Applying an Unfolding Method with Monte Carlo Simulations of IceCube Electron (anti-)Neutrinos induced Cascade Data

Iowa State University of Science and Technology

Colton Addis

Dr. Joanna Kiryluk, Dr. Zelong Zhang, Zheyang Chen

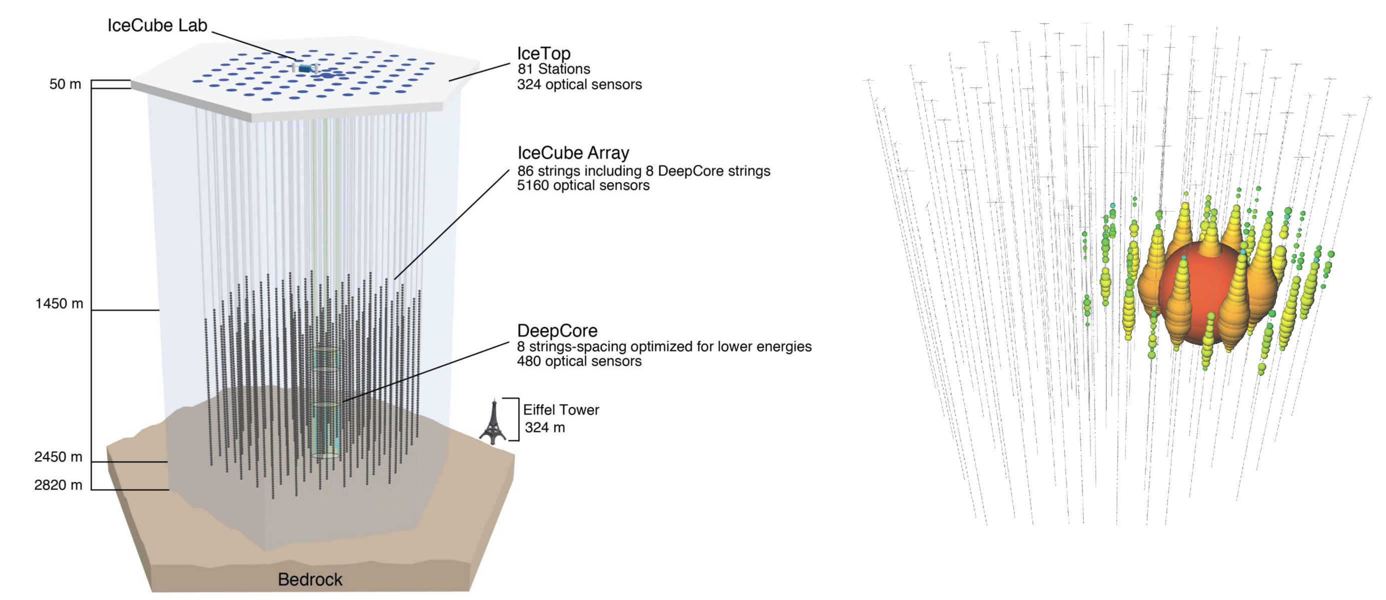
Stony Brook University

Abstract

IceCube Neutrino Observatory is a cubic kilometer detector located at the South Pole in Antarctica. In this project we will utilize Monte Carlo Pass2 high energy neutrino induced cascade data, re-optimize energy and zenith binning and study unfolding of true electron (anti-)neutrino kinematic variables. The neutrino Monte Carlo generators used in NuGen, studies comparing to Lepton Injector are in progess.

Introduction

- IceCube is a km^3 of ice in Antartica, with 5160 optical modules [1] used as a neutrino telescope.
- It detects Cherenkov light produced when a neutrino interacts with quarks and antiquarks in the target nucleons.
- There is a systematic source of error from the process of reconstructing the energy and direction of neutrino from Cherenkov light.
- To recover true variables from these reconstructed ones, unfolding is used.



Credit: IceCube Collaboration

Unfolding

- Unfolding is a process that takes a distribution of reconstructed data and finds the true distribution from that and needs an unfolding matrix.
- To create an unfolding matrix, Monte Carlo simulations of events are used where both true and reconstructed values are known.
- The unfolding matrix is the probability an event in a reconstructed bin is in some true bin.
- Below is an unfolding test with toy data, with the data sampled from 3 normal distributions and a unform background.

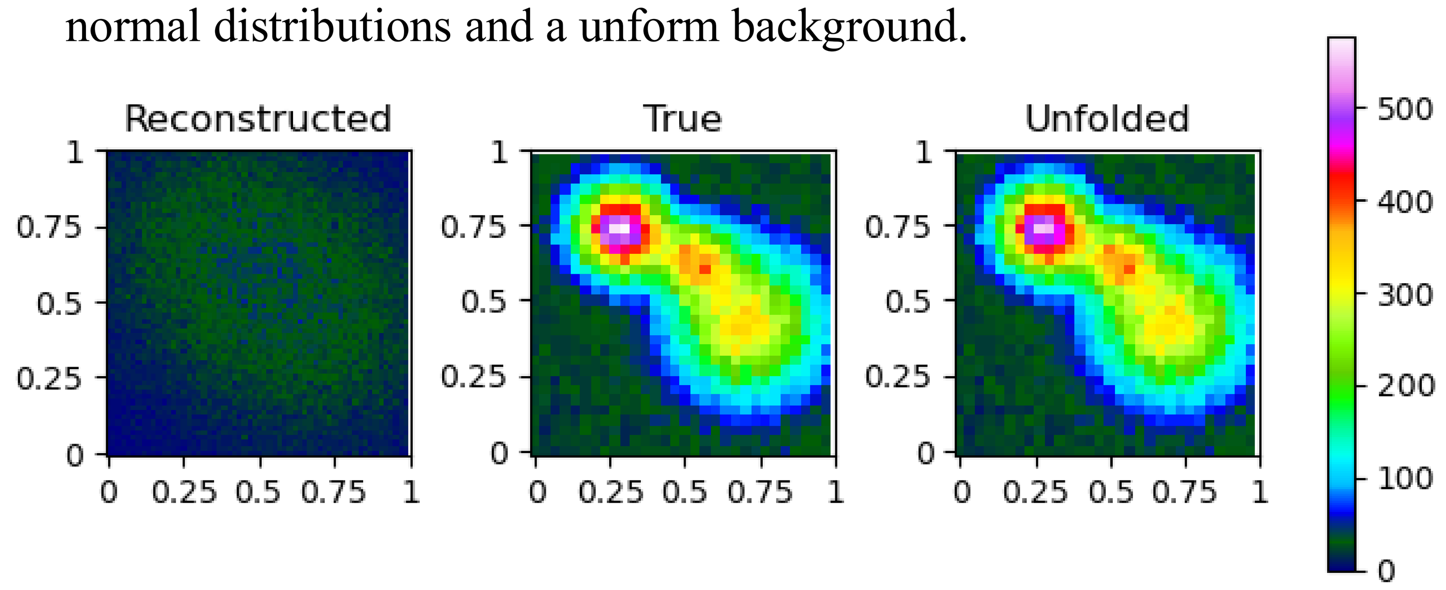


Figure 1: A test of the Richardson-Lucy unfolding algorithm with toy data. The reconstructed data has an assumed resolution of 0.25 and no bias. 100 iterations was used for the unfolding algorithm. The reconstructed data is in 60 x 60 bins, and the true and unfolded data is in 30 x 30 bins.

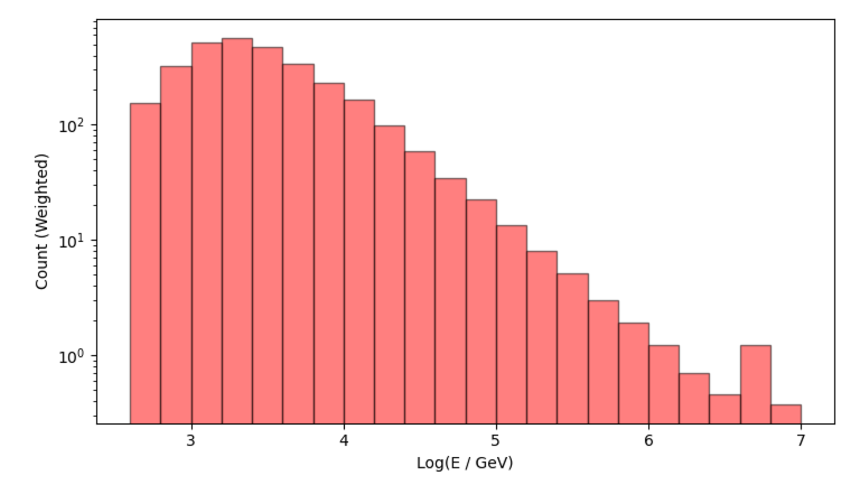
The Unfolding Algorithm

- The Richardson-Lucy unfolding algorithm [2] is used.
- The unfolding matrix (A) is created from the probability a value in a true bin j is in some reconstructed bin i.
- It starts with a guess of the true distribution, then in iterations makes an estimate of the reconstructed distribution (d_i) from the current unfolded distribution ($\hat{\theta}_j$) and the unfolding matrix.
- Each iteration the algorithm attempts to minimize the difference between the estimate and unfolded distributions.
- α is a vector that represents the acceptance loss when making the unfolding matrix.

$$d_i^{(k)} = \sum_{j=1}^{N} A_{ij}\hat{\theta}_j^{(k)} . \qquad \hat{\theta}_j^{(k+1)} = \sum_{i=1}^{M} A_{ij}\hat{\theta}_j^{(k)} \frac{d_i}{d_i^{(k)}} / \alpha_j .$$

Binning Optimization

- The parameters of interest are the energy of the neutrino and zenith angle. They are unfolded together as a 2D histogram, with log(E) as the x-axis and cos(Z) as the y-axis.
- A bin width of .2 was chosen for log(E), corresponding to 22 bins
- A bin width of .5, or 4 bins, was chosen for cos(Z)
- This was done by looking at the resolution of the reconstructed data, and by looking at the minimum number of events for the bins on the edge of the histogram.
- The graph on the right shows the energy data binned, with a timespan of 11 years, as it is for unfolding.



Results on NuGen Monte Carlo Simulated IceCube Data

- Using optimized number of bins and iterations, the result is below.
- A timespan of 11 years was used when weighting.
- The results were tested against a χ² distribution and a good agreement was found.

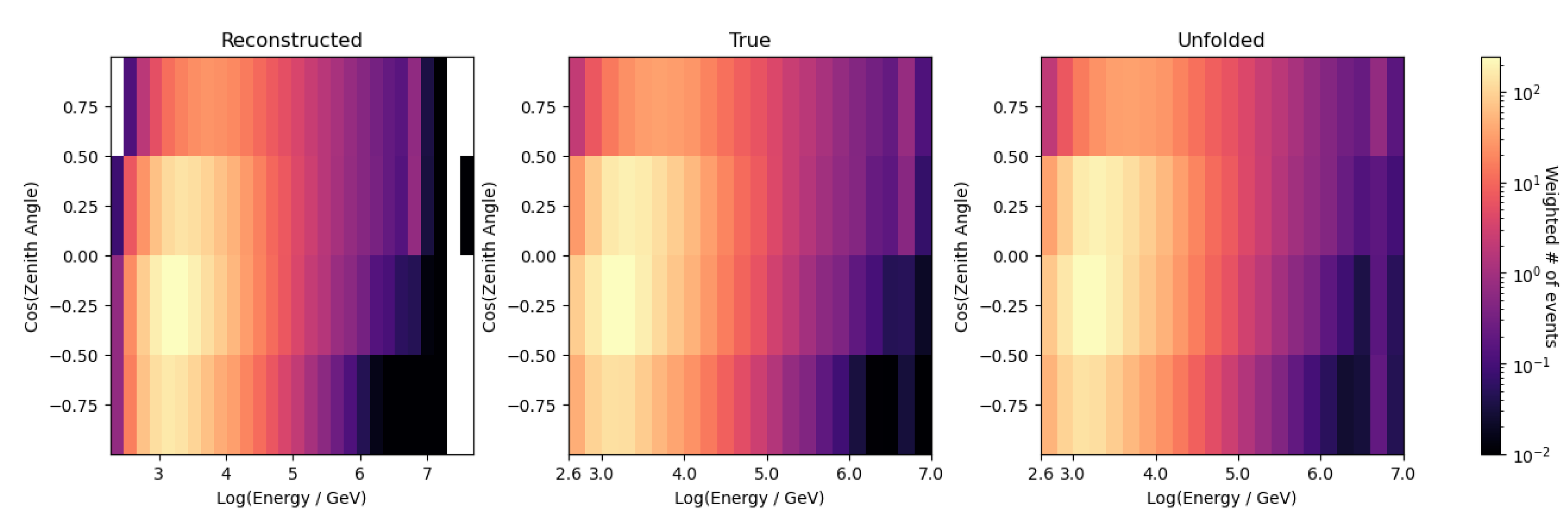


Figure 2: The final results of unfolding the Monte Carlo simulated data. The bin width here is .2 for log(E) space, and .5 for cos(Z) space. 4 iterations of the unfolding algorithm were run.

Optimizing Unfolding Iterations

The number of iterations with a small bias, which decreases through iterations, and a small variance, which increases through iterations, needs to be found. To do this, a log-likelihood ratio is used to estimate the bias from the reconstructed values, because the true values are unknown [3]. The difference between the reconstructed and unfolded is found, and when the log-likelihood passes a critical value the iterations are stopped. To get the critical value, a Poisson sample is taken from the reconstructed 100,000 times, and the log-likelihood is found for each time. When the log-likelihood passes half of that distribution, where p = 0.5, double the iterations is run. This results in 4 iterations being run when used on the simulated data.

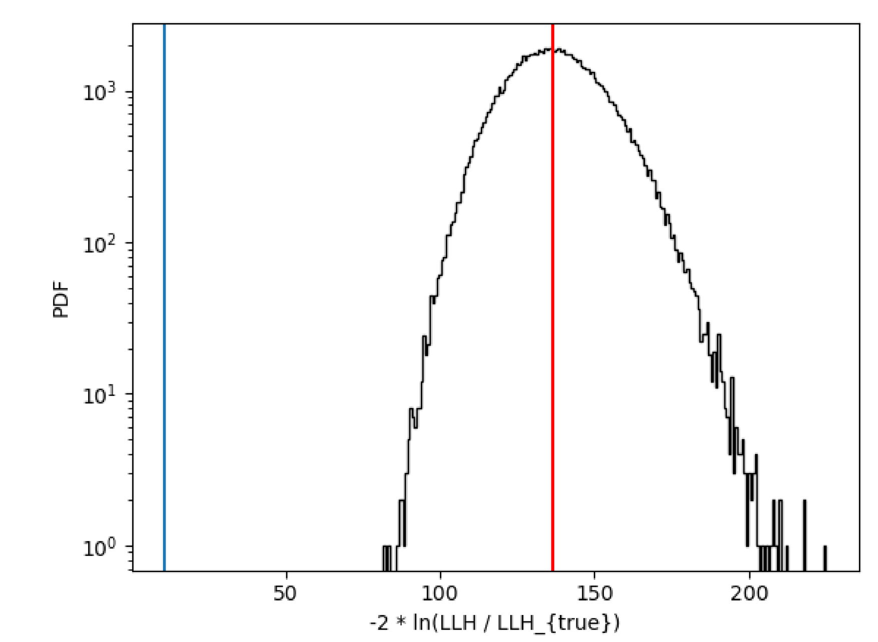


Figure 3: Graphing showing sampling log-likelihood ratio (black), and the log-likelihood ratio of the unfolding after 2 iterations. Since the unfolding is past the critical value (red), the algorithm will run double the iterations and stop

Discussion/Conclusion

- The unfolding algorithm takes a reconstructed distribution and finds the true distribution associated with it.
- To create the unfolding matrix, NuGen Monte Carlo simulations where the true values are known were used. Studies with Lepton Injector are in progress.
- Bin sizes were found by looking at resolution and minimum statistics.
- Iterations count was found from log-likelihood ratios minimizing bias while keeping variance low.
- Future work could be done to implement error analysis when unfolding.
- The algorithm could also be tested with different flavors of neutrinos.

References

[1] F. Halzen, S.R. Klein, IceCube: An Instrument for Neutrino Astronomy, (2010) . https://doi.org/10.48550/arXiv.1007.1247.

[2] G. Zech, Iterative unfolding with the Richardson-Lucy algorithm, (2013). https://doi.org/10.48550/arXiv.1210.5177.

[3] Y. Xu, Measurement of the High Energy Neutrino-Nucleon Cross Section Using Neutrino-Induced Electromagnetic and Hadronic Showers Observed in Five Years of IceCube Data (2019)

Acknowledgements

Thanks to IceCube Stony Brook Group for data and support, and thanks to my mentor, Dr. Kiryluk

IceCube Neutrino Observatory