Event reweighting and generative models

In neutrino experiments

IPA workshop on Machine Learning for particle physics and astrophysics

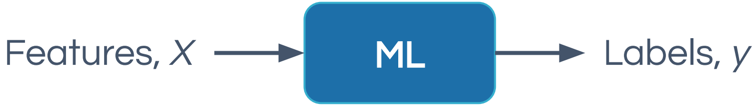
March 21, 2023

LABORATÓRIO DE INSTRUMENTAÇÃO
E FÍSICA EXPERIMENTAL DE PARTÍCULAS
partículas e tecnologia

Cristóvão Vilela

Reweighting and generating data

- Machine learning often used to **summarise** multidimensional data.
  - Given high-dimensional representation of the event X, produce a classification score y.

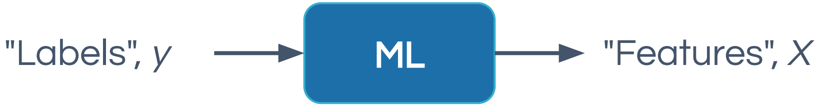


Features, X

ML

Labels, y

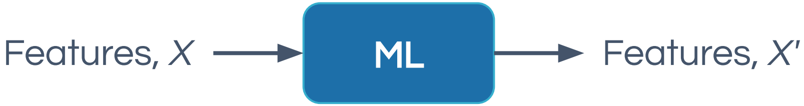
- But we may be interested in **generating** some data X, starting from the summary y, or some modified data X', starting from our nominal data X.



"Labels", y

ML

"Features", X



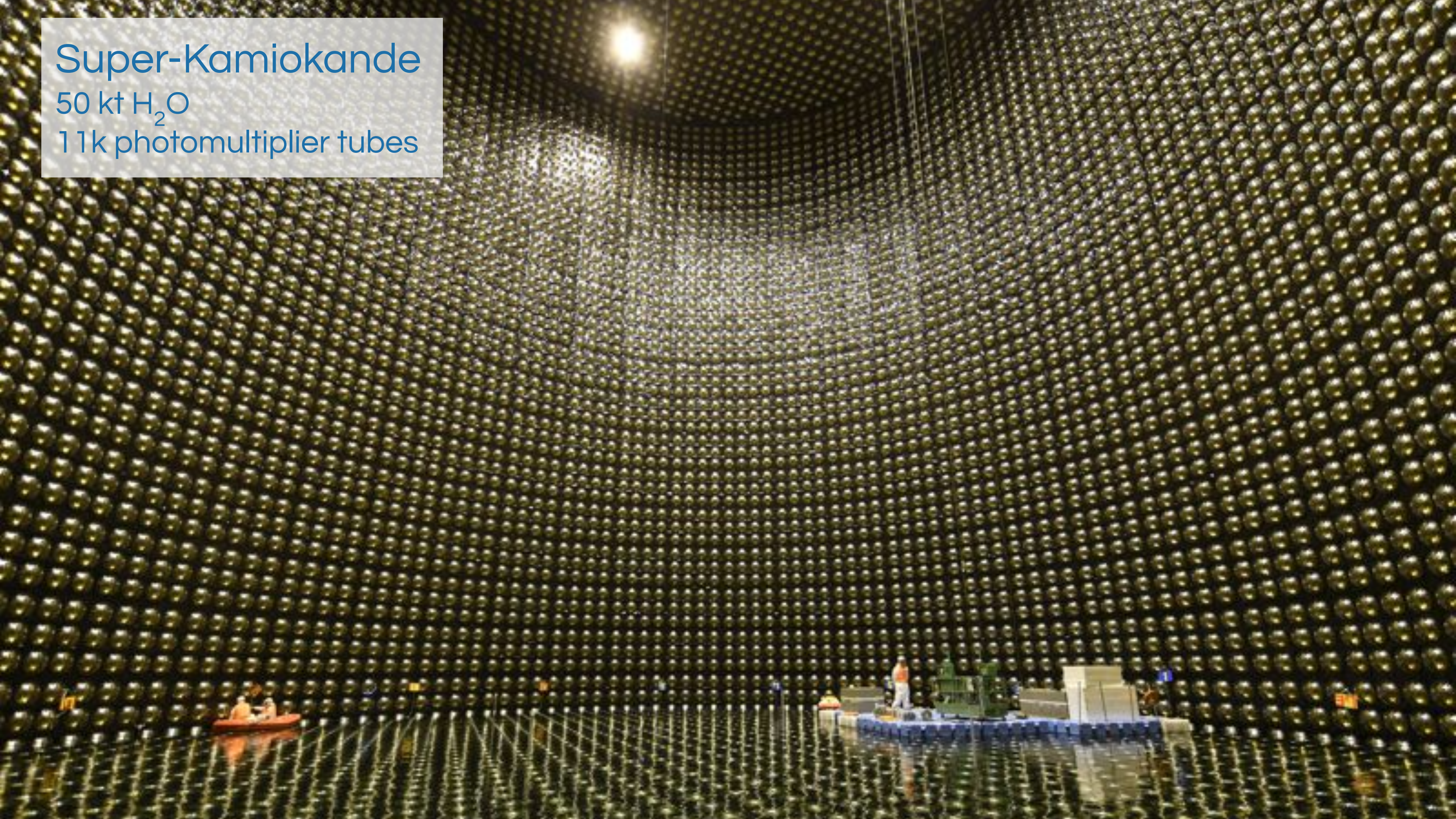
Features, X

ML

Features, X'

- I will briefly describe three applications of ML in neutrino physics where we are interested in obtaining a detailed description of events, X, rather than the summary, y.
  - A generative model for Super-Kamiokande events.
  - Reweighting between neutrino interaction models in DUNE.
    - This is actually a classification task.
  - Translating DUNE events from the near to the far detector.

Super-Kamiokande event reconstruction

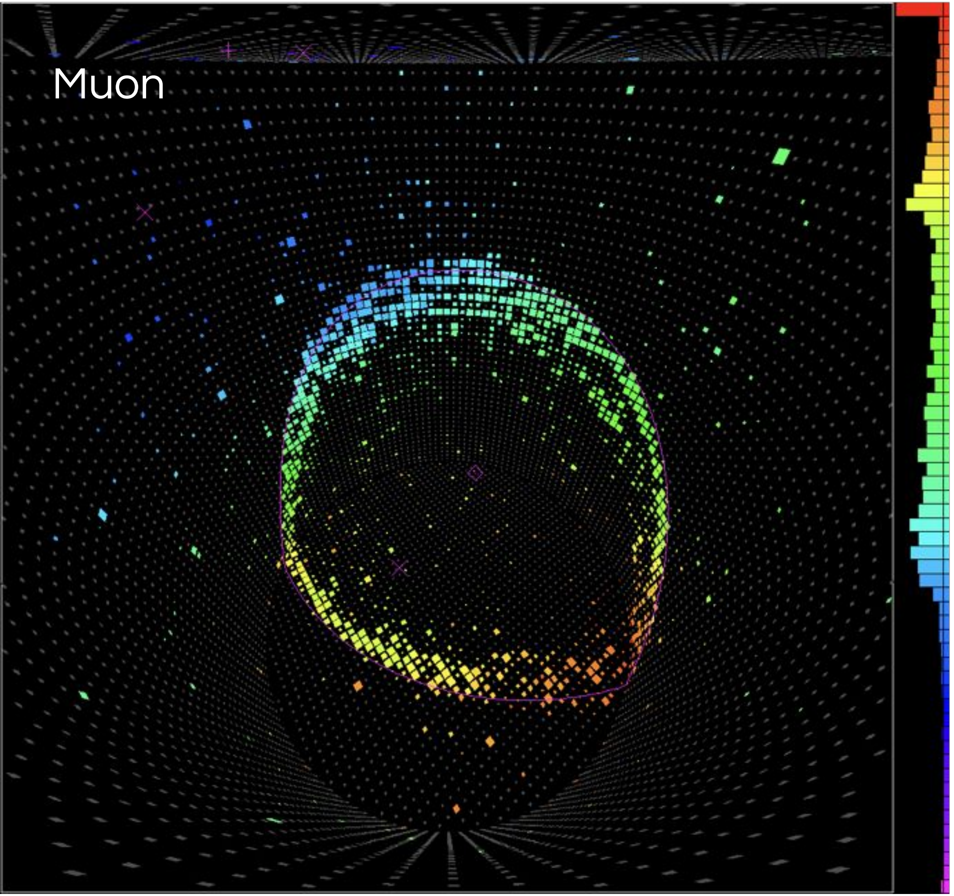


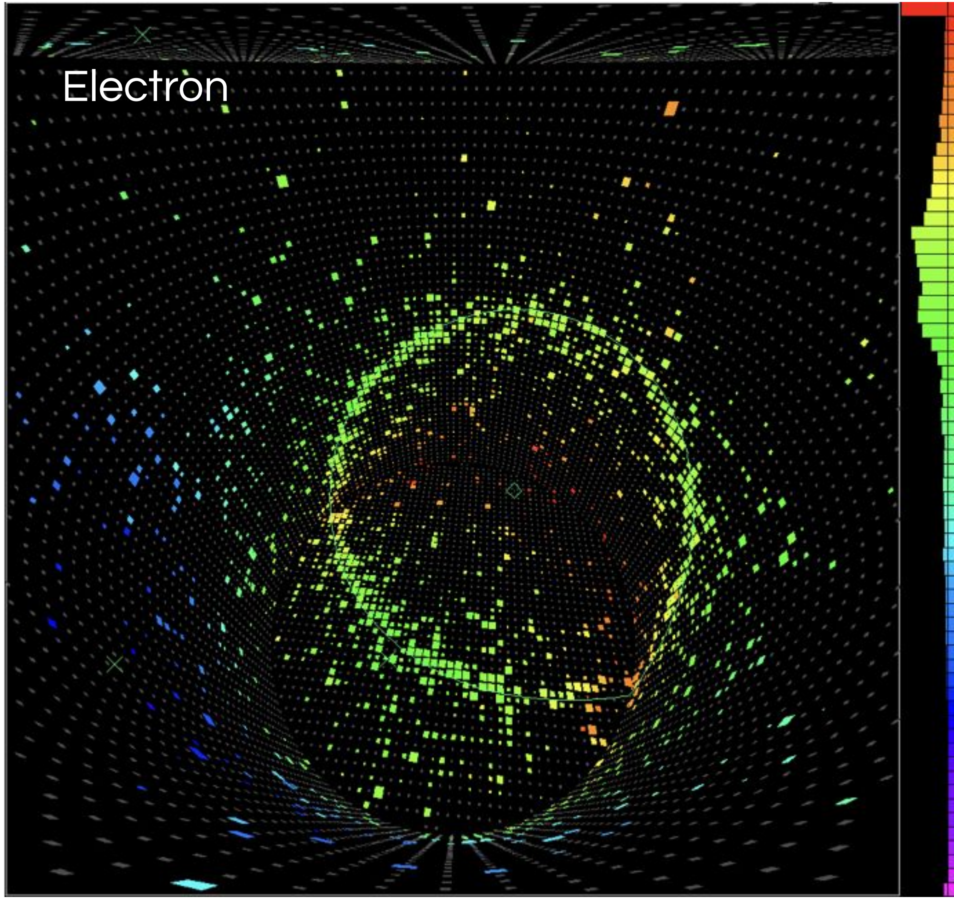
Super-Kamiokande

50 kt H_2O

11k photomultiplier tubes

Particle identification at Super-K





Super-K event reconstruction

Use information from sensors both with and without registered hits in the event

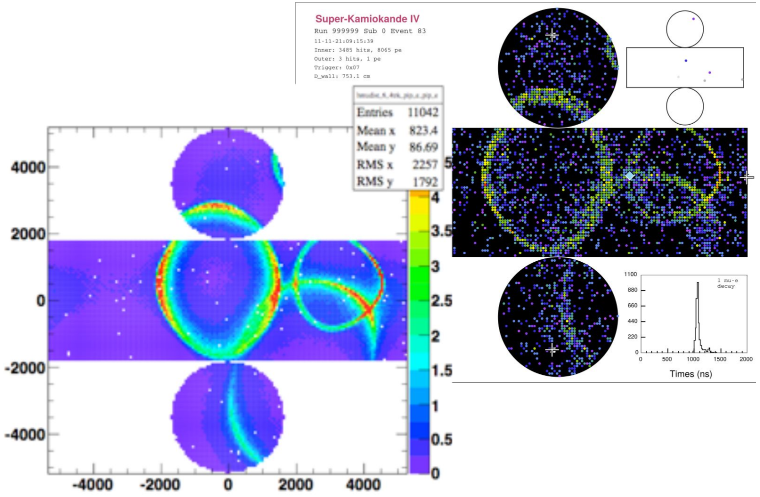
$$\boxed{L\left(\mathbf{x}\right)} = \boxed{\prod_{i_{unhit}} \mathrm{P}\left(i_{unhit}|\mathbf{x}\right) \prod_{i_{hit}} \mathrm{P}\left(i_{hit}|\mathbf{x}\right)} \boxed{f_q\left(q_i|\mathbf{x}\right)} \boxed{f_t\left(t_i|\mathbf{x}\right)}$$

Build likelihood function for event hypothesis **x**

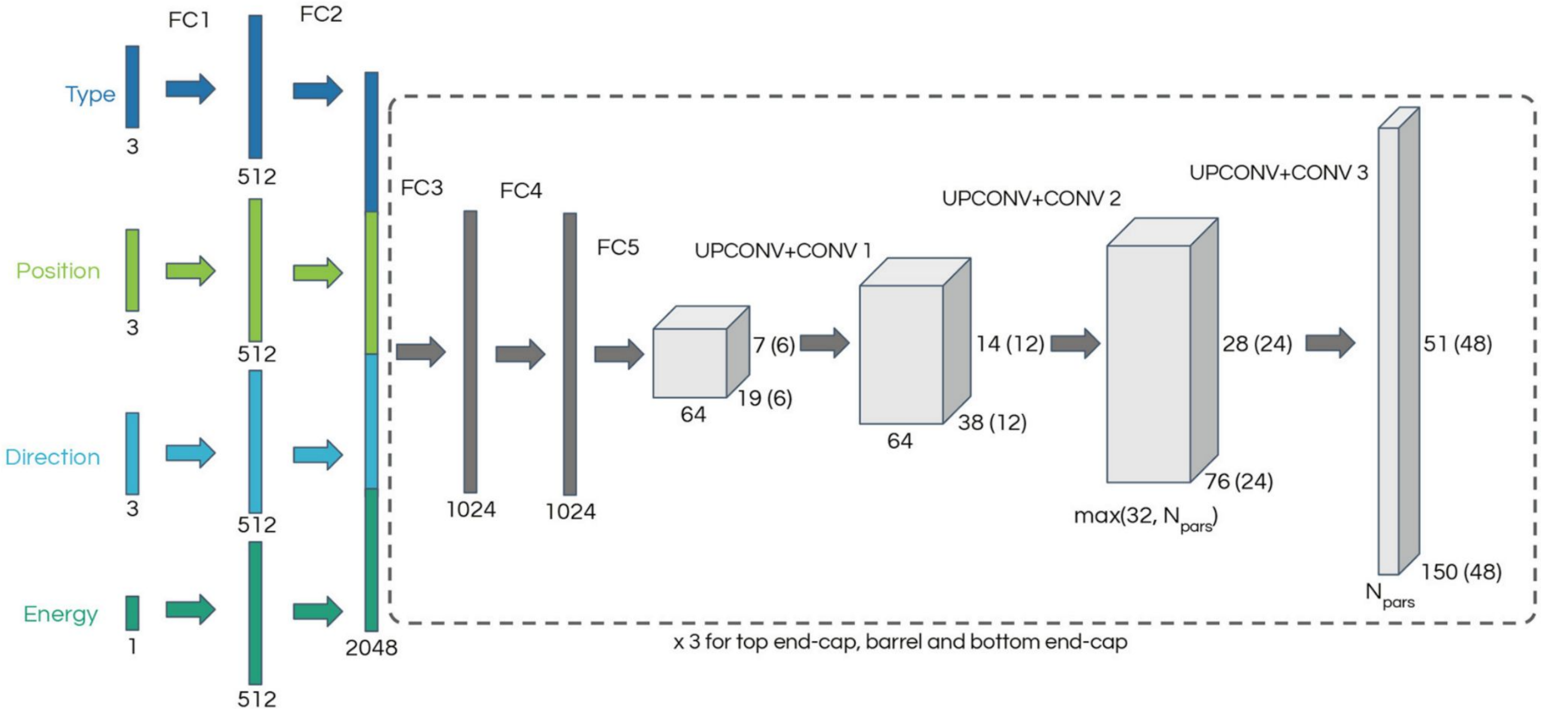
For hit photosensors:

Compare observed charge to prediction and Compare hit time to prediction

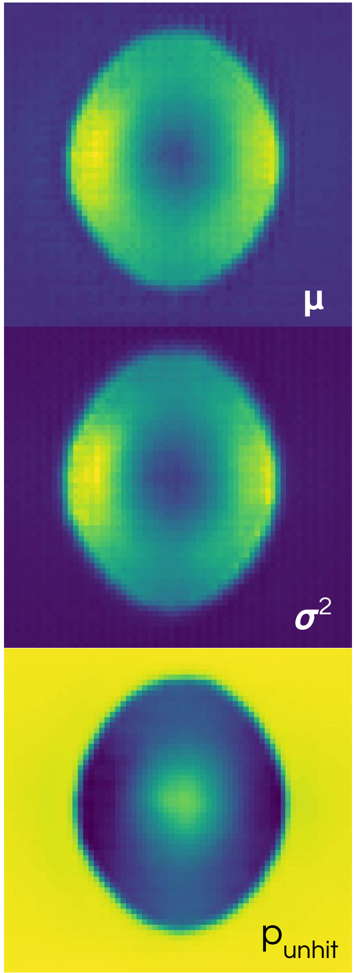
- Maximise the likelihood for hit pattern as a function of particle kinematics.
- "Traditional" approach has shortcomings due to curse of dimensionality.
    - Very difficult to model reflection of light in the detector in few dimensions.
- Use **neural network** to **generate** probability density function at each photosensor.



Generating PDFs for Super-K

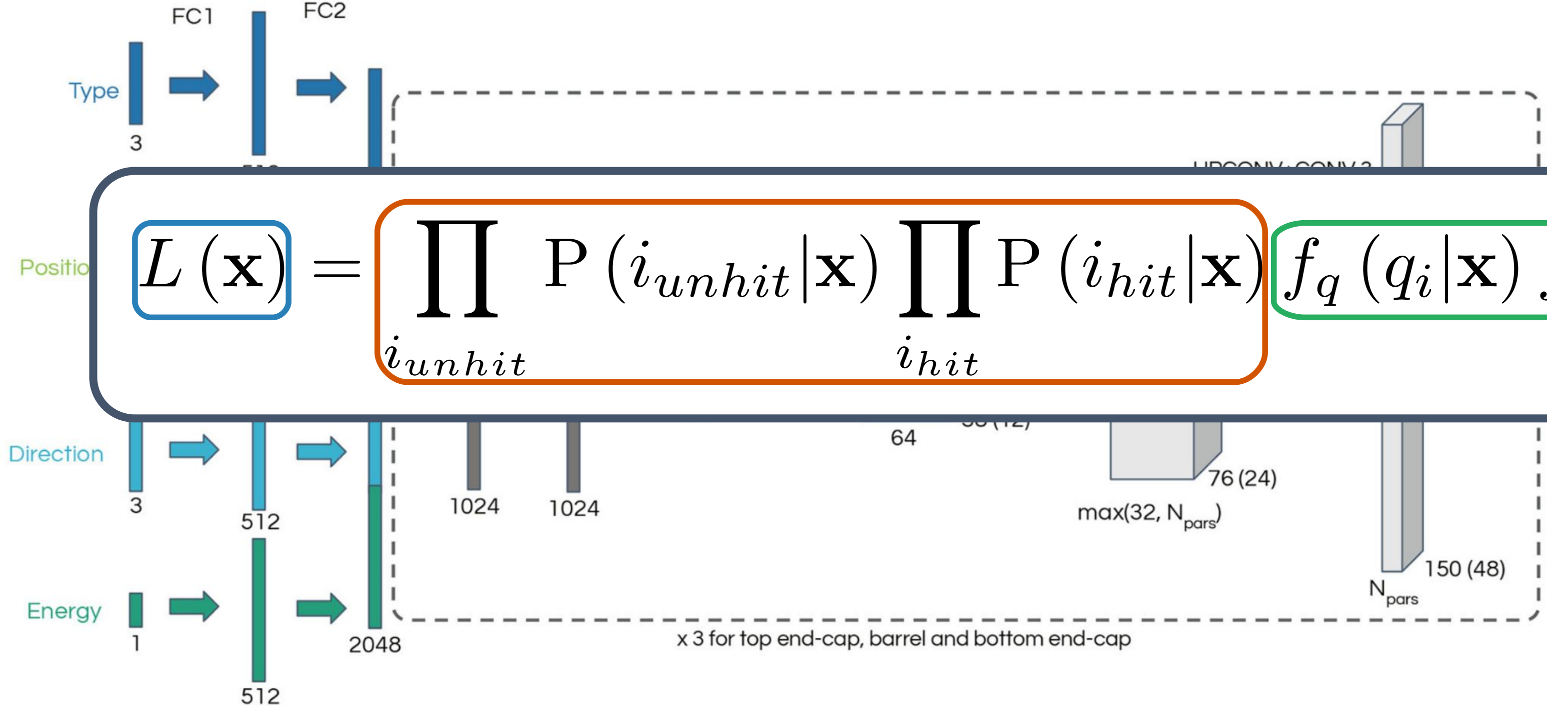


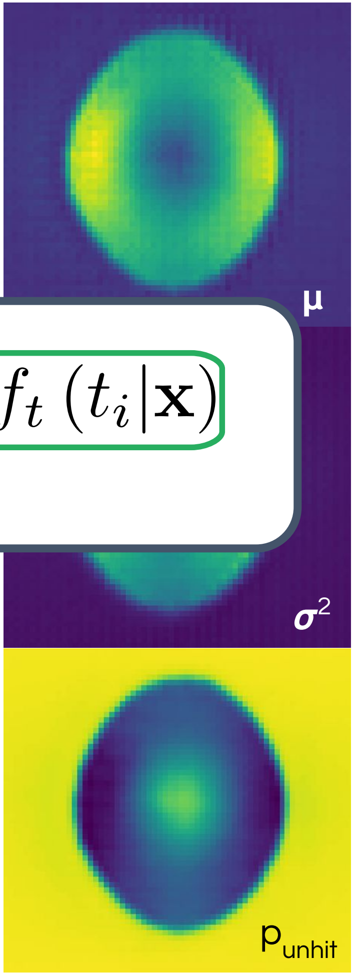
$$\text{Loss} = -\log L = \sum_{i} -\log P_{unhit}(y_i) + \sum_{i_{hit}} -\log p_{qt}(q_{i_{hit}}, t_{i_{hit}})$$



Front. Big Data 5:868333

Generating PDFs for Super-K



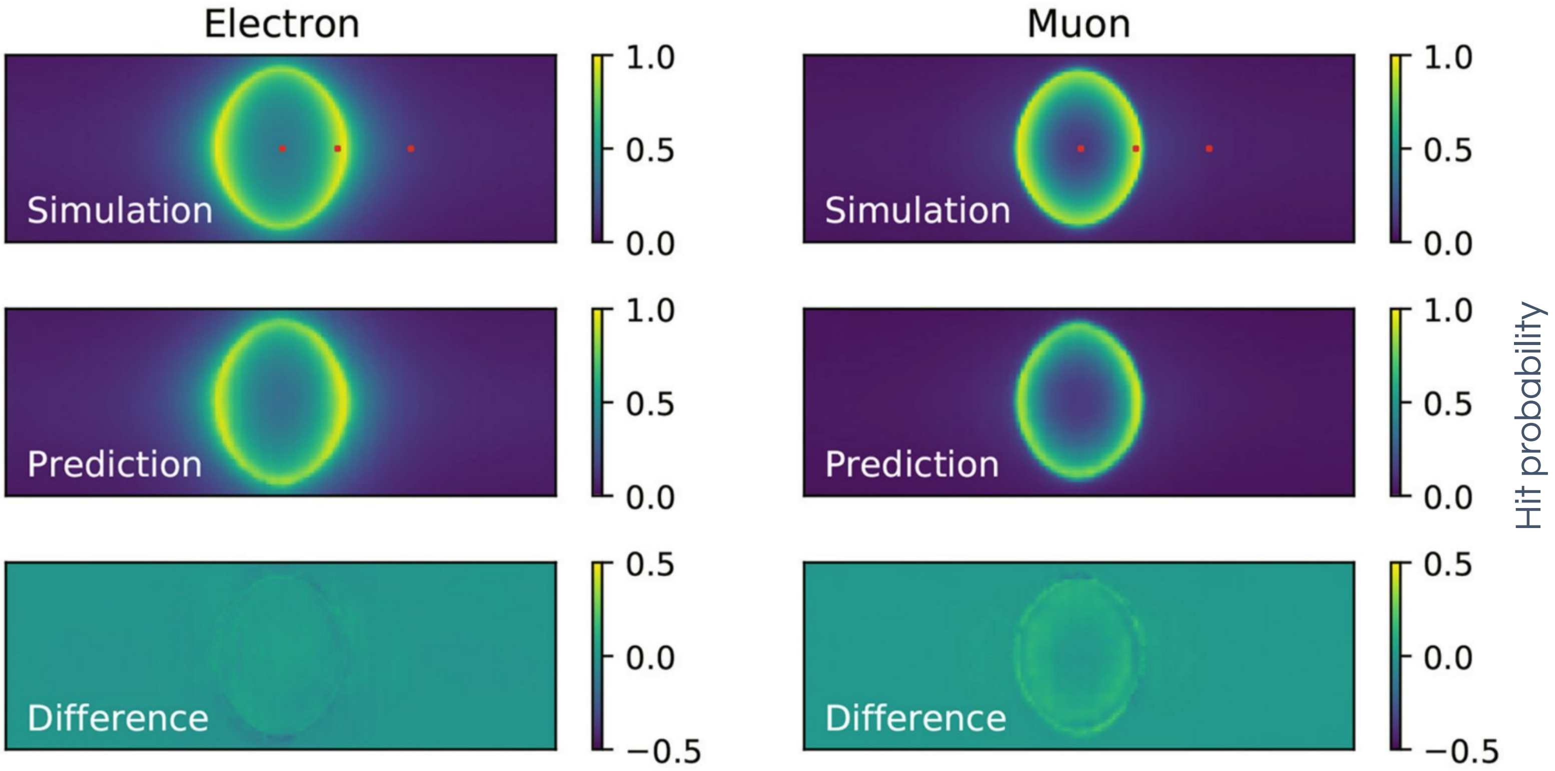


$$\boxed{L(\mathbf{x})} = \boxed{\prod_{i_{unhit}} \mathrm{P}\left(i_{unhit}|\mathbf{x}\right) \prod_{i_{hit}} \mathrm{P}\left(i_{hit}|\mathbf{x}\right)} \boxed{f_q\left(q_i|\mathbf{x}\right) f_t\left(t_i|\mathbf{x}\right)}$$

$$\boxed{\text{Loss}} = \boxed{-\log L = \sum_i -\log P_{unhit}(y_i)} + \boxed{\sum_{i_{hit}} -\log p_{qt}(q_{i_{hit}}, t_{i_{hit}})}$$

Front. Big Data 5:868333

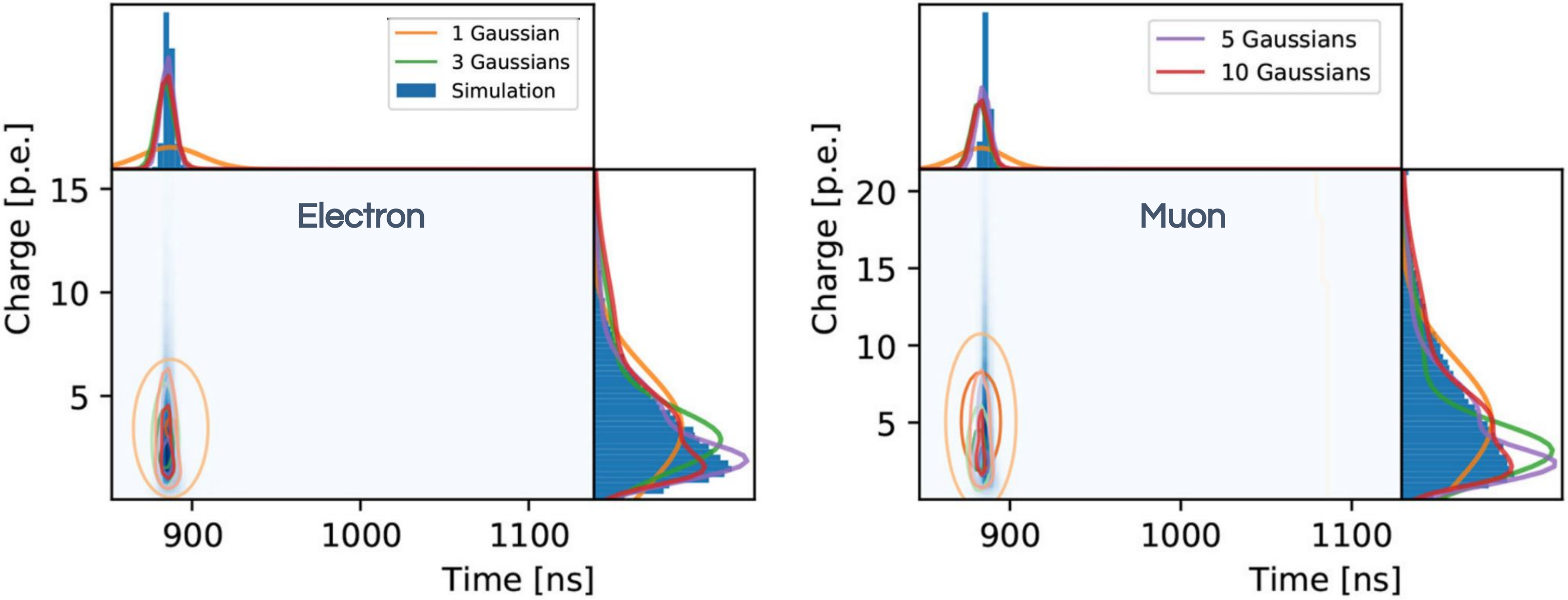
Hit probability



- Compare neural network output to average of 50k events generated with the same parameters.

Front. Big Data 5:868333

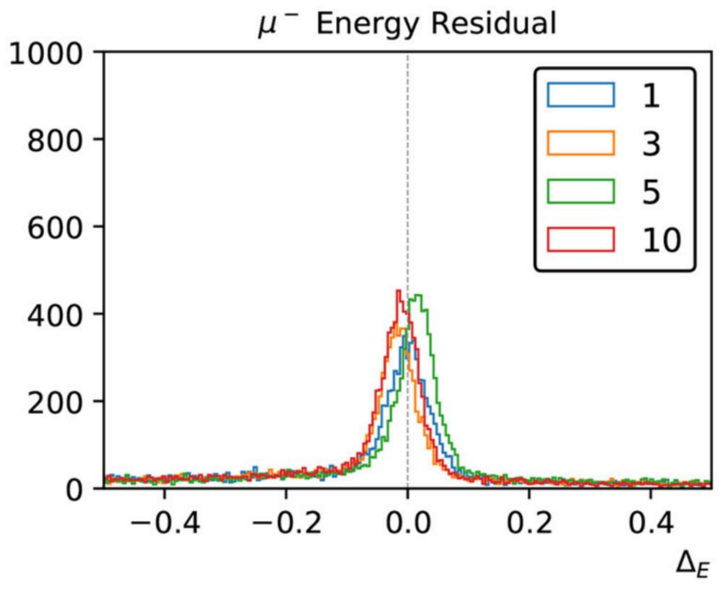
Charge and time PDF



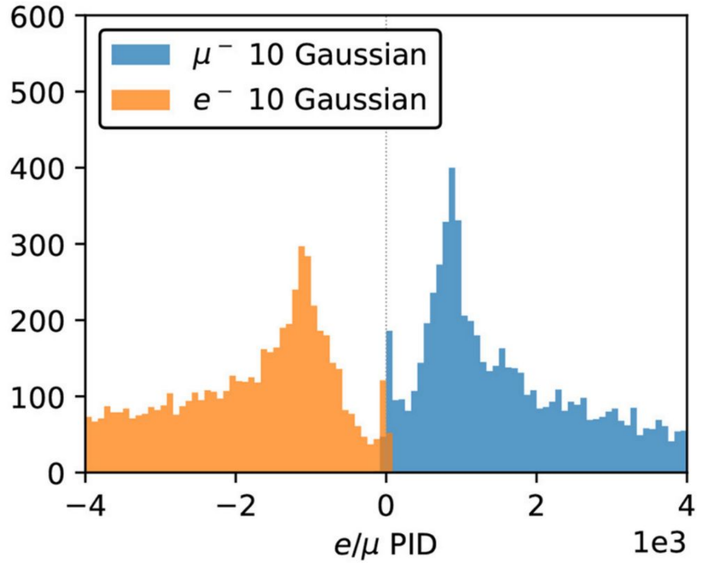
PMT at the edge of the Cherenkov ring.

[Front. Big Data 5:868333]

Energy reconstruction



- Reconstruct the particle energy by maximizing the likelihood.



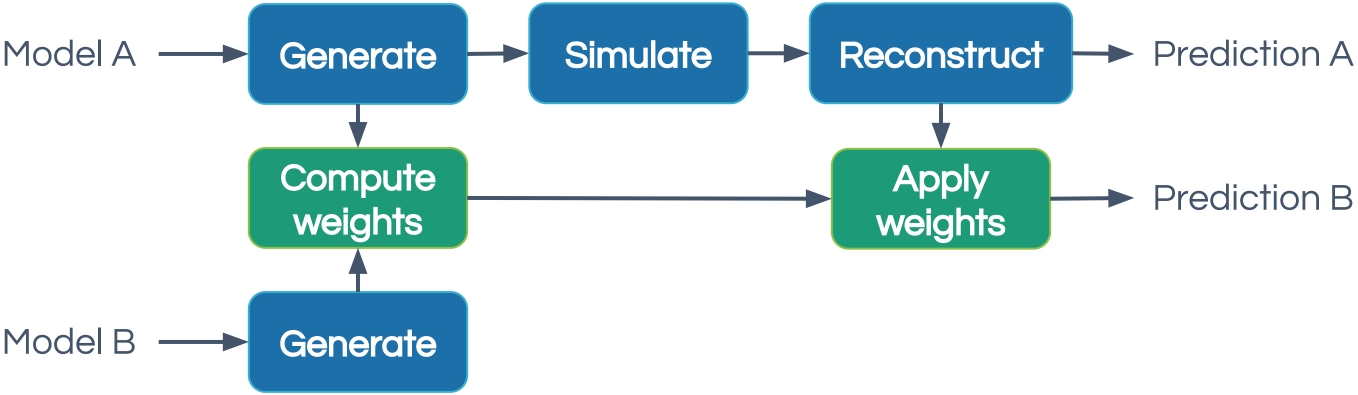
- Take likelihood ratios between different hypotheses to identify particles.

Front. Big Data 5:868333

DUNE event reweighting

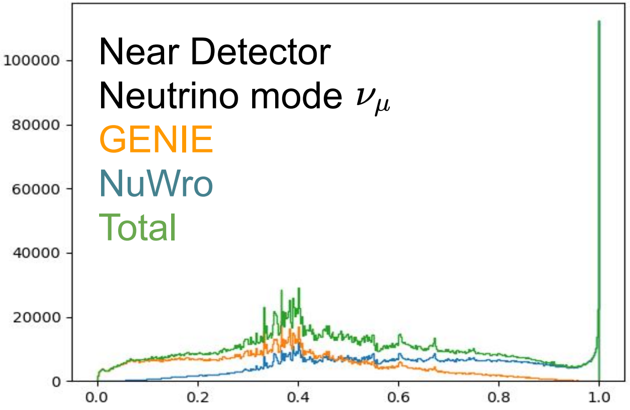
Event reweighting: what is it good for?

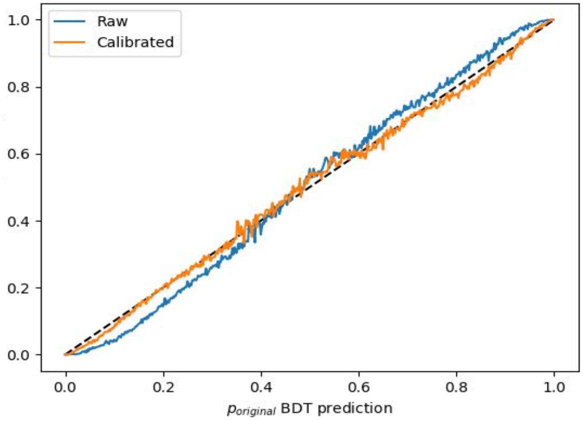
- We use Monte Carlo methods to predict what our data should look like under different hypotheses.
- Roughly factorizes into:
  a. **Generate** neutrino-nucleus interactions (fast)
  b. **Simulate** the detector response (slow)
  c. **Reconstruct** the events (slow)
- There are significant uncertainties in the neutrino-nucleus interaction models, so we want to test the impact of different models on our sensitivity.
  - However, re-running the full simulation chain is often prohibitively expensive.

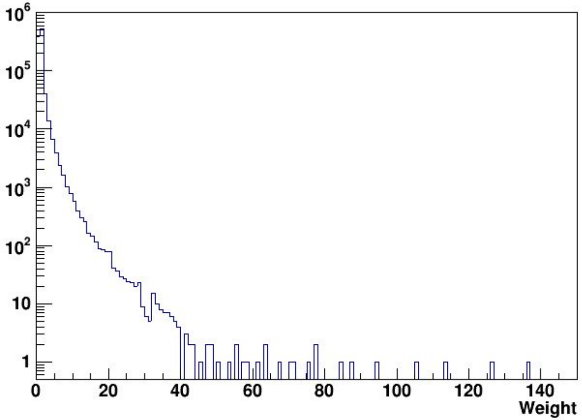


Reweighting between models

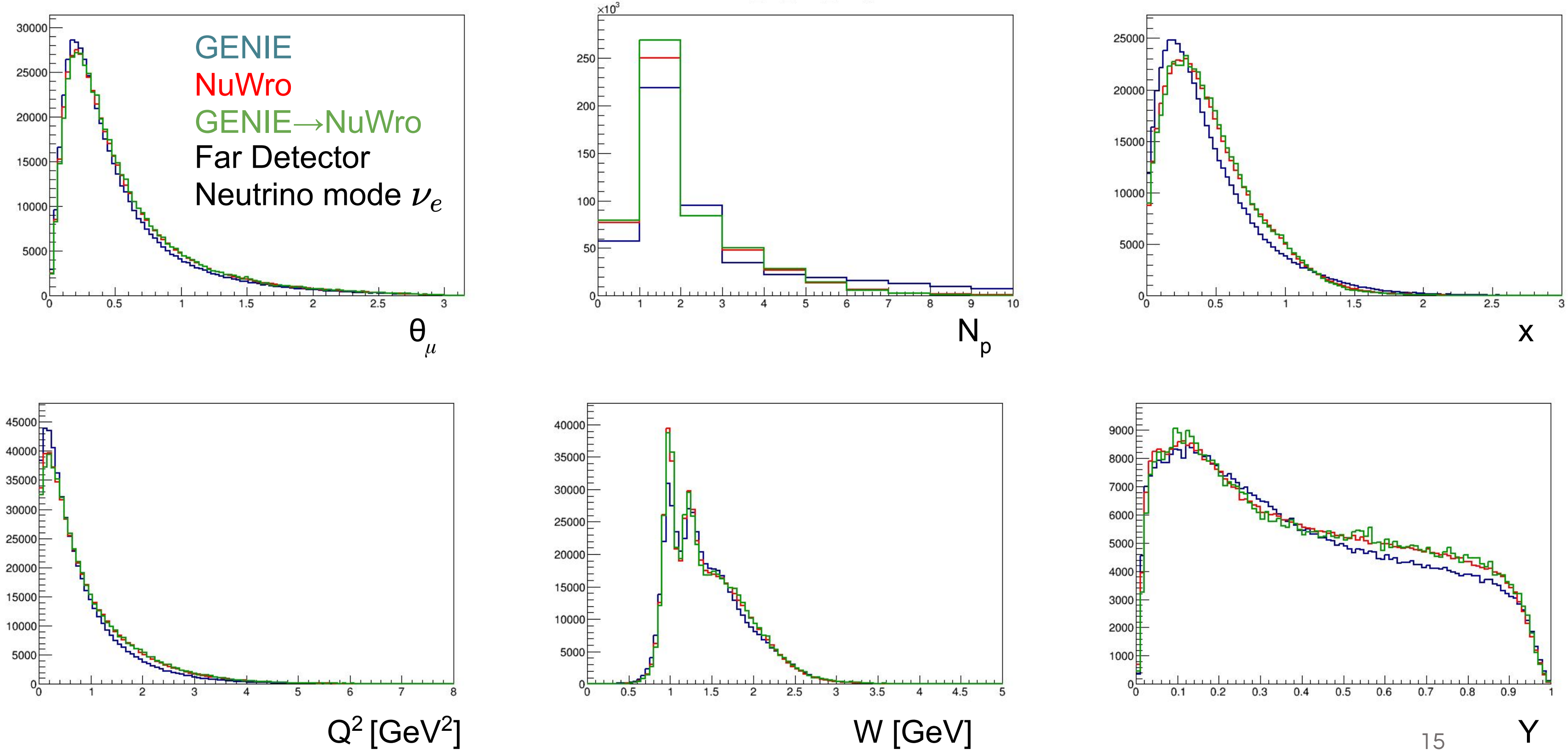
- Traditional reweighting methods make use of histograms.
  - Allows for reweighting up to 2 or 3 dimensions but not more.
  - Can only be used to reweight low-dimensional parts of the model that can be factorized from the rest of the model.
- But we can **reweight in high number of dimensions** using ML.
- DUNE TDR: **boosted decision trees** used to **classify** generated events between two different models using 18 variables for describing the events.
  - Classifier output can be interpreted as a **probability** and expressed as a **weight**.





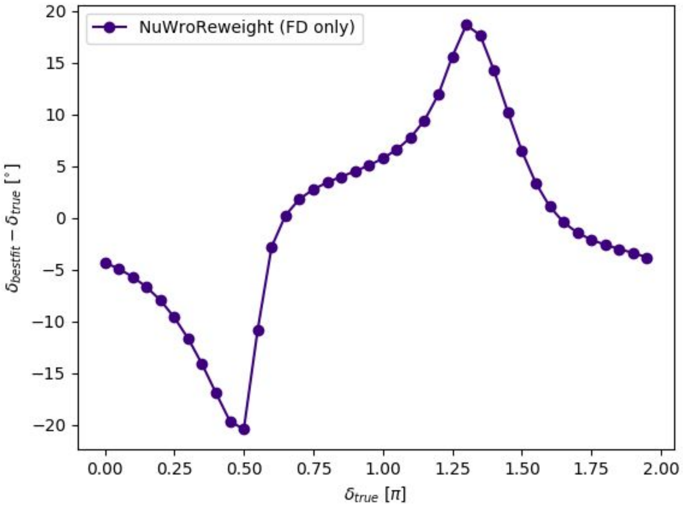


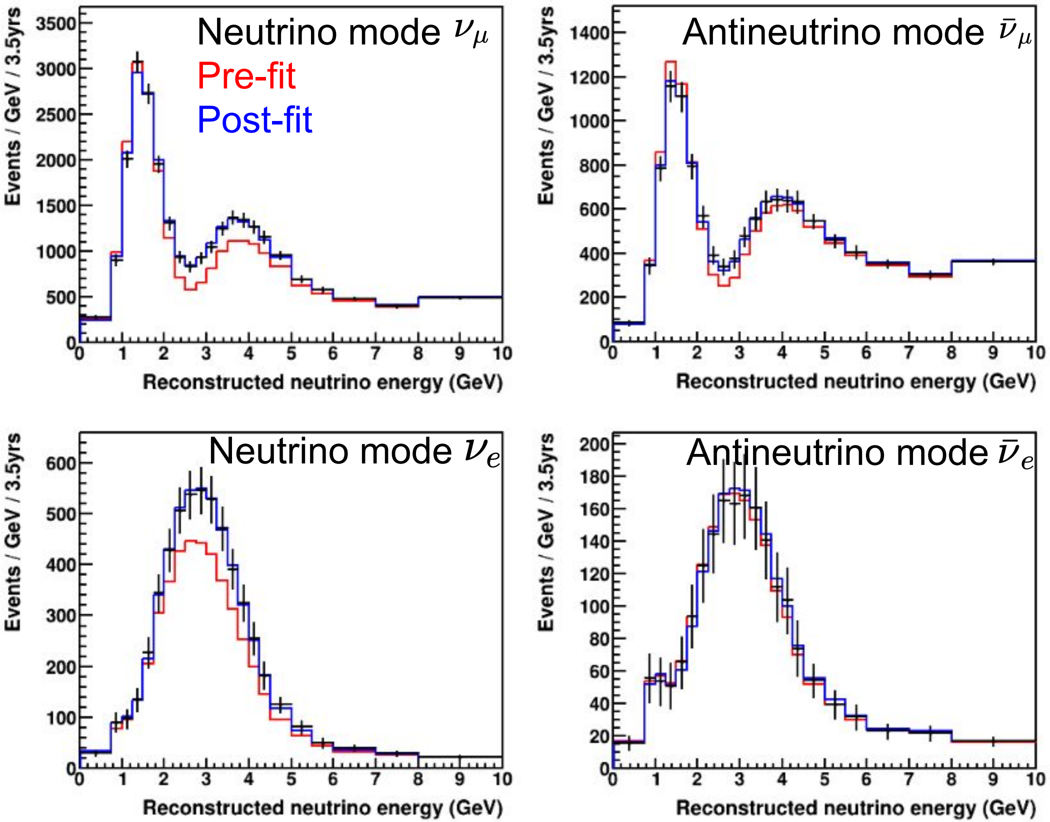
Reweighting examples



Check model impact on analysis

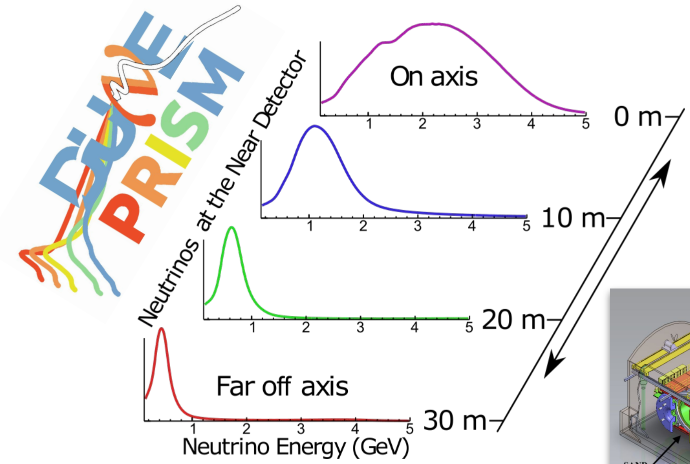
- Run neutrino oscillation analysis on reweighted simulation.
- Get significantly different results if the wrong model is assumed.
    - But near detector saves the day!
- Important result, only possible using ML.

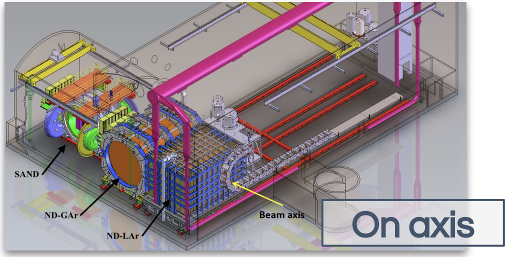


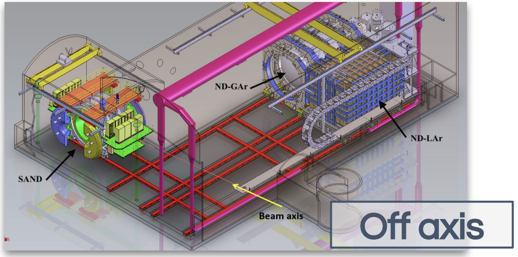


DUNE near to far detector translation

Precision Reaction-Independent Spectrum Measurement

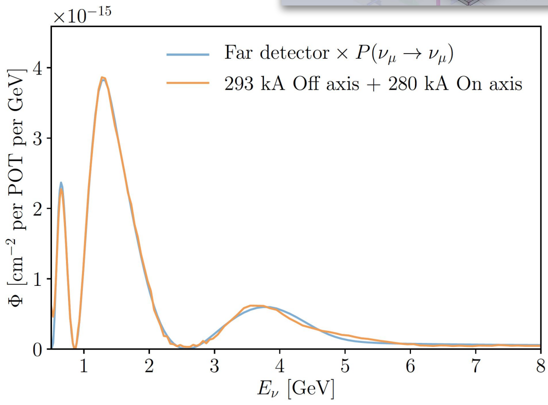




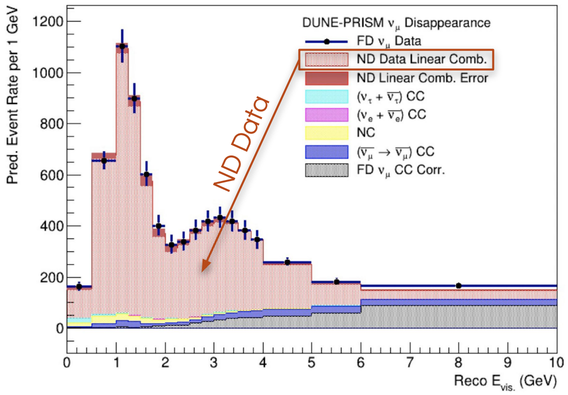


- Data-driven DUNE analysis possible using the DUNE-PRISM moveable detector.
- For this to work, we need to be able to **compare near** detector data to **far** detector **data**.

Combine DUNE-PRISM **fluxes**



Combine DUNE-PRISM **data**

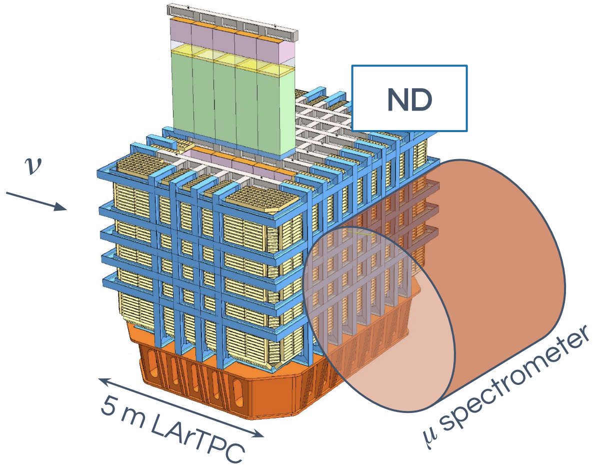


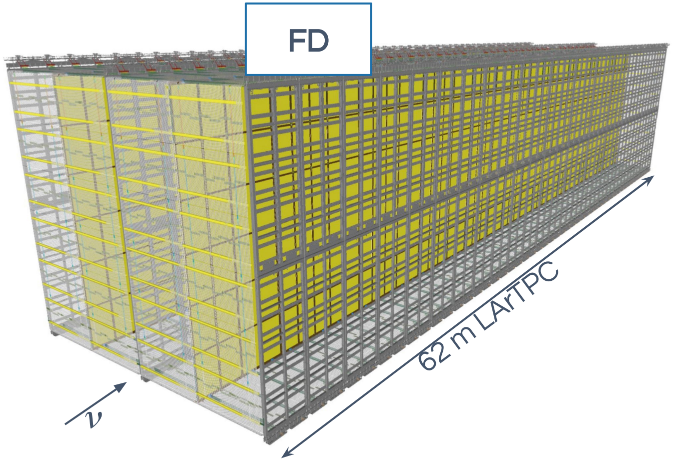
Detector response in data-driven analyses

67 ton LArTPC with 3D **pixel** read-out, **segmentation,** and downstream **muon spectrometer**

$$R^{\mathrm{ND}}\left(\mathrm{E}_{\nu}, \mathrm{E}_{\mathrm{reco.}}\right) \neq R^{\mathrm{FD}}\left(\mathrm{E}_{\nu}, \mathrm{E}_{\mathrm{reco.}}\right)$$

10 kton LArTPC with 3 x 2D **wire** read-out and very large drift volumes



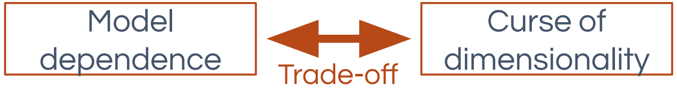


- Need a **model-independent** method to account for differences in the **detector responses.**
  - If an **ND** event had occurred instead in the **FD**, what would be its reconstructed energy at the FD?

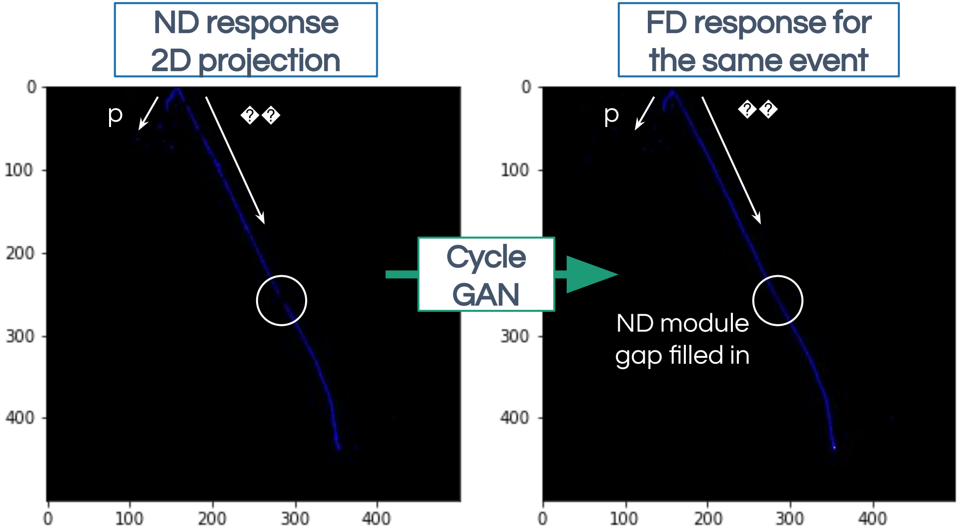
Learning the differences between ND and FD

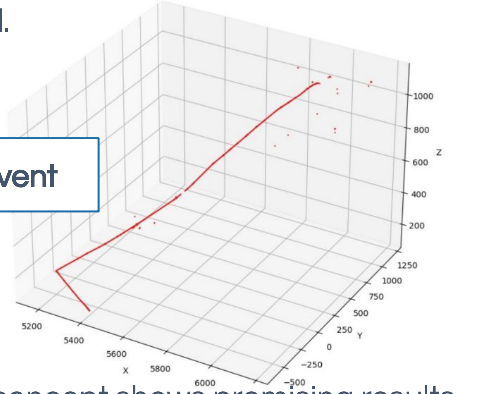
CycleGAN arXiv:1703.10593 (2017)

- Traditional approach: response matrices out of high-level reconstructed variables.
    - Leads to model dependence.



- Reduce model dependence by using **image-to-image translation** techniques to generate FD-like events from ND events at **hit level.**





- 2D proof-of-concept shows promising results.
- Next step:
    - ND 3D readout → FD 3x2D readout
    - Needs novel neural network **architectures.**
    - Work in progress with R. Radev (CERN).

Summary

- Lots of interesting applications of machine learning in neutrino physics.
- Showed three examples of applications not focused on classification:
    - Generative model for Super-Kamiokande events.
        - Produces probability density functions for the detector sensors given an event hypothesis.
    - Event reweighting for DUNE.
        - Reweight between interaction models in many dimensions.
        - Enables important studies that would not be possible without ML.
    - Near-to-far detector event translation in DUNE.
        - Get far detector response given a near detector event.
        - Exploit full information content of the data to avoid model dependence.
        - Work in progres...

Extra

PDF parameterisation

- Requirements:
  - Statistically robust (i.e., integrates to 1) and smooth (for gradient descent)
  - General-ish – do not assume what the PDFs look like *a priori*
- Strategy:
  - Use combinations of Gaussians (a la Gaussian Mixture Model)
    - 2x1D or 2D, correlated or uncorrelated.

$$-\log p_{qt}(q_{i_{hit}}, t_{i_{hit}}) = -\sum_{i_{hit}}\Bigg[\sum_{j}^{N}\Bigg(\log(n_j) - \log(\sqrt{2\pi}\sigma_{q_j}) - \frac{(q_{i_{hit}}-\mu_{q_j})^2}{2\sigma_{q_j}^2} - \log(\sqrt{2\pi}\sigma_{t_j}) - \frac{(t_{i_{hit}}-\mu_{t_j})^2}{2\sigma_{t_j}^2}\Bigg)\Bigg]_{i_{hit}}$$

Uncorrelated

Note: q-t correlations can still arise from the different means of each component.

$$f(\boldsymbol{\eta}|\boldsymbol{\theta}) = \sum_{j}^{N}\frac{n_j}{(2\pi)|\Sigma_j|^{1/2}}\exp\left(-\frac{1}{2}(\boldsymbol{\eta}-\boldsymbol{\theta}_j)^{\mathrm{T}}\Sigma_j^{-1}(\boldsymbol{\eta}-\boldsymbol{\theta}_j)\right)$$

Correlated

Extra DOF for each component: the q-t correlation.

$$\Sigma^{-1} = \begin{pmatrix}\alpha_{11} & 0\\ \alpha_{12} & \alpha_{22}\end{pmatrix}\begin{pmatrix}\alpha_{11} & \alpha_{12}\\ 0 & \alpha_{22}\end{pmatrix}$$

Work with components of the Cholesky-decomposed triangular matrix.

Add conditions to the alphas to guarantee covariance matrix properties.

Front. Big Data 5:868333