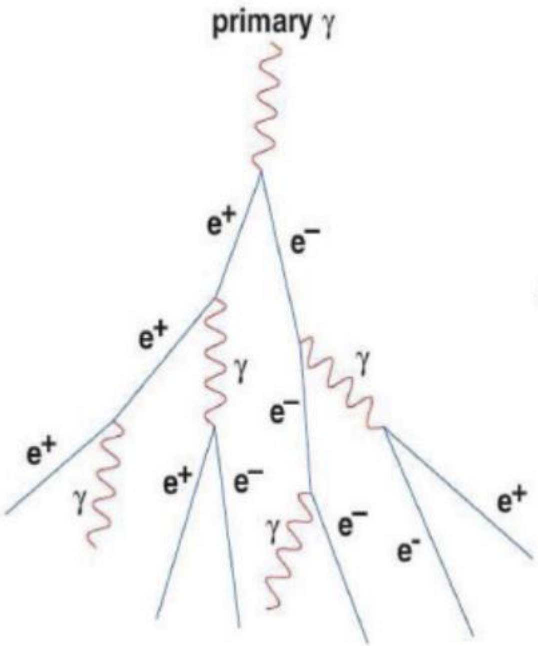
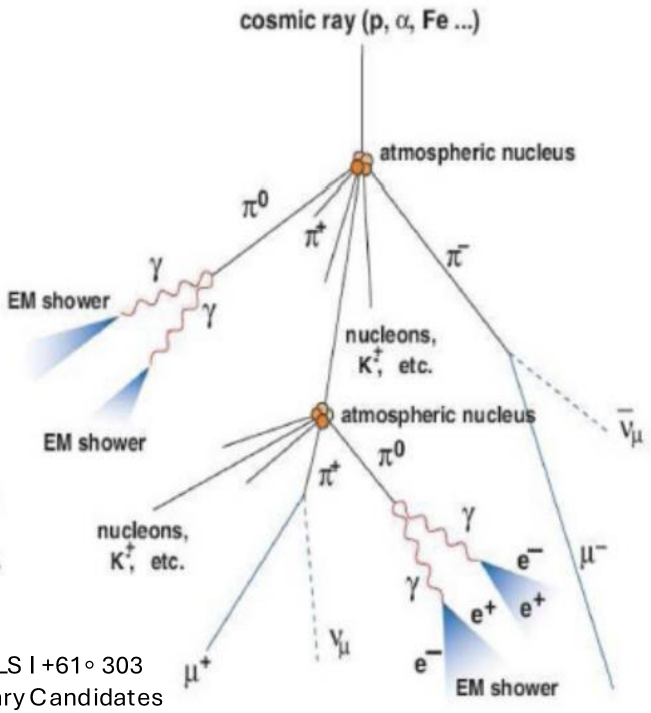
IceTop gamma-hadron separation and angular error estimation using machine learning techniques

- Sebastian Vergara Carrasco
- University of Canterbury, Christchurch, New Zealand

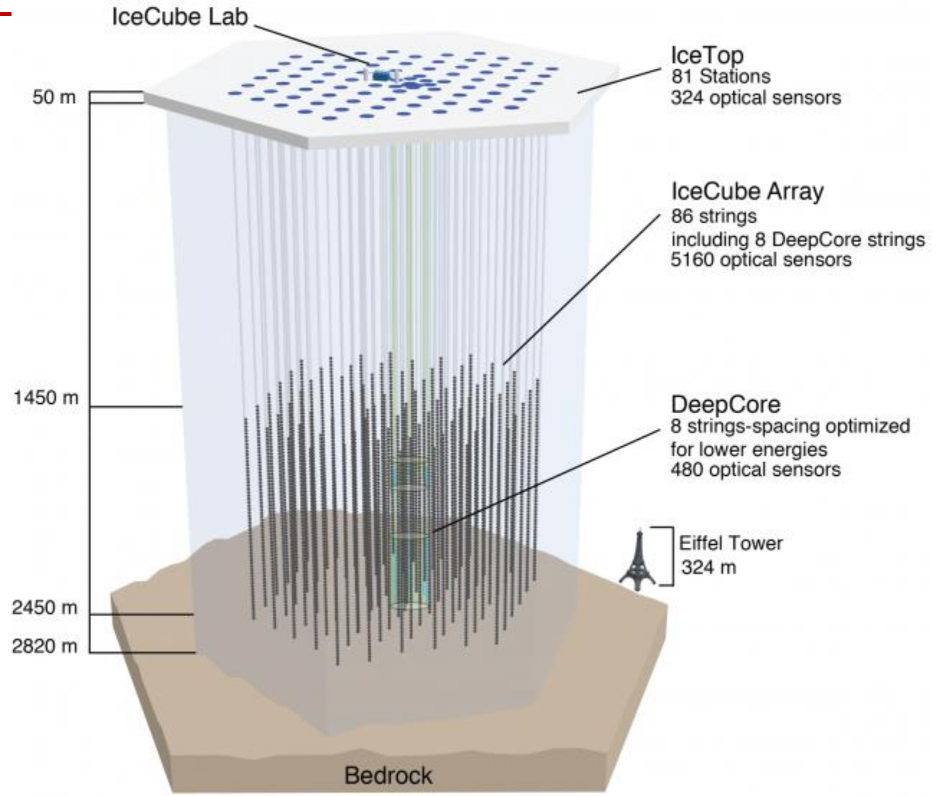
- Hybrid Workshop on Machine Learning for Cosmic Particles, Delaware, 27-31 Jan 2025

Introduction





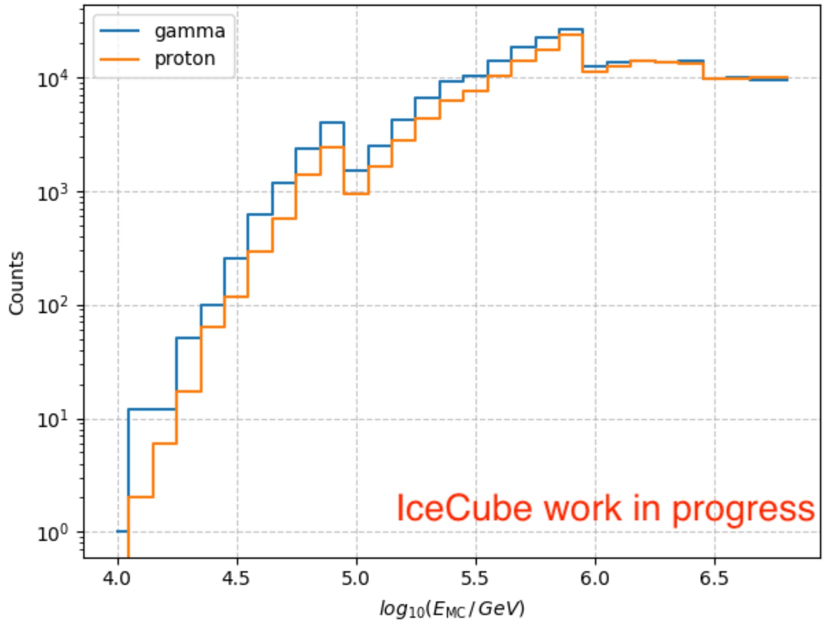
Ref: Multi-year Campaign of the Gamma-Ray Binary LS I +61° 303 and Search for VHE Emission from Gamma-Ray Binary Candidates with the MAGIC Telescopes, 10.13140/RG.2.1.4140.4969



- Pulses in IceTop are either classified as HLC (two tanks hit within 1 microsecond in one station) or SLC (only one tank hit in a station).
- Observables such as the direction, primary energy proxy and shower core position are reconstructed using IceTop HLC pulses.

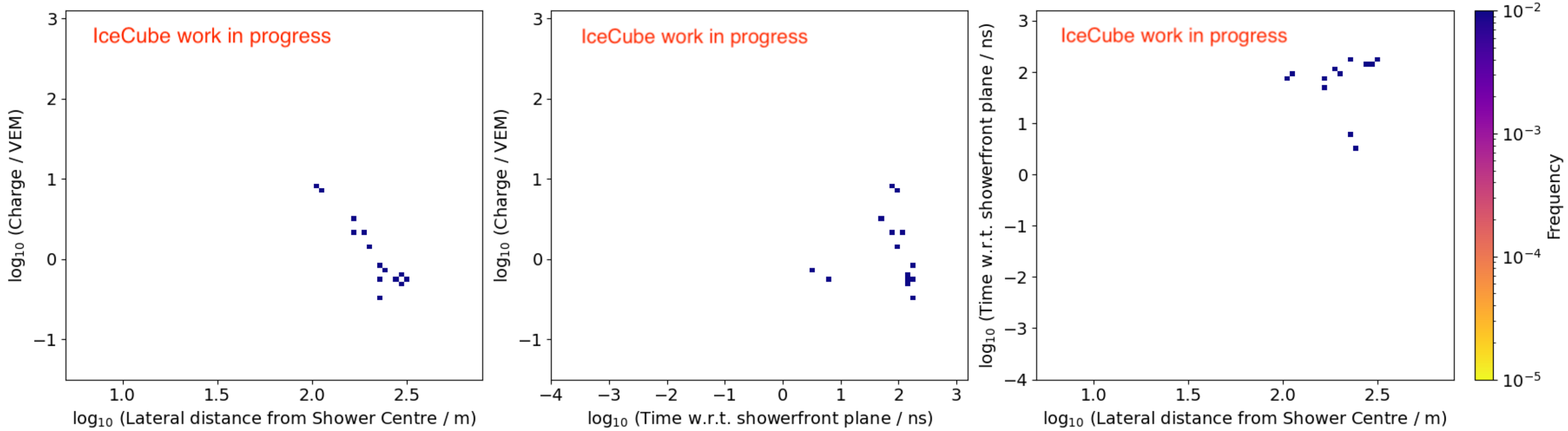
Dataset

- Gamma-induced and proton-induced air shower simulations produced by Federico Bontempo from $4.0 \leq \log_{10}(E/GeV) \leq 7.0$ for 2012.
- Using sibyll2.3d as hadronic model
- This data is regularly split into energy bins of 0.1 in $\log_{10}(E/GeV)$. A certain energy bin will be referred to as E bin number, e.g. E6.9 represents the energy bin $6.9 \leq \log_{10}(E/GeV) \leq 7.0$.
- Standard IceTop quality cuts are applied throughout, these are:
  - Radius < 500 m
  - Zenith < 38 degrees
  - Fit status = OK
- This means the number of events after quality cuts is:
  - Gamma: 238528
  - Proton: 208203
  - Total: 446731

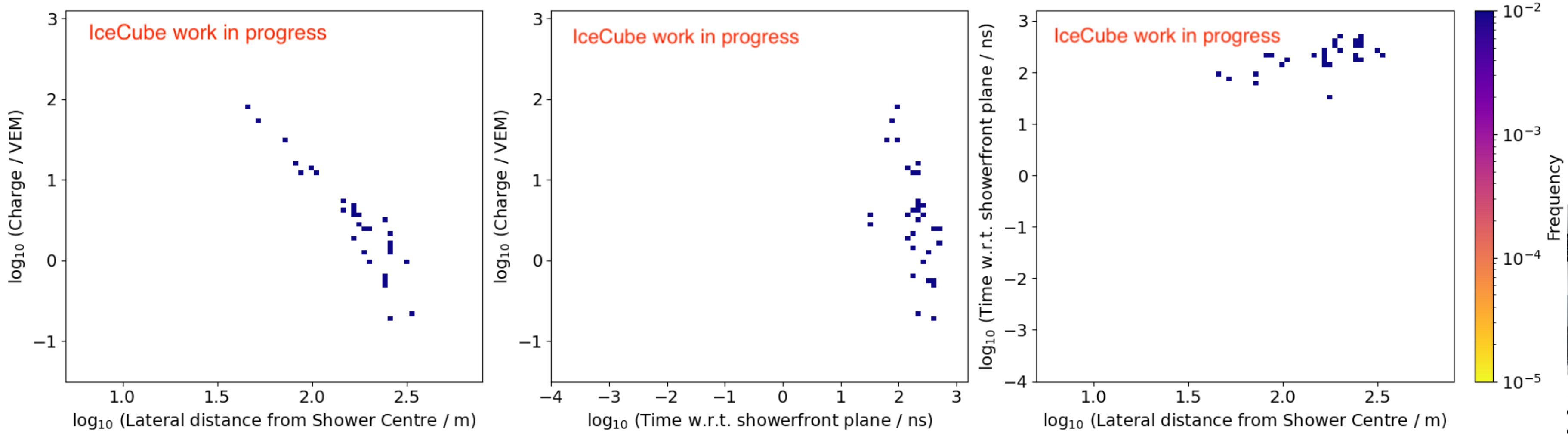


CNN for separation - inputs

Single gamma event

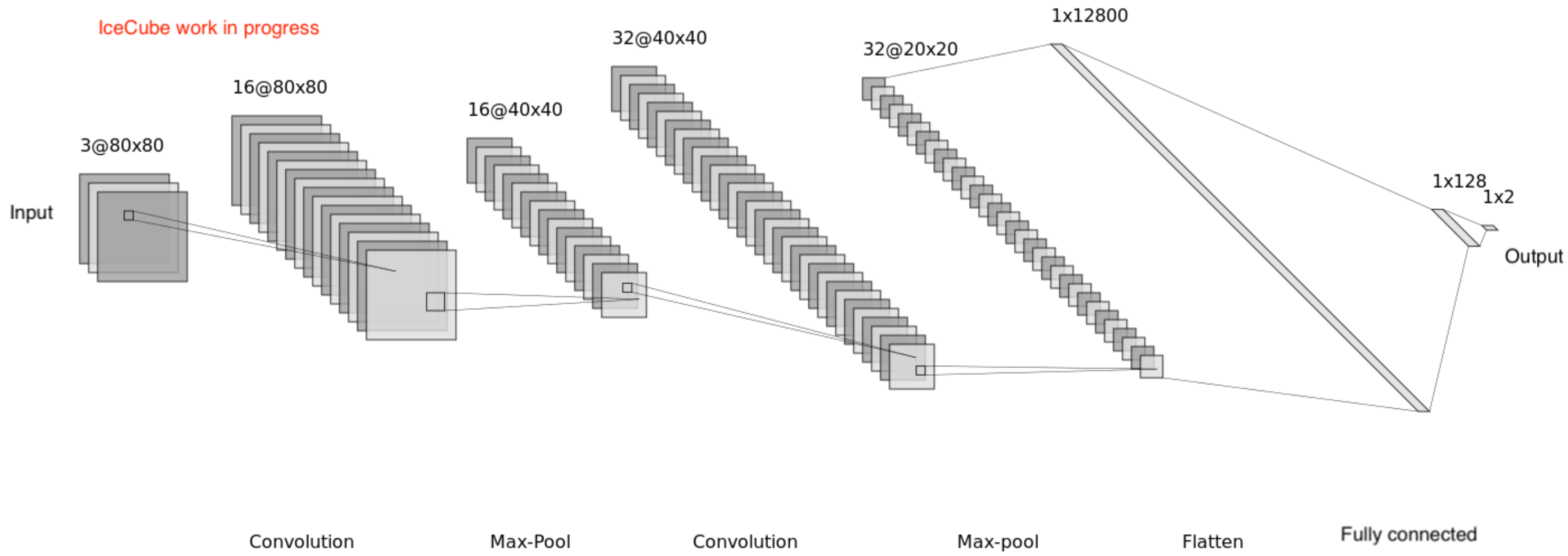


Single proton event



CNN Architecture

- Final two layers are both fully connected layers. Output layer has two outputs, proton probability and gamma probability.
- Regularization includes dropout layer, weight decay and a learning rate scheduler.
- Using cross entropy loss, initial learning rate of 0.001.



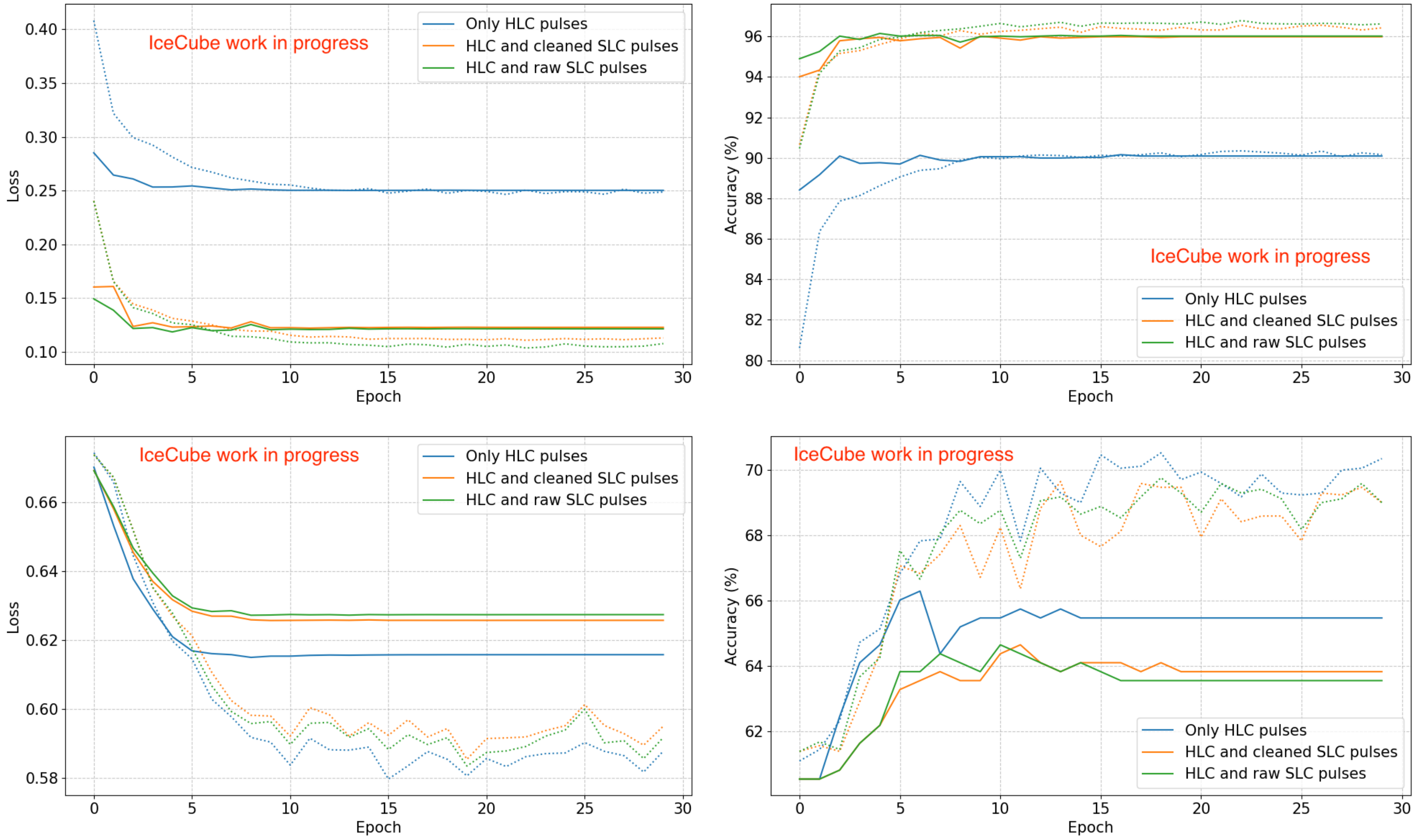
Each $\log_{10}(E/GeV)$ bin is has its own training with three separate models.

CNN results

E6.9 bin model

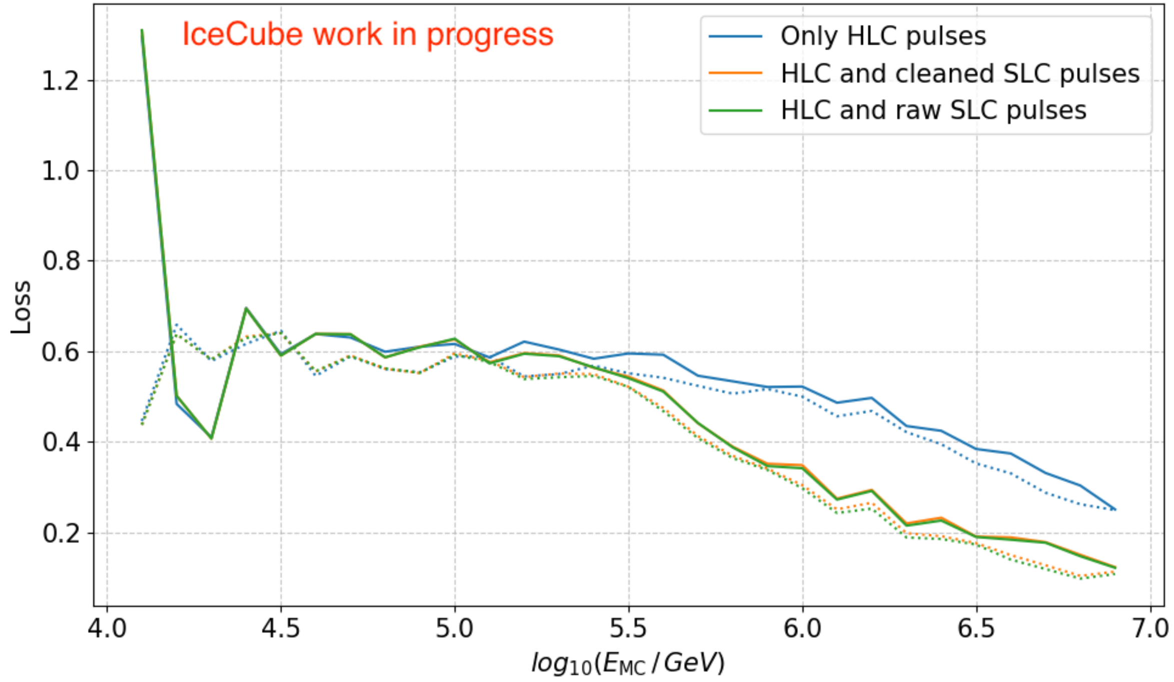
- Dotted line is training, solid line is validation

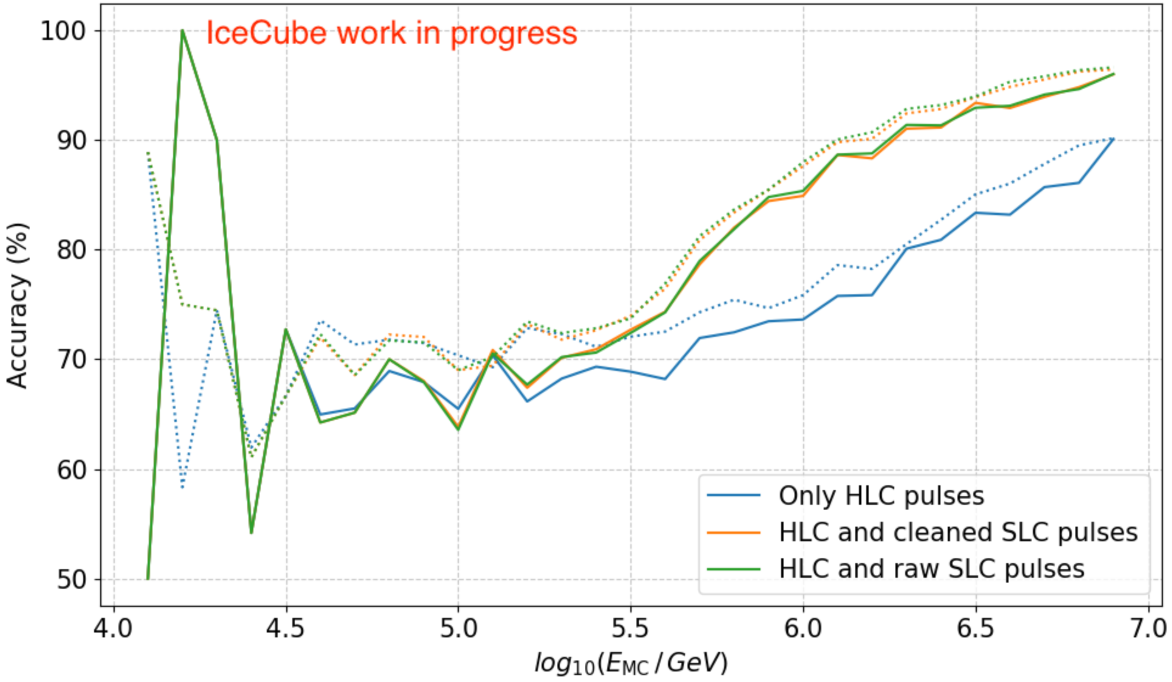
E5.0 bin model



CNN results

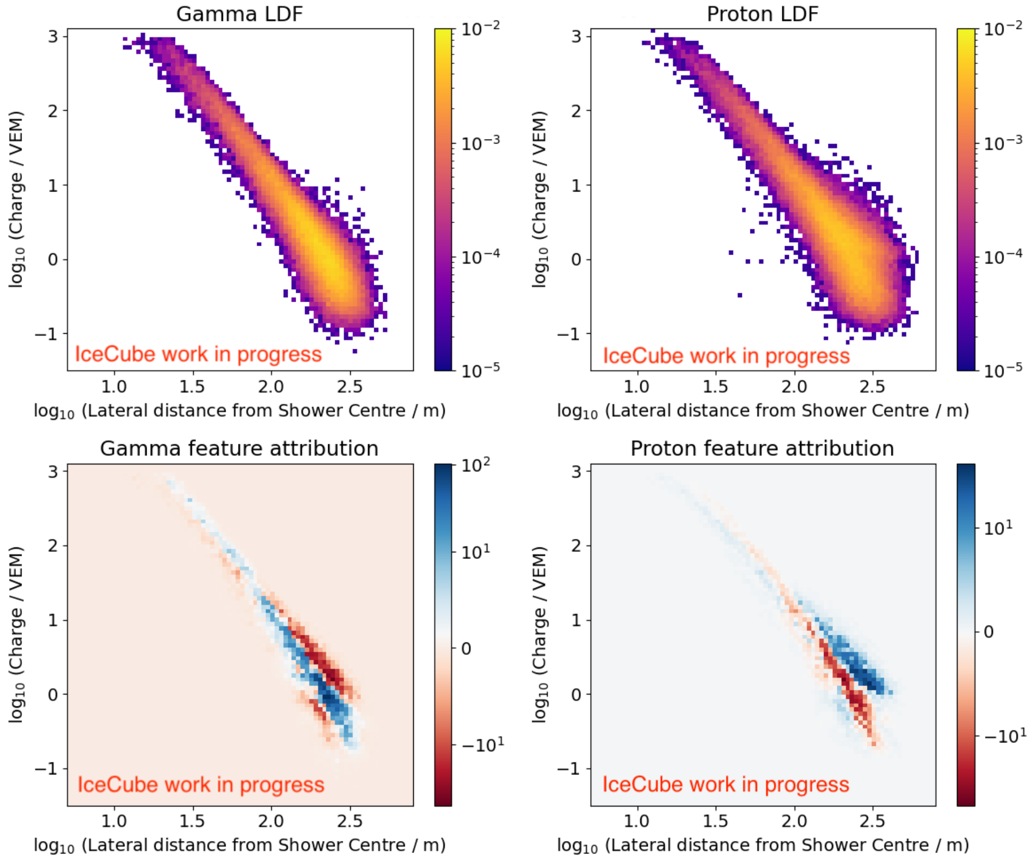
- Final metric value after 30 epochs per energy.



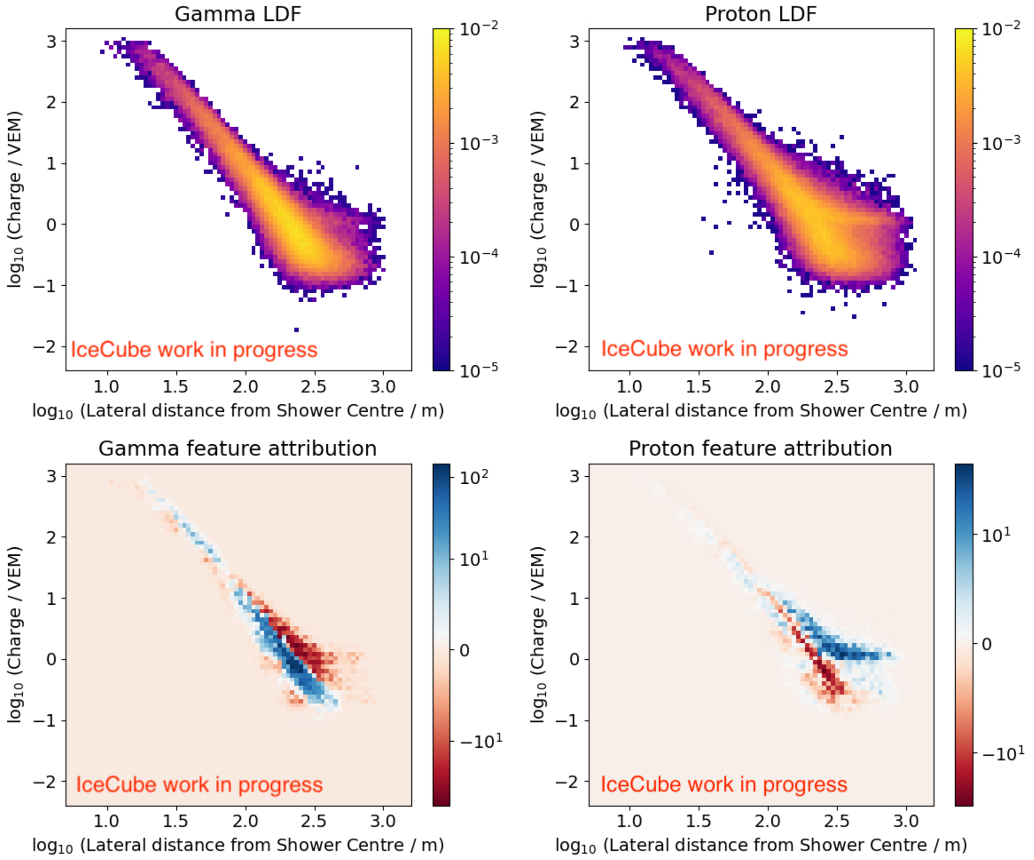


CNN feature attribution

- Shows what areas the CNN focused on. Blue being positive, red being negative. E6.9 bin.



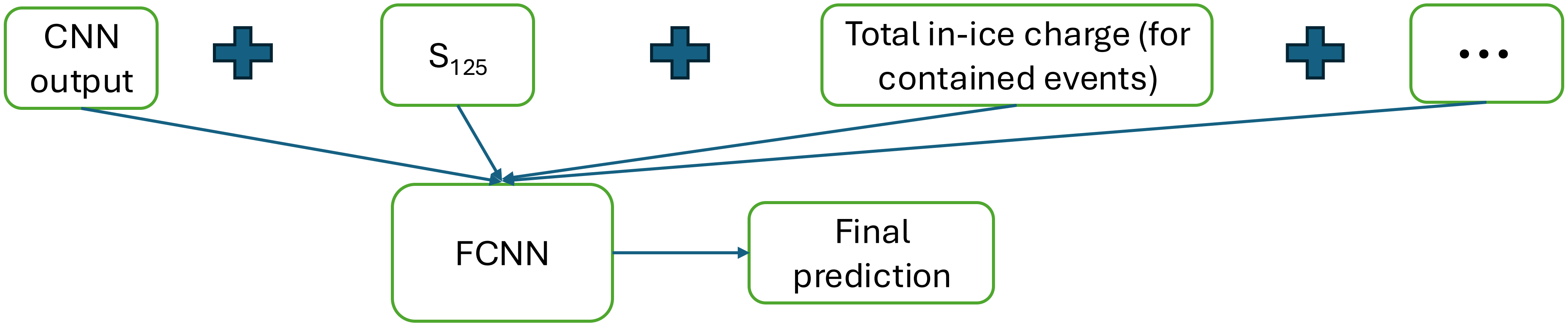
HLC only



HLC and cleaned SLC

What next?

- Use the probability output of the CNN as an input into a FCNN – can directly input into previous models.



- Ensure this provides the strongest result for uncontained events - test on real data and train on contained/uncontained events.
- Possibly create two models, one for contained one for uncontained?

Angular error estimation

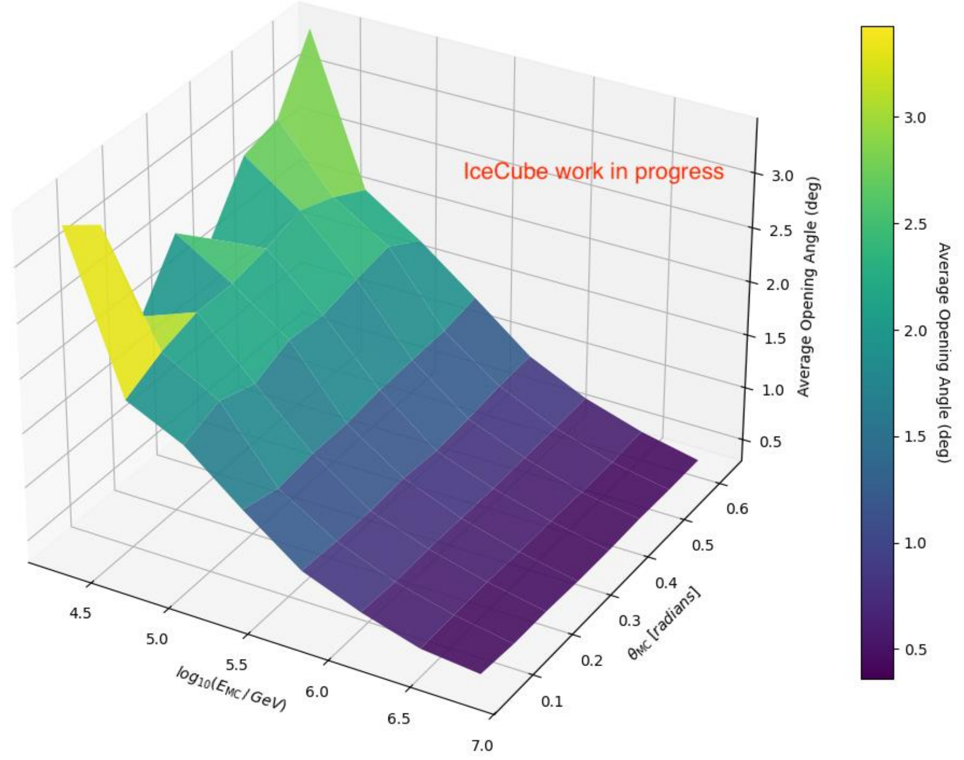
- Is it possible to estimate the accuracy of our reconstructed direction on an event-by-event basis?

$$\boldsymbol{n} = \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} \sin(\theta) \cdot \cos(\phi) \\ \sin(\theta) \cdot \sin(\phi) \\ \cos(\theta) \end{bmatrix}, \qquad \theta_{\text{res}} = \cos^{-1}\left(\boldsymbol{n}_{\text{true}} \cdot \boldsymbol{n}_{\text{reco}}\right) \cdot \frac{180}{\pi}.$$

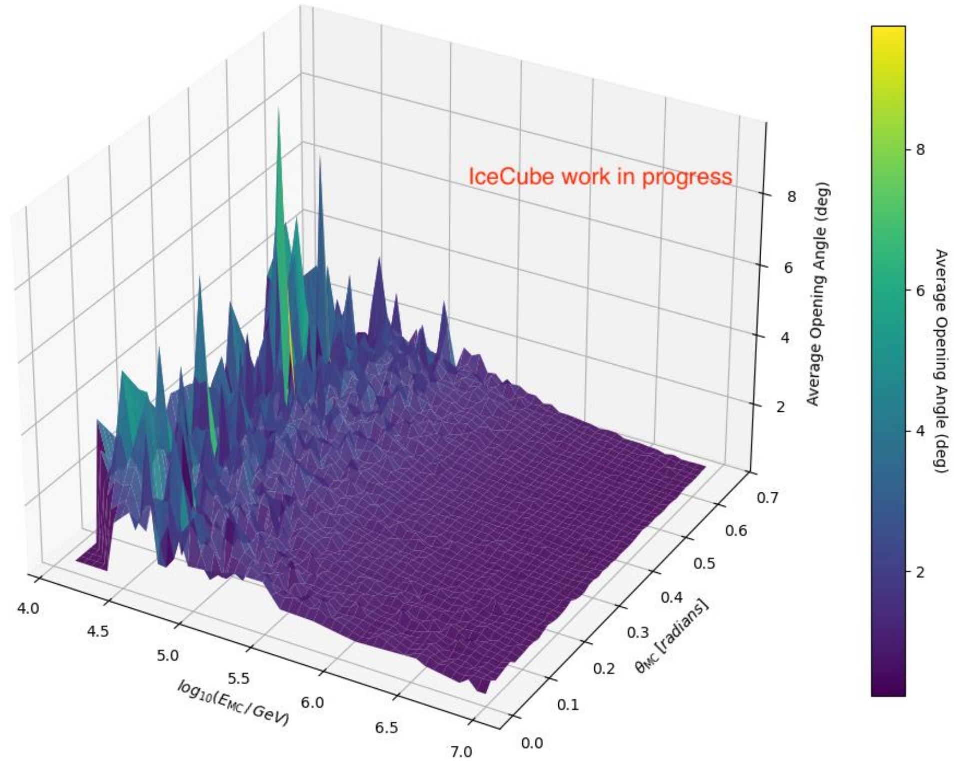
- To try and capture a general trend for opening angle (angular error) by creating a spline over specific parameter spaces. Namely:
  - Energy
  - Reco zenith
  - S_{125} (energy proxy)
  - Chi2 time (from direction fit)
  - True zenith
- Also experimented in log space.
- Within the parameter space we take the average angular error values of all events falling in a specific bin, ranging our number of bins for each space from 5 to 50.

Angular error spline fit

- Example: true energy vs true zenith spline approximation for different bin values. Is there a better way?



8 bins



50 bins

BDT for angular error

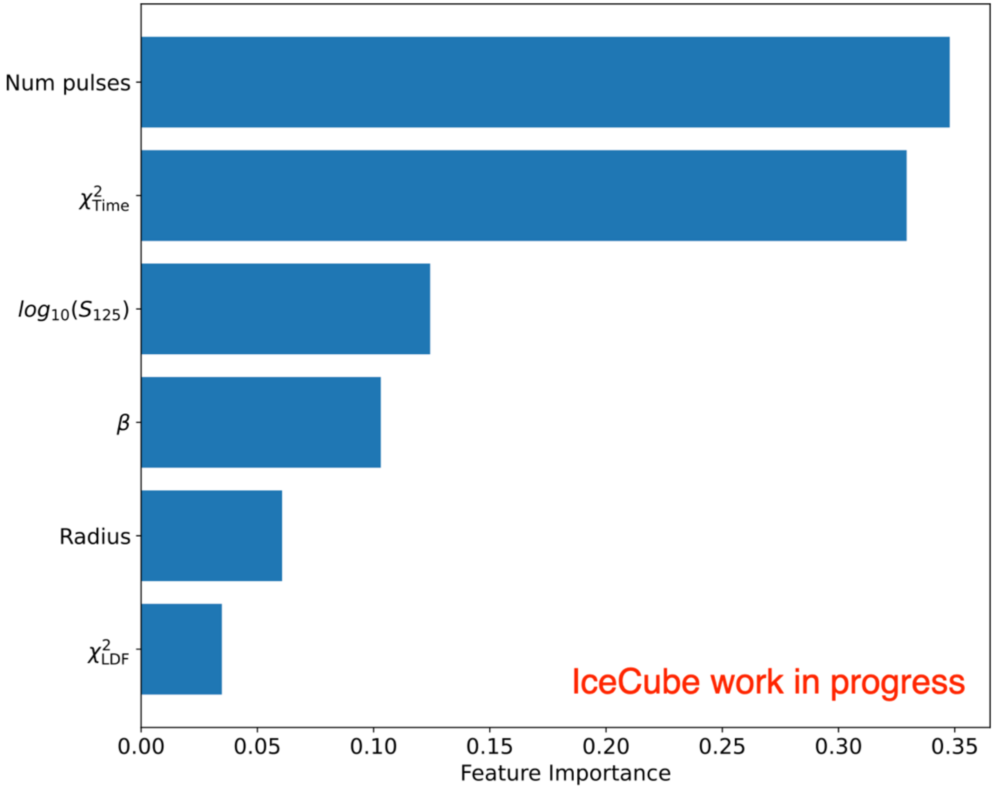
- Trying to get a BDT to estimate the angular error of directional reconstruction.
- Only using HLC pulses for the gamma dataset.
- Varied many of the feature inputs, based on feature important plots and parameter distributions.
- Also tried regularization techniques – such as varying values for L1 (lasso) and L2 (ridge).
- To optimize hyperparameters, used in-built randomized search and grid search from XGBoost. These include:
  - Num estimators
  - Colsample by tree
  - Learning rate
  - Alpha (L1)
  - Max depth
  - Lambda (L2)
  - Subsample
  - Min child weight

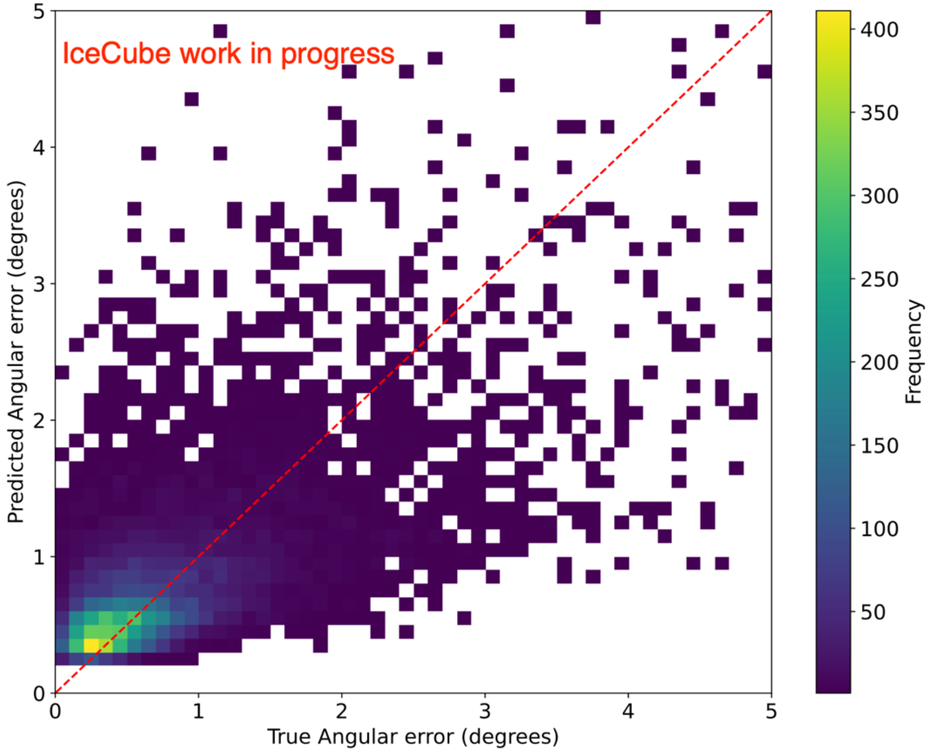
BDT results – basic

Test Root Mean Squared Error (RMSE): 0.6504 degrees
Test R-squared (R^2): 0.4529

Train Root Mean Squared Error (RMSE): 0.6359 degrees
Train R-squared (R^2): 0.5144

- Num estimators: 80
- Learning rate: 0.1
- Max depth: 5
- Subsample: 1.0
- Colsample by tree: 0.8
- Alpha: 1
- Lambda: 1.5
- Min child weight: 3



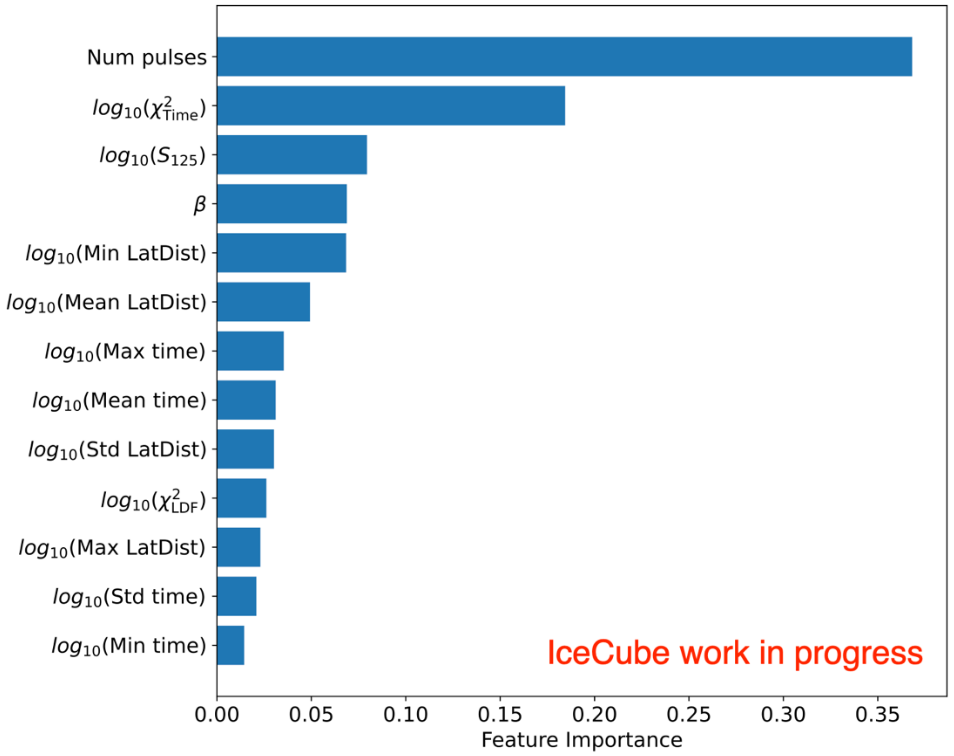


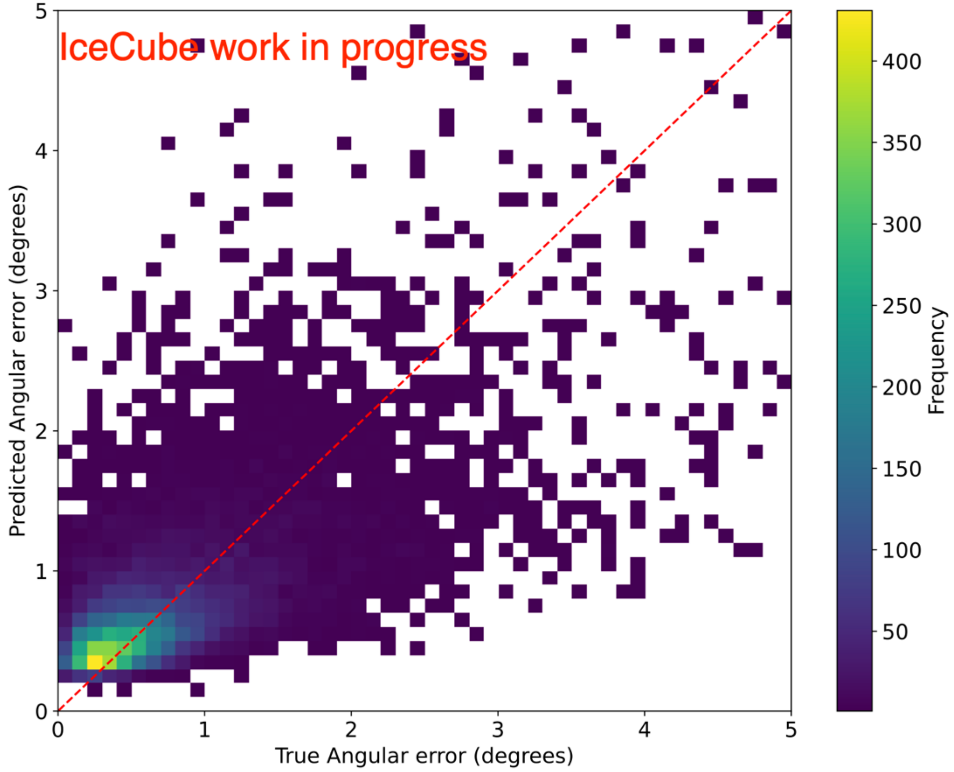
BDT results – extra pulse info

Test Root Mean Squared Error (RMSE): 0.6357 degrees
Test R-squared (R^2): 0.4773

Train Root Mean Squared Error (RMSE): 0.6025 degrees
Train R-squared (R^2): 0.5641

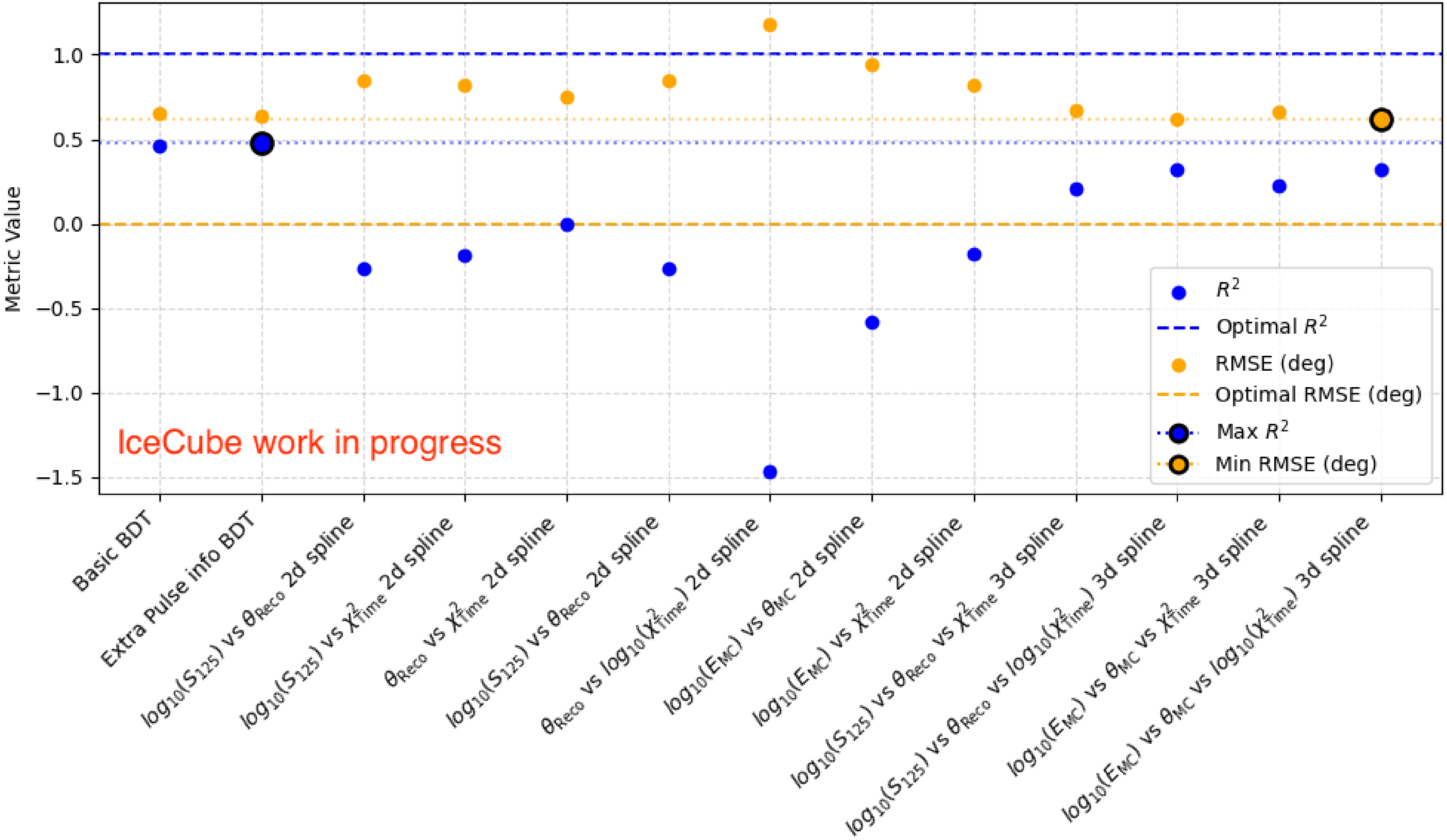
- Num estimators: 200
- Learning rate: 0.05
- Max depth: 5
- Subsample: 0.9
- Colsample by tree: 1.0
- Alpha: 1
- Lambda: 1.5
- Min child weight: 5





ICECUBE

BDT results – compare to spline fit



Summary and outlook

CNN for gamma/hadron discrimination:

- Using the distributions of charge, time and lateral distance in a CNN for gamma/hadron separation gives promising results as an initial prediction.
- Using SLC pulses within the CNN gives further improvements on discrimination at higher energies, still struggling at lower energies.
- Next step is to integrate into previous methods using in-ice within a FCNN, compare results.

BDT for angular error estimation:

- Works better than sampling from multidimensional splines, but still not necessarily a great result.
- Requires testing on real data, specifically the reconstructed parameter distributions.

Backup slides

Previous work

1. Search for PeV Gamma rays and astrophysical neutrinos with IceTop and IceCube – Hershal Pandya PhD.
   - Created the IT-LLHR method. Our method is based off of this approach.
   - He created probability distribution functions (PDFs) using a certain percentage of the data to form the hypothesis, then compared each event bin by bin to form the likelihood value.

$$L_{QR}(\text{event}|H) = \prod_{\mathrm{i}=1}^{162} P\left(Q_{\mathrm{i}},\ R_{\mathrm{i}}|\ H\right),$$

$$\Lambda_{QR} = \log_{10}\left(\frac{L_{QR}(\text{event}|H_{\gamma})}{L_{QR}(\text{event}|H_{CR})}\right),$$

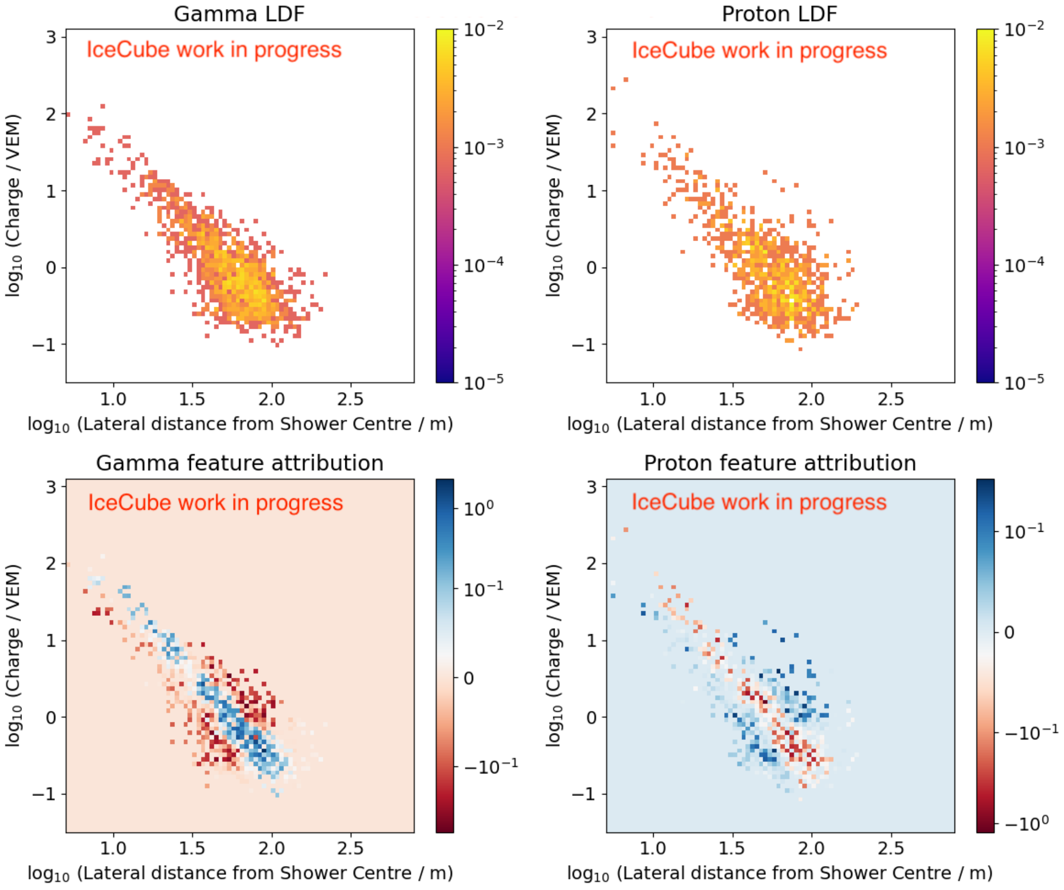
$$\Lambda = \Lambda_{QR} + \Lambda_{Q\Delta T} + \Lambda_{\Delta TR}$$

2. Search for PeV Gamma rays with the IceCube observatory – Zachary Dean Griffith PhD.
   - Focused on using ML for gamma-hadron separation, specifically a random forest using multiple reconstructed variables, also Hershals IT-LLHR

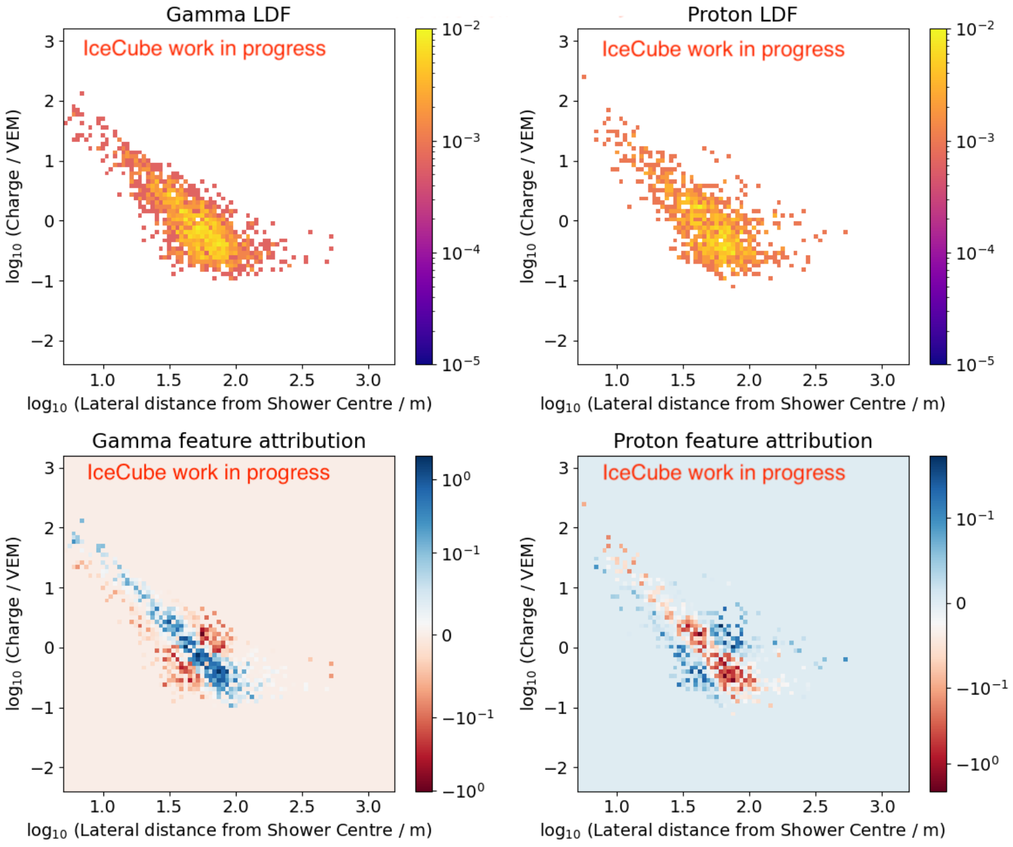
3. Federico Bontempo PhD.
   - Continued using ML for gamma-hadron separation, expanding on previous work. Used 2d surface maps in a CNN as in input for a fully connected NN.

CNN feature attribution

- E5.0 bin

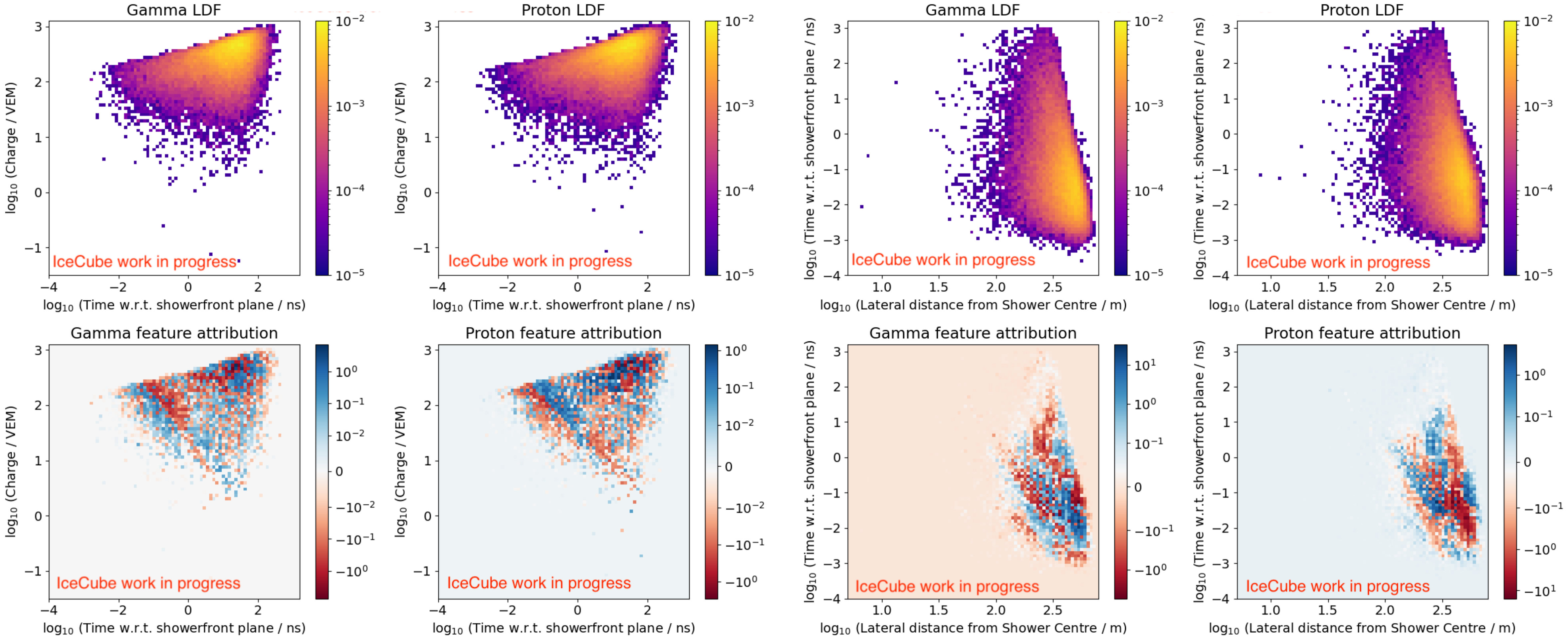


HLC only



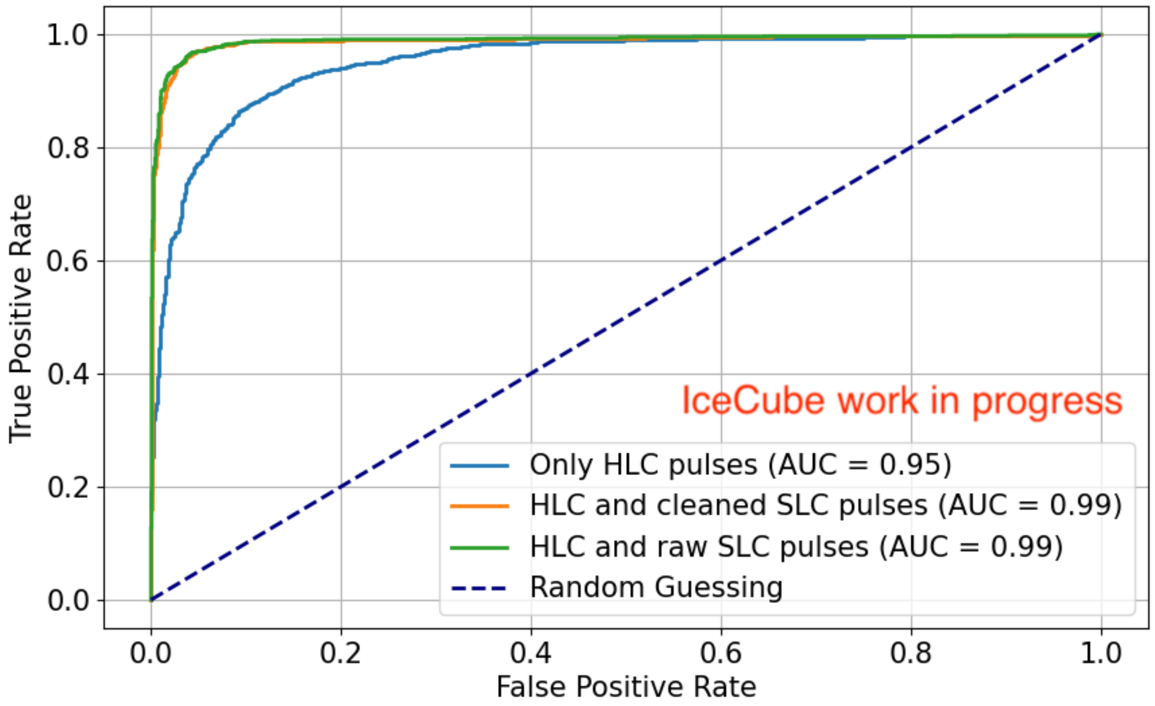
HLC and cleaned SLC

Extra feature attribution distributions



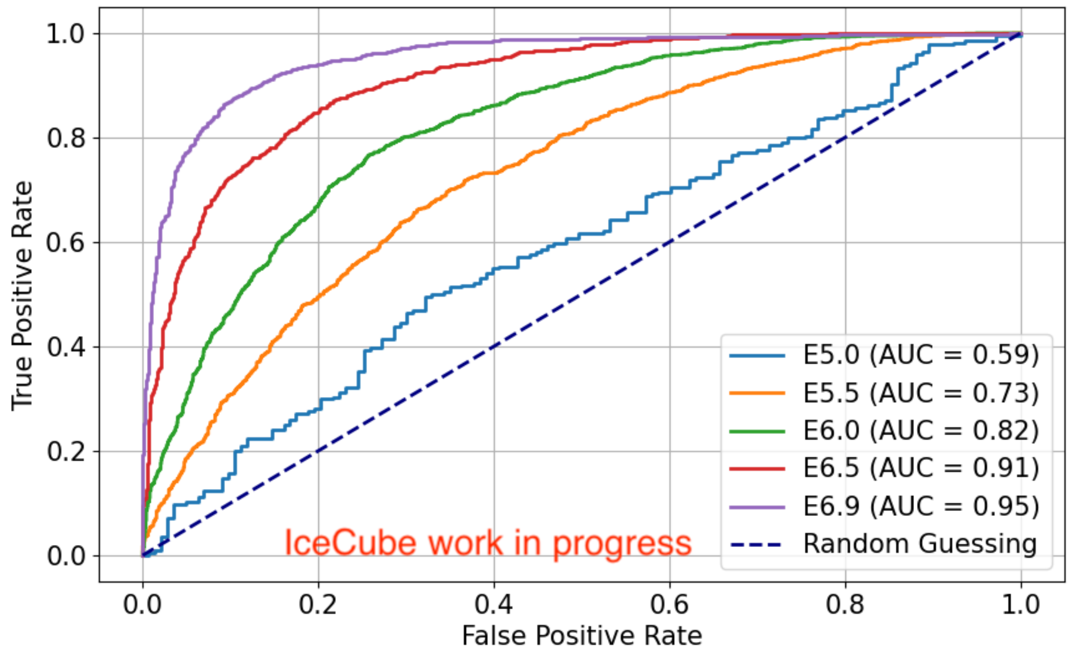
ROC curves

- Receiver operating characteristic (ROC) curves comparing E6.9 models. As gamma is given the label 1, it is the positive case.
- Generated by varying the classification threshold. Thus:
  - bottom left - threshold is extremely high - everything is classified as negative
  - top right – threshold is extremely low - everything classified as positive.
- True Positive rate = True Positives / (True Positives + False Negatives)
- False Positive rate = False Positives / (False Positives + True Negatives)
- Area under the curve closer to 1 indicates a much better model

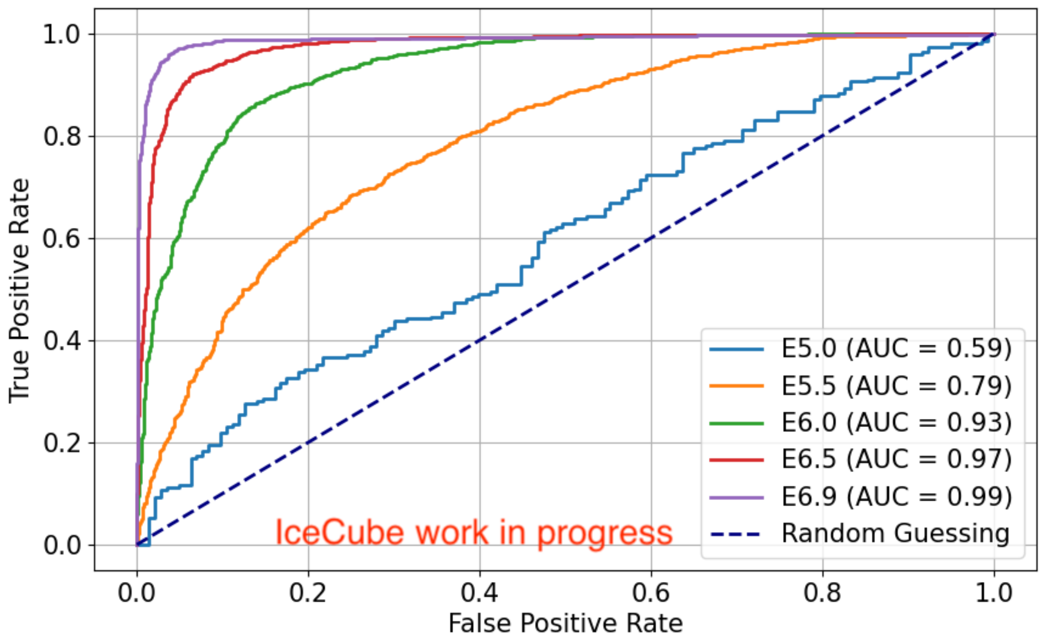


ROC curves

- What about other energies?



HLC only



HLC and cleaned SLC

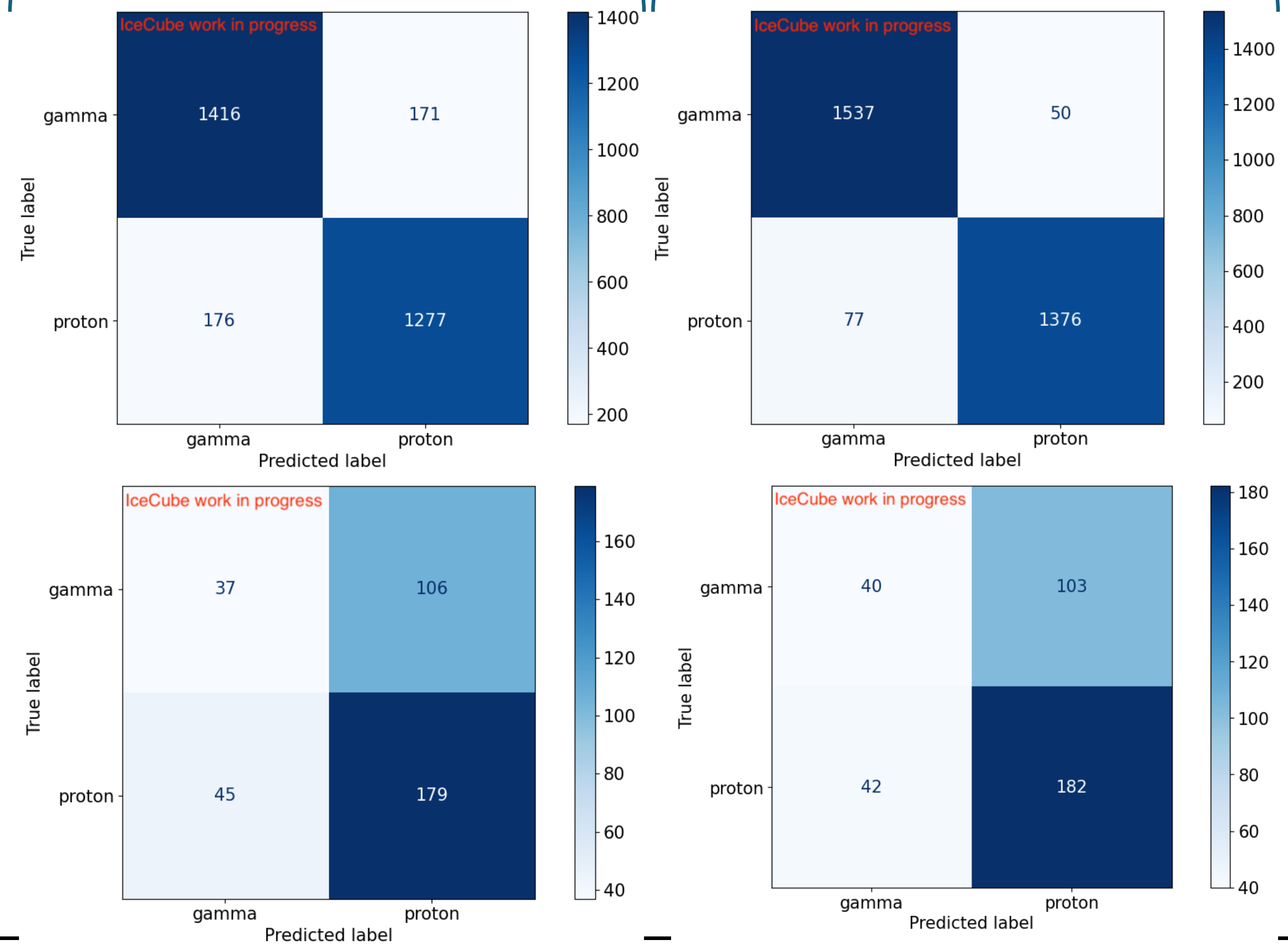
Confusion matrices

HLC only

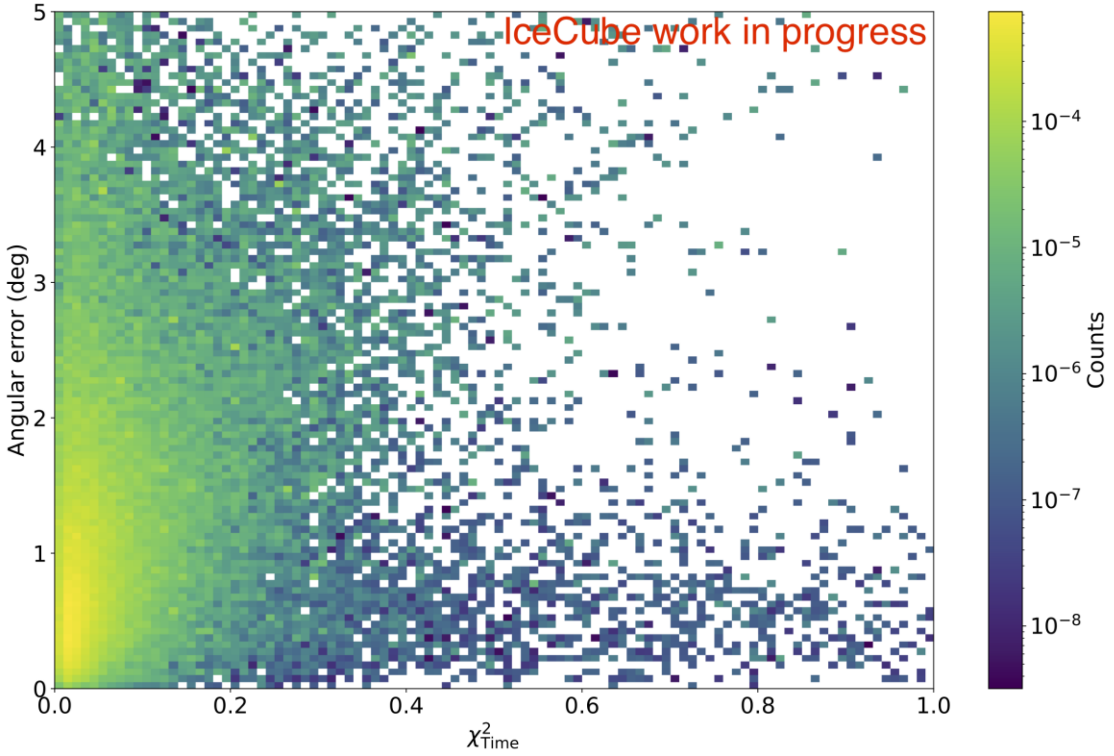
HLC and cleaned SLC

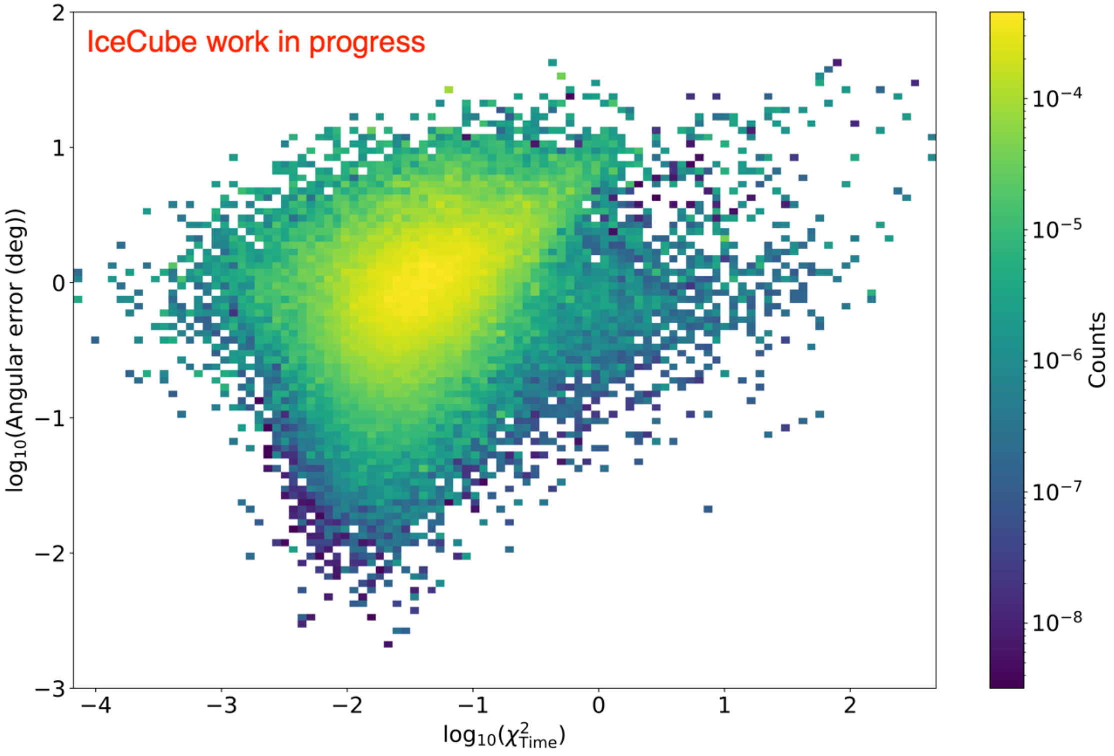
E6.9 bin model

E5.0 bin model



Chi2 distribution





Log scale seems to have a more linear relationship, something to explore!