Uncertainty Quantification

in Reconstruction in the DUNE-ND

D. Douglas
For the SLAC Neutrino Group & ML Initiative

Neutrino Physics and Machine Learning

August 23, 2023

Uncertainty Quantification

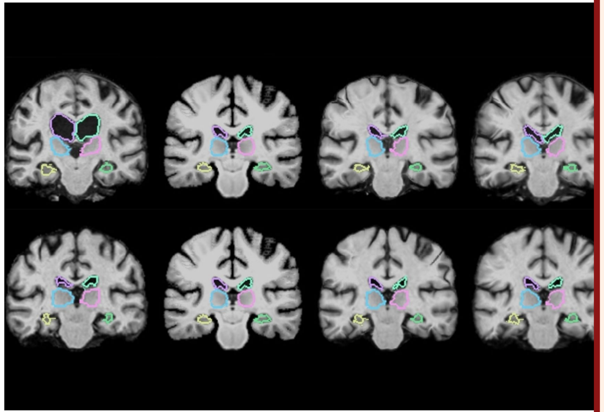
Why UQ in ML models?

In short: high-regret problems

Faster analysis of medical images

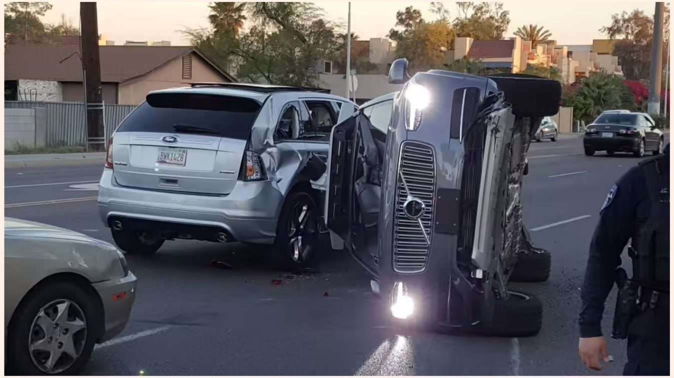
Algorithm makes the process of comparing 3-D scans up to 1,000 times fast

Rob Matheson | MIT News Office
June 18, 2018



Uber self-driving car crashes during US tests

Collision threatens plans to bring high levels of autonomy into commercial use

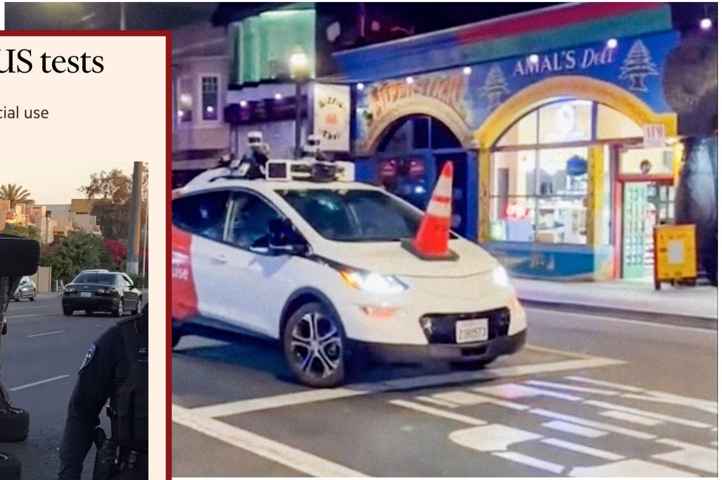


The Uber SUV on its side following the accident in Tempe, Arizona © Reuters

The Self-Driving Cars Wearing a Cone of Shame

There's a brilliant activist campaign to stop San Francisco's autonomous vehicles in their tracks.

BY ALISON GRISWOLD JULY 11, 2023 • 10:45 AM



n, in a way, it is). Screengrab from TikTok/Safe Street Rebel

Uncertainty in ML

<u>Aleatoric</u> - Uncertainty arising from inherent randomness of sampling.

Aleator - one who rolls the dice

<u>Epistemic</u> - Uncertainty arising from choices of model which do not fully describe a modelled process

https://arxiv.org/abs/1703.04977


arXiv > cs > arXiv:1703.04977

Search...

Help | Advanc

Computer Science > Computer Vision and Pattern Recognition

[Submitted on 15 Mar 2017 (v1), last revised 5 Oct 2017 (this version, v2)]

What Uncertainties Do We Need in Bayesian Deep Learning for Computer Vision?

Alex Kendall, Yarin Gal

There are two major types of uncertainty one can model. Aleatoric uncertainty captures noise inherent in the observations. On the other hand, epistemic uncertainty accounts for

How to Characterize Uncertainty in a Model

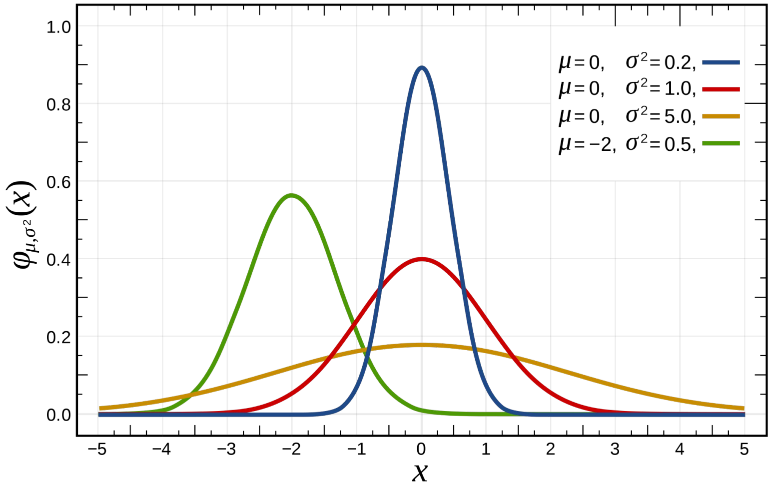
Aleatoric - The model can predict a distribution which maps out the underlying distribution of the training set instead of a modal outcome – Probabilistic Neural Network

Epistemic - One method of approximating a bayesian NN: the model can be altered at inference time by enabling high-probability (50%) dropout layers – Monte Carlo Dropout

What is a PNN?

A Probabilistic Neural Network (PNN) aims to predict an underlying distribution from which a given image might be sampled. This is a way of estimating heteroscedastic (different for each input) aleatoric uncertainty

This is implemented by predicting both a mean and an uncertainty term for each voxel and using a NLL loss

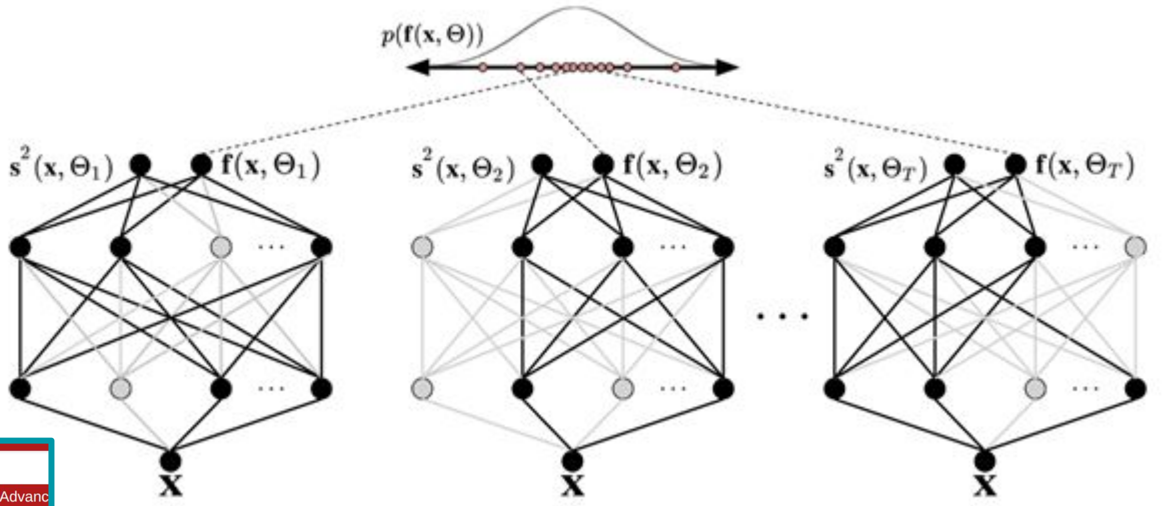


predicted

$$\log \mathfrak{L} = \sum_{i \in \text{voxels}} -\frac{1}{2}\frac{(\boxed{\hat{E}_i} - E_i)^2}{\boxed{\hat{\sigma}_i^2}} - \log\boxed{(\hat{\sigma}_i)}$$

What is Monte Carlo Dropout?

Monte Carlo Dropout is a method for stochastically changing your model in order to approximate a posterior distribution of a model's prediction

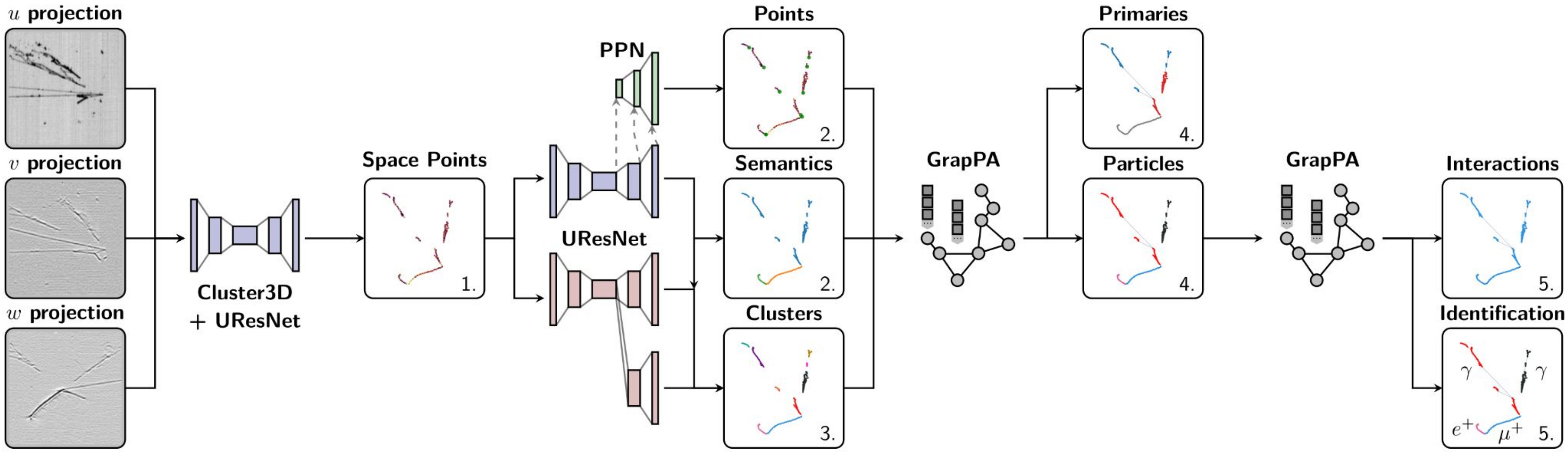


https://arxiv.org/abs/1506.02142

ML in LArTPC's

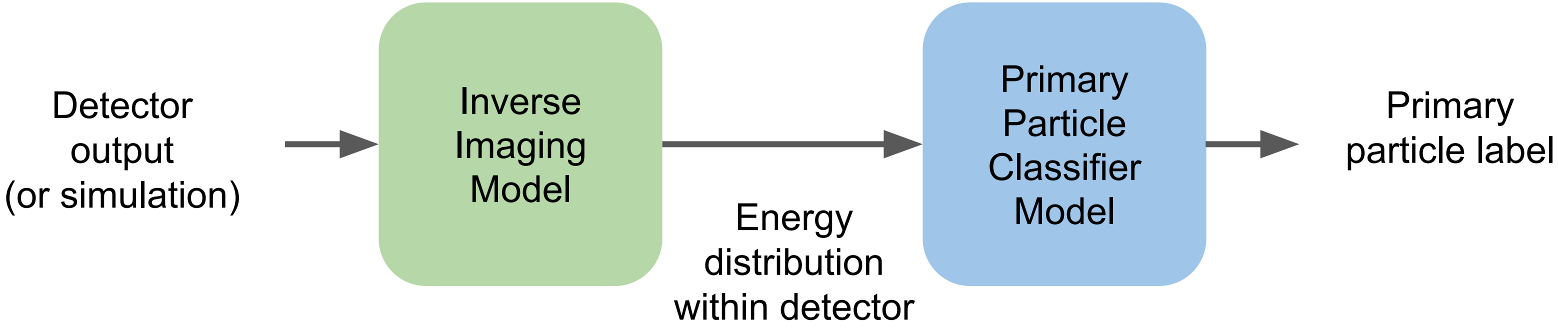
How Confident Are We?

Our reco chain is models after models after models after models…



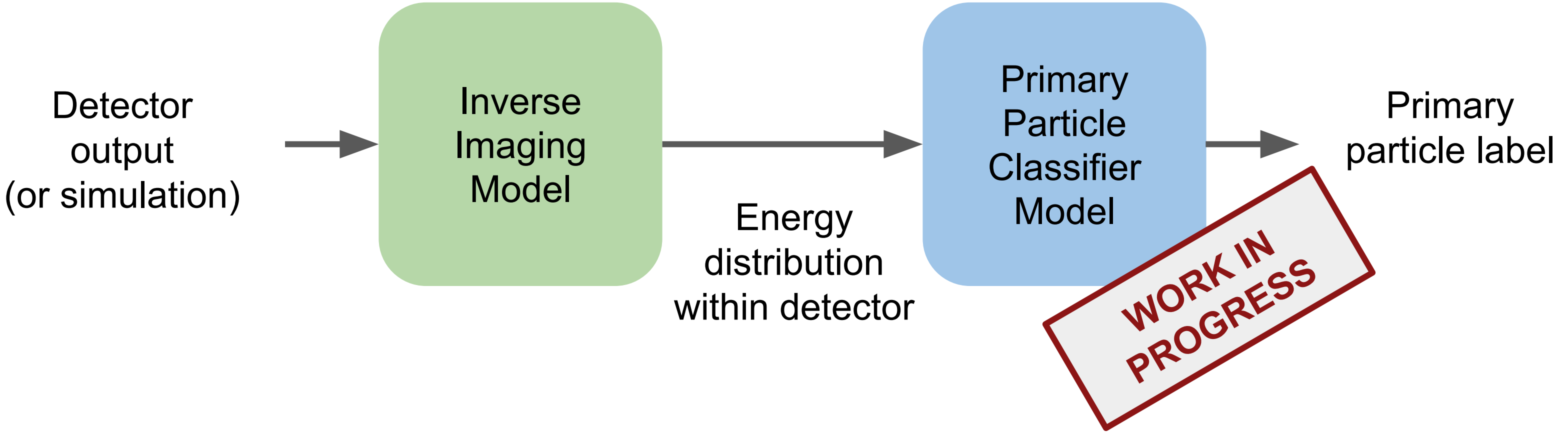
Let’s Simplify: 2-Model UQ

Question: Does including uncertainty in the predictions of an upstream model *at training time* improve the accuracy or robustness of a downstream reconstruction model?



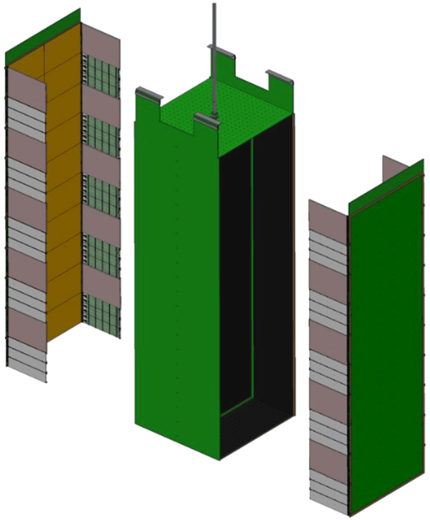
Let's Simplify: 2-Model UQ

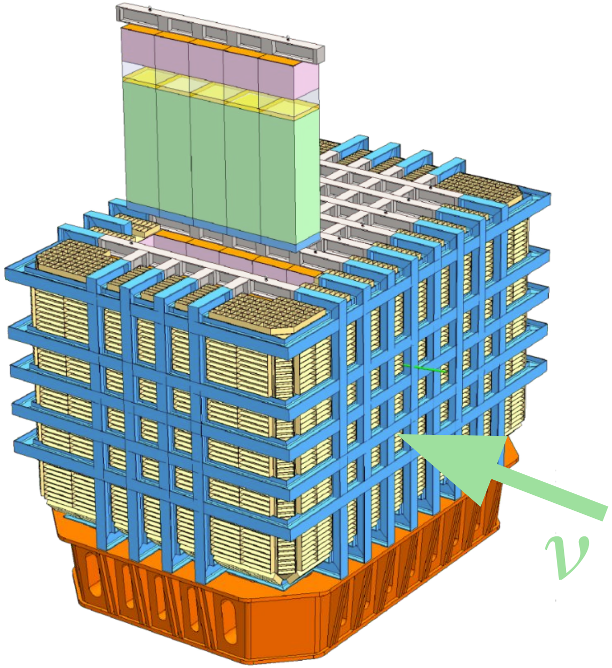
Question: Does including uncertainty in the predictions of an upstream model *at training time* improve the accuracy or robustness of a downstream reconstruction model?



`larnd-sim` Inverse Solver Model

The DUNE Near Detector LArTPC

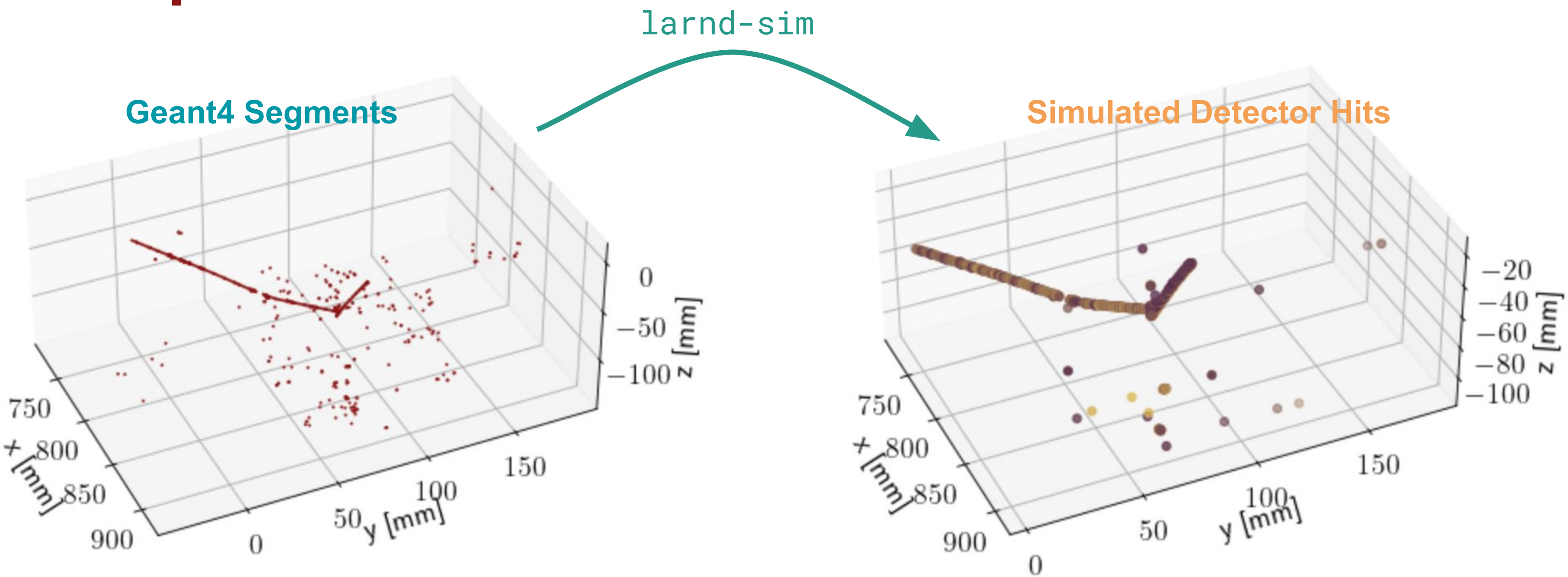




Consists of a 5 × 7 array of dual-TPC modules positioned ~570m from the neutrino beam source at FNAL.

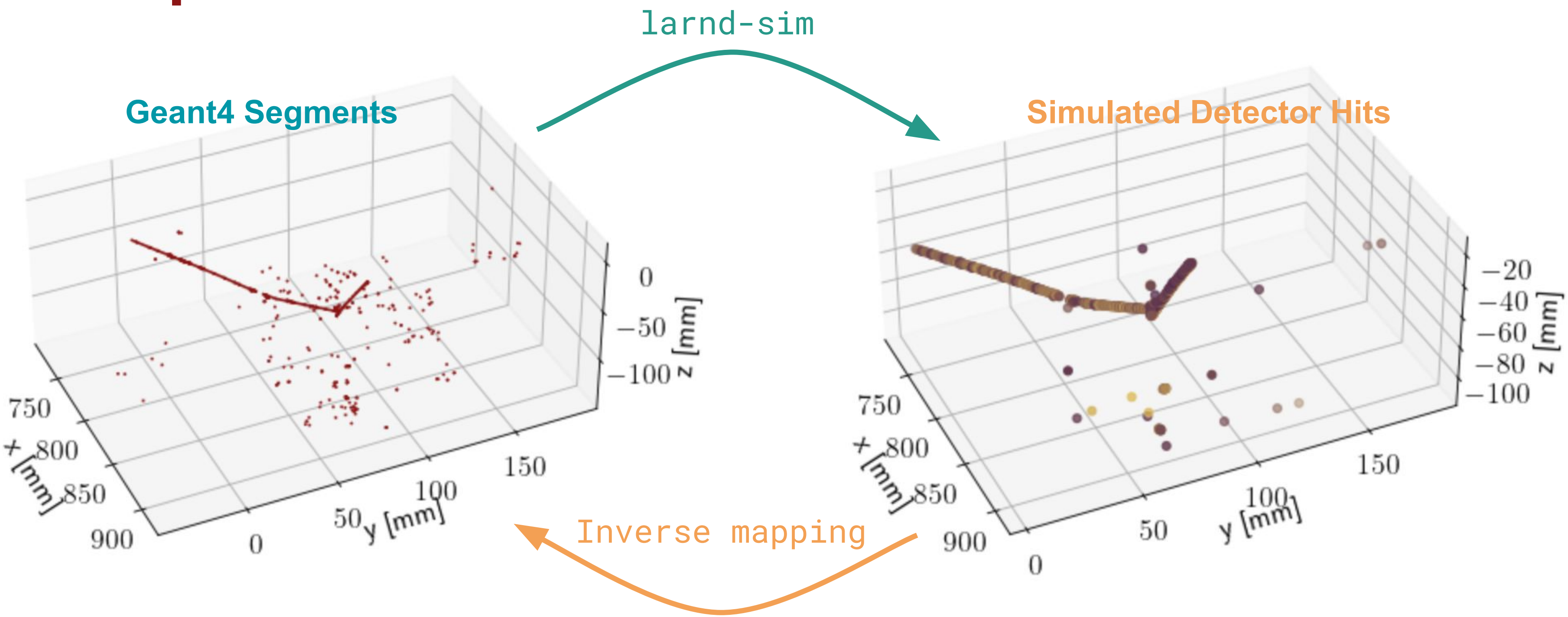
Each 1m × 1m × 3m module has two anode planes made of 4.434 mm square pixels.

The Spaces



larnd-sim
Geant4 Segments
Simulated Detector Hits
x [mm]
750
800
850
900
y [mm]
0
50
100
150
z [mm]
0
−50
−100
−20
−40
−60
−80

The Spaces

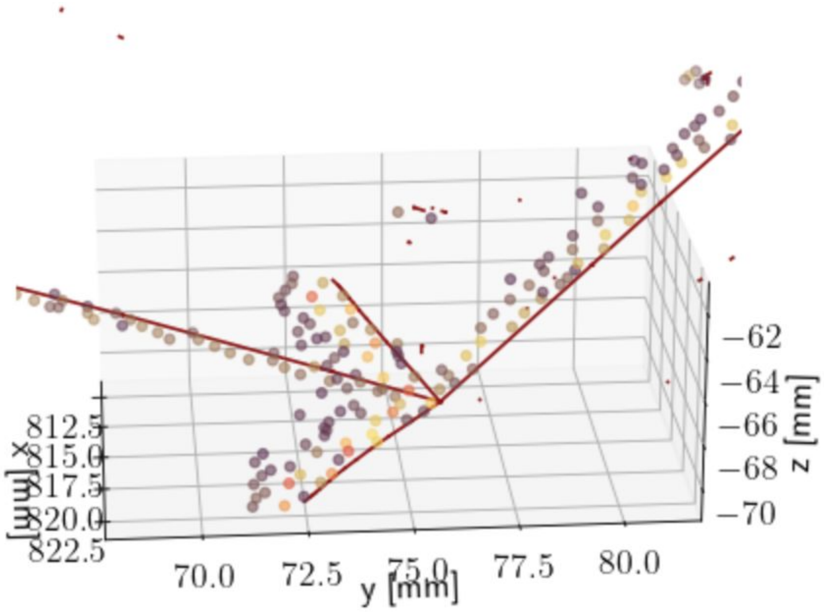


larnd-sim
Geant4 Segments
Simulated Detector Hits
Inverse mapping
x [mm]
750
800
850
900
y [mm]
0
50
100
150
z [mm]
0
−50
−100
x [mm]
750
800
850
900
y [mm]
0
50
100
150
z [mm]
−20
−40
−60
−80
−100

Building the Training Data

Hits (scatter points) and Geant4 segments (lines) are not perfectly aligned, particularly along the drift direction, where induction tends to produce “early” (~0.5 mm) hits.

To accommodate the mismatch in domains of the sparse images, zero-padding is added around the two point clouds (simulated hits and voxelized Geant4 tracks)



Training Set

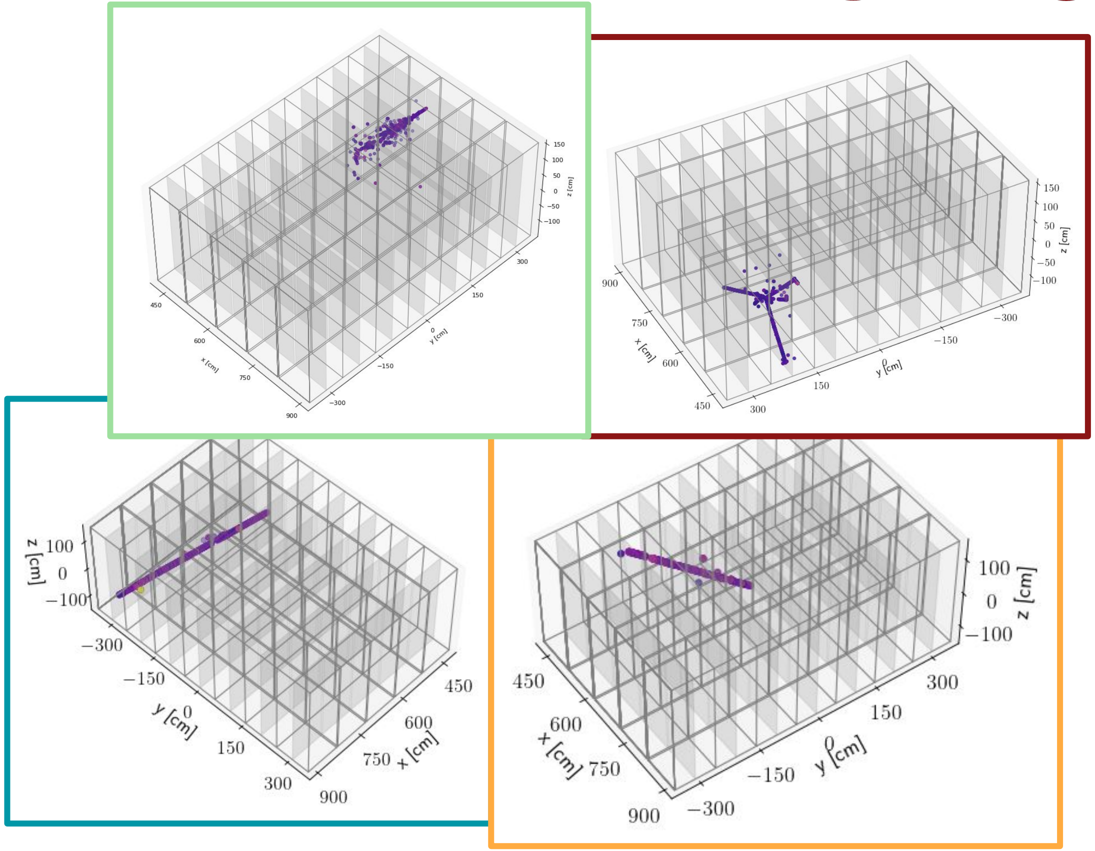
Training on single-primary-particle images in DUNE ND-LAr geometry

Primary particles are:

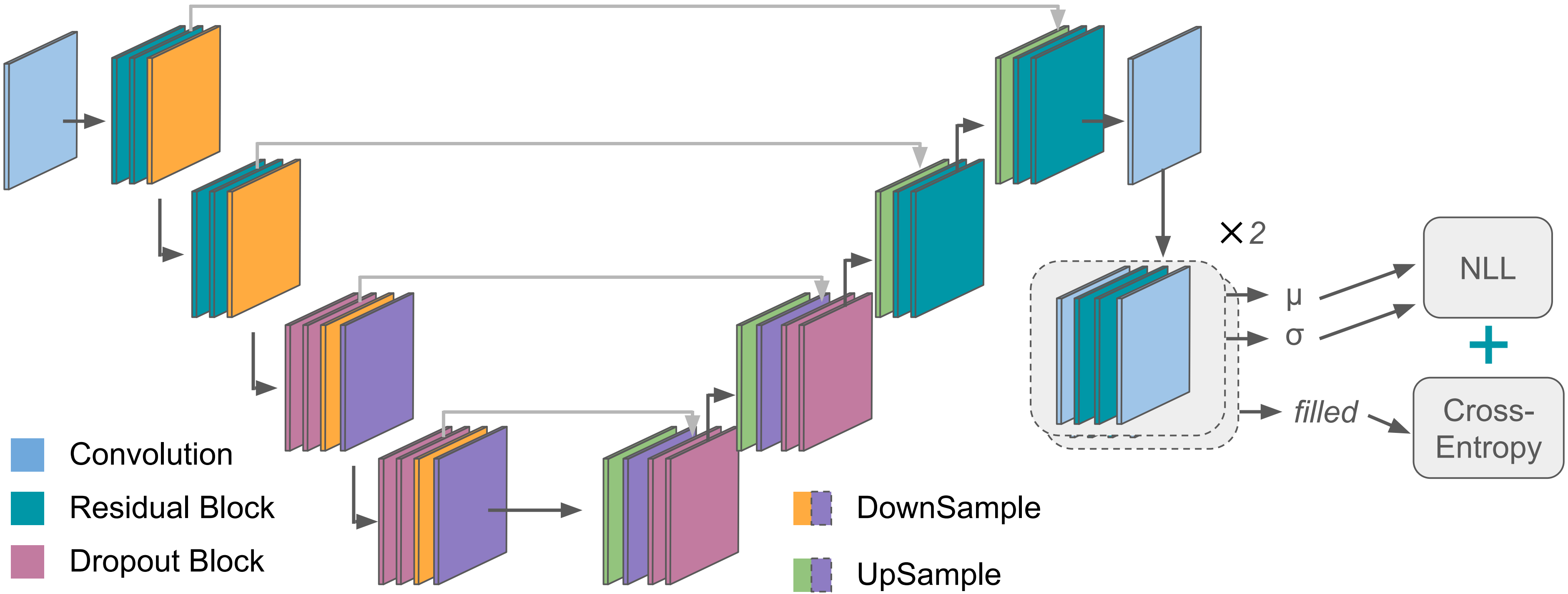
π+, γ, e, μ, p

Primary Energies of 20 MeV - 500 MeV

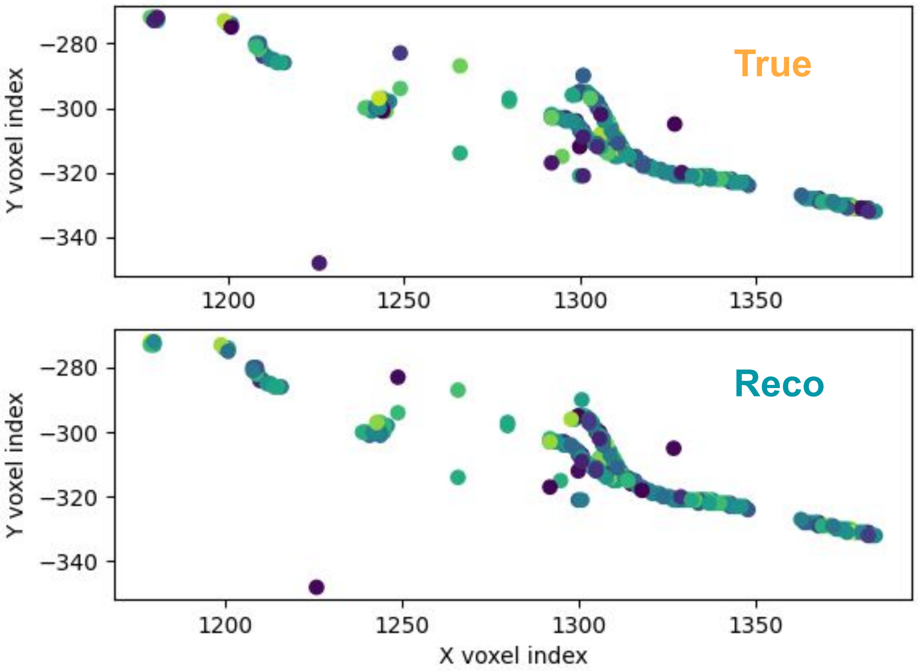
Ground Truth is the voxelized (pixel-sized voxels) G4 input



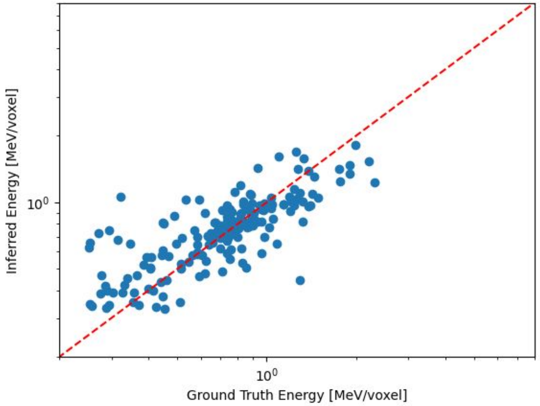
Upstream Model: Inverse Imaging DUNE ND-LAr



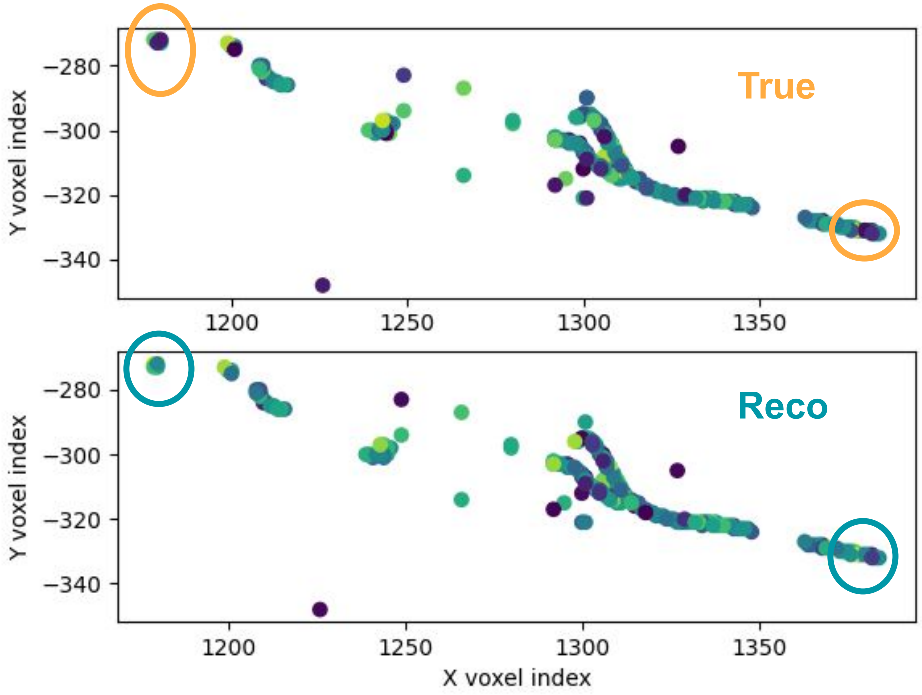
Running the Inverse Model on a Single Event



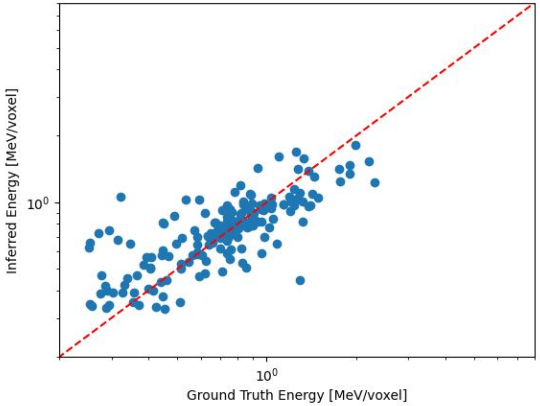
Voxel-to-voxel inference works well overall, but fails in areas with high levels of stochasticity, and disconnected blips



Running the Inverse Model on a Single Event



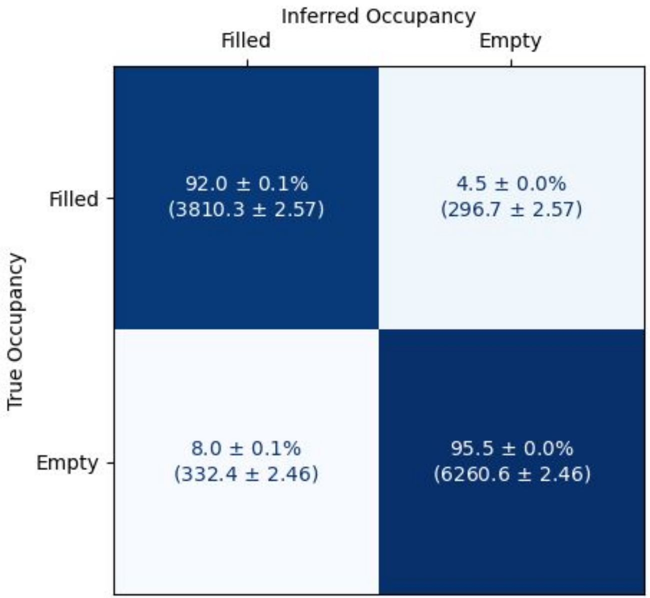
Voxel-to-voxel inference works well overall, but fails in areas with high levels of stochasticity, and disconnected blips



Voxel Occupancy

Occupancy output quickly converges to 92% true positive rate

The epistemic uncertainty on this classification scheme (derived from MC Dropout) is ~0.1%



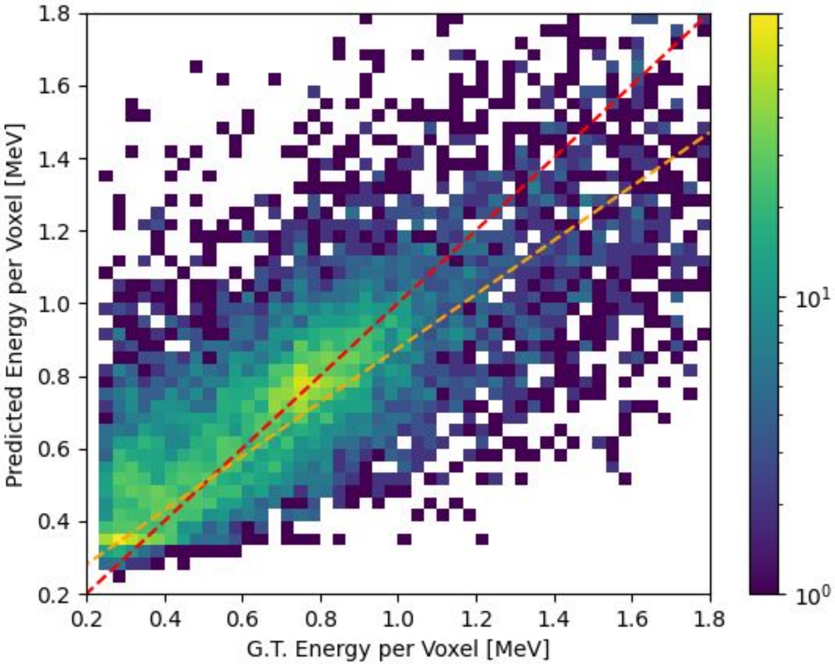
Distribution of Regressed Voxel Means

Good voxel-to-voxel matching with some obvious errors

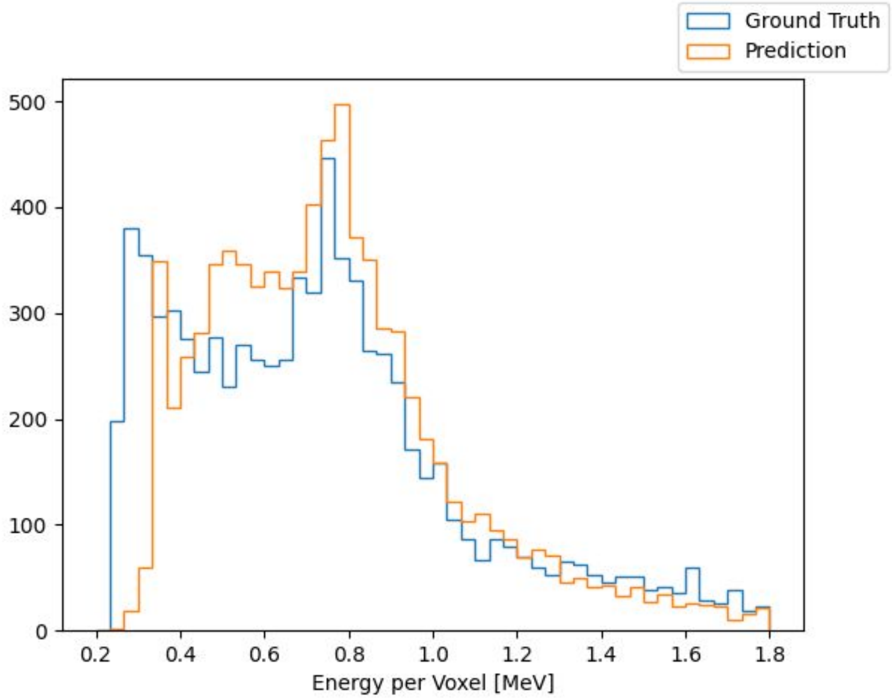
low-G.T. voxels tend to be over-predicted

Higher- (and less common) valued G.T. voxels are under-predicted

This seems to be a regression towards the mean value



1-D Spectra of Voxels



Some specific features which are hard to map out

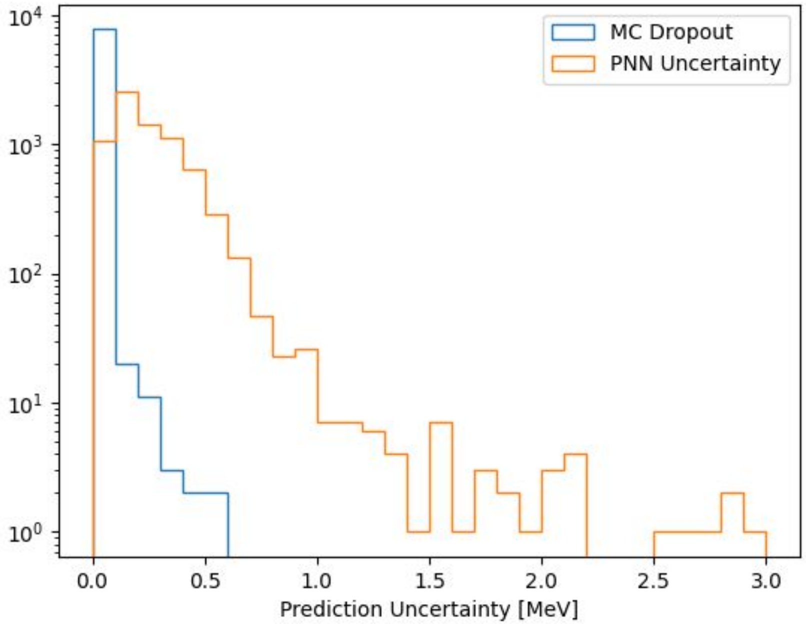
MIP peak is over-populated in prediction

Above-threshold continuum voxels show a feature (distribution of voxel crossing angles) which is not well-replicated by the model

Regressed Voxel Uncertainties

Comparing MC Dropout-derived uncertainty and PNN output uncertainty indicates that aleatoric uncertainty is the dominant source of prediction error for this model

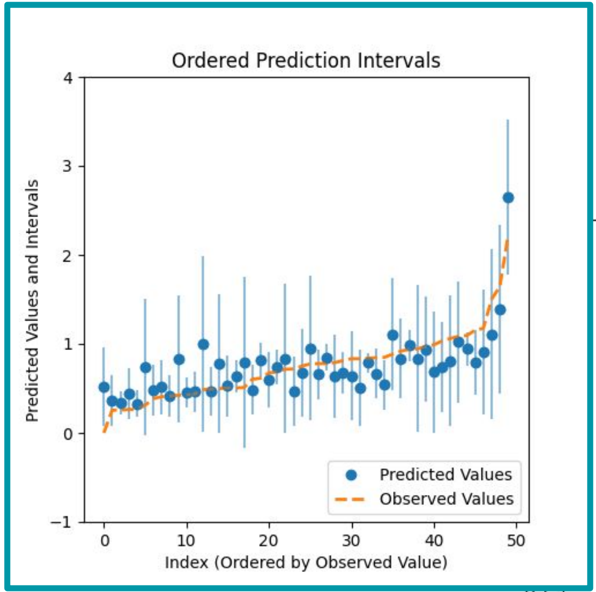
This is typical for regression-type problems

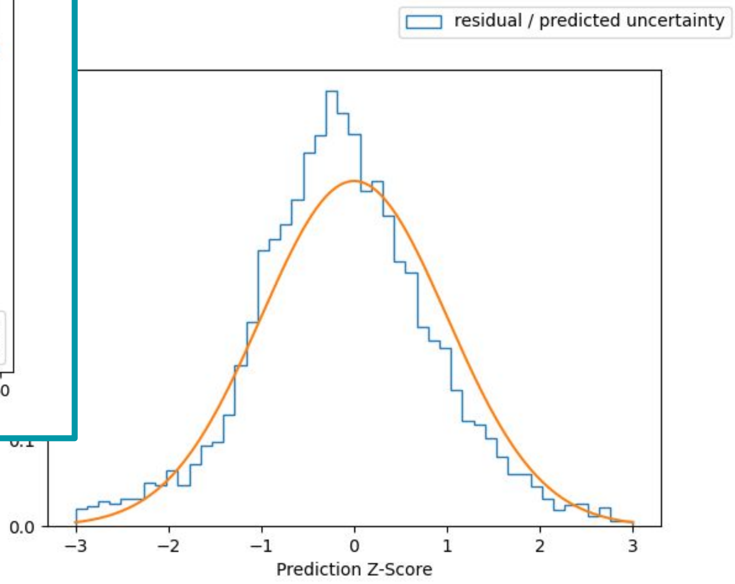


Regressed Voxel Uncertainties

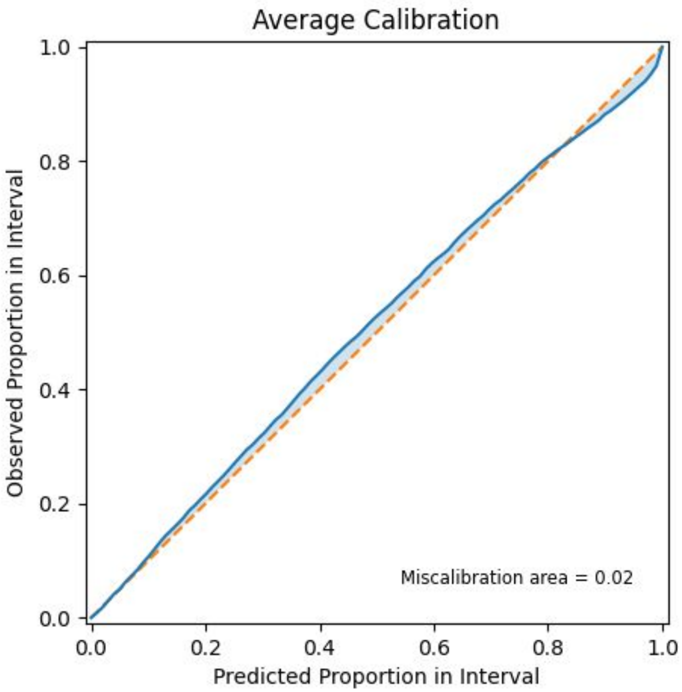
The model predicts appropriately large uncertainties when it is unsure of a voxel's value

Errors are not *exactly* normally distributed





Predicted Uncertainty Calibration



This model is well-calibrated for non-zero voxels,

but lacks sharpness

Accuracy Metrics

MAE	0.224
RMSE	0.411
MDAE	0.125
MARPD	31.391
R2	0.621
Correlation	0.791

Sharpness Metrics

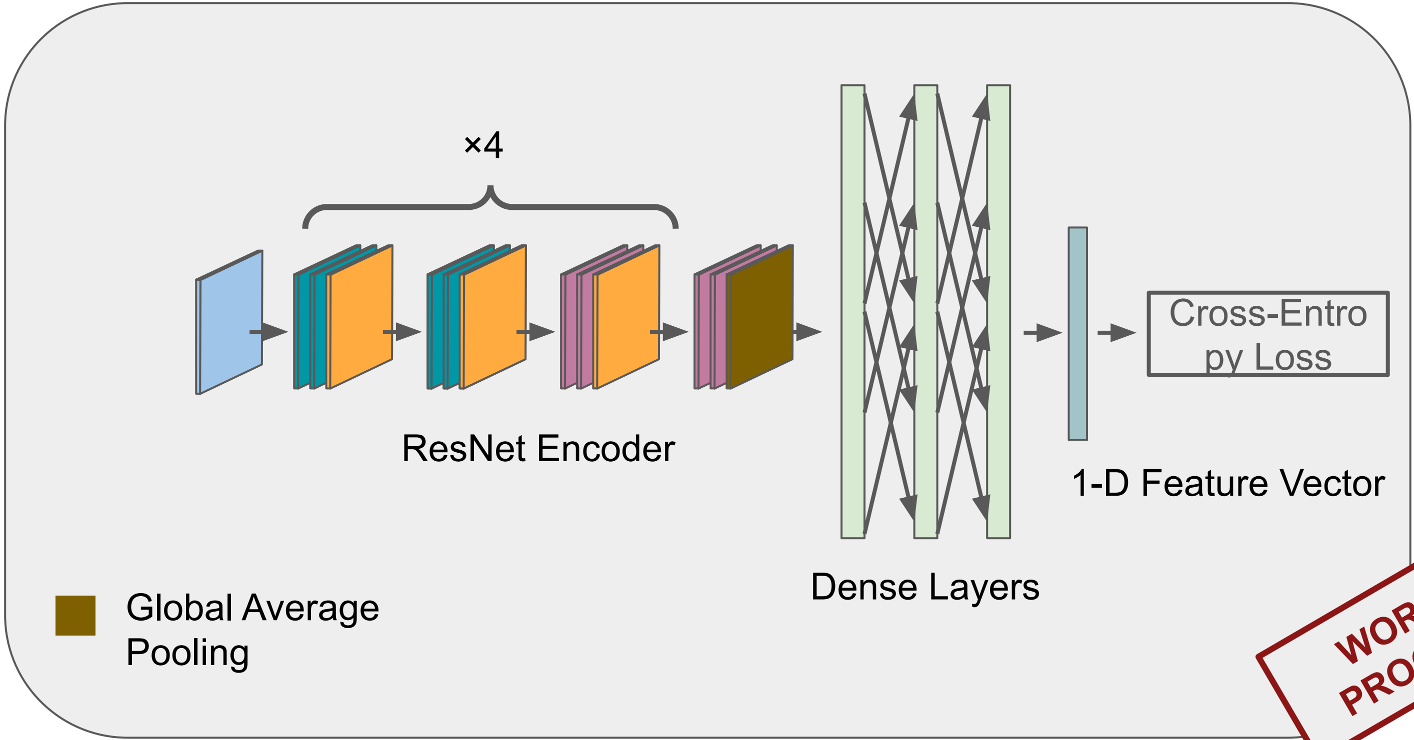
Sharpness	0.439

Scoring Rule Metrics

Negative-log-likelihood	-0.020
CRPS	0.163
Check Score	0.082
Interval Score	0.834

https://github.com/uncertainty-toolbox/uncertainty-toolbox

Next Up: Uncertainty-Enabled Reco Model



Downstream analysis model is still under construction!

Please stay tuned!

WORK IN PROGRESS

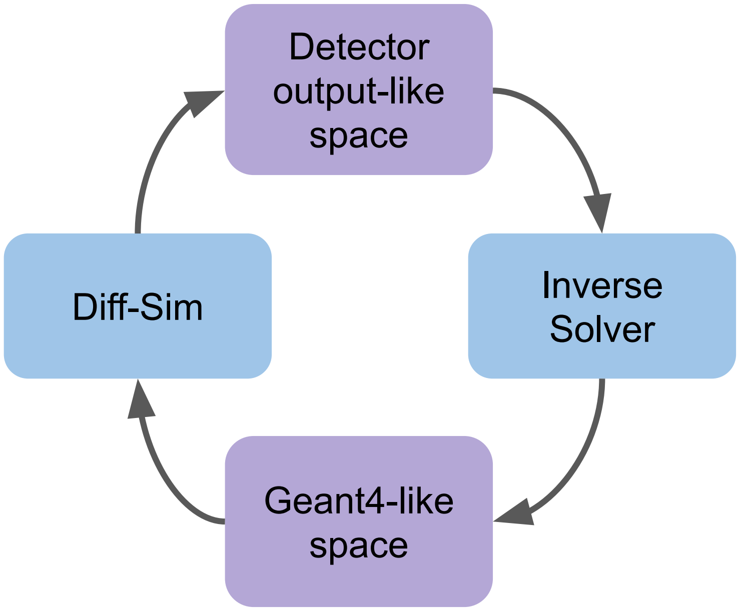
Downstream Applications Of Inverse Model

Unsupervised Learning with Differentiable Sim

Using a simulator with auto-differentiation (See Yifan Chen's talk earlier today),

A network like this can be trained unsupervised, evaluating losses in an Geant4-like space, and propagating gradients through the simulator

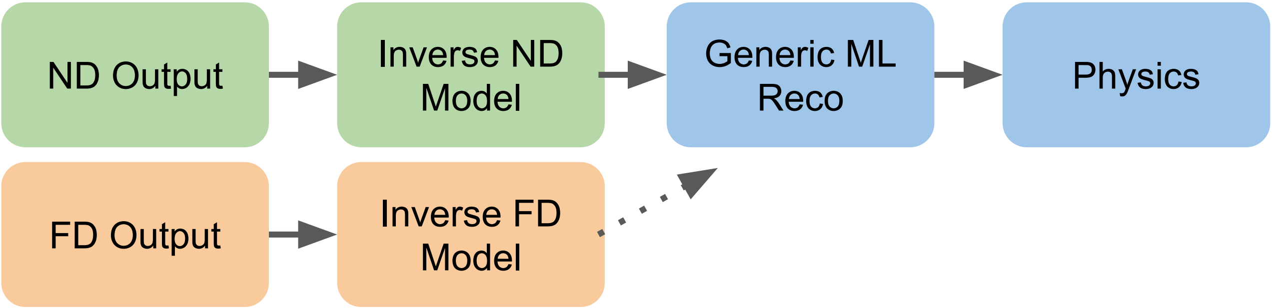
Alternatively, we can deploy this loop directly on data! Must be careful to avoid inducing bias in calibration, etc.



Detector-agnostic Reconstructions

If detector-specific responses can be unmapped into generic edep-sim like format, they can be the first translation layer for a more generic ML reco

This reconstruction can be more robust and shared between detectors/experiments, reducing duplication of effort



Acknowledgements

Thank you to K. Terao, D.H. Koh, A. Mishra, D. Ratner for assistance in developing both the problem and the solution!

Thank you to ZOOX, who funded this work in part

Thank you, the audience for your attention!

Summary

- Basic uncertainty quantification can be a simple modification to your existing models, no need for a full Bayesian NN!
- Toolkits like the uncertainty toolbox exist and are developed with simple interfaces for machine learners, particularly in physics
- The inverse mapping model described here is a toy for understanding uncertainty quantification in chained models, but is also aimed at interfacing with differentiable simulators and downstream (detector-agnostic) reconstructions

BACKUP

SLAC

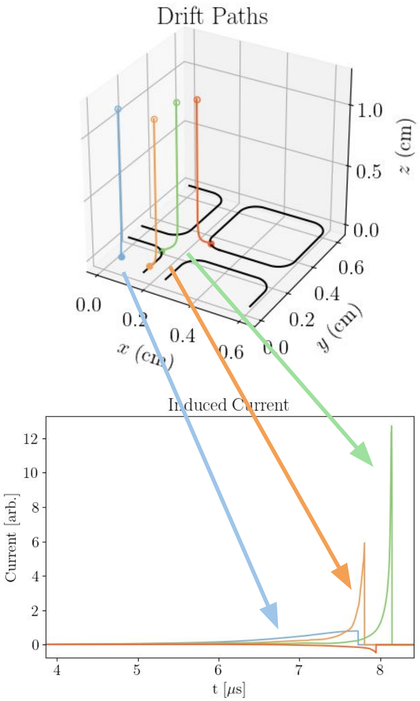
Hit Formation

Charge clouds drift to the anode plane

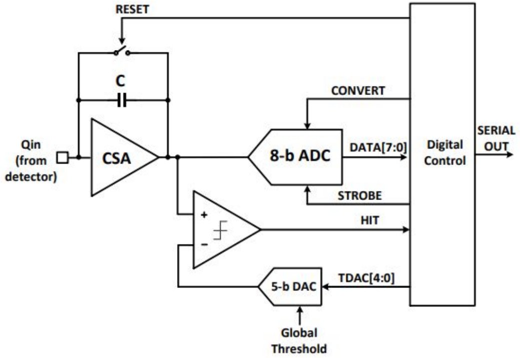
Voltage is induced on the surfaces of electrodes

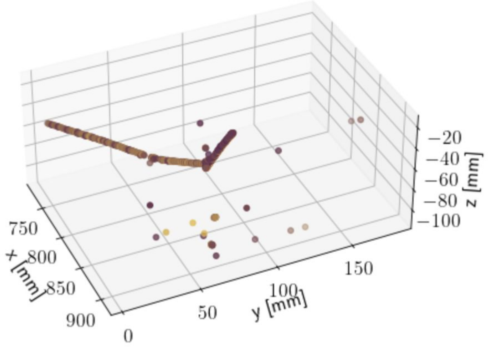
Pixel electronics register a “hit” and digitize charge after a threshold is reached + 8 clock cycles (10 MHz)

Measurement is (pixel address, timestamp, ADC value)



Σ

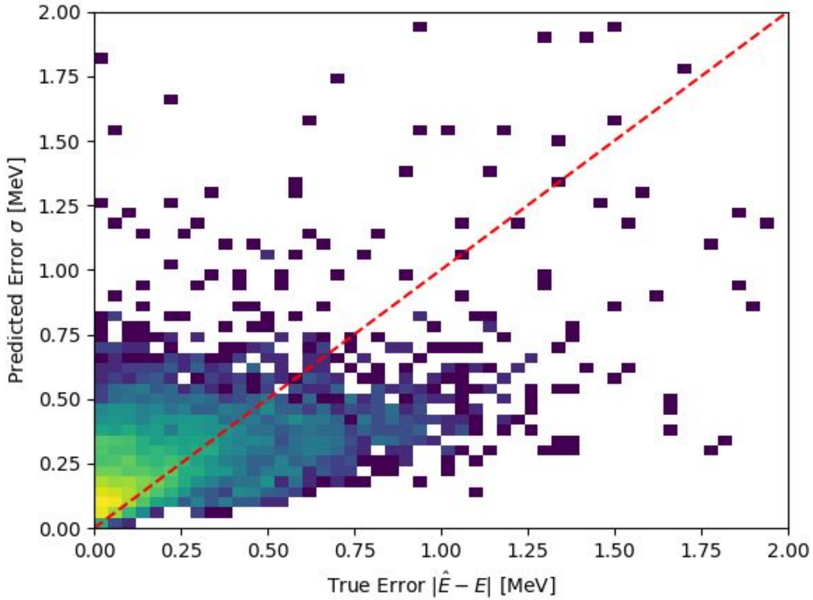




Regressed Voxel Uncertainties

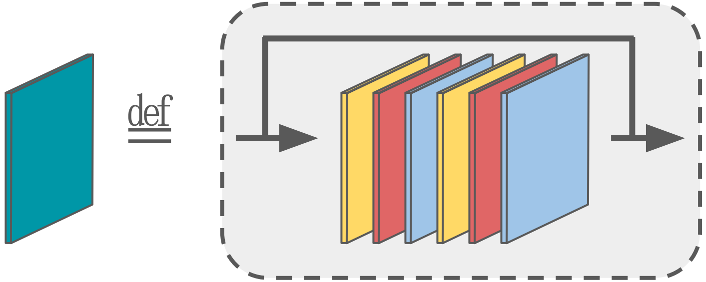
Model is well-calibrated: 68% of the predicted errors cover the observed errors

The predicted error shows a dependence upon the voxel value and the magnitude of the true error

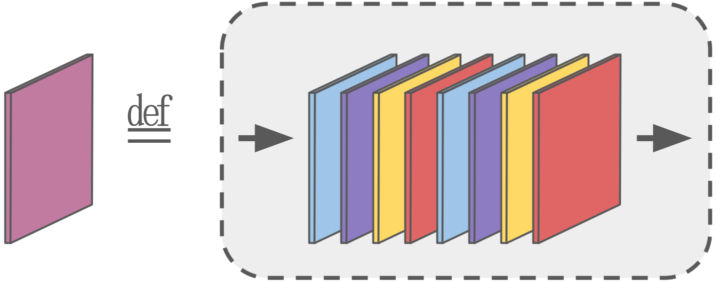


Block Definitions

Block Definitions



Residual Block



Dropout Block

BatchNorm

ReLU

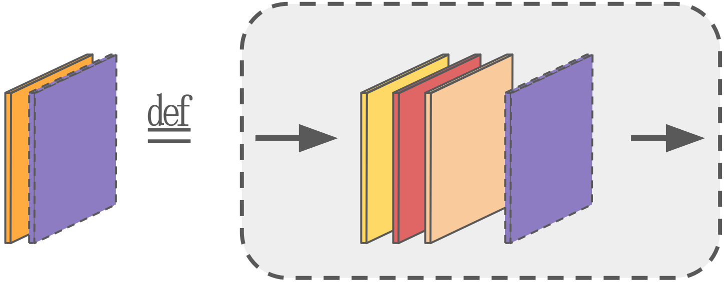
Convolution

Dropout

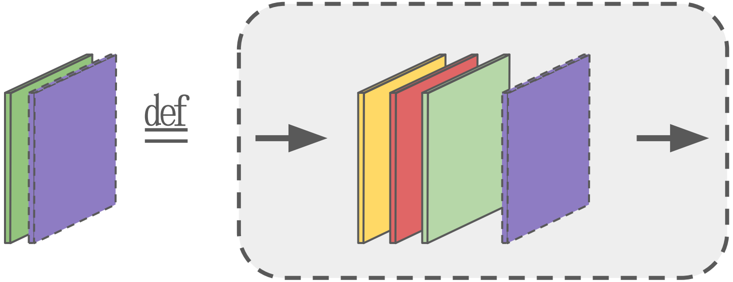
Conv 2×2×2, Stride 2

Conv 2×2×2 Transpose

Block Definitions, Continued



DownSample Block



UpSample Block

BatchNorm	Convolution	Conv 2×2×2, Stride 2
ReLU	Dropout	Conv 2×2×2 Transpose

Dropout Layers are included if the previous block has a dropout layer