

Using Machine Learning to Improve the Performance of the Cosmic Ray Veto

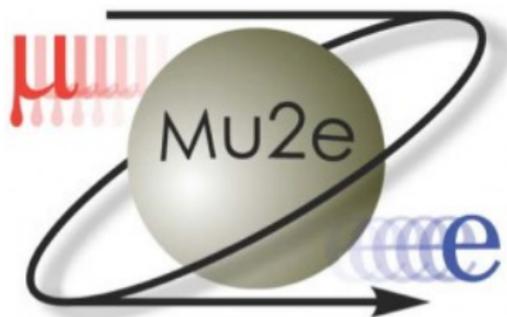
Mohit V. Srivastav

Undergraduate Physics & Computer Science Student at the University of Virginia
For the Mu2e Collaboration

mvs2dq@virginia.edu

July 14, 2021

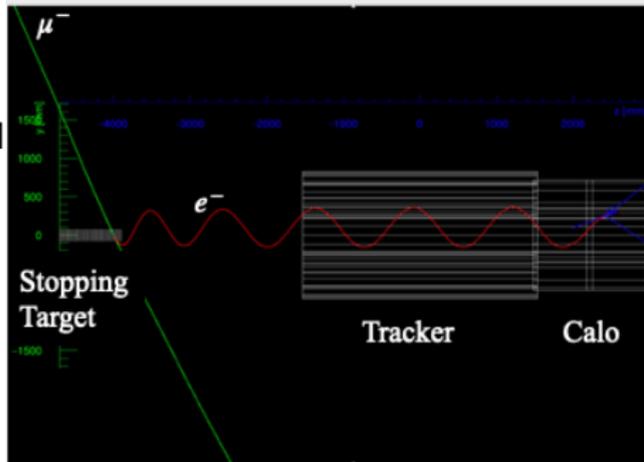
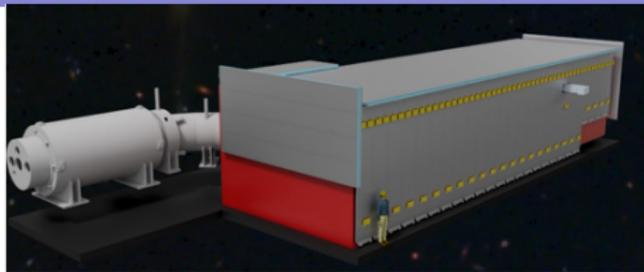
The Mu2e Experiment



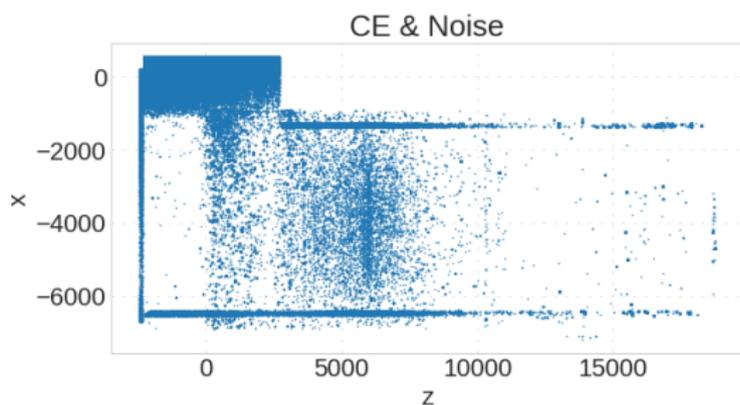
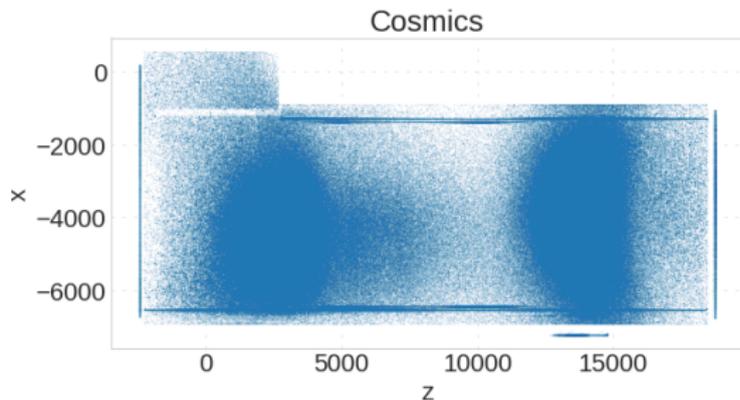
- Consistent of multiple institutions across the globe centered at Fermilab
- Looking for neutrinoless conversion of a muon to an electron
- If found, it would be evidence of new physics beyond the Standard Model

The Cosmic Ray Veto

- On average, cosmic-ray muons will produce 1 event/day that cannot be distinguished from a real conversion electron
- The CRV identifies and rejects cosmic ray muons that produce conversion-like backgrounds during offline analysis
- The CRV is consistent of 4 layers of scintillator counters, read out with SiPM photodetectors
- A track stub in 3/4 of the layers localized in space/time produces a "veto", or rejection of the event
- To reduce the number of background events < 0.1 of all events requires a muon detection efficiency ~ 0.9999

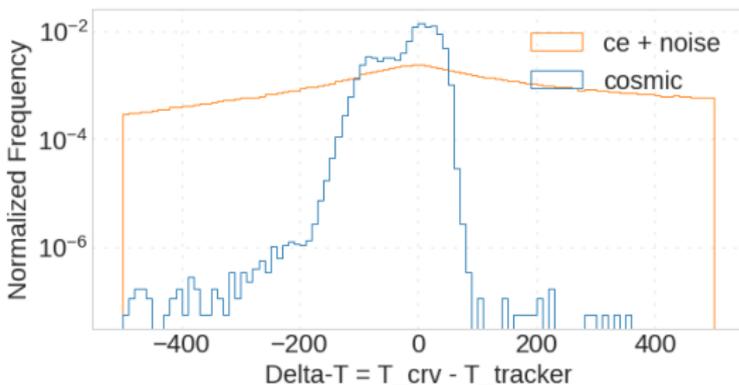


Data



- "Cosmics" refers to simulated cosmic ray muons
- CE & Noise distribution consistent of Conversion Electron (CE) events overlaid with Beam-induced CRV noise
- Large number of noise events that look like cosmic ray muon events, making them harder to distinguish

Variable Distribution Example

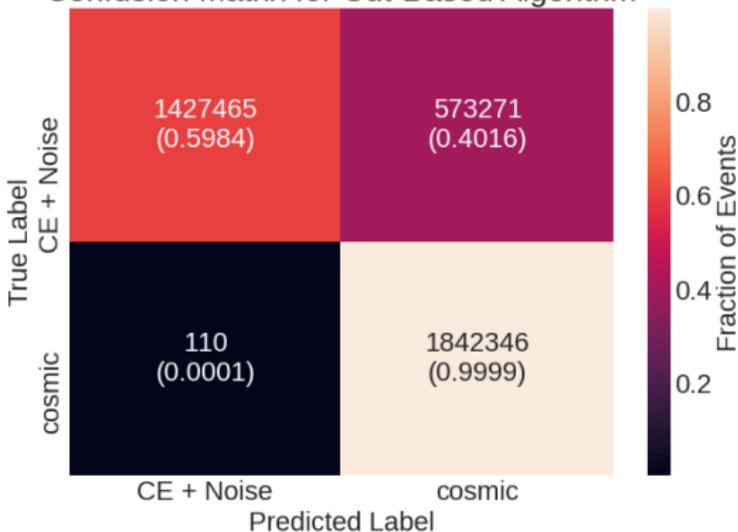


- An example of one of the reconstructed variables the model uses
- The differentiation between cosmic events and CE/noise events is evident in their distributions

Value Cut Algorithm Performance

- Identifies whether something is a cosmic based upon hard cuts on certain values, and categorizing them that way
- Currently quite good at cosmic identification - is used as the benchmark for future performance

Confusion Matrix for Cut-Based Algorithm



- Parentetical values of matrix represent fraction of events in that category
- Diagonal values are successful predictions
- Anti-diagonal entries are incorrect predictions

Objectives with This Study

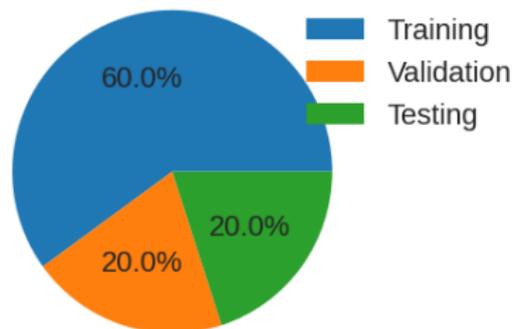
- Previous studies looked at images of events using a Convolutional Neural Network (CNN) - the current study will be using reconstructed numeric variables from events
 - Numeric variables lend themselves to Deep Neural Networks instead of CNNs

- Maximize the CE/Noise identification rate to minimize CRV deadtime
 - Still match the cosmic rejection efficiency of the cut-based algorithm (0.9999) such that the important function of the CRV is not compromised
 - The cut-based algorithm is currently the default one - these explorations look to improve its performance

Dataset Creation

- Input data standardized such that $\mu = 0$ and $\sigma = 1$ for all input variables
- Data was then shuffled and divided into training, validation, and testing datasets
- Each set of input variables had a corresponding set of target output variables
 - Output variable =
$$\begin{cases} 0 & \text{if CE/Noise} \\ 1 & \text{if cosmic} \end{cases}$$

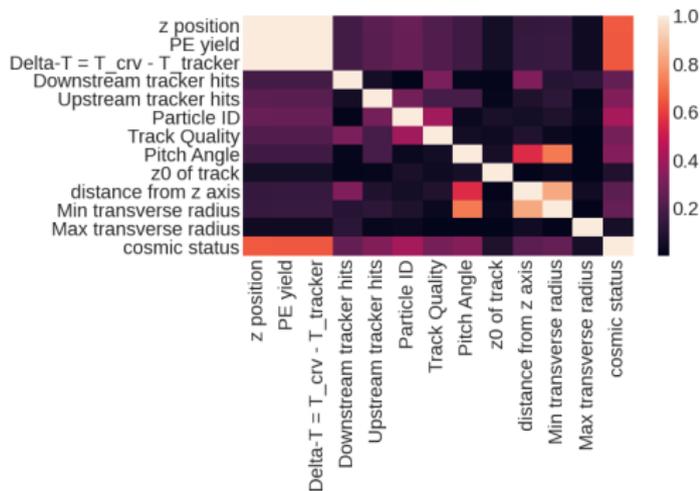
Size Split of Data



- 3,294,653 events total pulled from root files using the Uproot library
 - 1,771,121 cosmic events
 - 1,523,532 CE/Noise events

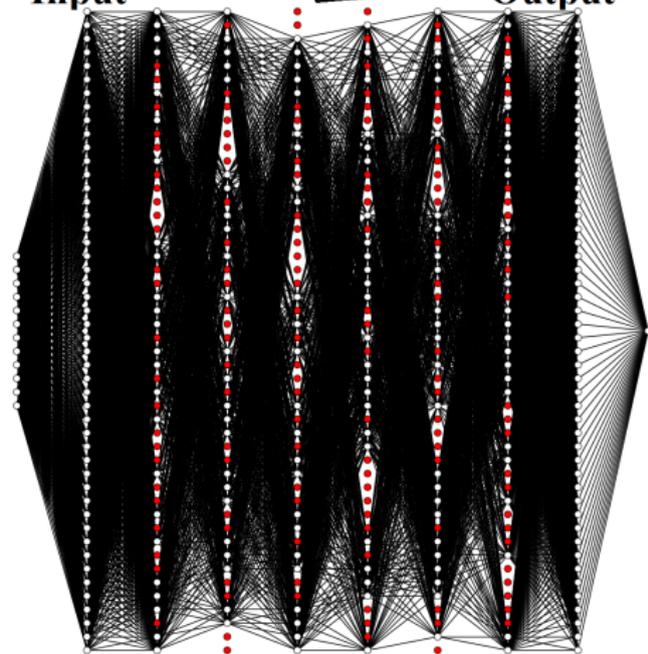
Variable Correlations

- Examples of input variables used are CRV Photoelectron (PE) Yield, Track Quality, and CRV z position
- Three variables at the top are CRV variables
- Their correlational values are the same due to the existence of events without CRV variables in the dataset
- Currently not a problem, but it will go away in the next iteration



Model Architecture

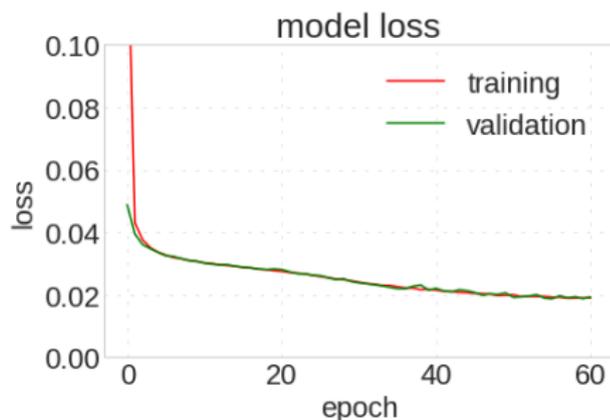
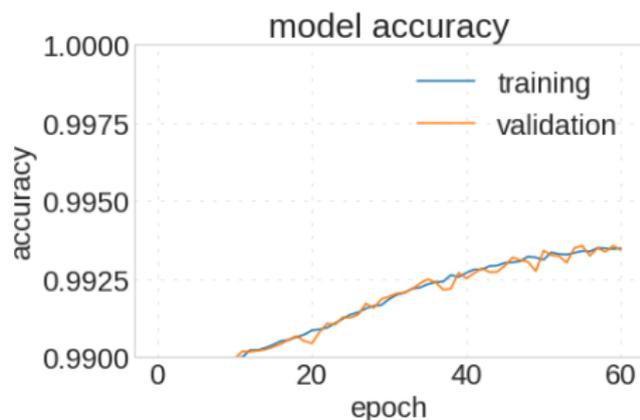
Input  Output



A rendition of the model with randomized dropout in red

- Utilizes the Keras package in Python
- Input layer of dimension 12
- Current depth of 8 with layer width of 48
- 6 Dropout layers with a rate of 0.5 serves 2 purposes
 - Prevent overfitting during training
 - MC Dropout statistics
- Utilizes the Binary Crossentropy loss function, and set to stop early if there is no improvement after 5 runthroughs

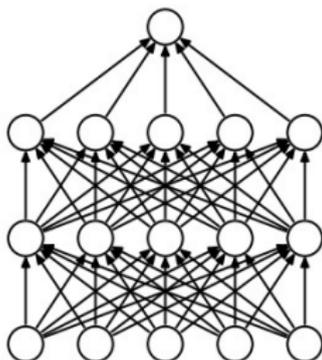
Training & Validation Metrics



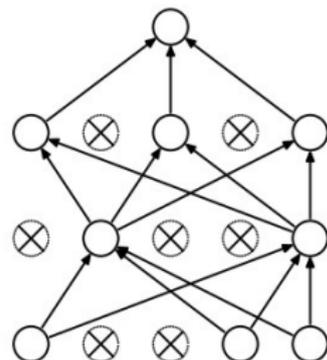
- Overall model accuracy ends at 0.99363 without risk of overtraining
- Training stops at 60 epochs, or runthroughs of the data

MC Dropout Usage

- Easy-to-compute method of determining epistemic uncertainty in a model
- Randomly disable a fraction of neurons in each layer each iteration
 - Mean of prediction iterations \rightarrow actual prediction
 - Variance of prediction iterations \rightarrow error in the prediction



(a) Standard Neural Net

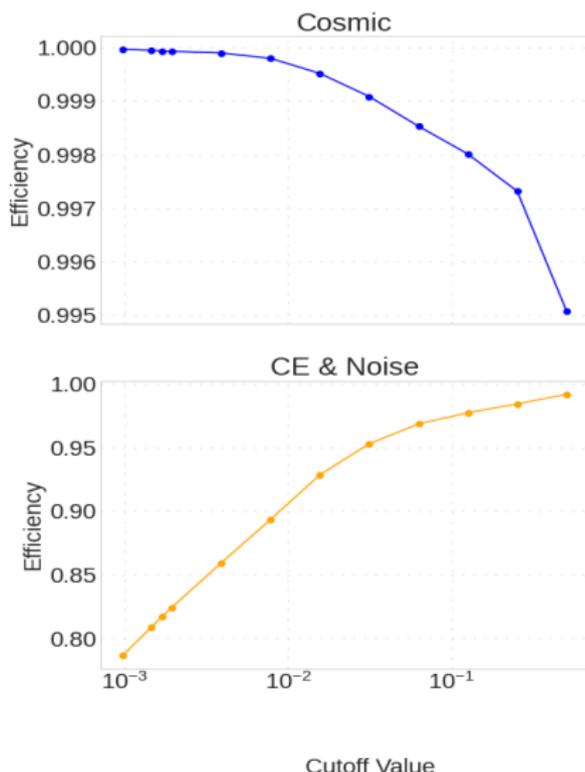


(b) After applying dropout.

Establishing a Cutoff Point

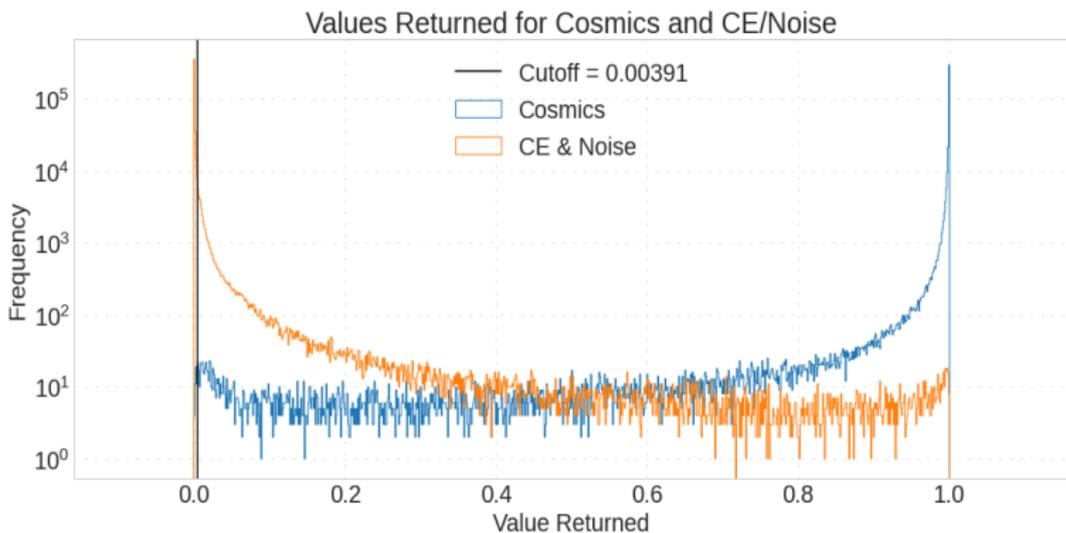
- Need to find a cutoff point such that cosmic identification is the same as cut-based algorithm
- The cutoff was found by conducting a binary search on the cosmic identification efficiency
- $classification = \begin{cases} CE/Noise & \text{if output} < \text{cutoff} \\ \text{cosmic} & \text{o.w.} \end{cases}$

Efficiency of Successful Identification VS Cutoff Value



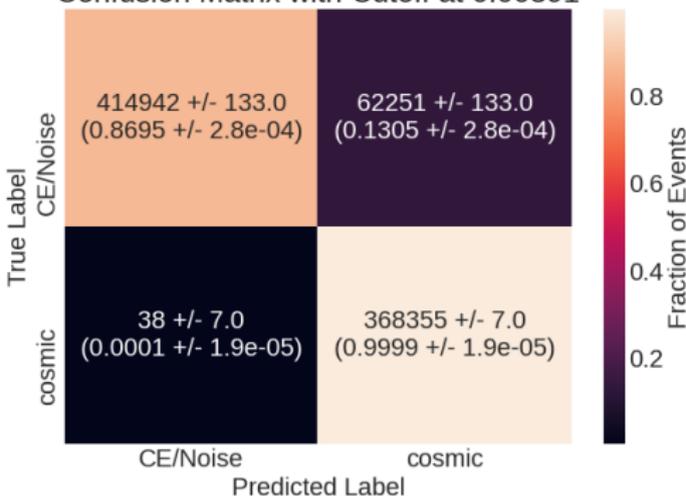
Output Distribution

- Colors represent the true label of the event
- Required cutoff is currently very small



Confusion Matrix

Confusion Matrix with Cutoff at 0.00391

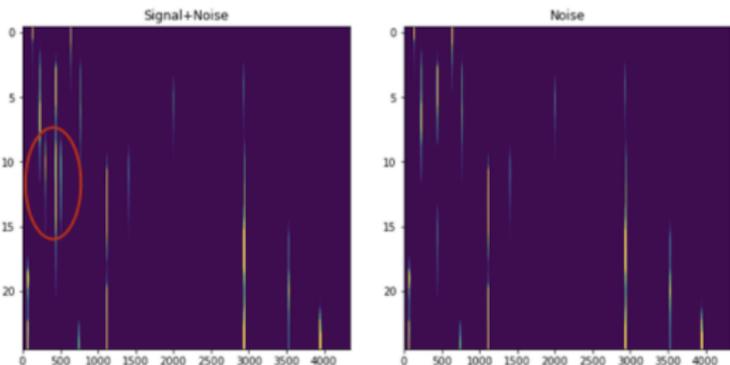
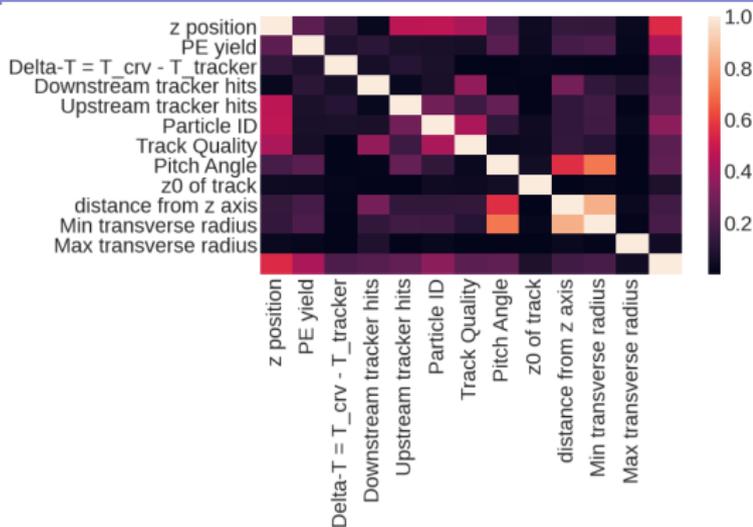


- With matching cosmic identification efficiency, the CE/Noise recognition is much better (~ 0.87 versus ~ 0.60)

- The error in each quadrant's value is found by calculating the quadrant values for each MC Dropout iteration, then getting their standard deviation
- The relative error is then taken by normalizing with the number of predictions within a row
- Much more uncertainty in the efficiency of CE/Noise than the cosmic identification efficiency due to low cutoff

Moving Forward

- Separating the algorithm into two event categories: with and without CRV coincidences
 - Preliminary findings show better results
- Testing robustness of model on variations in detector performance (i.e. changing light yield)
- Compare results to the CNN previously developed, as well as recurrent neural network using Long Short-Term Memory (LSTM) in the future



Links to Useful Resources



Yarin Gal & Zoubin Ghahramani (2016)
[Dropout as a Bayesian Approximation](#)



Filip Široký (2020)
[Bayesian Approximation Using Dropout During Inference](#)



E. Craig Dukes (2017)
[Cosmic Rays are a Pain: The Mu2e Cosmic Ray Veto](#)



[Uproot Library](#)
[Uproot Documentation](#)



[Keras Library](#)
[Keras Documentation](#)

Questions



Above: [XKCD 1838](#)

Right: [XKCD 2451](#)

