(aka, make the computers do it!)

- Use a Gradient Boosted Classifier to come up with discrimination between alpha and neutron particles.
- Started with PICASSO-Like data and plan to include PICO data in the future
- Organizes based on features we choose, so we will be able to tell what specifically about alpha and neutron signals is different.
  
- Work is being done in parallel with work by Kyle Yeates, who is developing a neural network to perform this same discrimination
- We expect the neural network to outperform the decision tree, but the results will show us the “best case scenario”

Me: \*uses machine learning\*

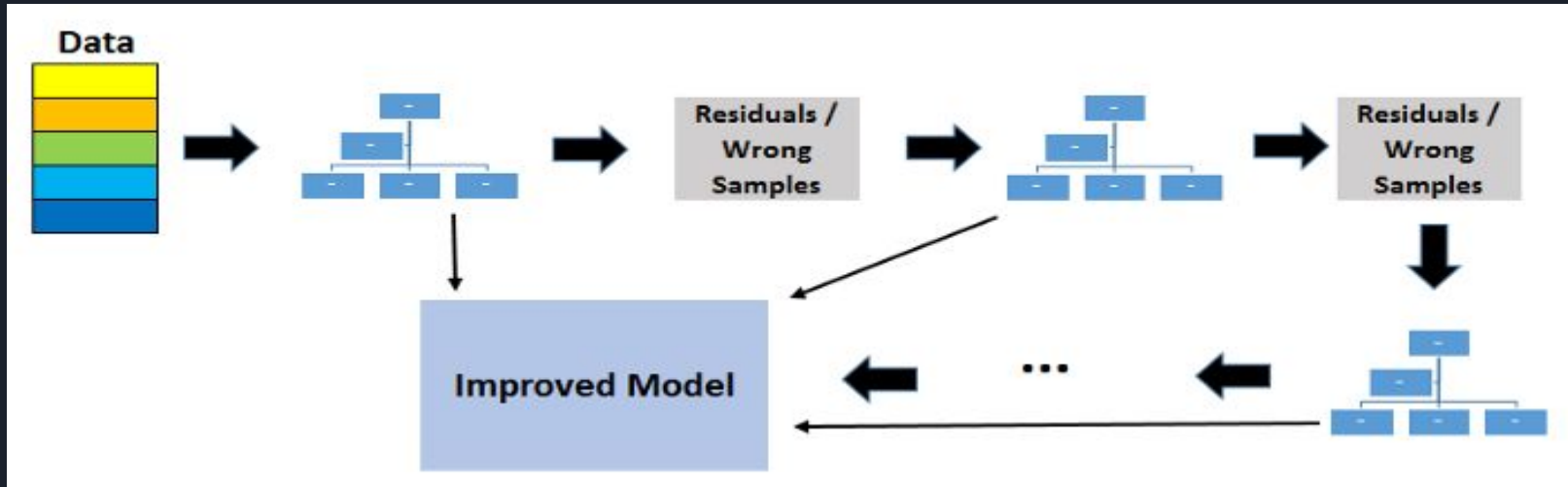
Machine: \*learns\*

Me:



# What is a Gradient Boosted Classifier?

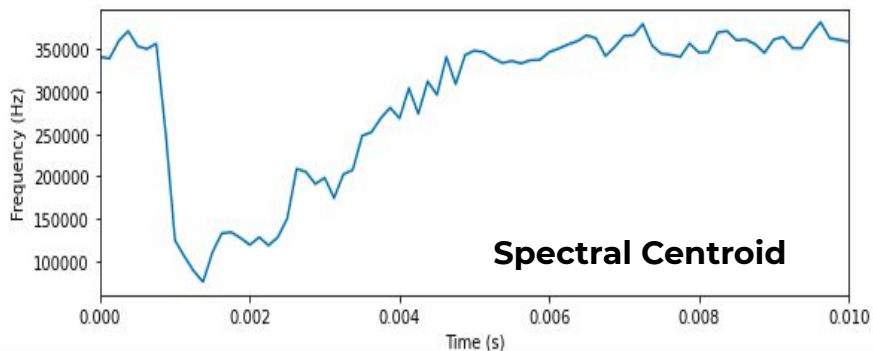
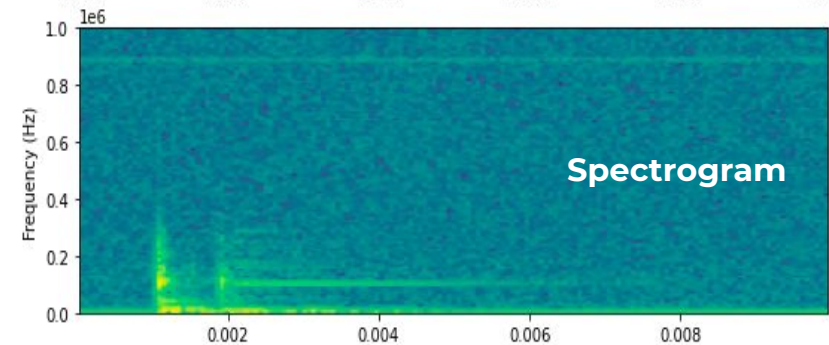
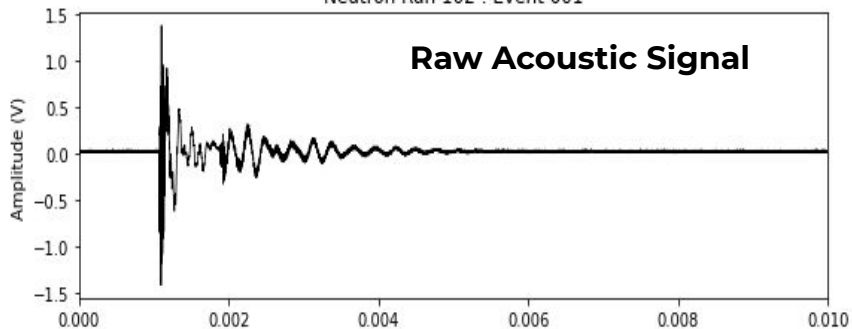
- Also known as a Boosted Decision Tree
- This classifier starts with a simple decision tree, which organizes data based on the answer to simple true/false questions
- Once a single tree is created it evaluates the model to find weak branches, and then creates a new tree that organizes that data
- This process repeats a number of times until the best possible model is created





# Building the Model

The background features a series of dark grey, 3D rectangular blocks arranged in a perspective view, receding towards the top right. Two blocks are highlighted: one is light green and the other is blue, both positioned in the middle-right section of the arrangement.



# Finding Features

## Audio Analysis

- Processed our acoustic signals as audio files, based on the theory that the alpha and neutron bubble events simply *sounded* different.
- Some features also chosen based on past observations made by others working with this data

## Final Features:

- Maximum amplitude of the signal
- Zero crossing rate of first 0.25ms of signal
- Total average Spectral Centroid
- Total average Spectral Bandwidth

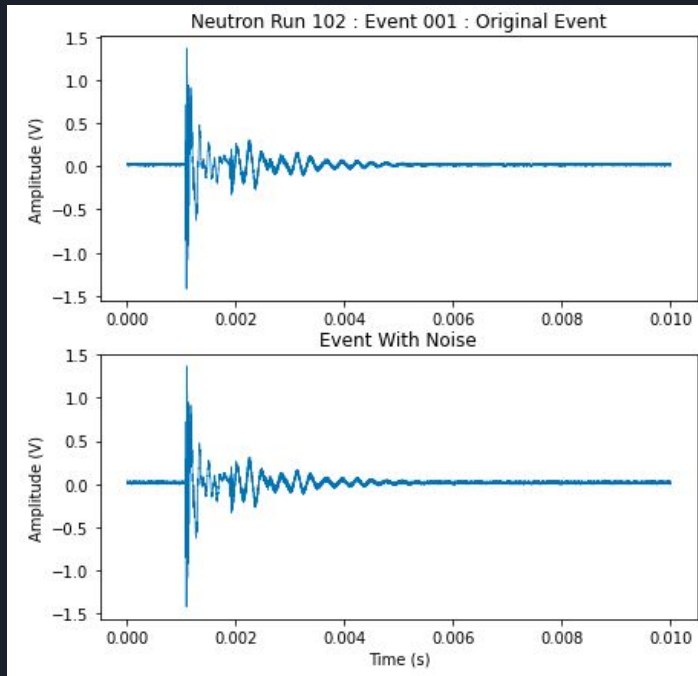
# The Nitty Gritty

## Balancing

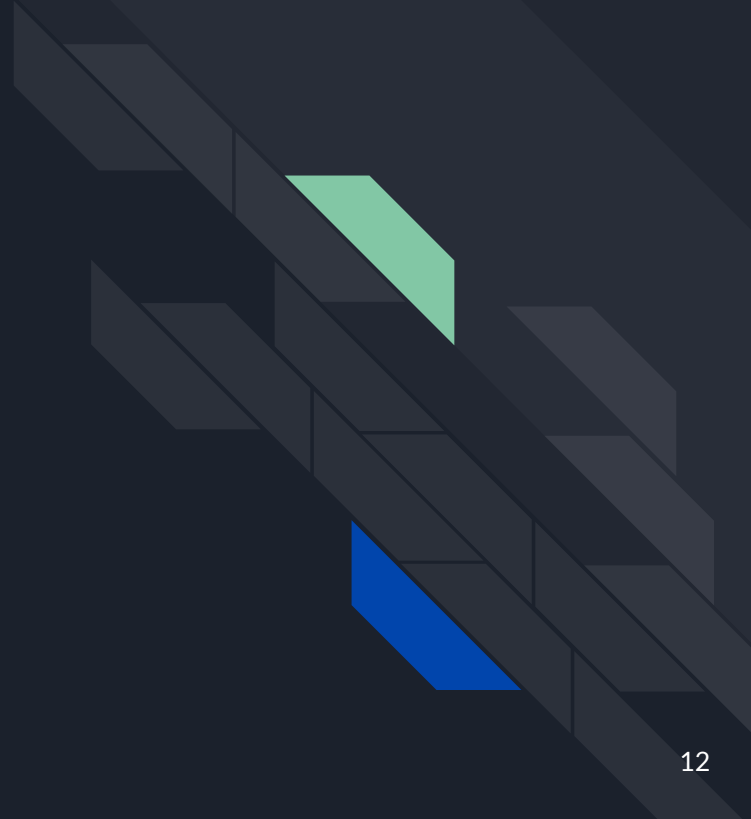
- Balancing is an important issue in any machine learning problem
- For much of our Picasso-like data, we have much more alpha data than we do neutron data
- We balanced this data by copying the neutron files and adding a small amount of noise to each file

## Parameters

- Finding proper parameters was another trial and error game
  - ❖ `learning_rate = 0.05`
  - ❖ `n_estimators = 20`
  - ❖ `max_depth = 5`
  - ❖ `min_samples_leaf = 300`
  - ❖ `min_samples_split = 140`



# Results: Round 1



Training Set:

Null Accuracy:

0.5278276481149012

Accuracy Score:

0.6687612208258528

Confusion Matrix:

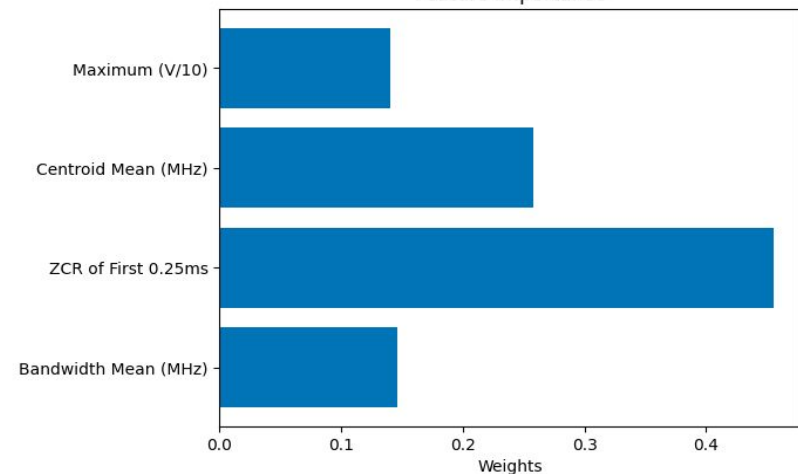
[[422 166]

[203 323]]

Classification Report:

|              | precision | recall | f1-score | support |
|--------------|-----------|--------|----------|---------|
| Alpha        | 0.68      | 0.72   | 0.70     | 588     |
| Neutron      | 0.66      | 0.61   | 0.64     | 526     |
| accuracy     |           |        | 0.67     | 1114    |
| macro avg    | 0.67      | 0.67   | 0.67     | 1114    |
| weighted avg | 0.67      | 0.67   | 0.67     | 1114    |

Feature Importance



# PICASSO-Like Data

# 67%

- Testing Set performed similarly with 64% Accuracy
- Null Accuracy: How the model could perform if it organized every event into the most frequency category
- Precision: Percentage of predicted group that was organized correctly
- Recall: Percentage of actual group that was organized correctly
- F1 - Score:  $(\text{Precision} \times \text{Recall}) / (\text{Precision} + \text{Recall})$

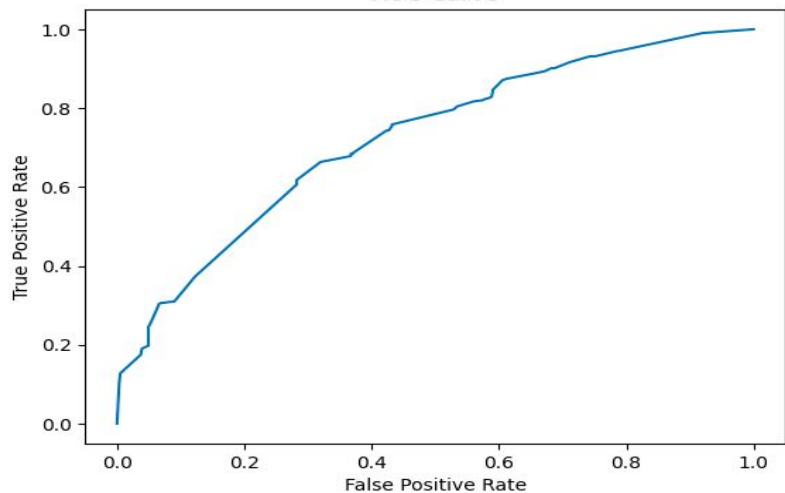
# PICASSO-Like Data

# 67%

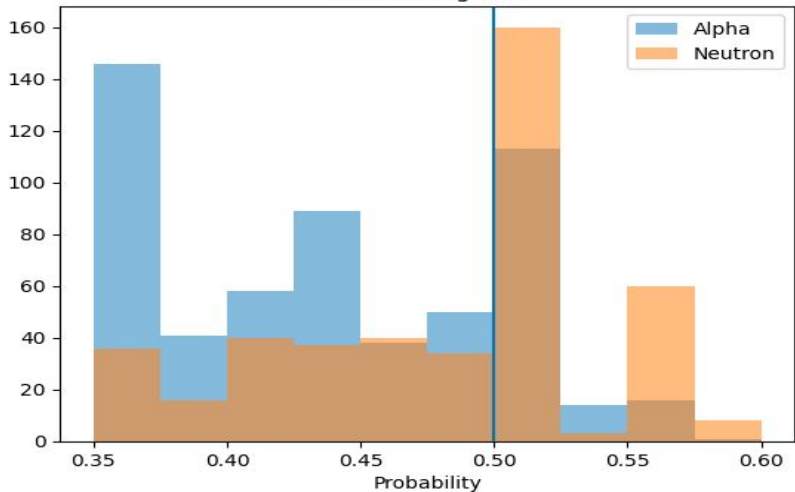
- Testing Set performed similarly with 64% Accuracy

- **ROC Curve:** Receiver Operating Characteristic curve
  - Representation of how a model can distinguish between true positives and true negatives
- **Histogram Analysis:**
  - The predicted probabilities for alpha and neutron induced events organized by their true groups

ROC Curve



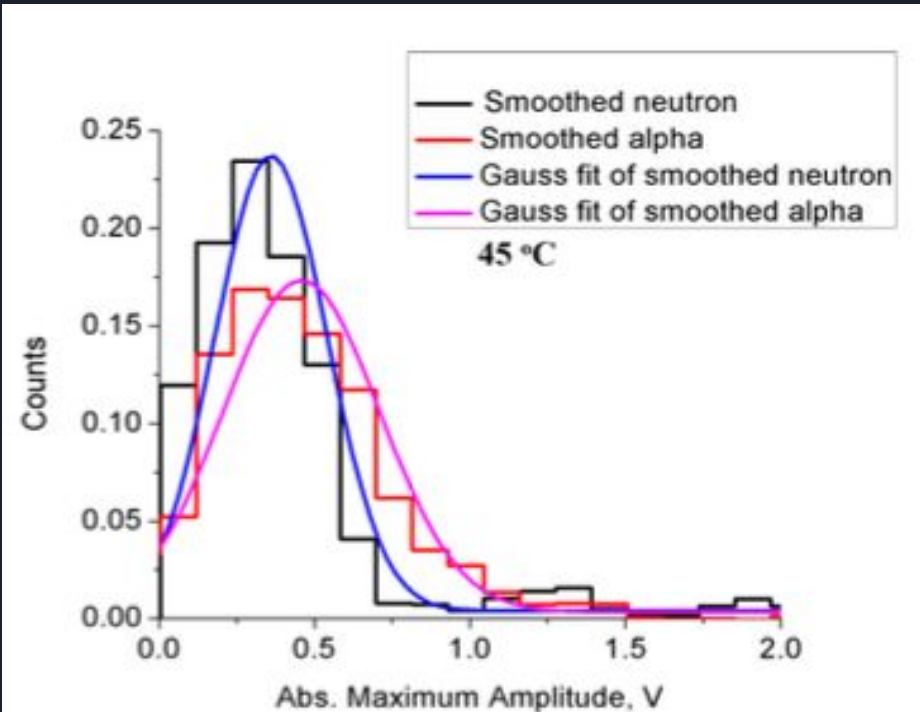
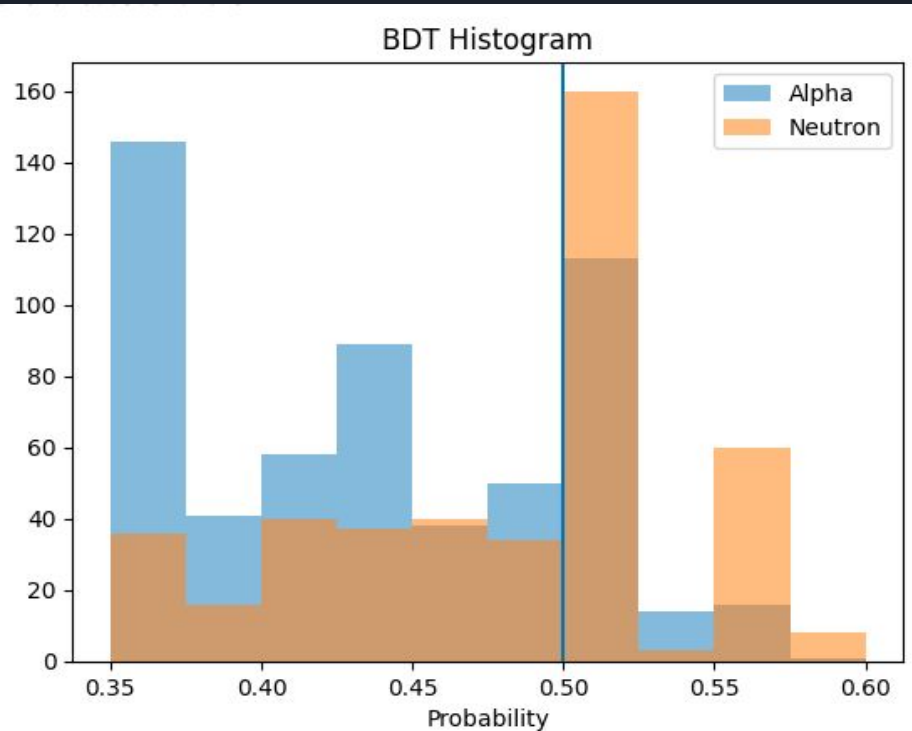
BDT Histogram



# Comparisons

With Navaraj Dhungana

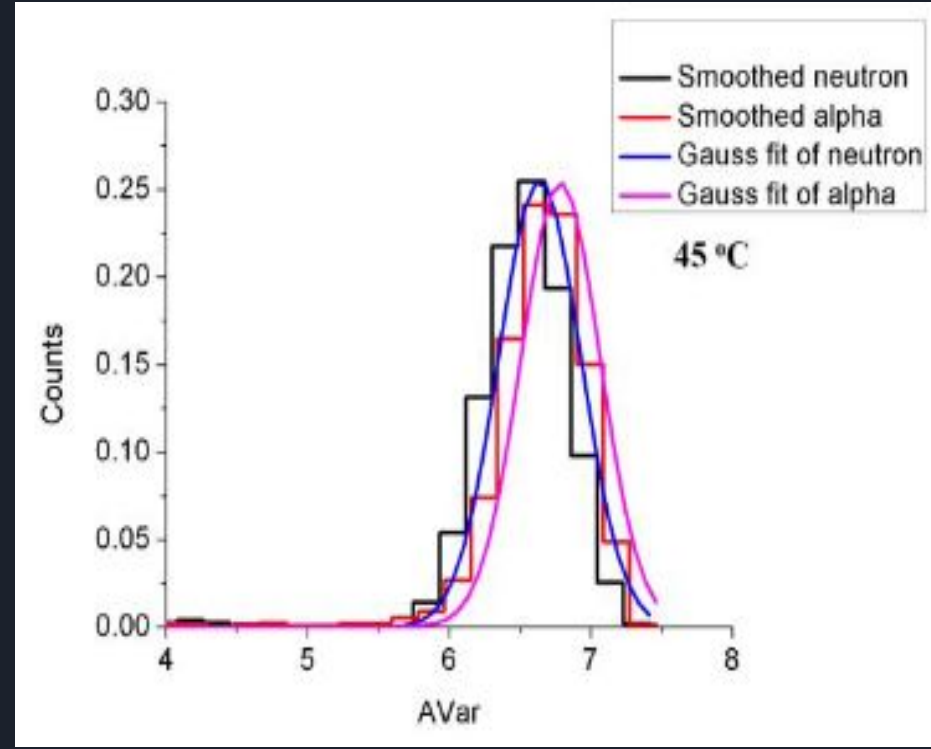
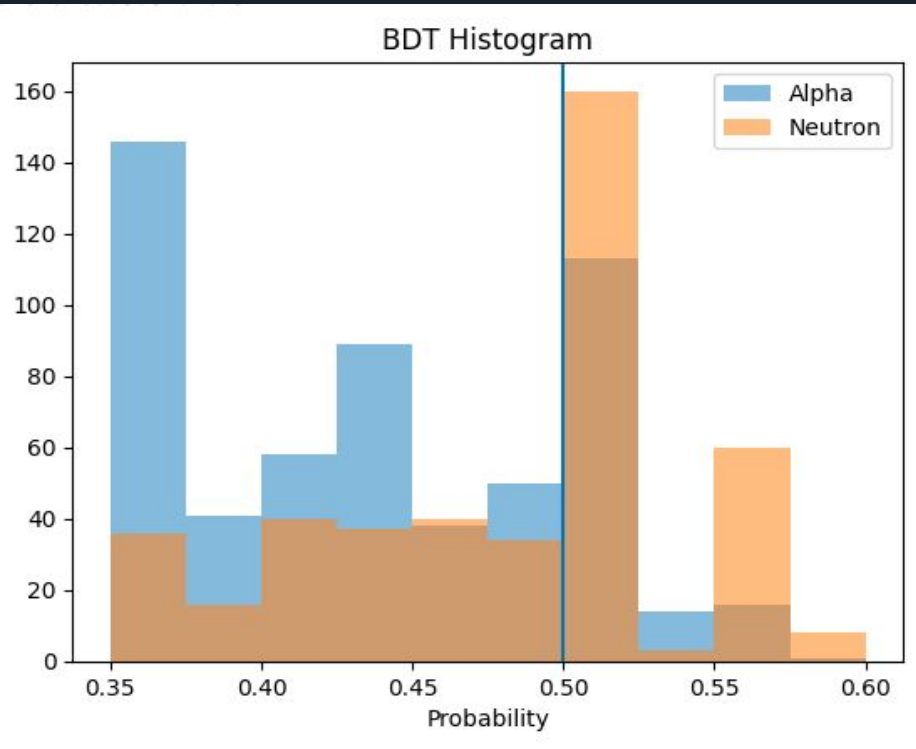
- Amplitude distribution of good alpha and neutron events
- A band pass 180 - 300 kHz Butterworth filter (built in MATLAB) has been applied to this data



# Comparisons

## With Navaraj Dhungana


- Distribution of AVAR between neutron and alpha events
  - Logarithmic sum of the amplitudes of the waveform, taken in time bins





# Continued Work

An abstract graphic on the right side of the slide. It features a series of dark grey, 3D-style rectangular blocks arranged in a diagonal line from the top right towards the bottom left. Two blocks are highlighted: one is light green and one is blue. The background is a dark blue gradient.

- 
- More features
  - Filter out bad signals
  - Taking care of data impurity
  - Direct comparison work between PICO and PICASSO data
  - Investigating whether focusing on specific parts of the signal will offer a better understanding of the bubble growth
  - Investigating how much changes in the environment affect the acoustics

We're not done yet!

Questions?