ARENA
SANTIAGO DE COMPOSTELA /2022

UNIVERSITY OF DELAWARE
BARTOL RESEARCH INSTITUTE

Deep Learning for the Classification and Recovery of Cosmic-Ray Radio Signals against Background Measured at the South Pole

Alan Coleman, Abdul Rehman, Frank Schroeder, Dana Kullgren

for the IceCube Collaboration

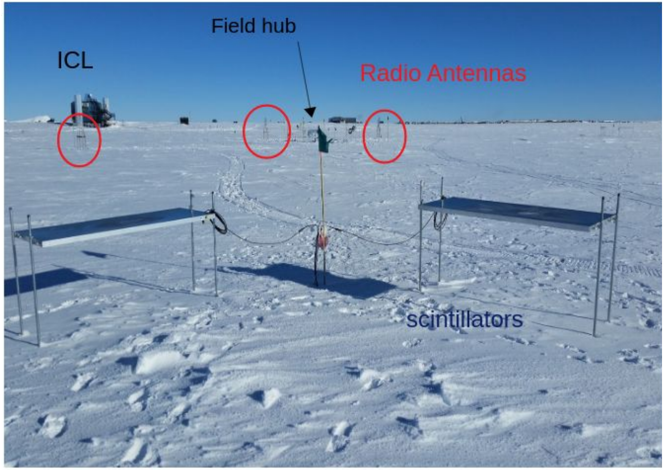
ARENA, SPAIN (Jun 7 – 10, 2022)

IceCube

Introduction

- Radio detection, like other techniques, has to deal with the continuous background.
- We are using ML (Convolutional Neural Networks) to try to mitigate the effects of background.
- Noise data used in this work is collected from the Prototype Station at the South Pole*.
- With this we aim to identify more radio signals.
- (In future) maybe we will be able to work on the stand alone radio trigger.

Full prototype station deployed in 2020



* Please see Hrvoje Dujmovic talk for more details on measured background.

Data Preparation

- CORSIKA → (IceTop atmosphere).
- CoREAS → Sibyll 2.3d.
- Thinning = 1e-06.
- Proton Primary, Energy = [10^{17}, $10^{17.1}$] eV.
- Zenith angles → [0., 0.8] in $\sin^2(\theta)$, random azimuth angles.
- For Noise: Measured background from **Prototype Station** at the **South Pole** (Soft triggered).
- Filtered band [100-350] MHz.

Data Processing Chain

CoREAS Traces

↓

Convolved with Detector Responses

↓

Coherence Cut on Pure Signal

↓

Measured Noise Added

↓

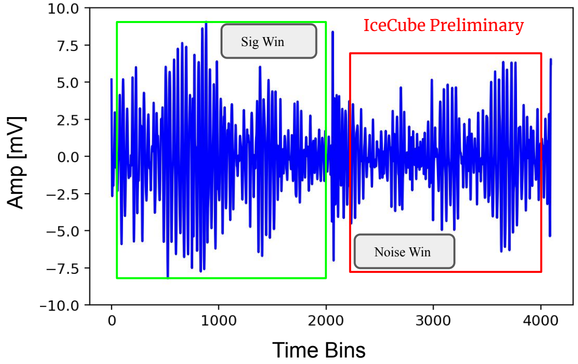
Traces Filtered

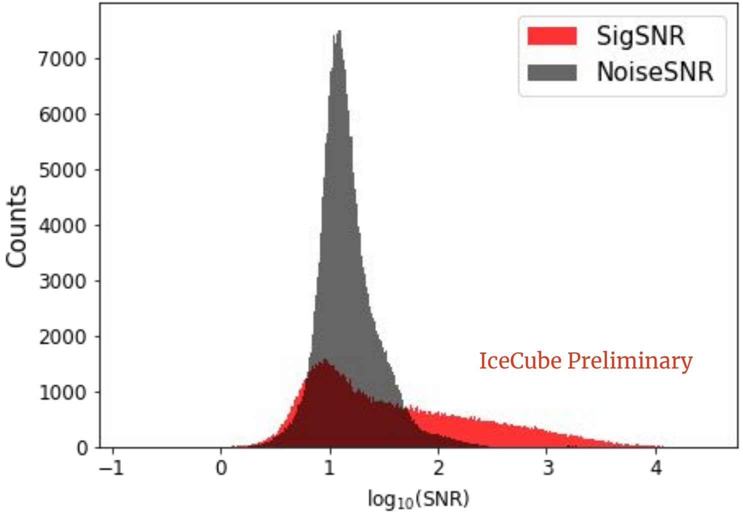
Dataset Distribution

- Signal to Noise ratio (SNR) is used to quantify the signal strength.
- Before adding noise signals are scaled by multiplying it with sqrt(10*A), where A is a random number b/w [-1, 2].
- About 300k Noise Only + 150k Signal traces.
- 80% for training, 20% for testing.

$$\mathrm{SNR} = \left(\frac{\mathrm{Signal}_{\mathrm{Peak}}}{\mathrm{Noise}_{\mathrm{RMS}}}\right)^2$$

Trace length = 1024 ns

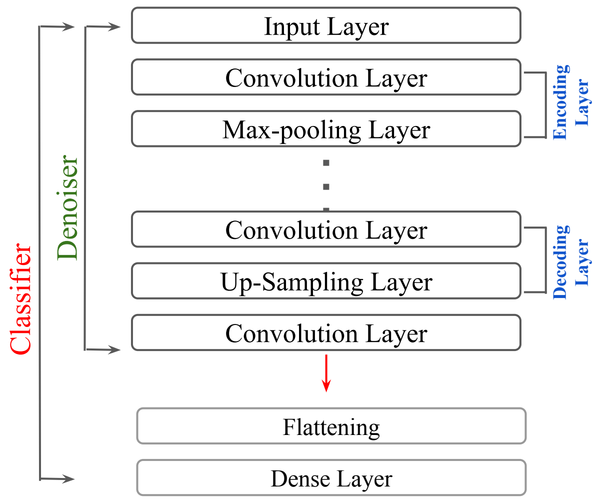




- Mean SNR of Noise ~ 20

Network Architecture

- Networks are based on Autoencoder technique.
- 1D convolutional layers are used with Max-pooling and Up-sampling layers to create encoding and decoding layers respectively.
- Classifier also includes Flattening and Dense layers.
- ReLu activation function is used in all except the last layer which uses Sigmoid (for Classifier) and Linear (for Denoiser) activation function.

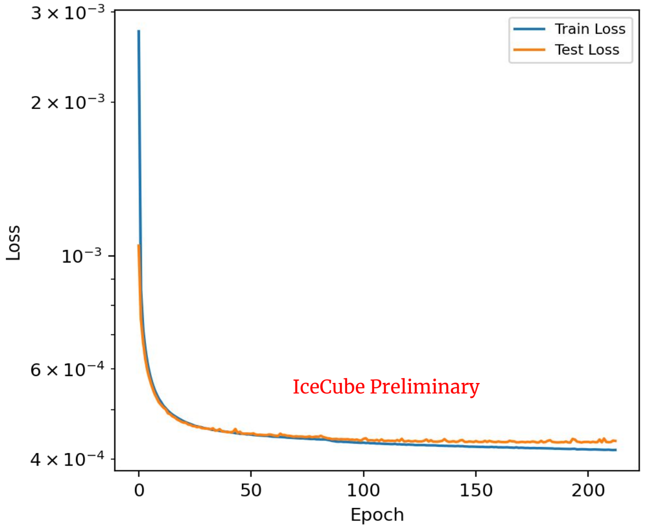


Model Architecture

Learning Curves

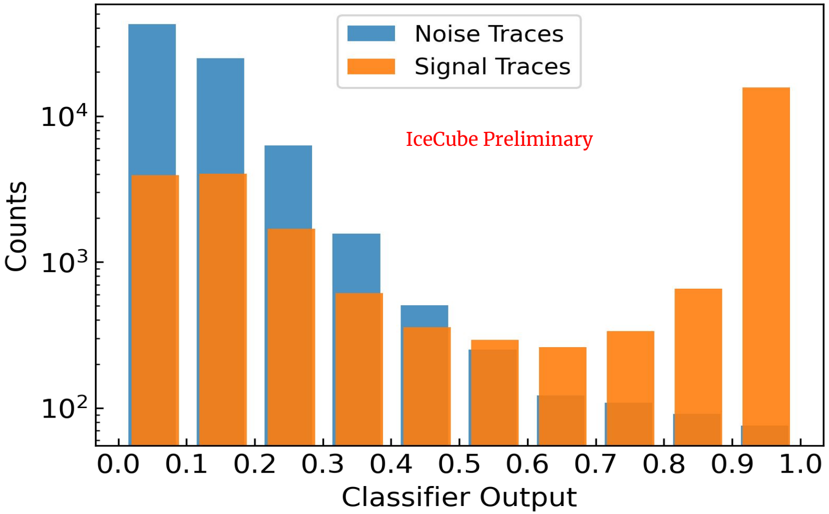
- Graph of loss curves for both training and testing set.
- Mean squared error (MSE) is used as a loss function.
- Early stopping is used:
  - Training is stopped if test loss is not decreased after 20 epochs.
- Same graph was observed for the Denoiser as well.
- The network are not overfitting (No significant difference in training and test loss).

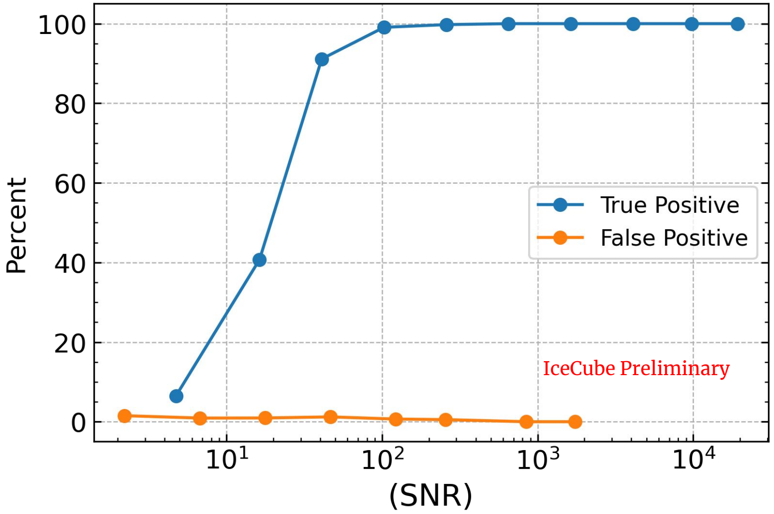
Classifier Loss



Classifier Results

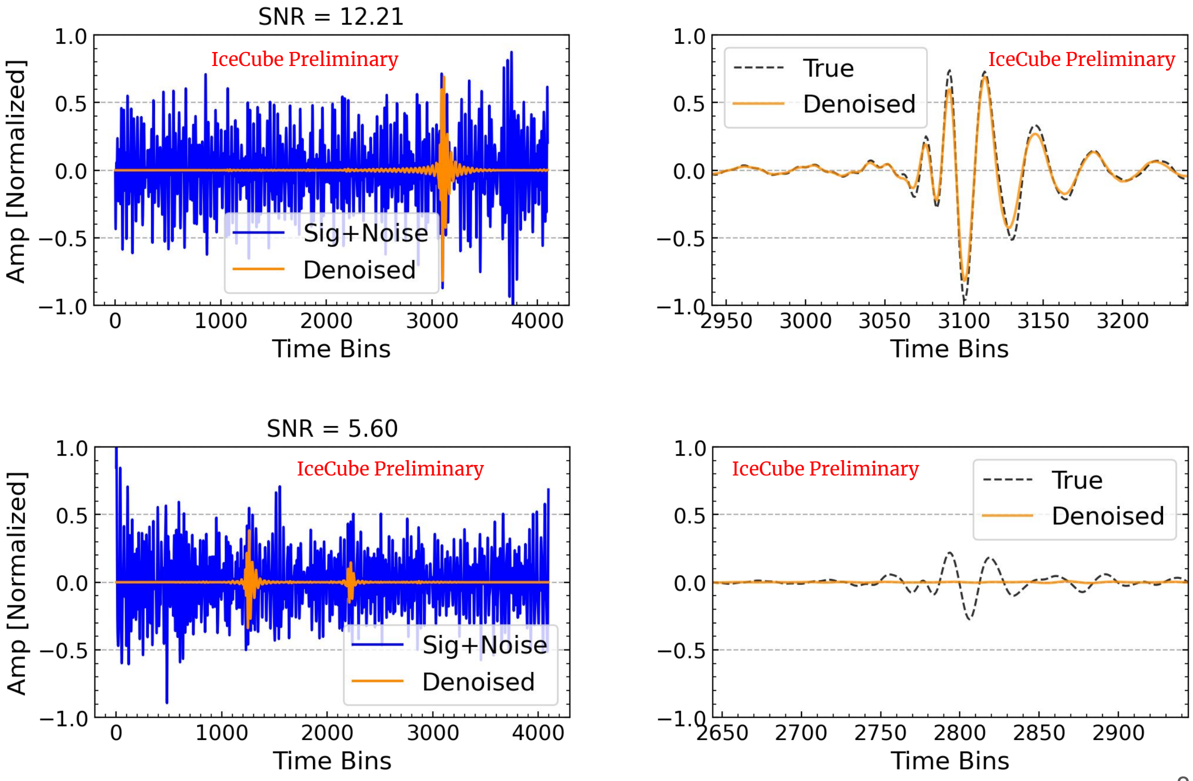
- ➢ **Best Classifier** → Layers=3, Filters=8, KernelSize~30-40ns.
- ➢ Classifier (CL) output shown in the bottom left plot. (Used **Sigmoid activation** function)
- ➢ A threshold cut on CL output ≥ **0.5** are used to make future plots.
- ➢ TP and FP rates (in percent), shown in the right plot.





Denoiser Results

- ➢ Classified Signal traces are passed to the Denoiser for cleaning.
- ➢ Example denoised traces shown here.
- ➢ 1st row shows a good denoising example where 2nd row shows a worse example.



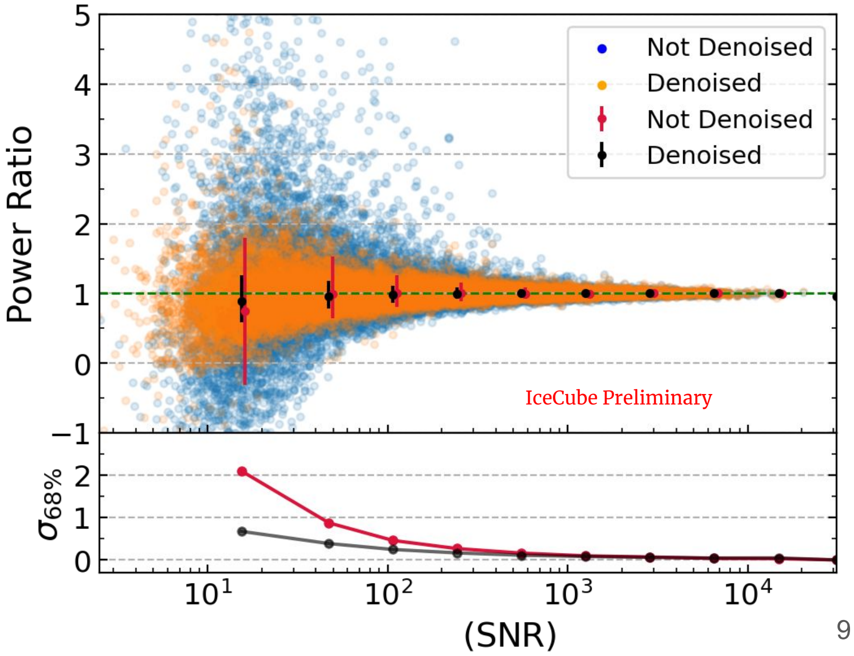
Accuracy Metrics

To get the accuracy of Denoiser we compute two quantities.

Shown here is the **Power ratio**, which is defined as:

$$\text{Power Ratio} = \frac{[P_S - P_N]_{\text{Measured}}}{[P_S - P_N]_{\text{True}}} .$$

P_S , P_N = power in the signal and noise window, respectively.

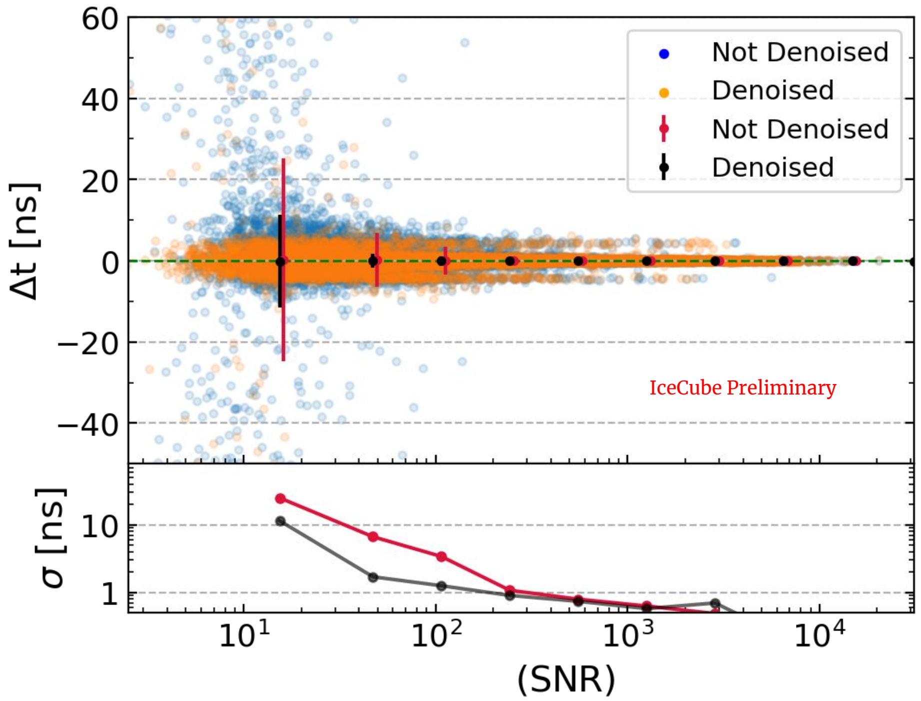


Accuracy Metrics

Shown here is the **Peak Time difference**, which is defined as:

$$\Delta t = T_{\text{measured}} - T_{\text{true}} \ .$$

Peak time is computed as the time of the hilbert peak

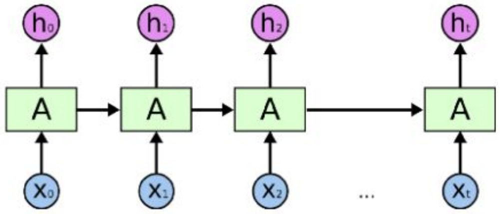


Summary & Outlook

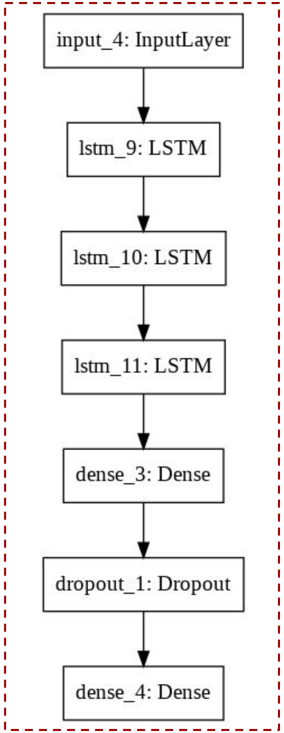
- ❑ Deep learning techniques are being used to mitigate the effect of background on the radio detection of cosmic-ray air showers.
- ❑ Background data from the prototype station at the South Pole is used.
- ❑ Many networks like CNNs and LSTMs are created and tested.
- ❑ Convolutional neural network with autoencoder architecture produce the best result.
- ❑ As a next step we will apply these trained networks to the recorded data.

Thank you!

Other Networks Tested



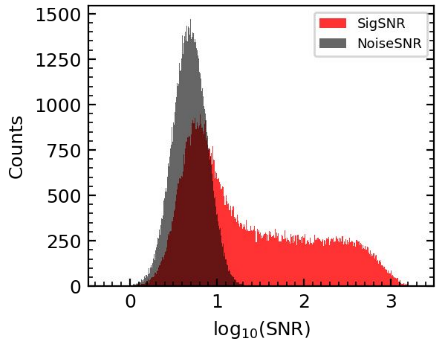
- RNNs process inputs in a **sequential manner**, where the information from the previous input is considered when computing the output of the current step.
- We have used the type of RNN called LSTM.
- LSTM's have the ability to preserve the long-term memory.
- 3 LSTM layers showed stable training but the results were not better in fact they were worse compared than the CNNs.
.

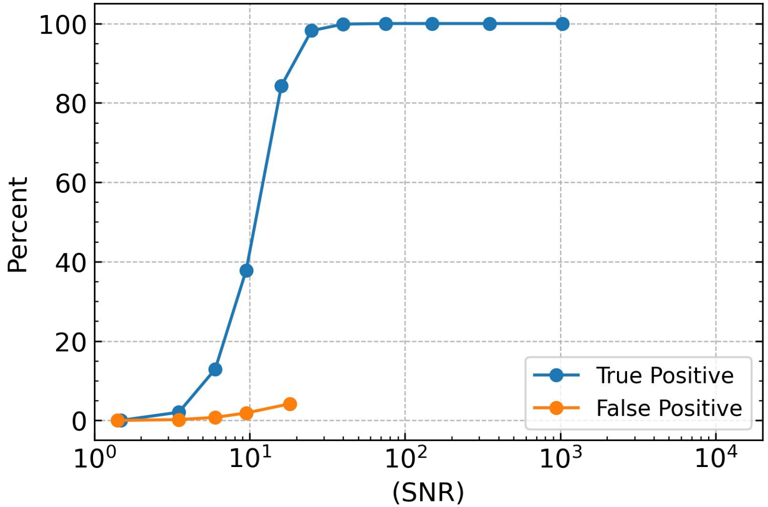


Results with Modeled Noise

- We have shown here results compiled with the modeled (CANE)* noise.
- Freq Band used for CANE noise [50-350]MHz.
- True positive and False positive rates vs SNR show here.

CANE SNR distribution used to train the models.





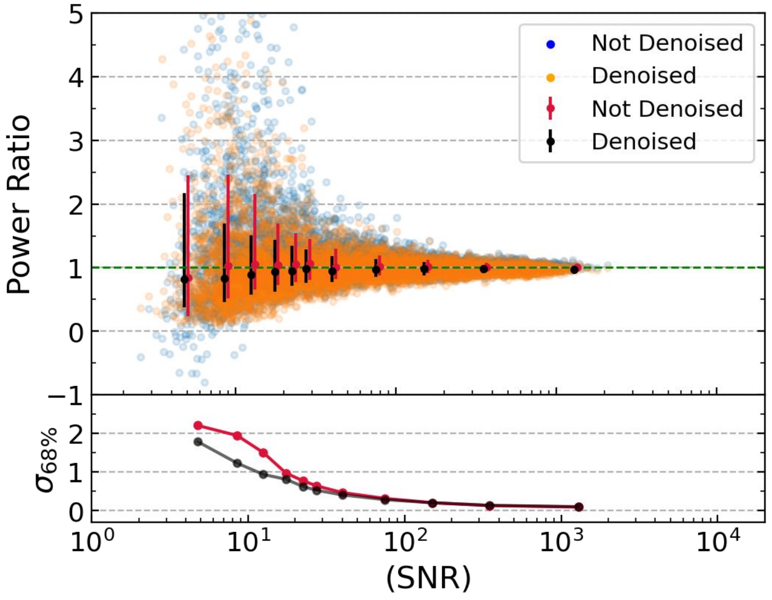
❏ Mean SNR ~ 7

* Please see PoS(ICRC2021)417 for more details on results with Modeled Noise.

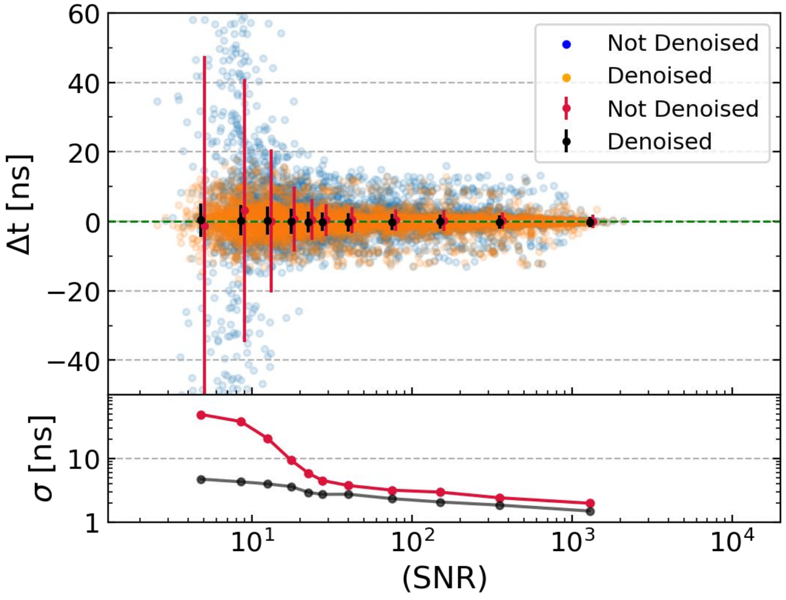
Results with Modeled Noise

Accuracy matrices computed for the modeled Noise.

Power Ratio vs SNR



Peak time difference vs SNR



* Please see PoS(ICRC2021)417 for more details on results with Modeled Noise.