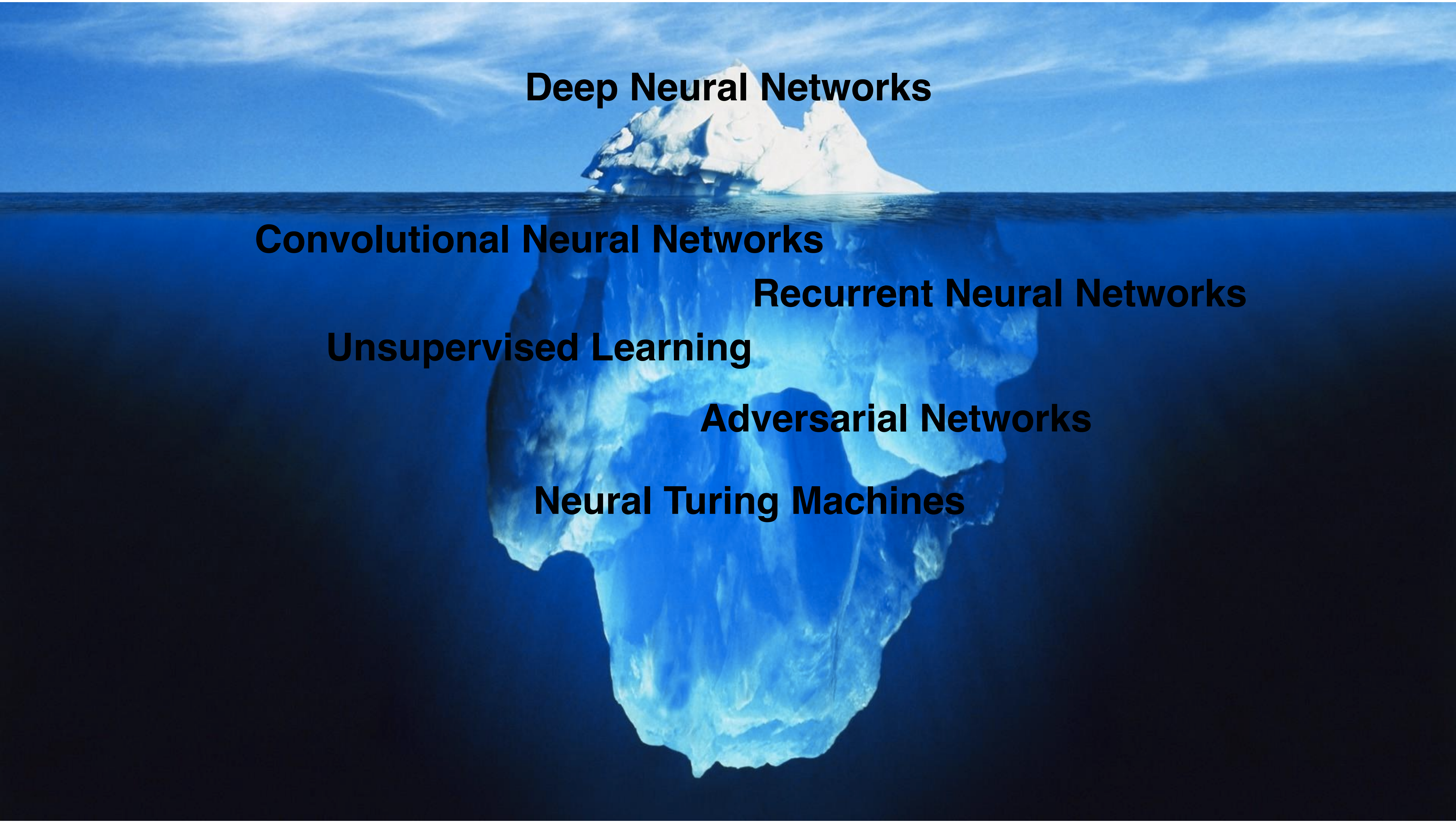
CVN at DUNE

Alexander Radovic
College of William and Mary
on behalf of the DUNE Experiment

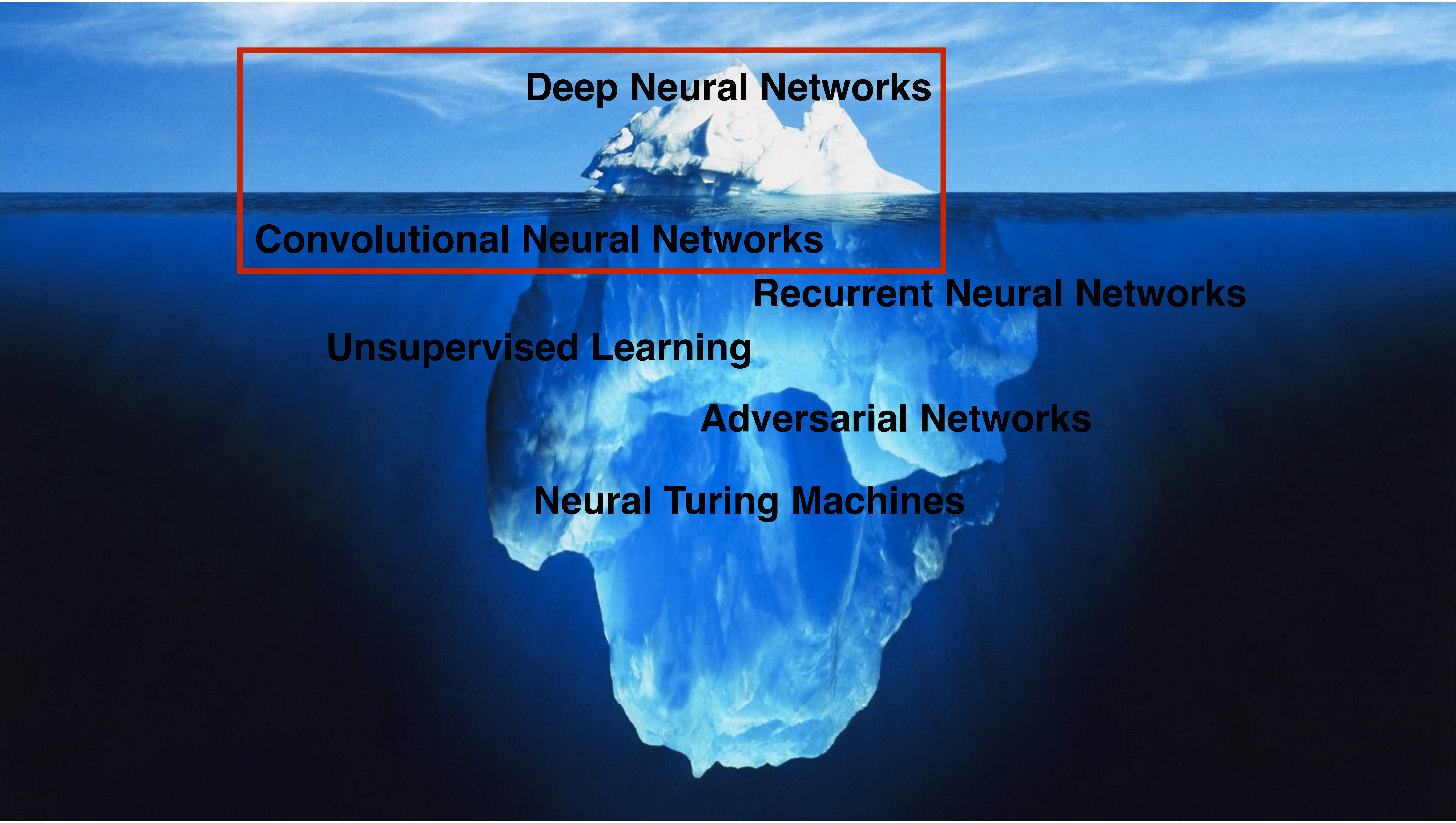
DUNE

WILLIAM & MARY
CHARTERED 1693

Deep Learning



Deep Learning



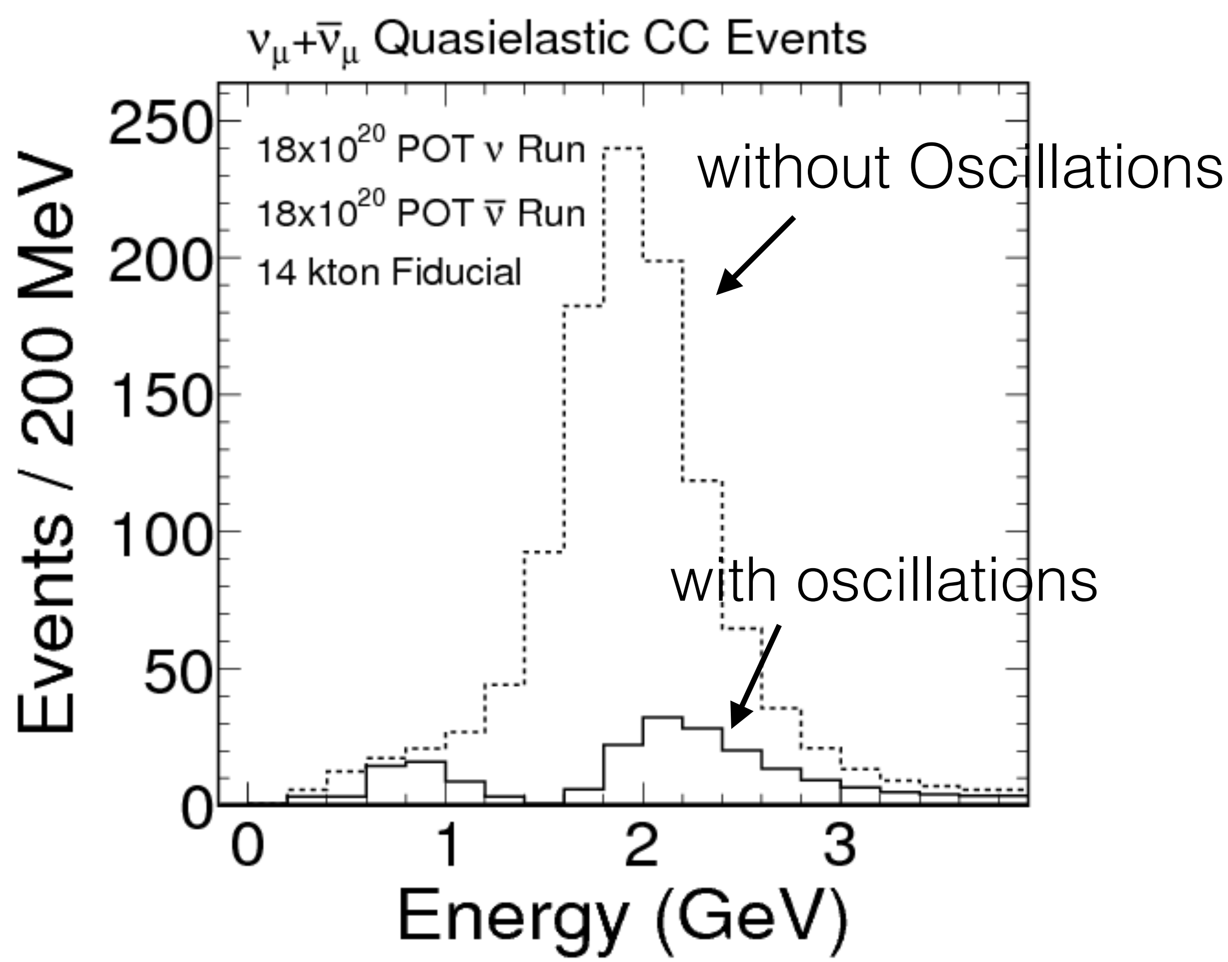
Introduction

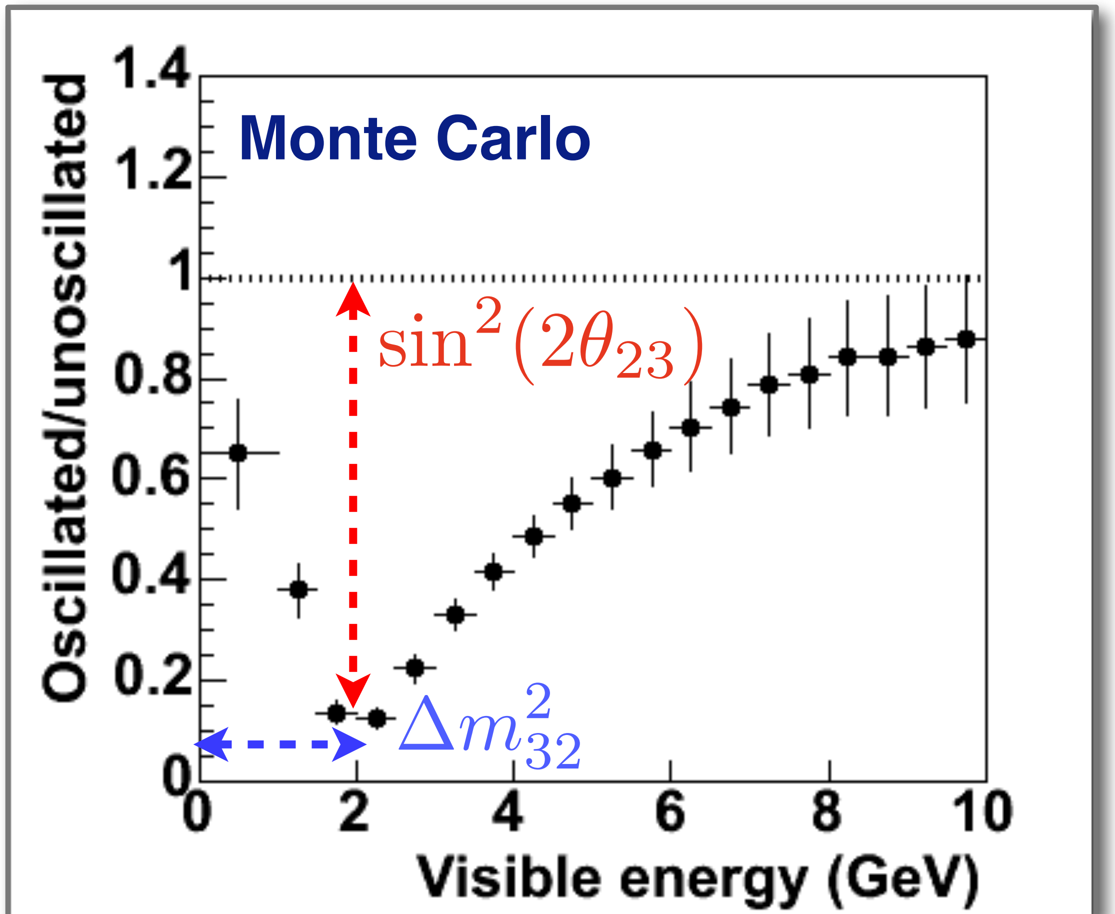
- Convolutional neural networks have proven to be extremely effective for event identification at the NOvA experiment: https://arxiv.org/abs/1604.01444
- MicroBoone has recently show us that liquid argon detector readout is a similarly excellent domain for CNNs: https://arxiv.org/abs/1611.05531
- This leads to a natural next question. Can we take the art based tools and expertise from the NOvA CNN, named CVN, and apply it on DUNE MC to rapidly produce a competitive PID?

Why Deep Neural Networks?

- Measuring neutrino oscillations is all about measuring how neutrinos change between different lepton flavor states as a function of distance traveled and neutrino energy.

$$P(\nu_\mu \rightarrow \nu_\mu) \approx 1 - \sin^2(2\theta_{23}) \sin^2\left(\frac{1.27\Delta m^2_{atm} L}{E}\right)$$



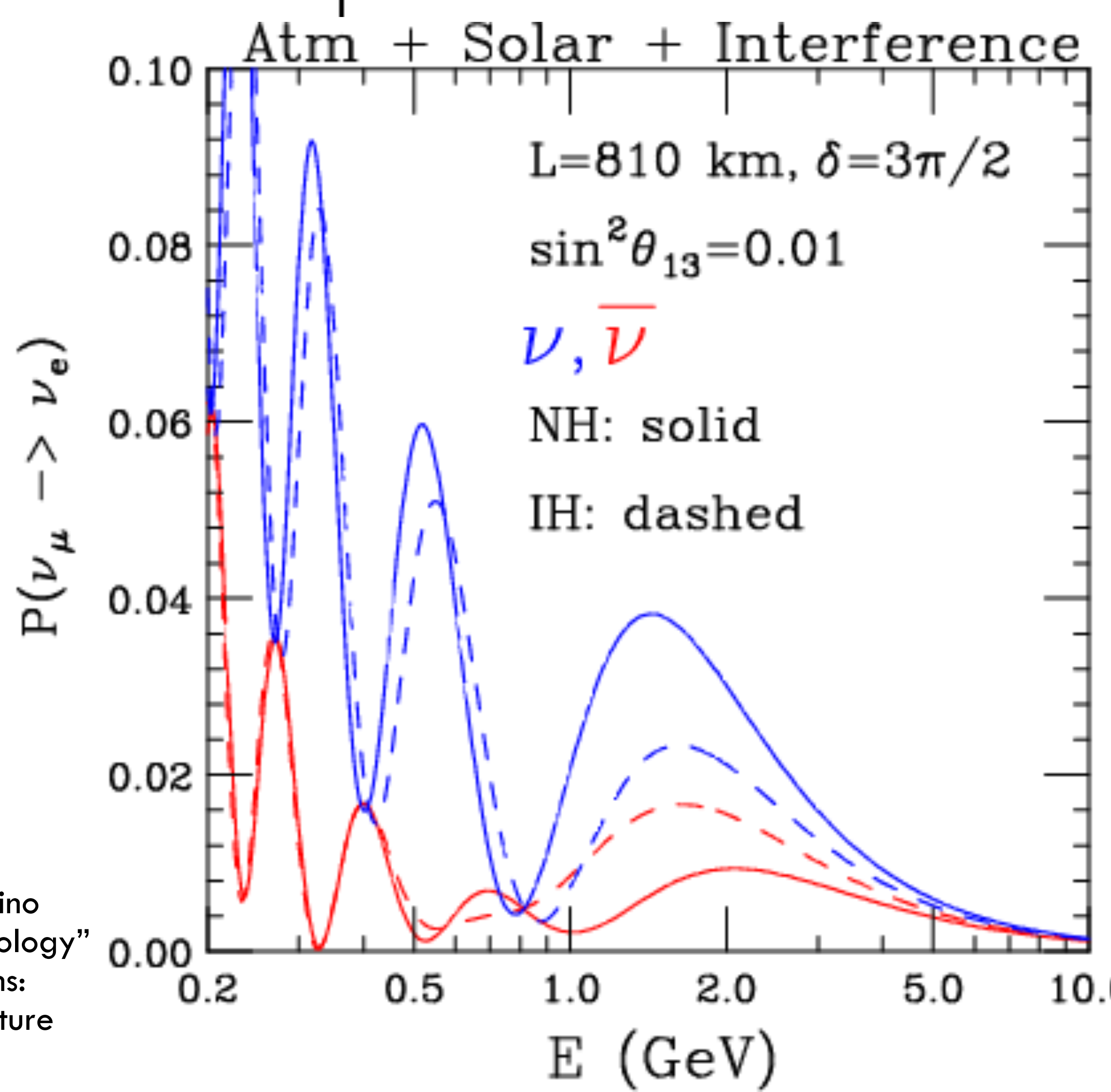


Why Deep Neural Networks?

- Measuring neutrino oscillations is all about measuring how neutrinos change between different lepton flavor states as a function of distance traveled and neutrino energy.

$$P(\nu_\mu \to \nu_e) \approx \left| \sqrt{P_{atm}} e^{-i\left(\frac{\Delta m^2_{32} L}{4E} + \delta_{cp}\right)} + \sqrt{P_{sol}} \right|^2$$

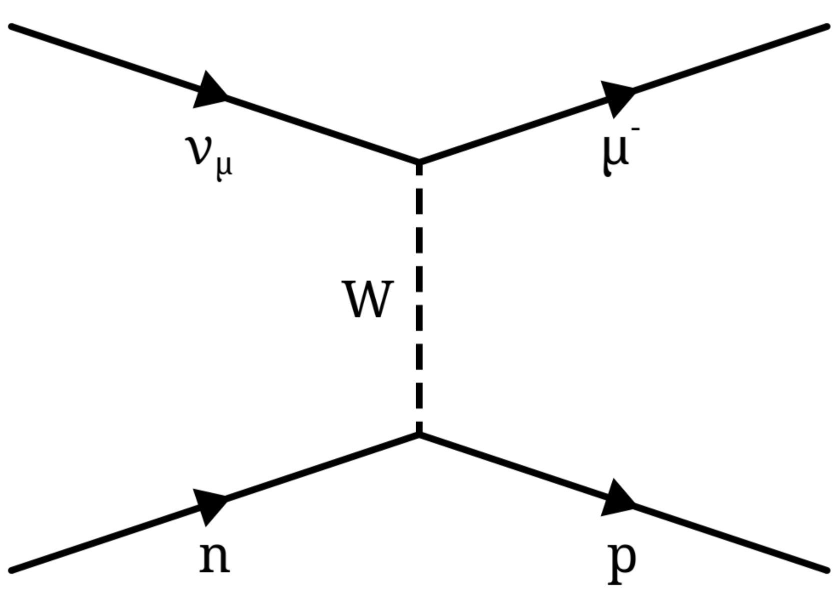
$$P_{atm} = \sin^2\theta_{23} \sin^2 2\theta_{13} \sin^2 \frac{\Delta m^2_{31} L}{4E}$$



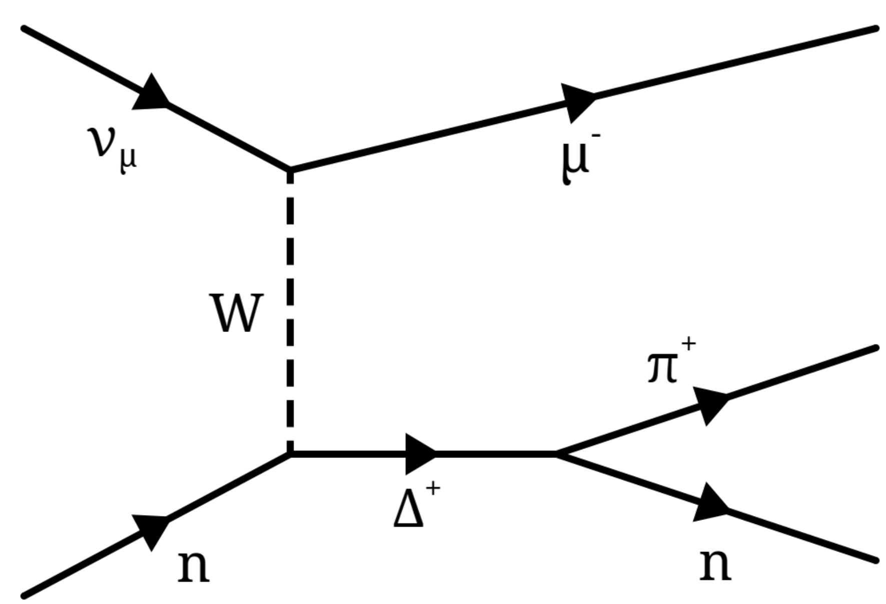
From S. Parke, "Neutrino Oscillation Phenomenology" in Neutrino Oscillations: Present Status and Future Plans

Why Deep Neural Networks?

- Any oscillation analysis can benefit from precise identification of the interaction in two ways:
  - Estimating the lepton flavor of the incoming neutrino.
  - Correctly identifying the type of neutrino interaction, to better estimate the neutrino energy, aka is it a quasi elastic event or a resonance event?



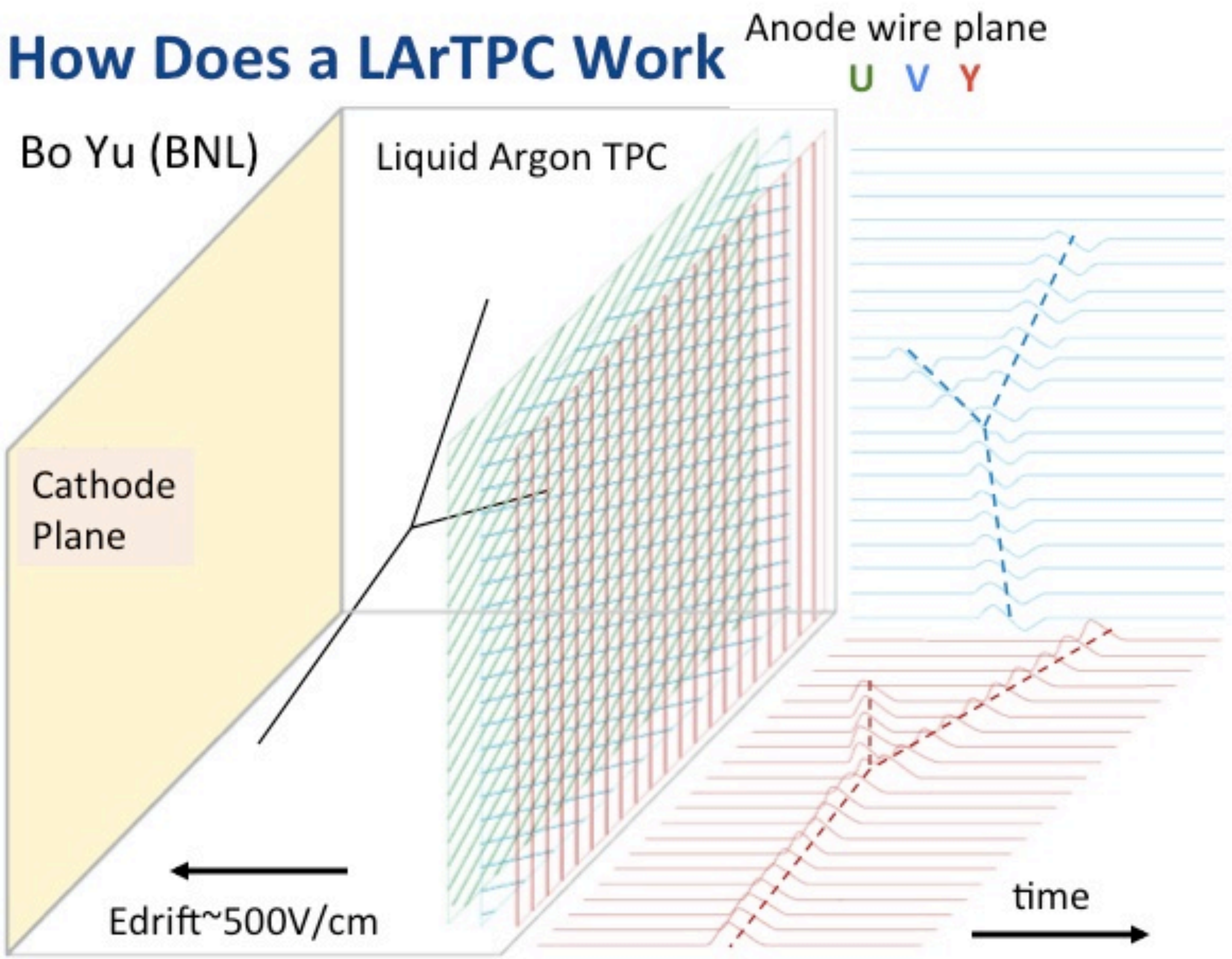
Quasi-Elastic



Resonance

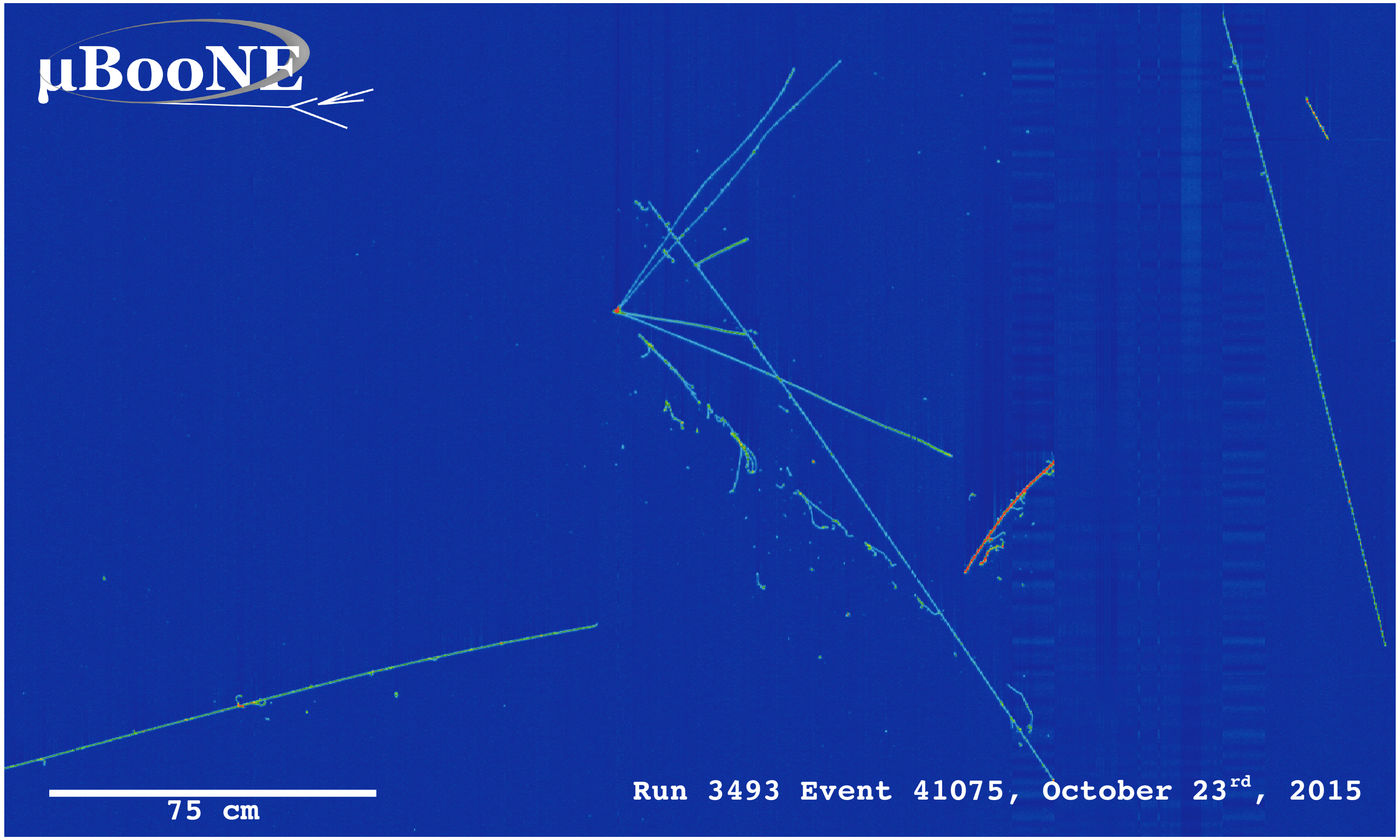
Why Deep Neural Networks?

- Liquid argon detectors are also the perfect domain:
  - Large ~uniform volumes where spatially invariant response is a benefit.
  - One, main, detector system.



Why Deep Neural Networks?

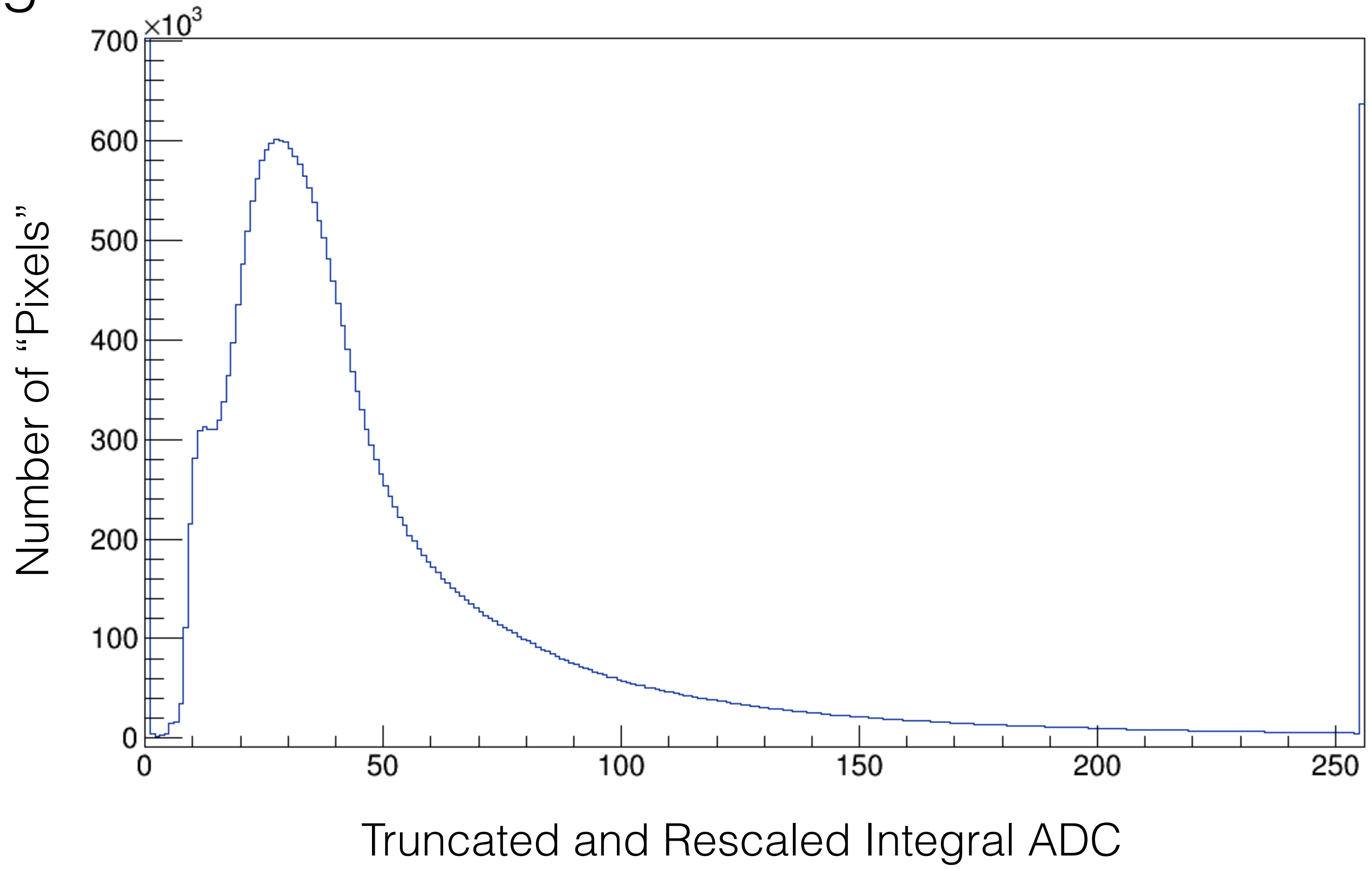
- Liquid argon detectors are also the perfect domain:
  - Large ~uniform volumes where spatially invariant response is a benefit.
  - One, main, detector system.



Deep Learning for Event Identification

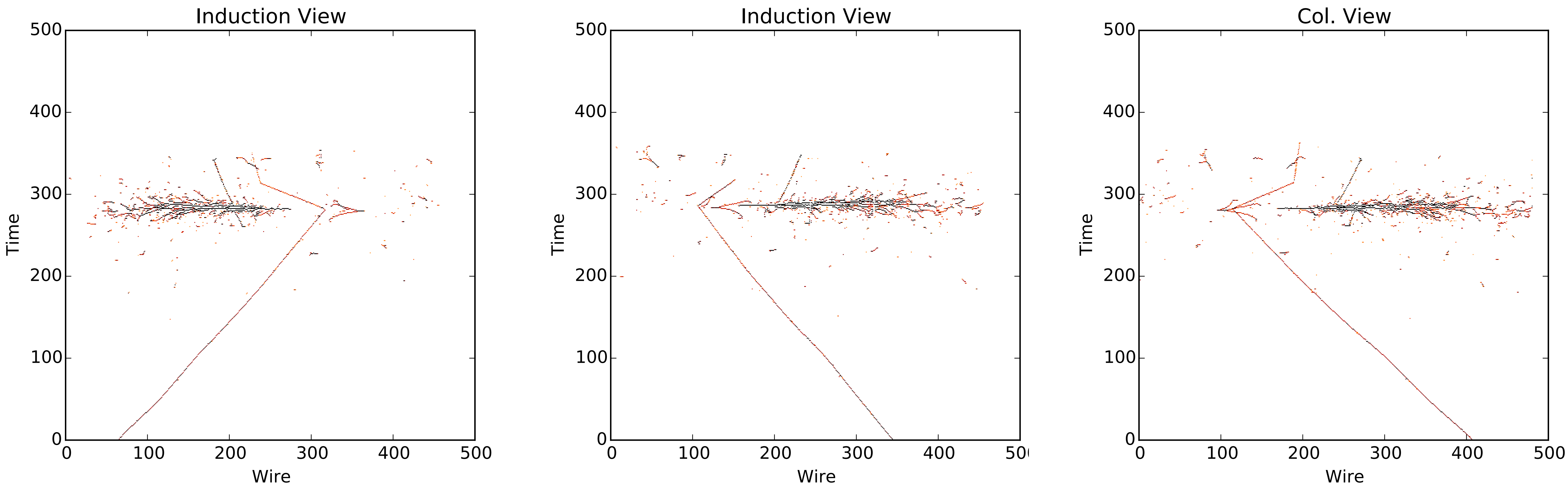
Our Input

- Each “pixel” is the integrated ADC response in that time/space slice.
- Rescale adc to go from 0 to 255 and truncate to chars for dramatically reduced file size at almost no loss of information. Final bin is overflow for >1000 adc.
- These maps are chosen to be 500 wires long and 1.2ms wide (split into 500 time chunks).
- At this stage we have a very simple object in the art files associated with each event, which we can keep for inference in ART later or export to a simple root ntuple for training



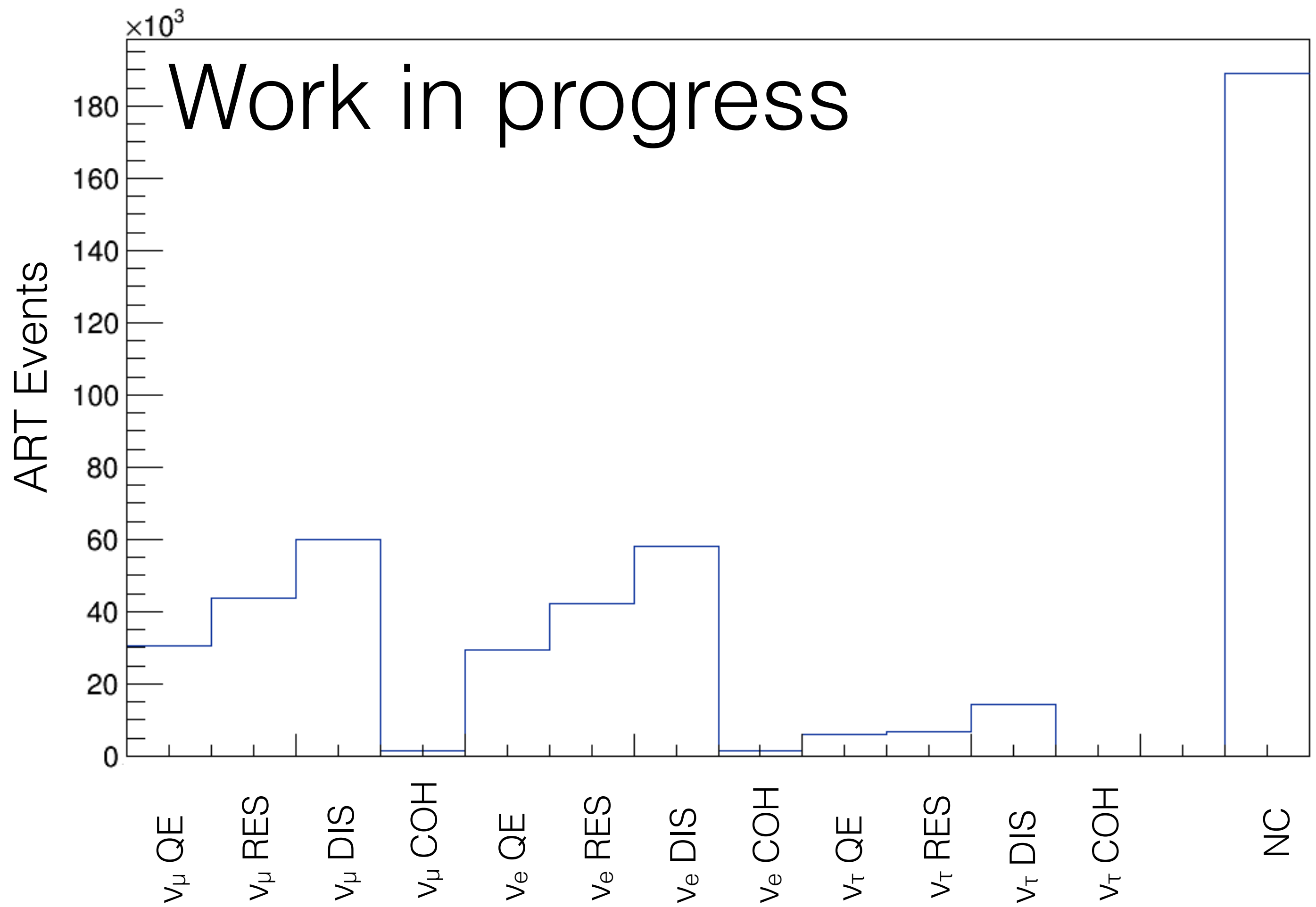
Our Input

- Each “pixel” is the integrated ADC response in that time/space slice.
- Rescale adc to go from 0 to 255 and truncate to chars for dramatically reduced file size at almost no loss of information. Final bin is overflow for >1000 adc.
- These maps are chosen to be 500 wires long and 1.2ms wide (split into 500 time chunks).
- At this stage we have a very simple object in the art files associated with each event, which we can keep for inference in ART later or export to a simple root ntuple for training



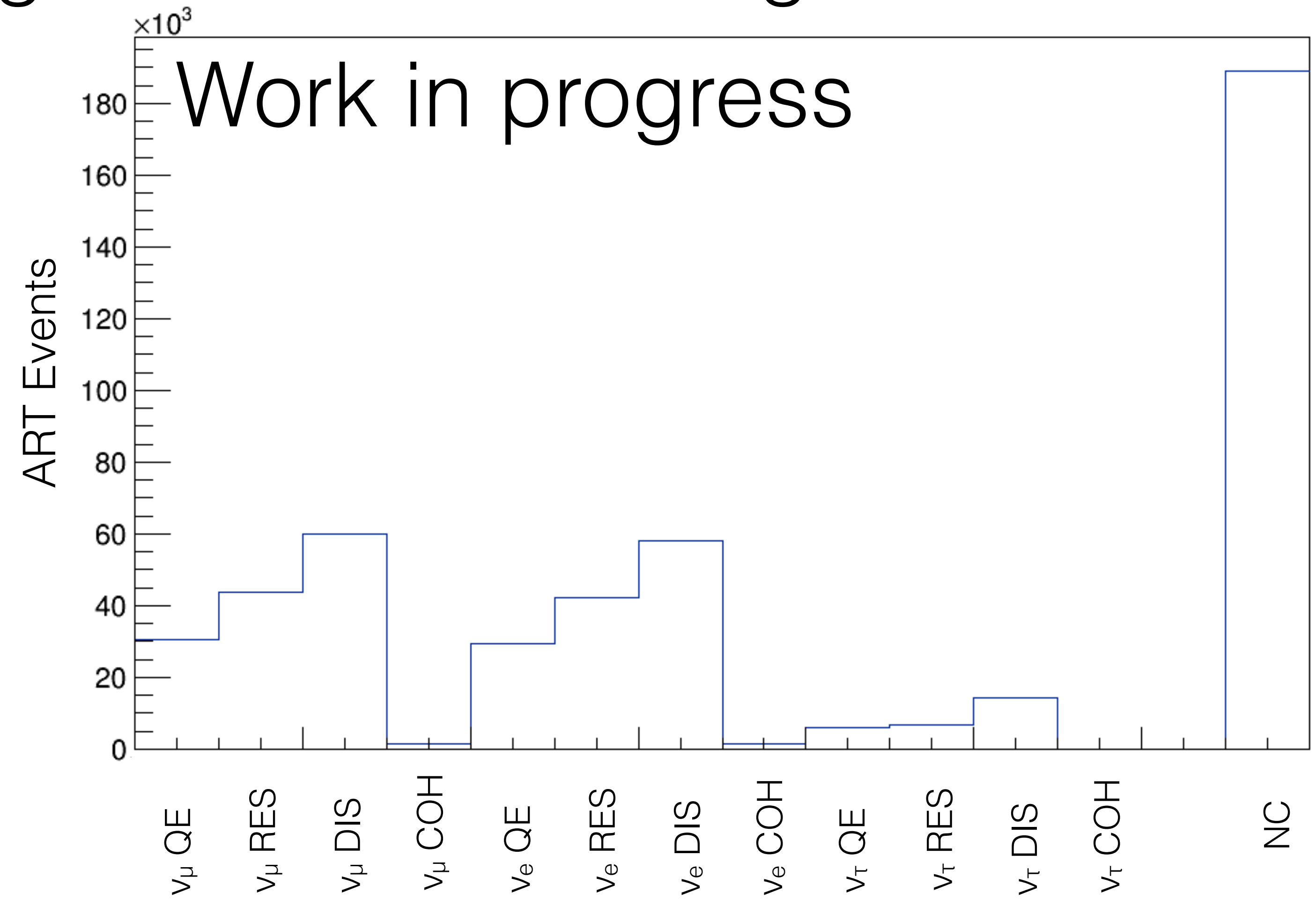
The Training Sample

- 1.2M events, only preselection requiring 100 hits split across any number of planes.
- Labels are from GENIE truth, neutrino vs. antineutrino is ignored.
- No oscillation information, just the raw input distributions.
- 80% for training and 20% for testing.
- Currently we store this sample in levelDBs



The Training Sample

- 1.2M events, only preselection requiring 100 hits split across any number of planes.
- Labels are from GENIE truth, neutrino vs. antineutrino is ignored.
- No oscillation information, just the raw input distributions.
- 80% for training and 20% for testing.

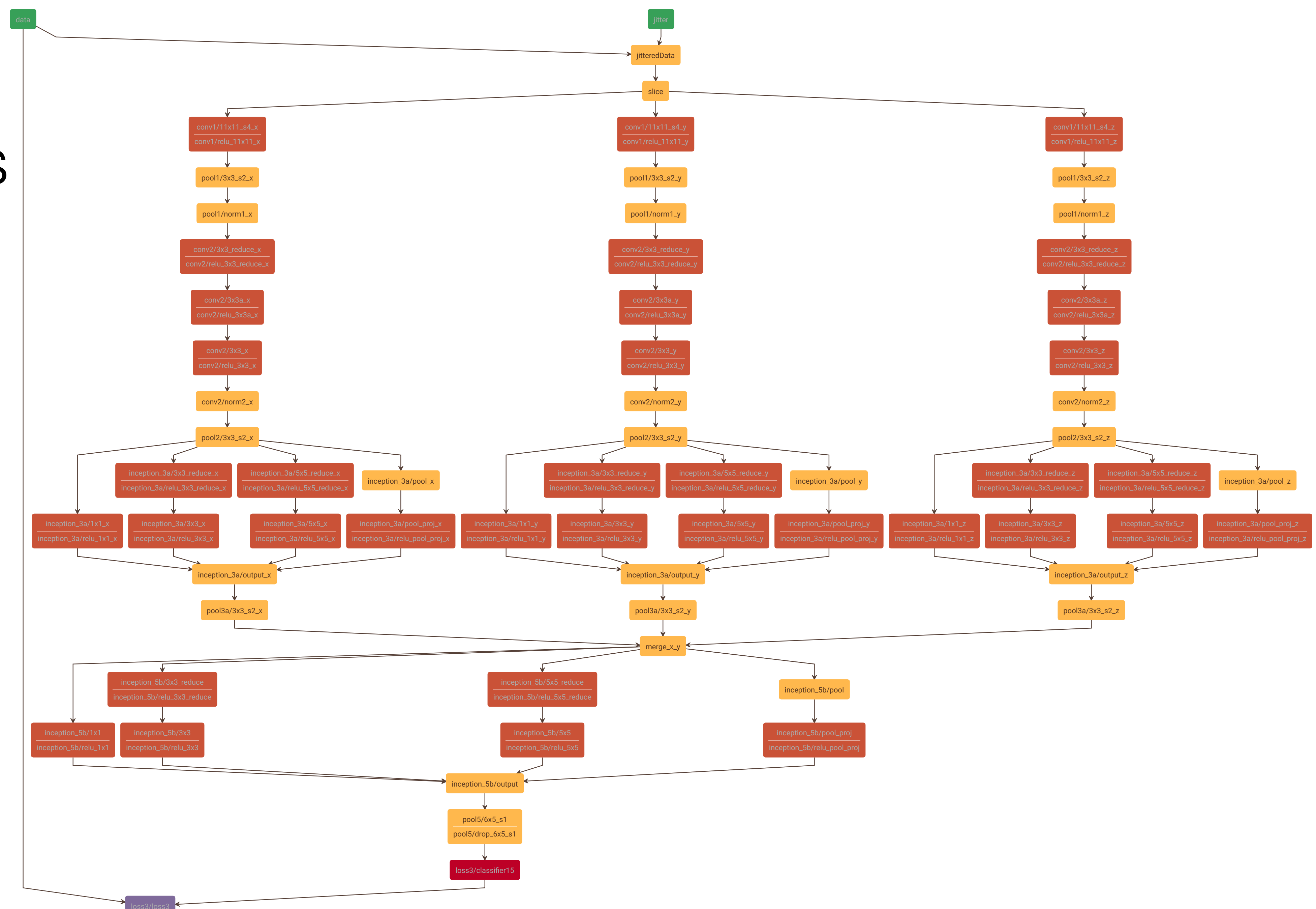


Our Architecture

Based on the NOvA CNN, named **CVN**. Small edits to better suit a larger input image and three distinct views.

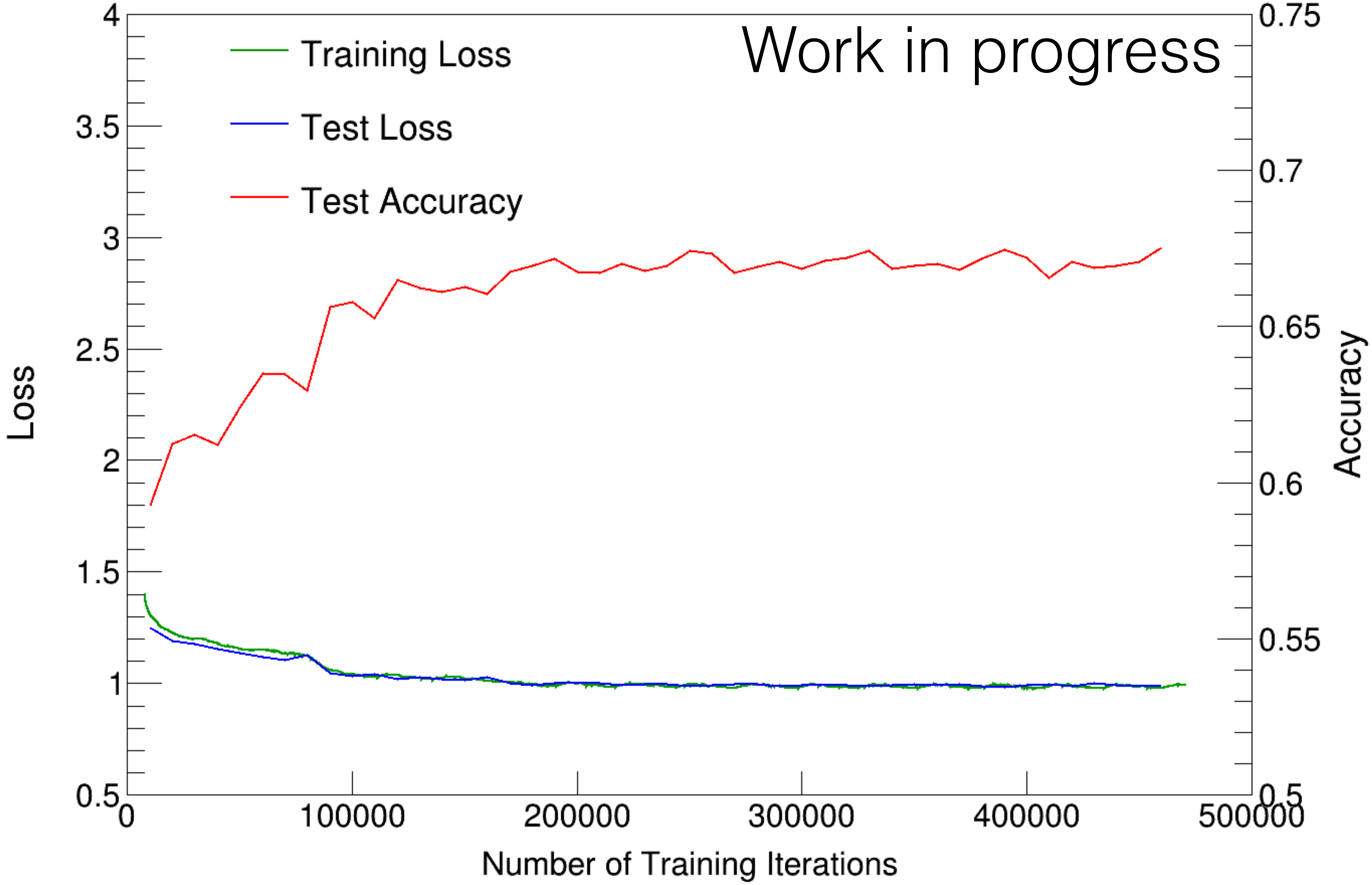
The architecture attempts to categorize events as $\{\nu_\mu, \nu_e, \nu_\tau\} \times \{QE, RES, DIS\}$, NC.

Built and trained the excellent CAFFE framework, latest stable release.

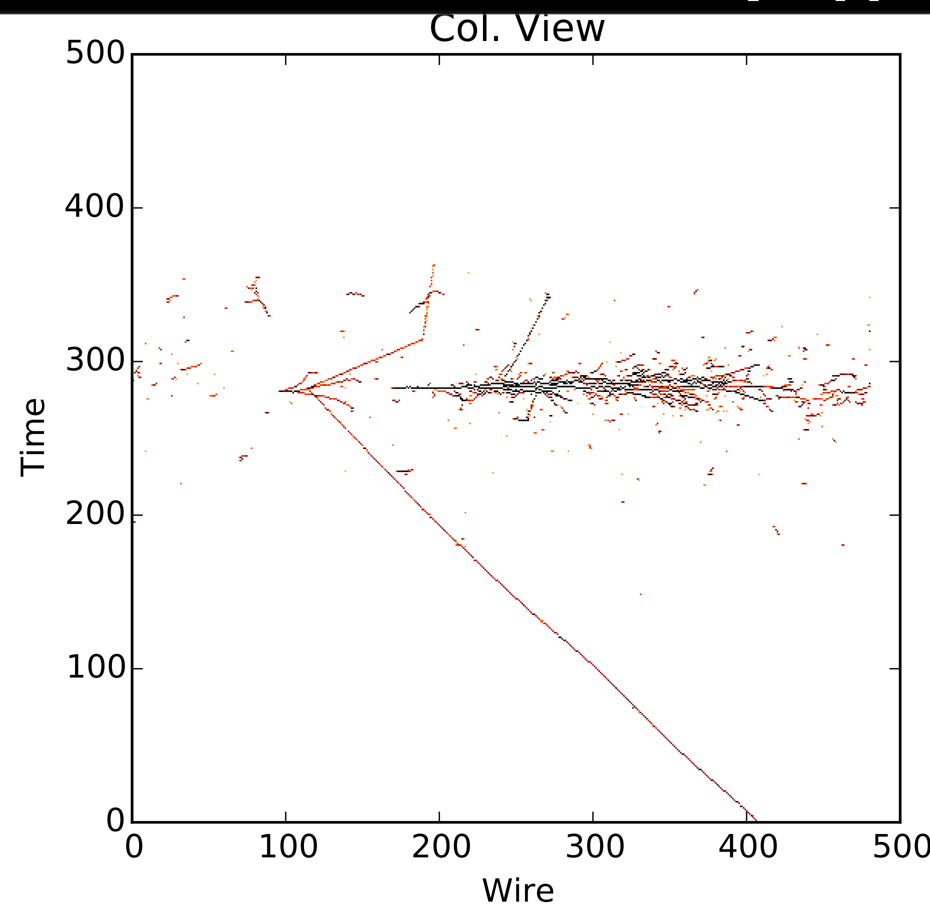


Training Performance

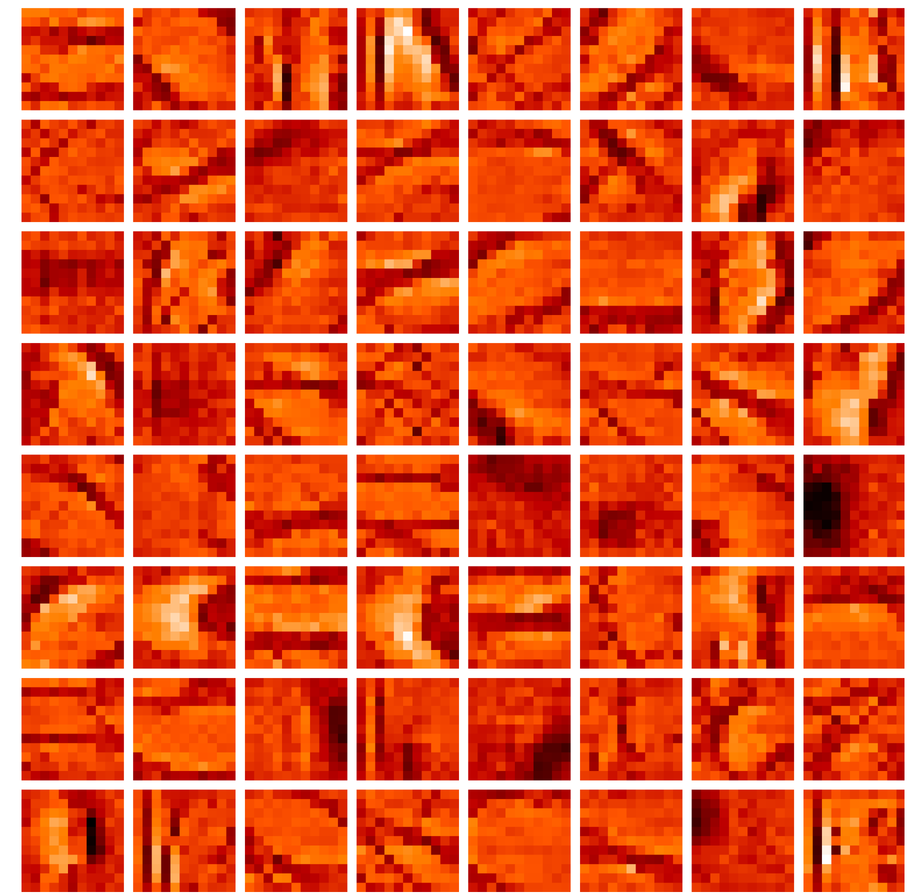
No sign of overtraining- exceptional training test set performance agreement!



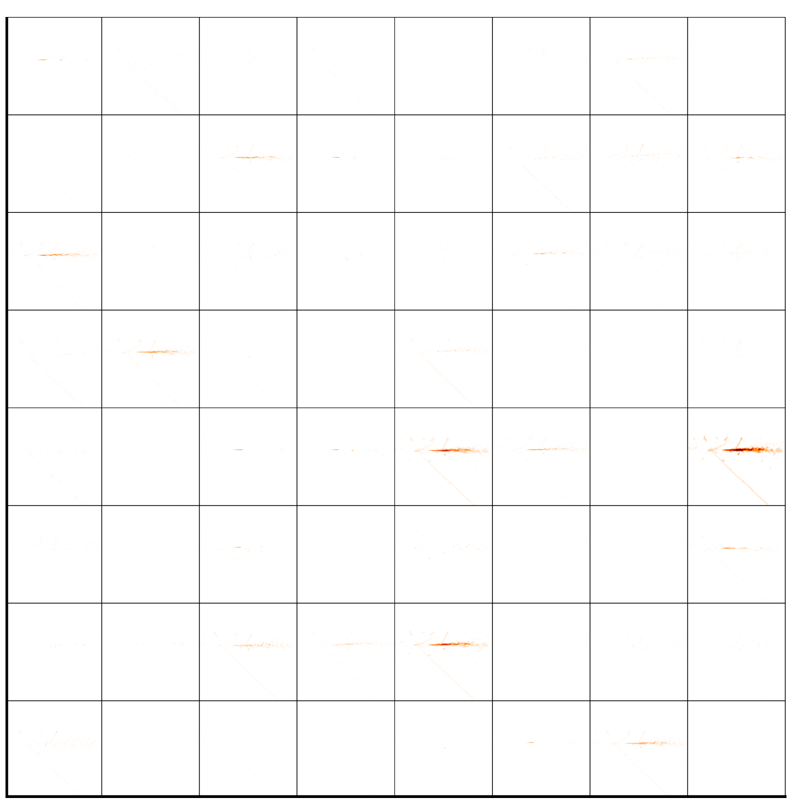
Example CVN Kernels In Action: First Convolution



X



=



Here the earliest convolutional layer in the network starts by pulling out primitive shapes and lines.

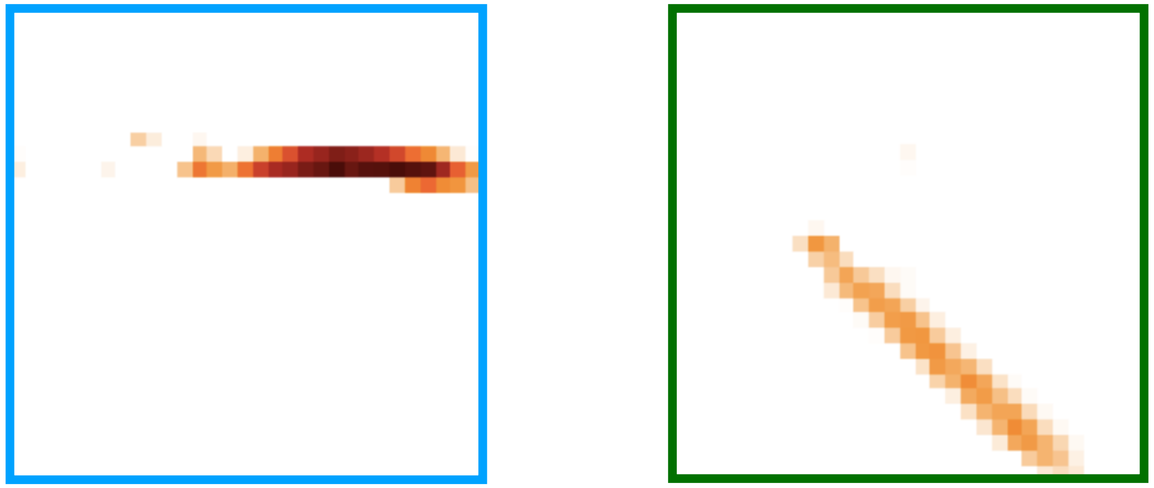
Already “showers” and “tracks” are starting to form.

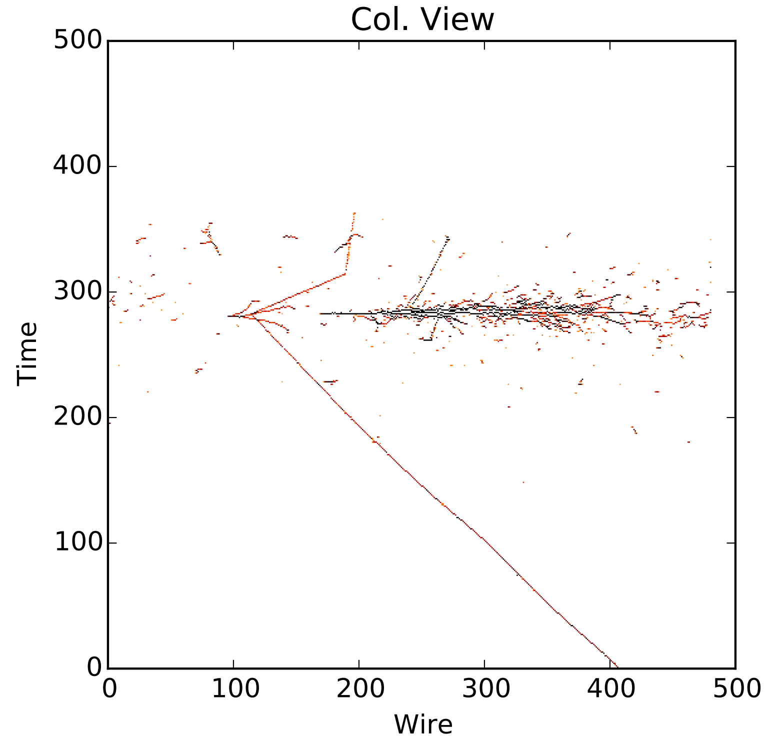
Example CVN Kernels In Action: First Inception Module Output

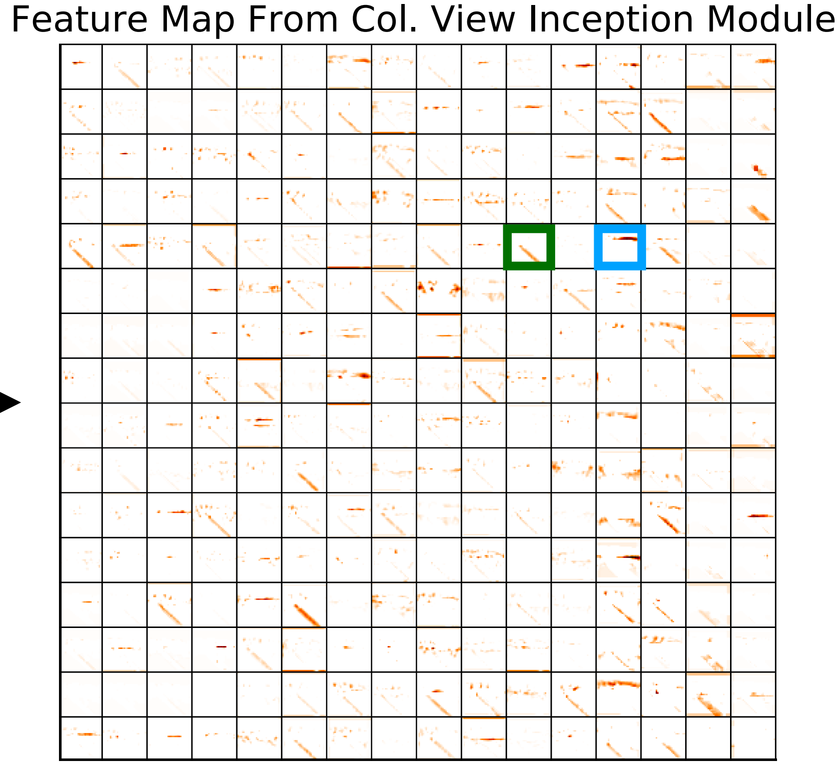
Deeper in the network, now after the first inception module we can see more complex features have started to be extracted.

Some seem particularly sensitive to muon tracks, EM showers.

True NuMu DIS Event





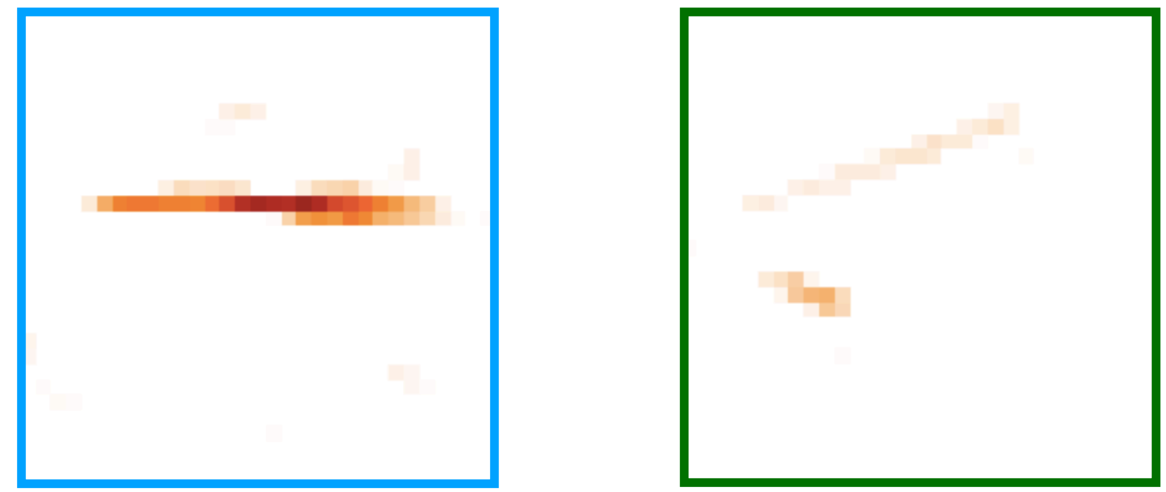


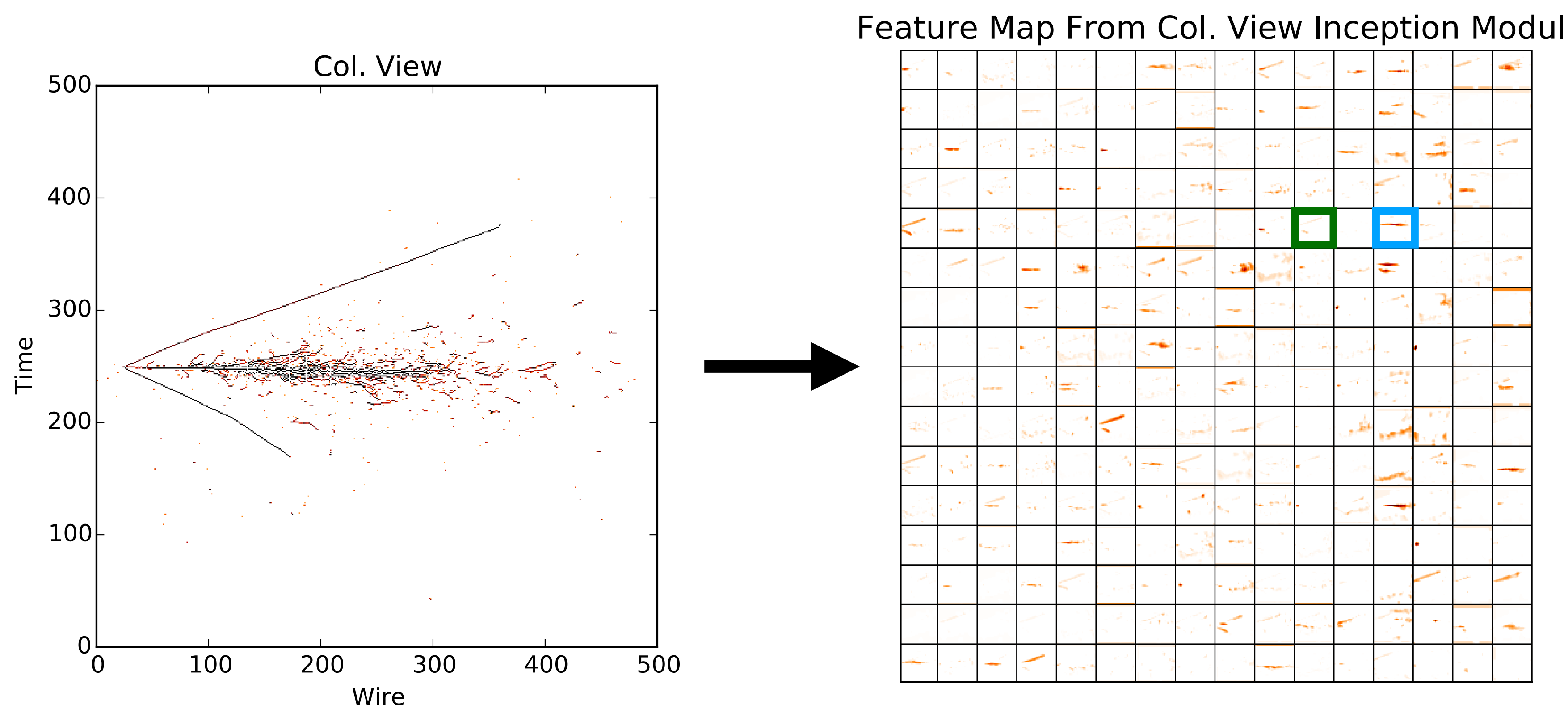
Example CVN Kernels In Action: First Inception Module Output

Deeper in the network, now after the first inception module we can see more complex features have started to be extracted.

Some seem particularly sensitive to muon tracks, EM showers.

True NuE COH Event

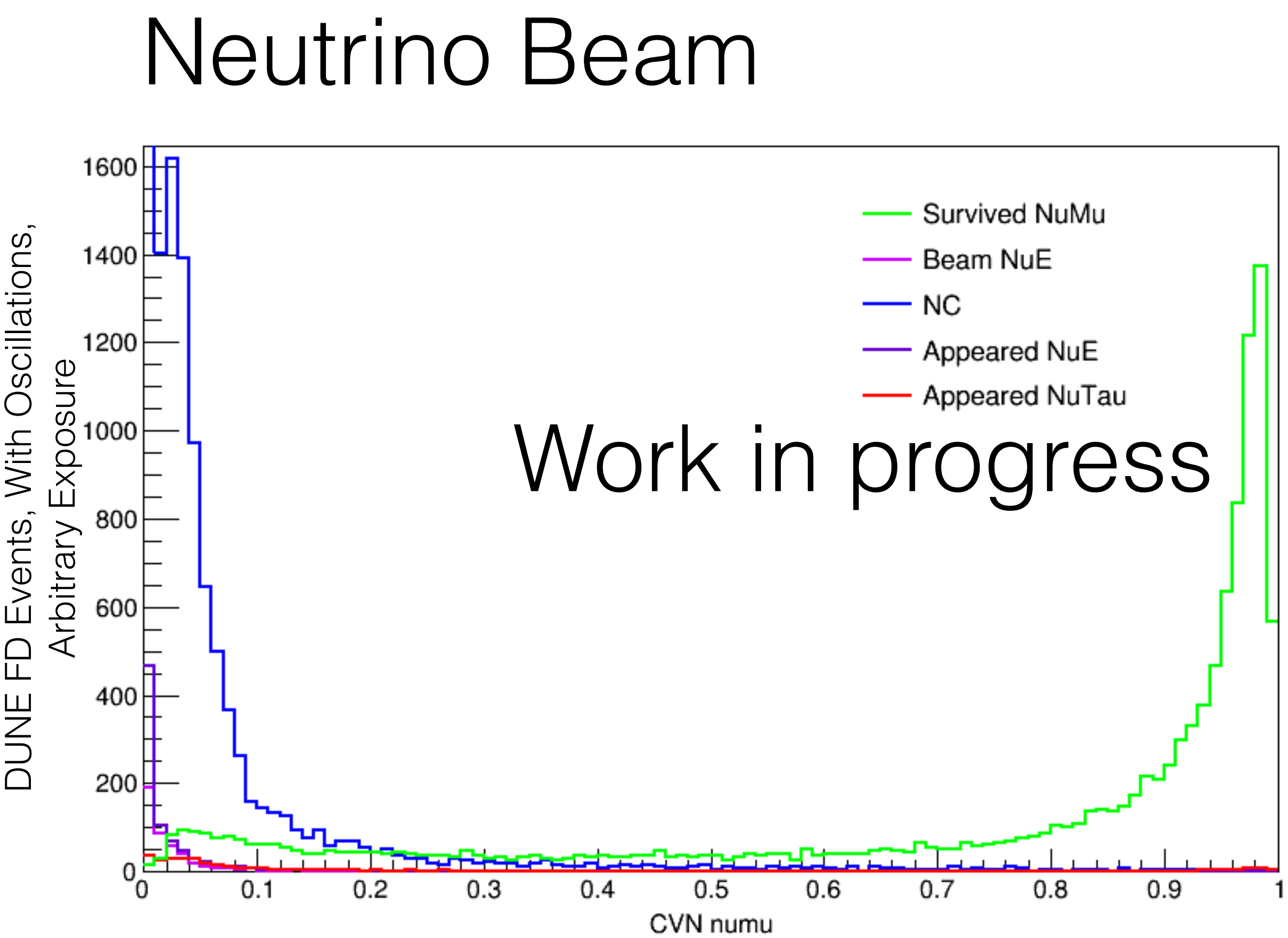


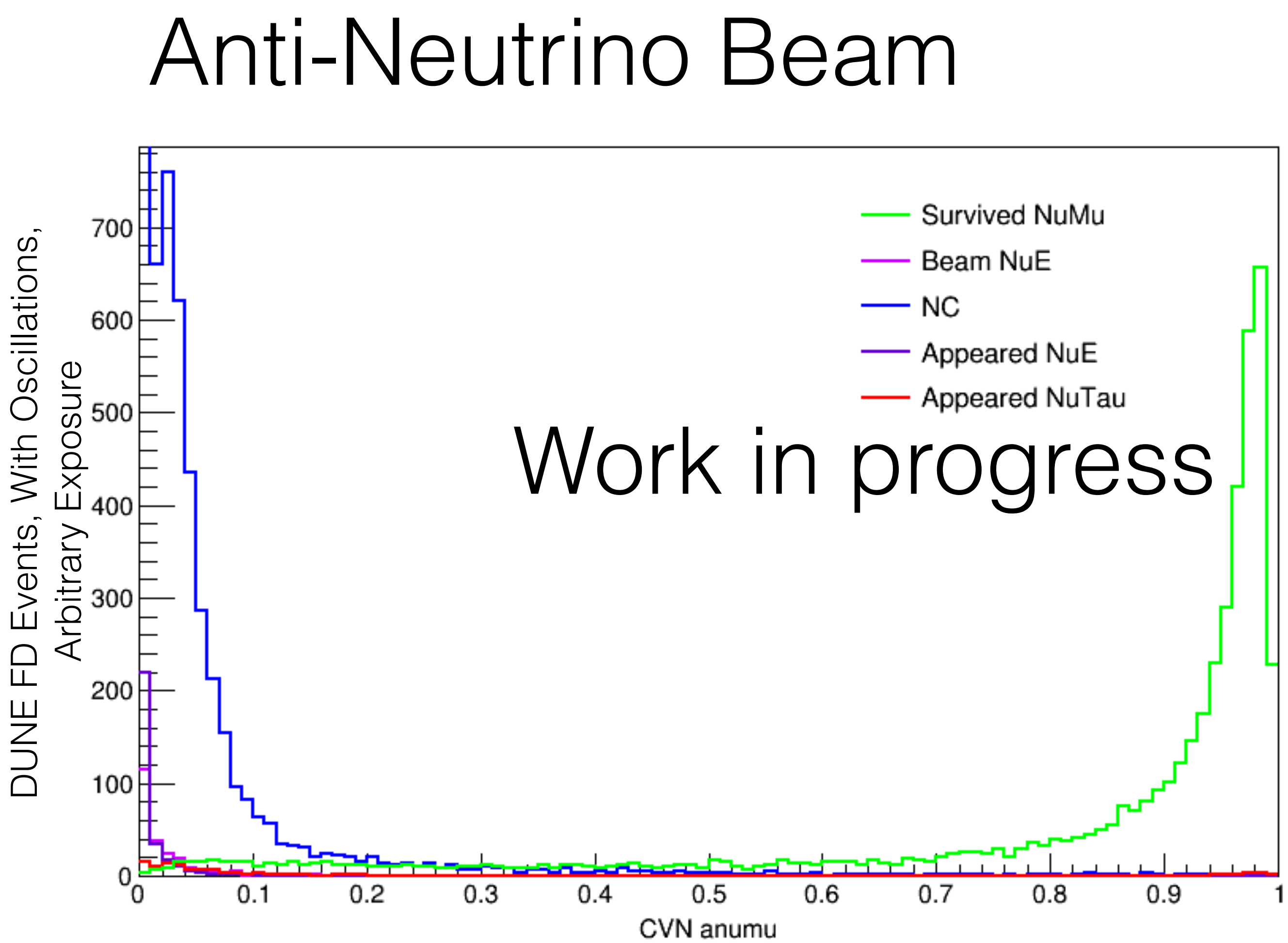


Inference

- After training the network using the caffe command line interface, we test its basic behavior using an array of pycaffe scripts.
- Then we run inference as part of a regular ART module, calling the caffe c++ API:
  - https://cdcvs.fnal.gov/redmine/projects/dunetpc/repository/revisions/develop/entry/dune/CVN/art/CaffeNetHandler.cxx
- It's much slower on a CPU, but still fast compared to most Liquid Argon Reconstruction, and the power of grid computing makes a huge difference.
- This makes it relatively easy to directly compare to more traditional tools, and to take advantage of the existing analysis framework.

NuMu PID

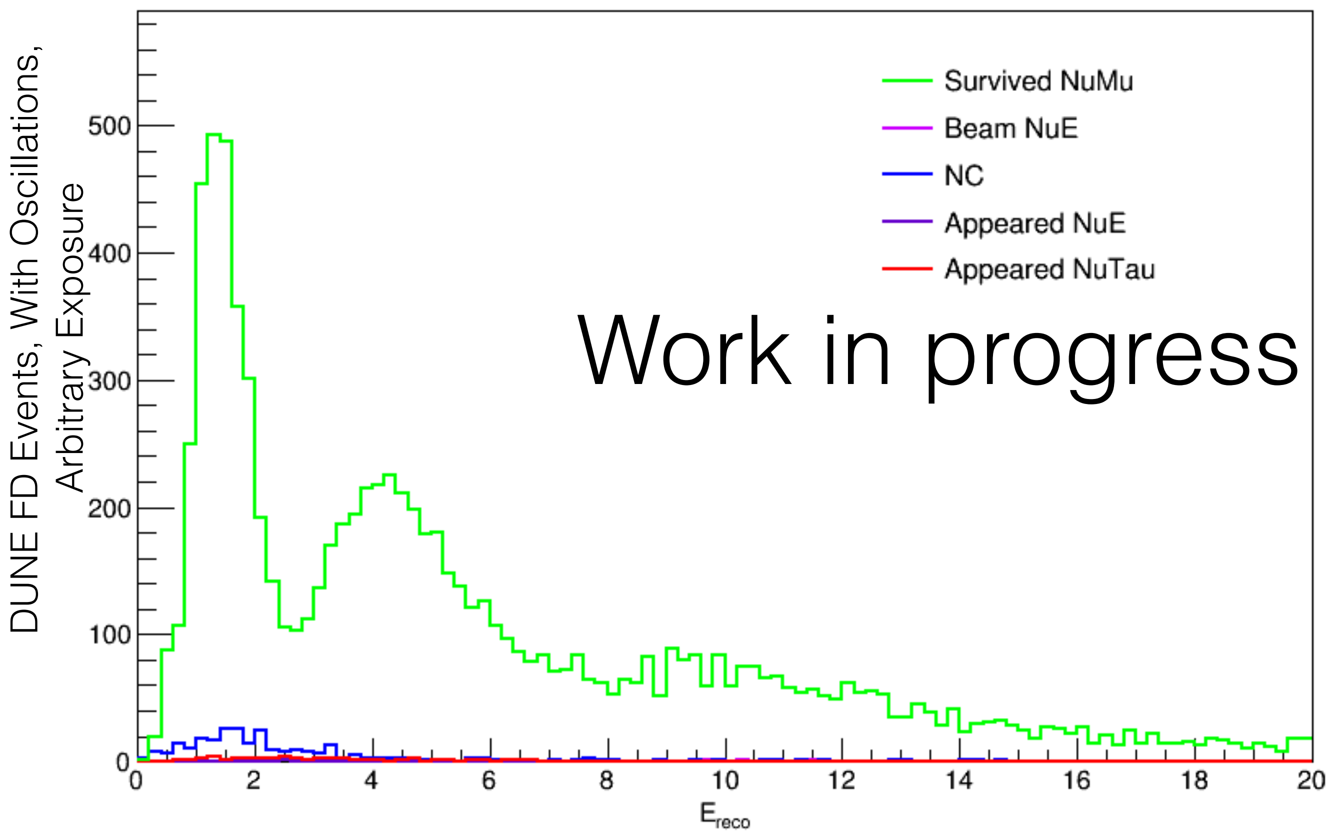




Cut at 0.5, guarantees no double counting due to sofmax output of CVN

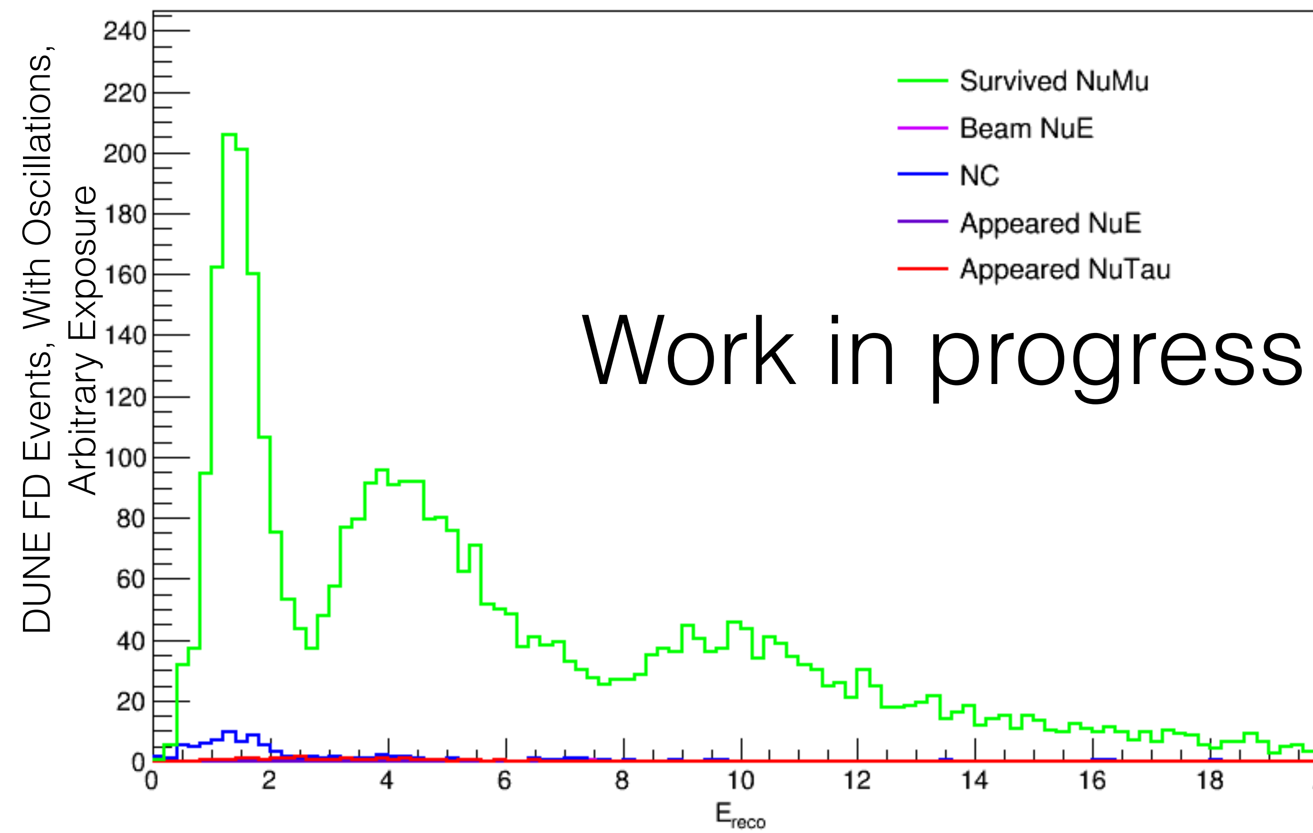
NuMu Selected Events, Reconstructed Energy Spectra

Neutrino Beam



	NuMu	Appeared NuE	Beam NuE	NC	NuTau
Efficiency	**80.6**				
Rejection		99.0	98.7	**97.6**	81.5

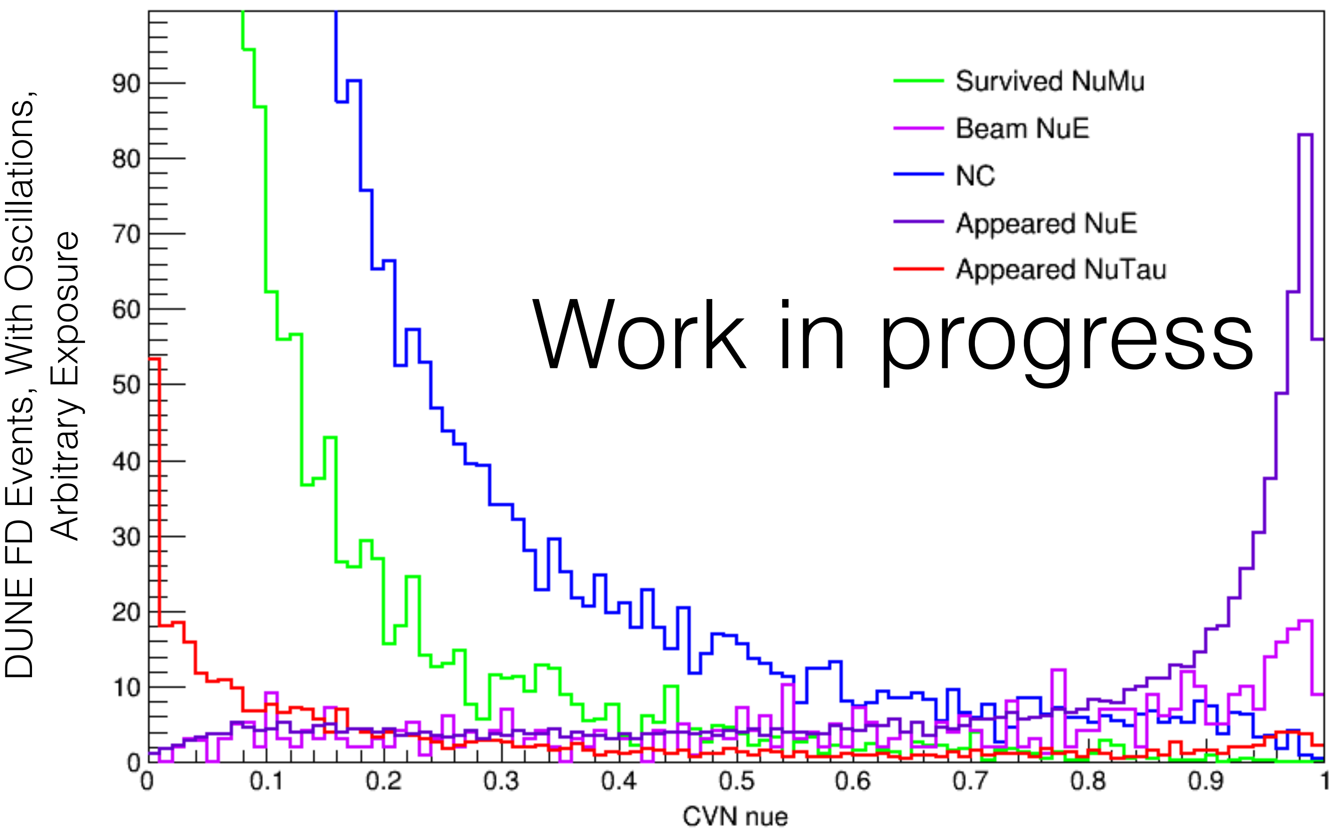
Anti-Neutrino Beam



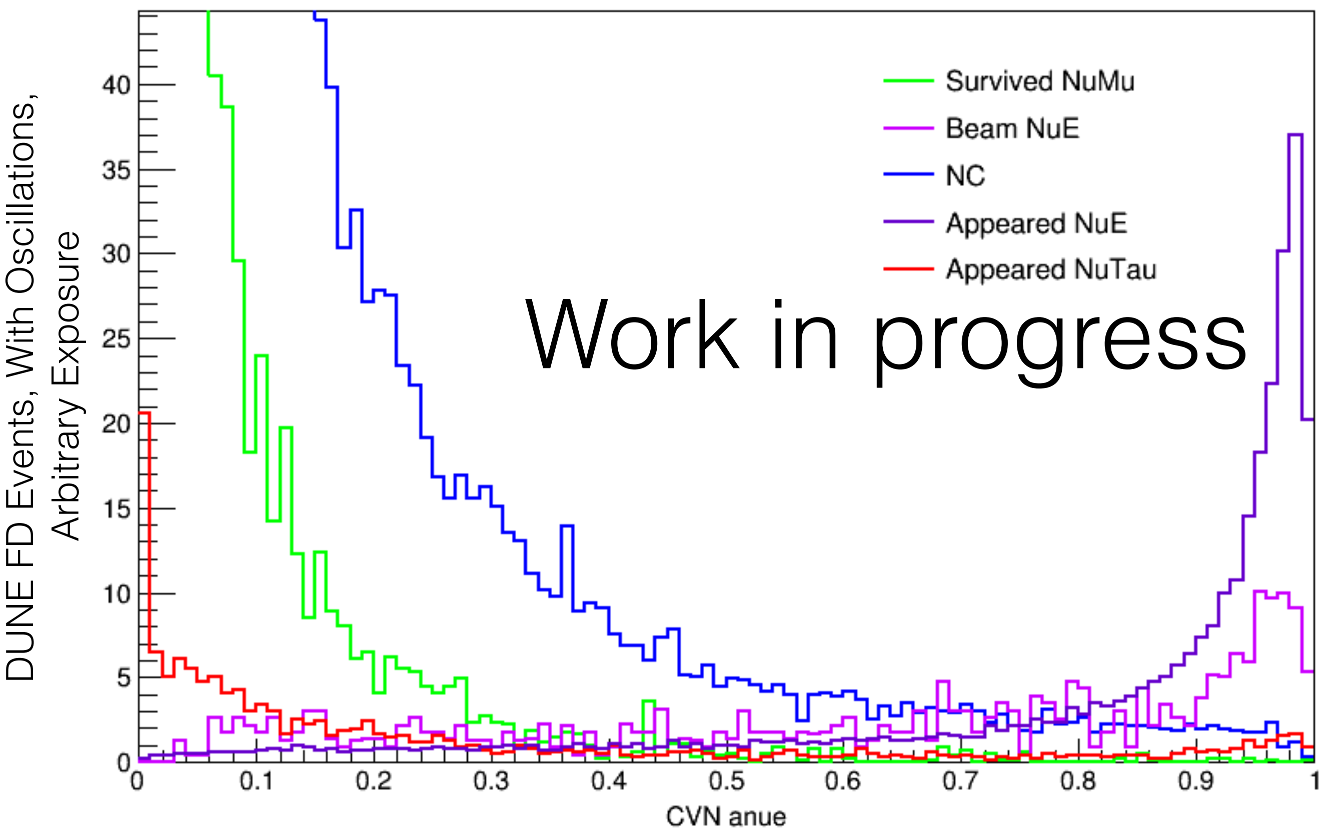
	NuMu	Appeared NuE	Beam NuE	NC	NuTau
Efficiency	**87.7**				
Rejection		99.6	99.3	**98.3**	81.4

NuE PID

Neutrino Beam



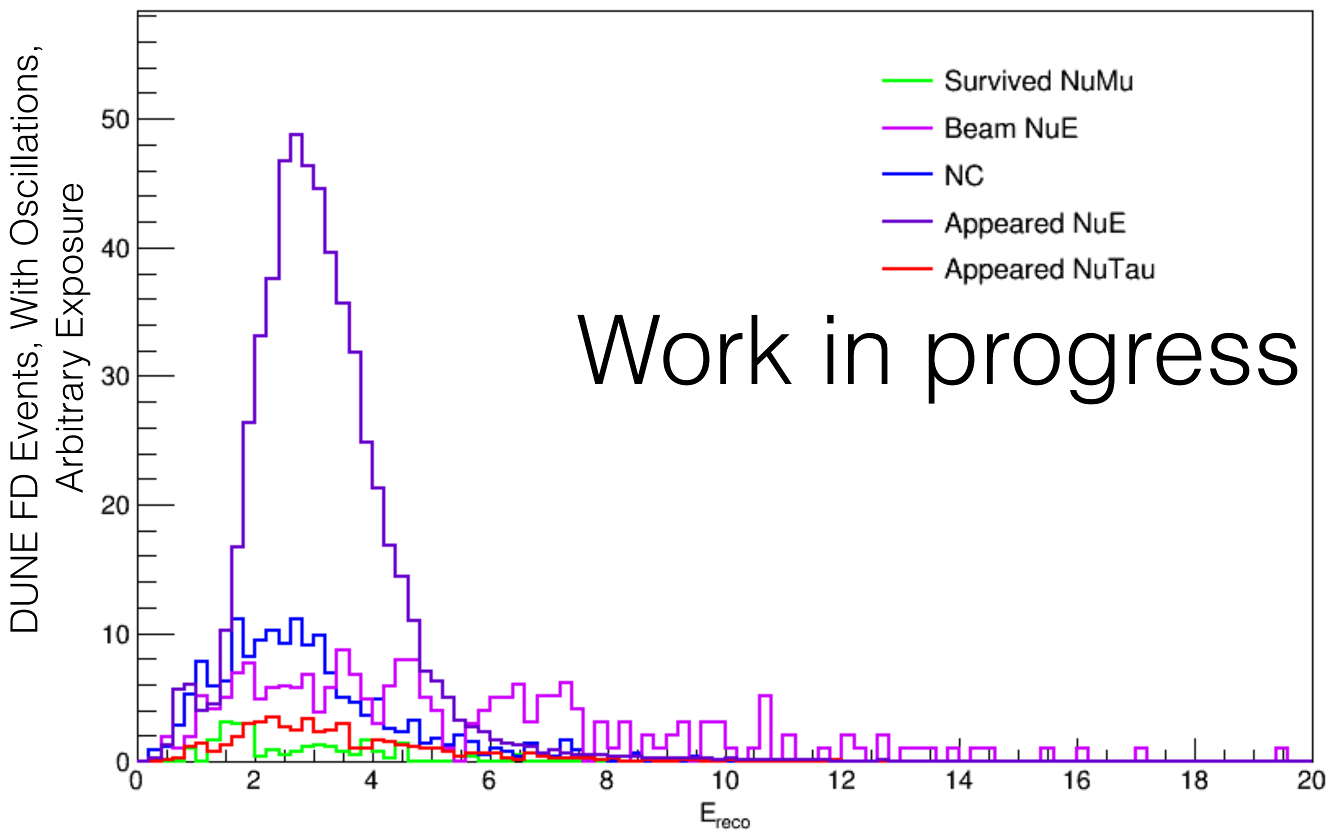
Anti-Neutrino Beam



Cut at 0.8, optimized for S/Sqrt(S+B)

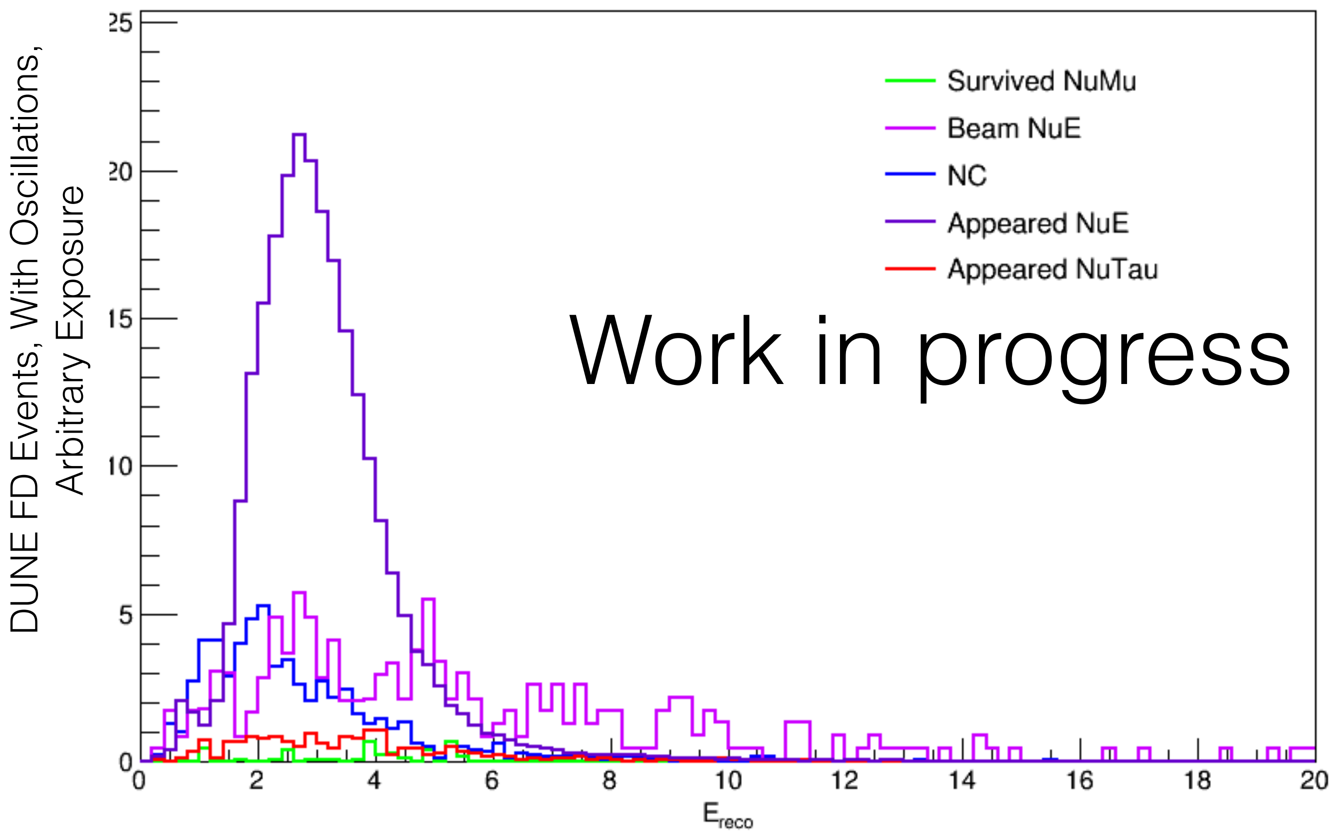
NuE Selected Events, Reconstructed Energy Spectra

Neutrino Beam



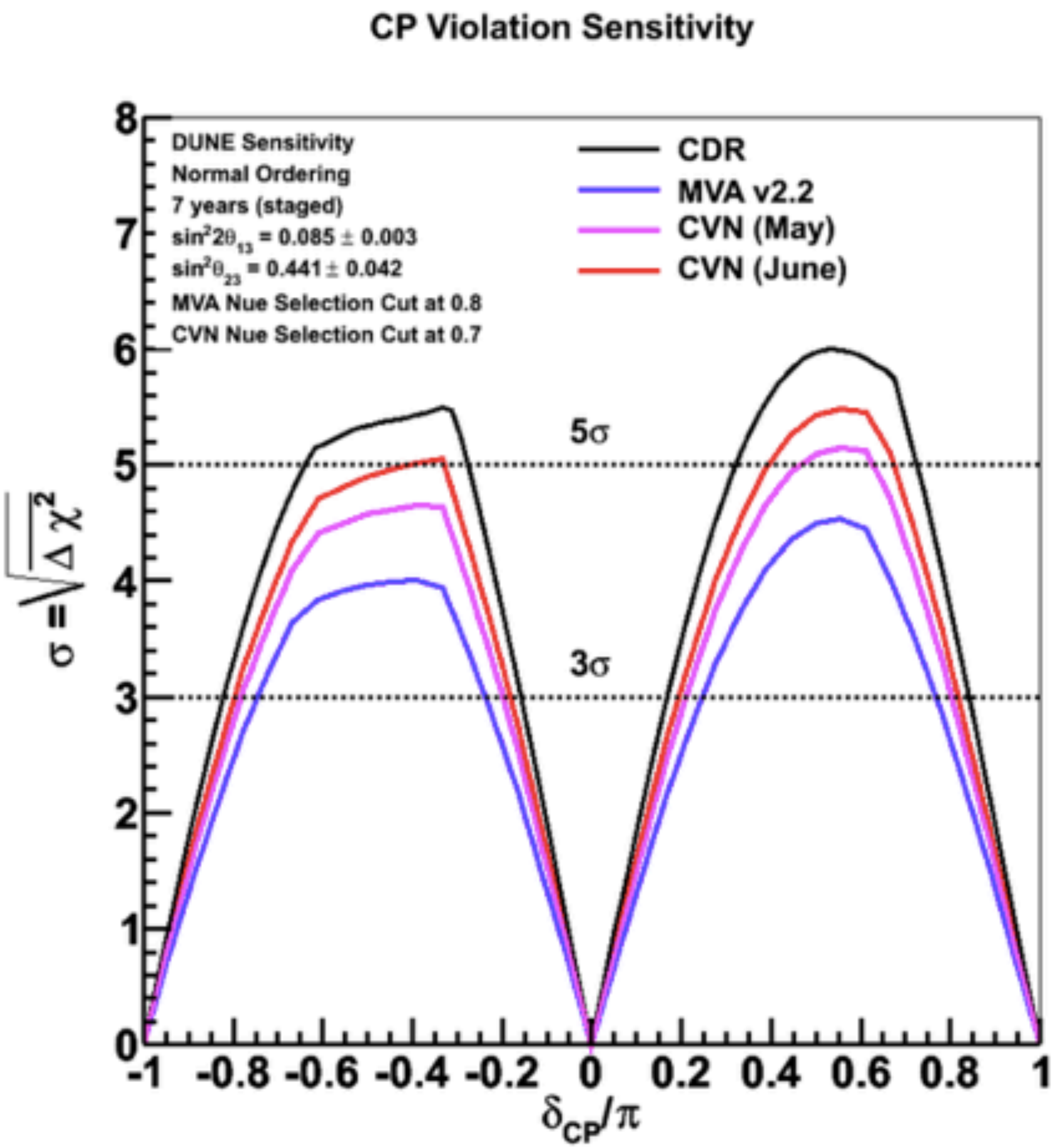
	Appeared NuE	NuMu	Beam NuE	NC	NuTau
Efficiency	**67.5**				
Rejection		99.8	52.1	**98.6**	85.8

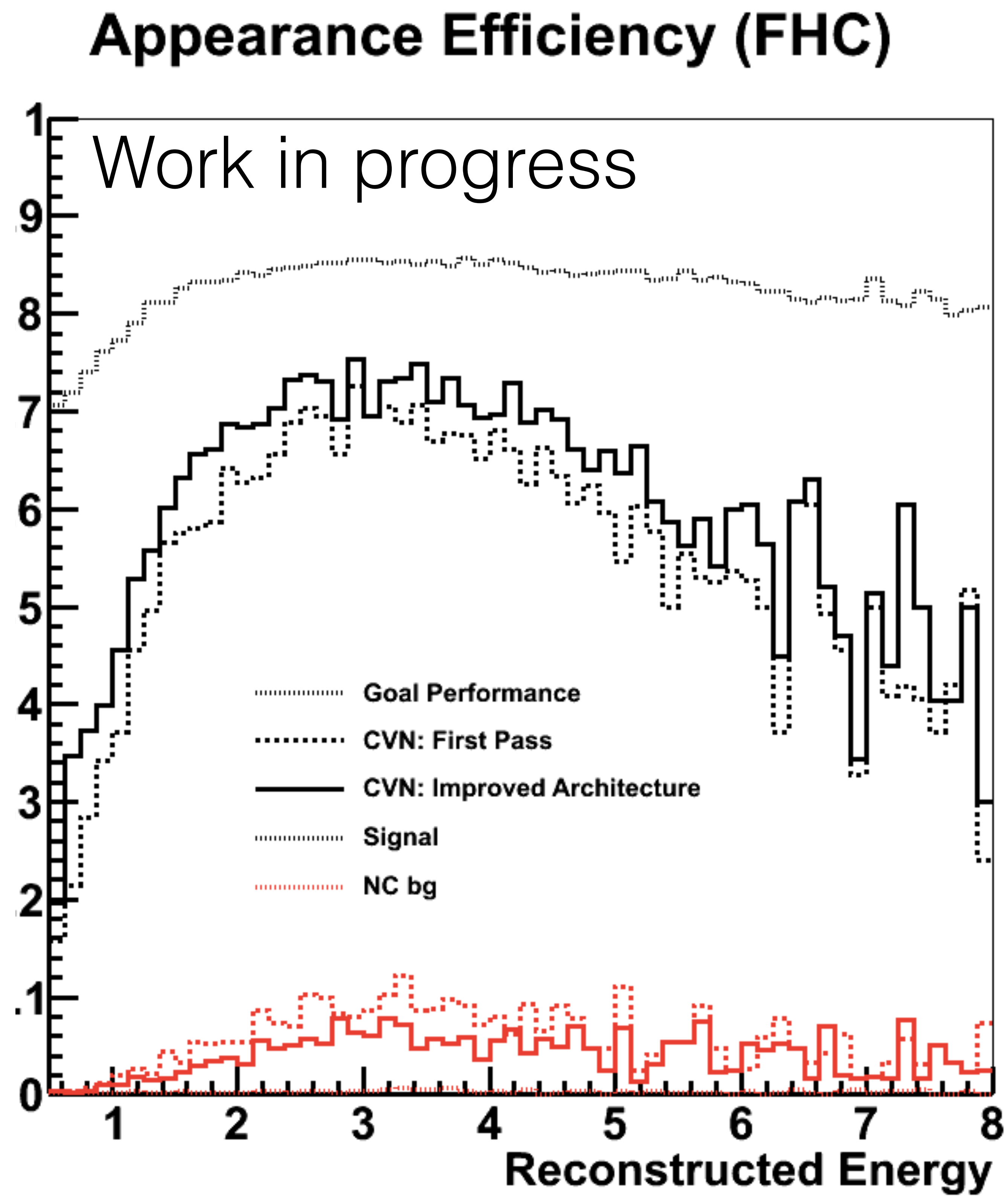
Anti-Neutrino Beam



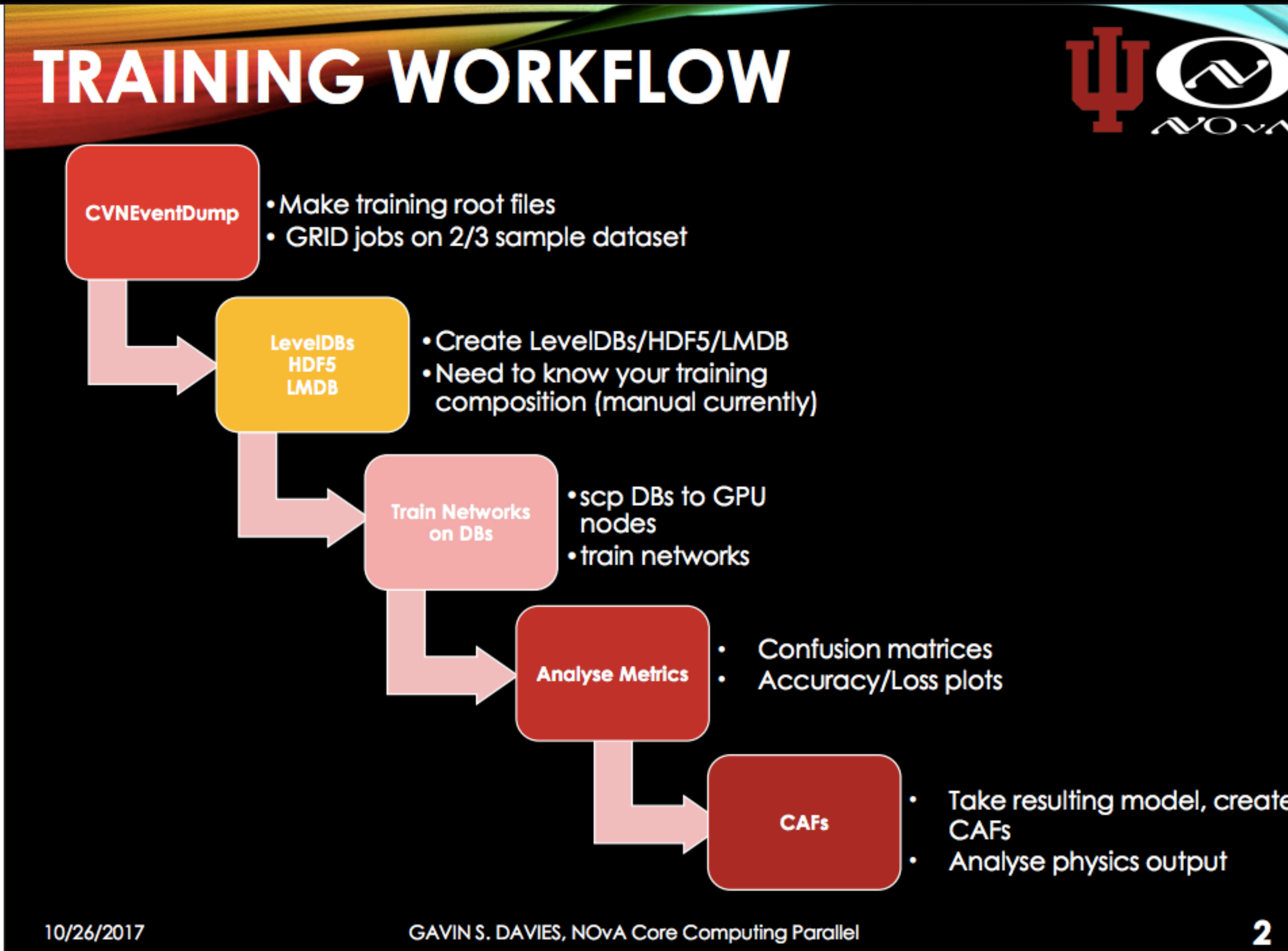
	Appeared NuE	NuMu	Beam NuE	NC	NuTau
Efficiency	**79.3**				
Rejection		99.9	48.2	**98.8**	87.6

The Bottom Line





Workflow Summary



Conclusions

- It works!
- Pros:
  - The Caffe c++ API makes it relatively painless to include in regular production, as longs as you have a valid UPS product (thanks Evan Niner & Lin Garen).
  - Caffe's model zoo has lots of great network examples, hugely helpful when getting started.
- Cons:
  - Awkward workflow for getting training samples out,
  - Hard to extend current file format beyond CNN for ID.
  - Net result is it slowed R&D.