Basics of Machine Learning and its Application in Neutrino Physics

Lecture 2: Applications in ν physics

Igor Ostrovskiy, IHEP

2nd JUNO Neutrino Summer School
Hangzhou, August 2025

Introduction

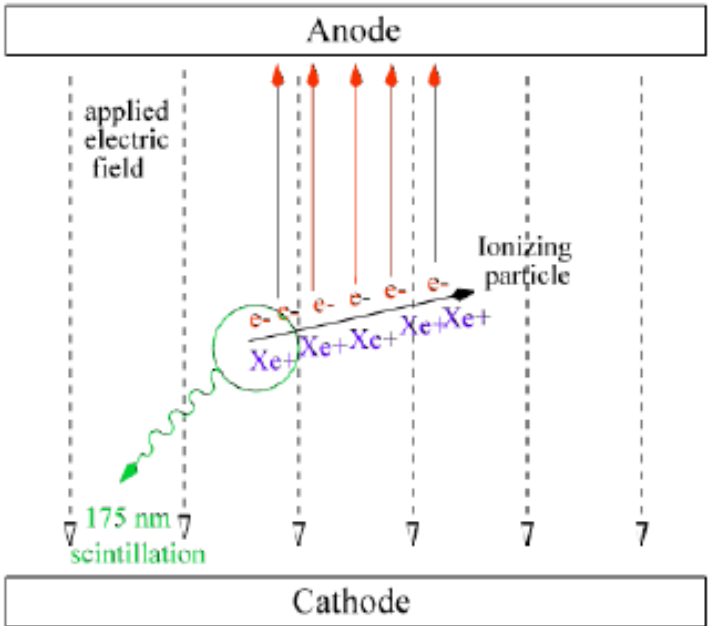
- Topics covered in lectures:
  - Lecture 2. Applications in neutrino physics
    - Case of EXO-200: Successful examples of MLP, CNN, GAN, and data-based training
    - Examples of much larger detectors: NOvA, SBND

Introduction

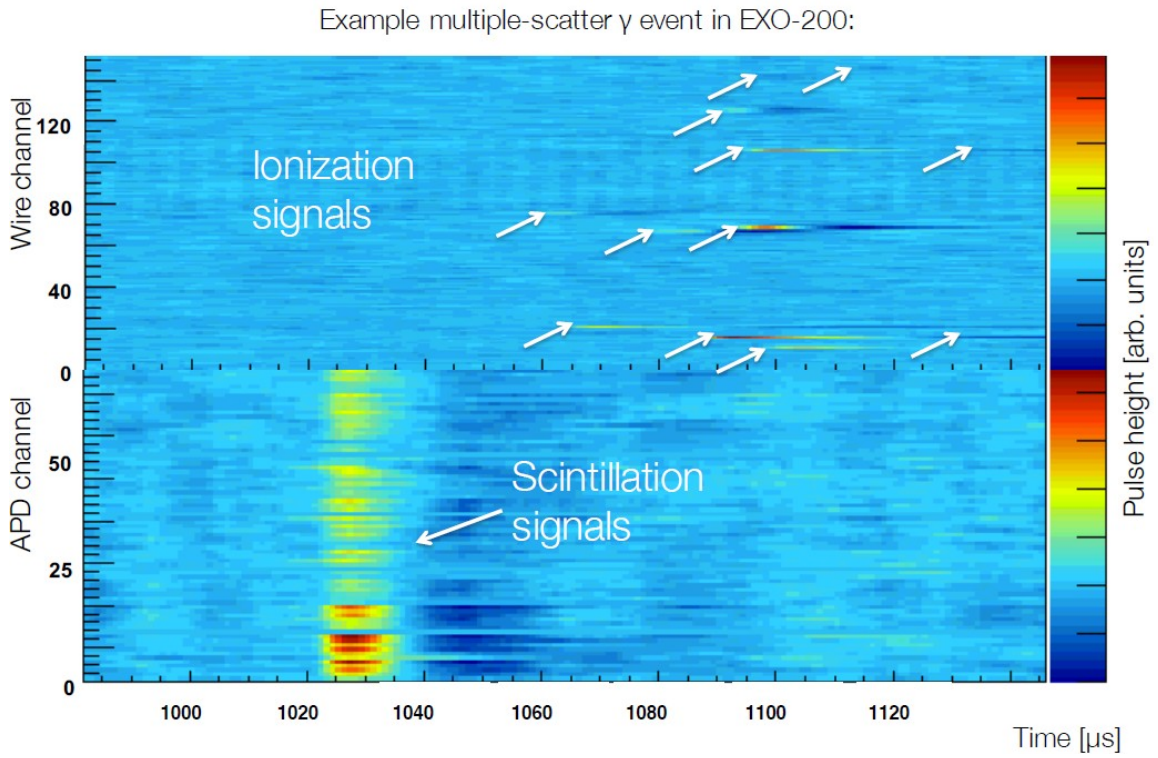
- For the tutorials/activity, I will use Jupyter notebooks and realistic data from a 0ν and another experiment
  - Password: cUaw (please don’t share further)
- Confirmed to work with:
  - python 3.7.1, torch 1.13.1, cpu
  - python 3.11.9, torch 2.5.1, gpu (cuda 12.5, Tesla T4)
  - python 3.12.11, torch 2.7.1, gpu (cuda 12.5, A100)
- You don’t have to run the scripts, can just follow the tutorial on the screen
  - Or feel free to join forces with others to be able to play with the scripts yourselves

EXO-200

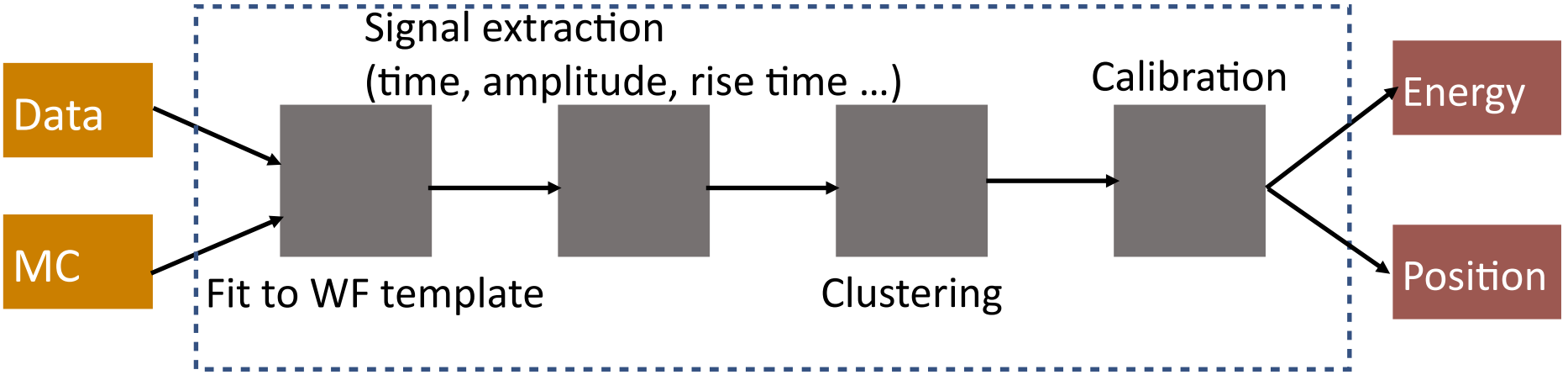
- Time projection chamber (TPC). Each side detects both charge and light
- 38x2 U-wire channels for charge collection
  - 800 e- noise per wire on average
- 38x2 V-wire channels for charge induction
  - Crossed at 60^{o} with U-wires
- 74x2 APD channels for light
  - Each channel is a chain of 7 LAAPDs
  - Cathode is mostly transparent (mesh)
  - Cylindrical Teflon reflector



EXO-200 data


Example multiple-scatter γ event in EXO-200:
Wire channel
120
80
40
0
APD channel
50
25
0
Ionization signals
Scintillation signals
1000
1020
1040
1060
1080
1100
1120
Time [μs]
Pulse height [arb. units]

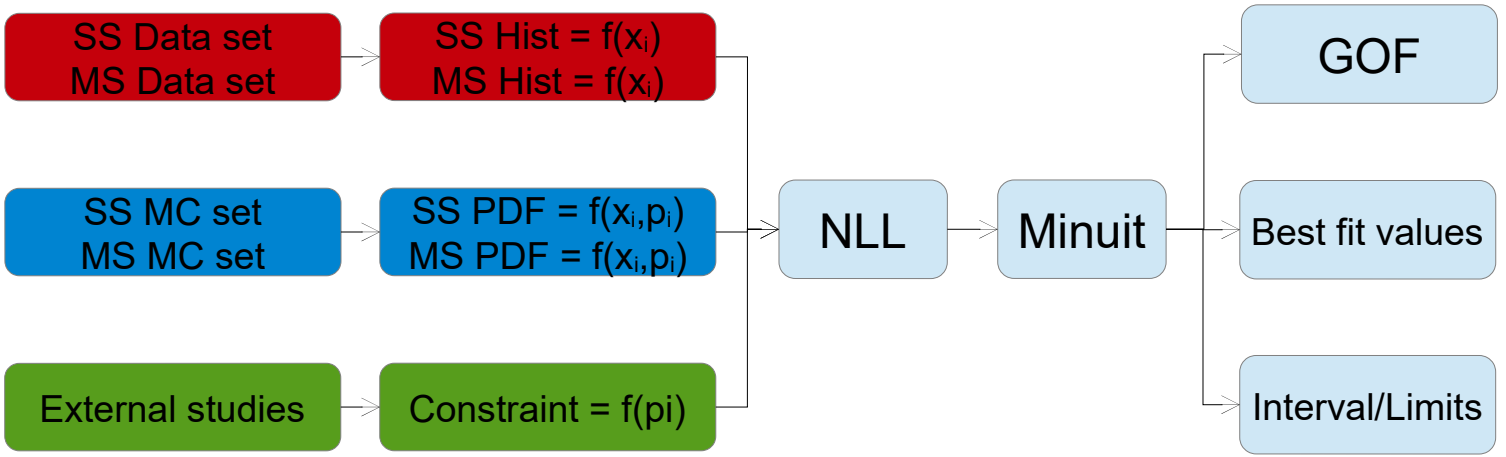
EXO analysis in broad strokes: reconstruction



- Multiple algorithmic steps
- Done by different people over the course of several years
- Imperfections in each step can add systematics

} “grey” boxes

EXO analysis in broad strokes: point/interval estimation



- Two classes of events, single-site (SS) and multi-site (MS)
- MC based PDFs, binned extended NLL with systematics constraints
- Profile likelihood for interval construction
- Systematics due to recon and MC errors. Measured or estimated using calibration data

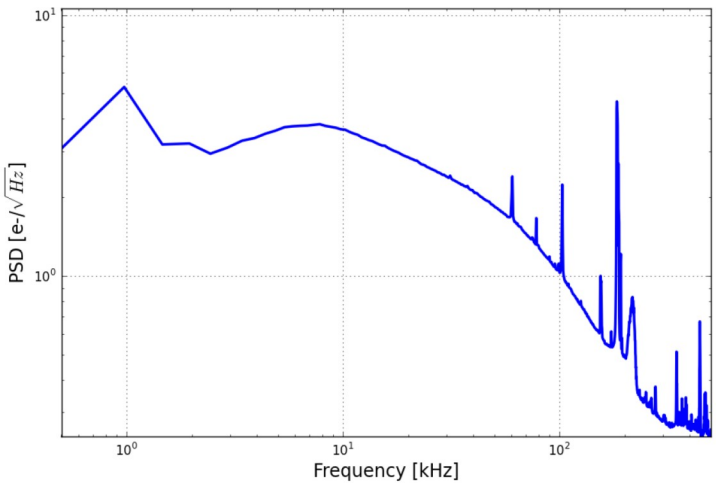
Deep Neural Networks in EXO

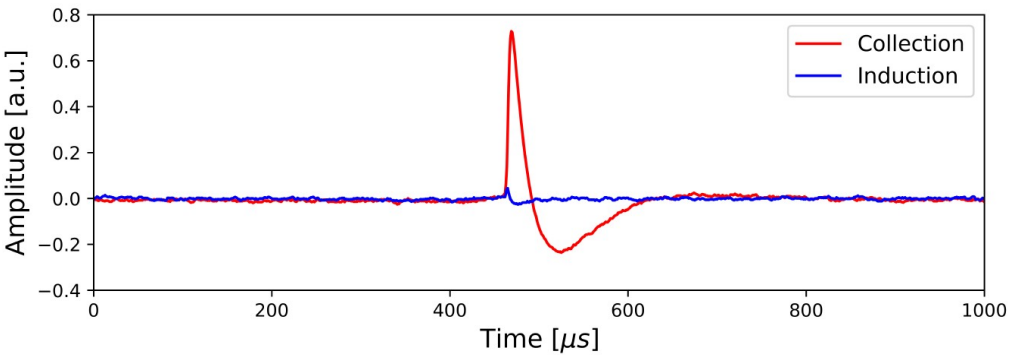
- Can circumvent intermediate steps and extract high level information directly from raw waveforms?
  - **YES**
- Can validate results on real detector data, not just MC?
  - **YES**
- Even then, if using MC truth during training, would be limited by how well MC models data (as some standard analysis steps are). Can reduce reliance on traditional MC?
  - **YES (Sometimes)**

JINST 13 P08023 (2018), Phys. Rev. Lett. 123 161802 (2019), JINST 18 P06005 (2023)

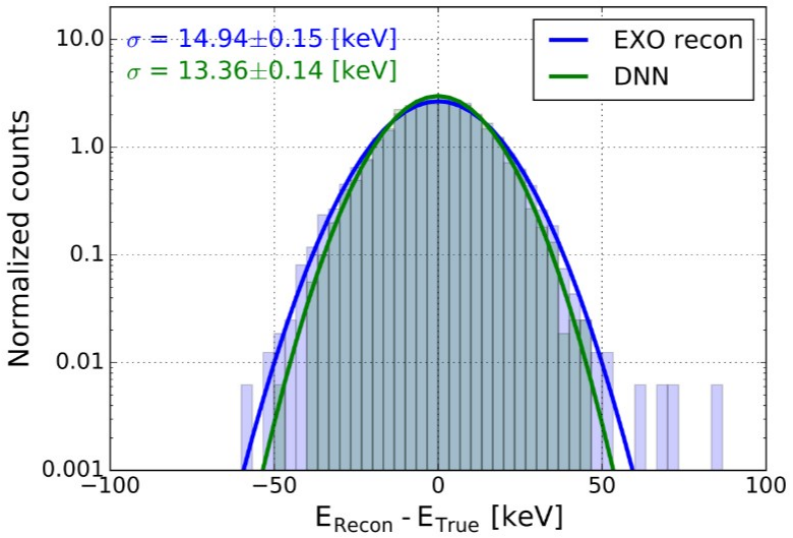
First application: charge energy reconstruction

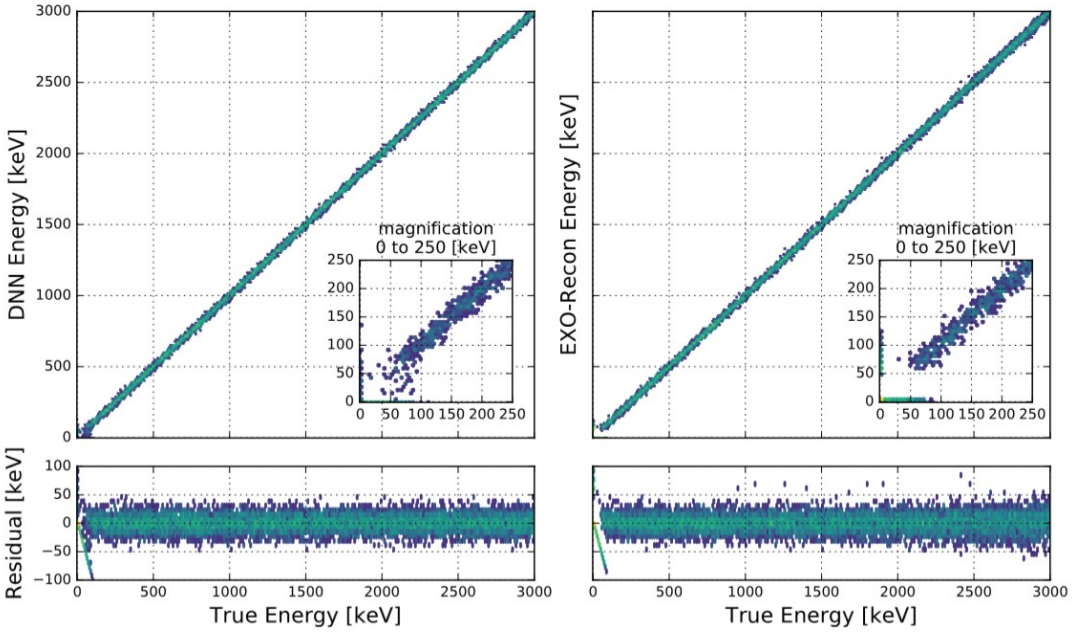
- The main challenges of charge reconstruction are nontrivial noise and disentangling U-wire signal into induction and collection
  - Somewhat covered by the yesterday's tutorial





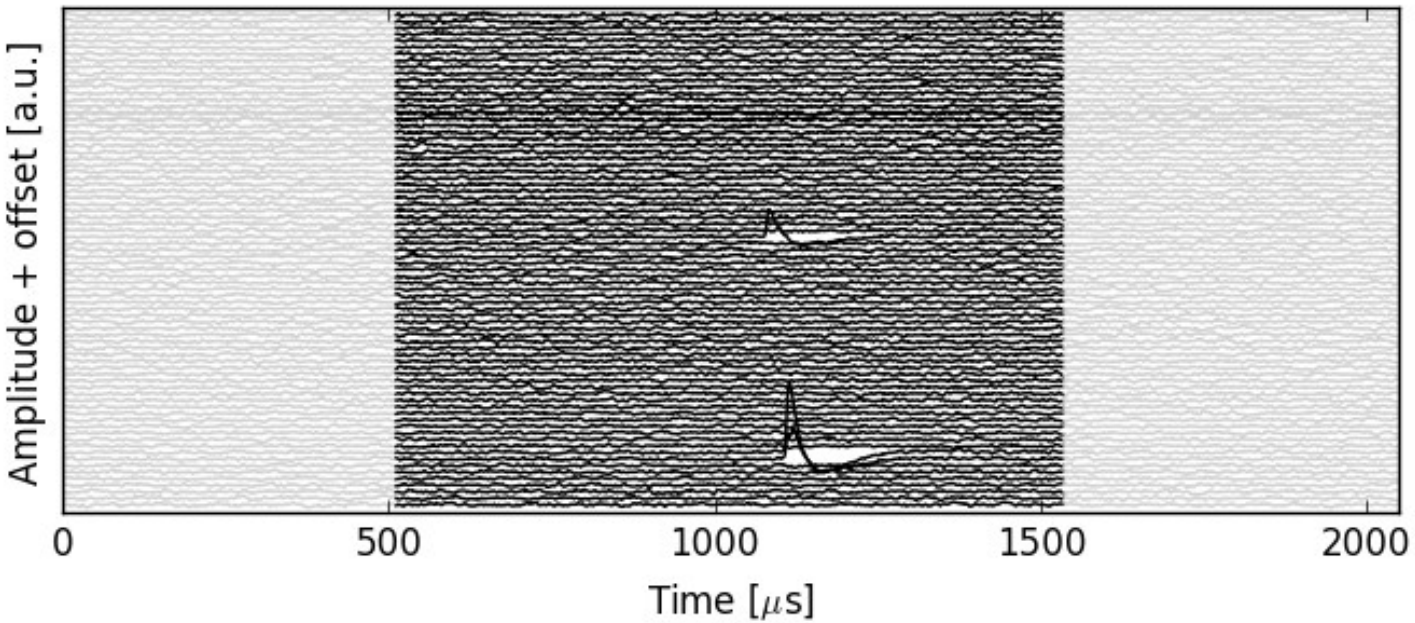
First application: charge energy reconstruction





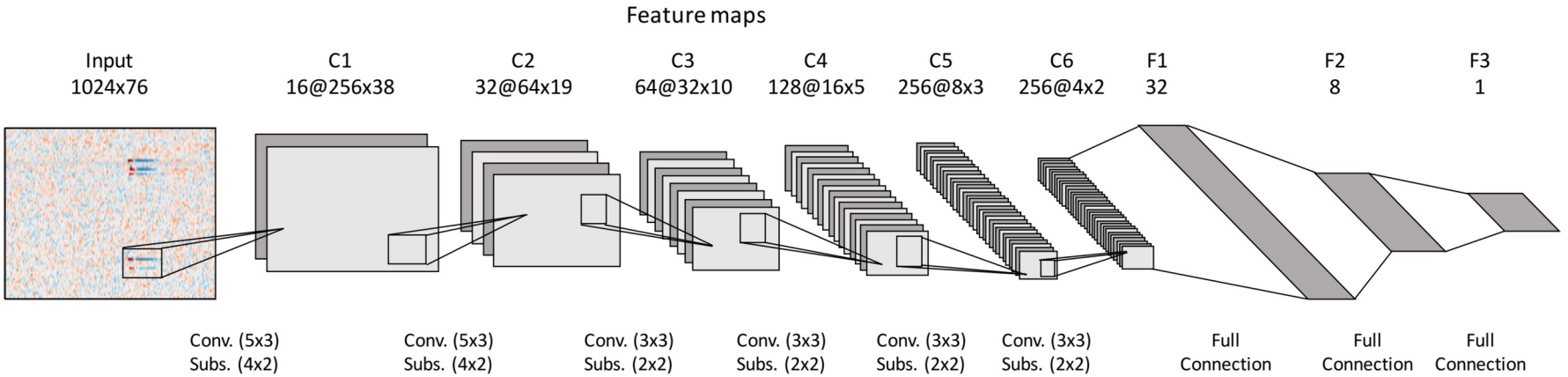
First application: charge energy reconstruction

- Now full events – all 76 U-wire waveforms (1024 time samples)
- Minimal Preprocessing: correct channel gains + crop waveforms



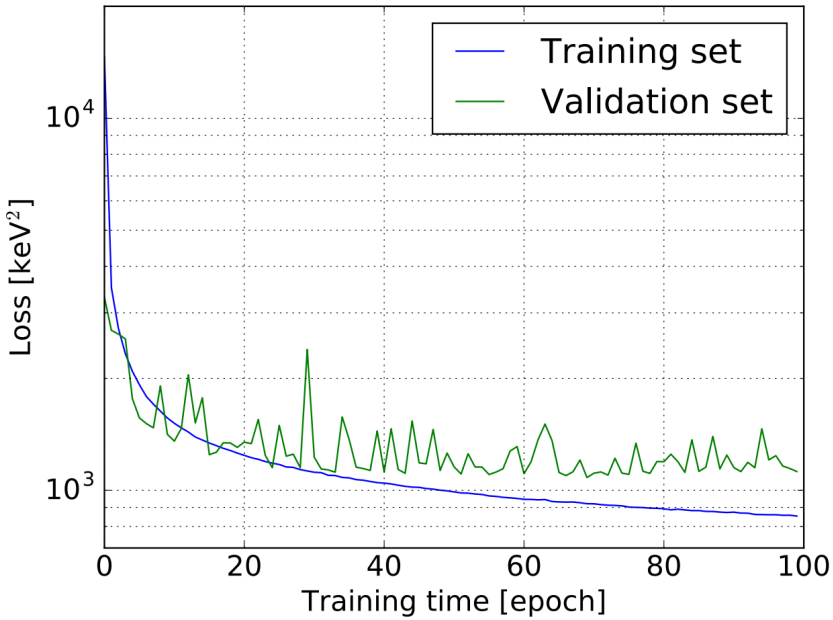
First application: charge energy reconstruction

- Input waveform image
- Convolutional part extracts features from image
- Dense part extracts target variable(s) from features



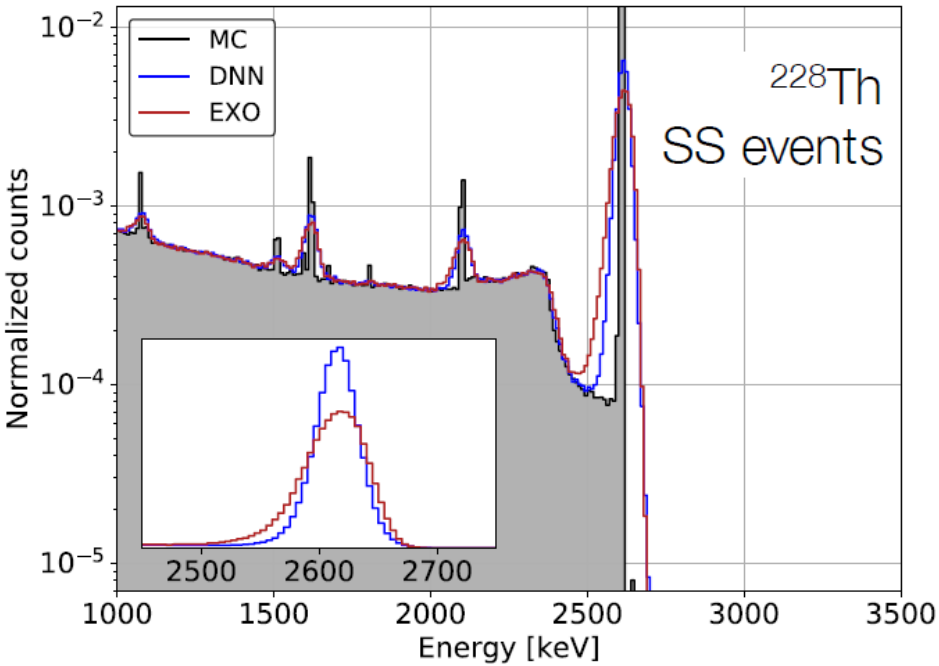
Charge reconstruction training details

- Training data:
  - Simulated events
  - Gamma ray source
  - Detector response uniform in energy
- Training:
  - 720 000 training events
  - 100 epochs
- Technical details:
  - Adam optimizer
  - Minimize mean square error
  - L2 regularization
  - ReLU activation
  - Uniform Glorot initialization



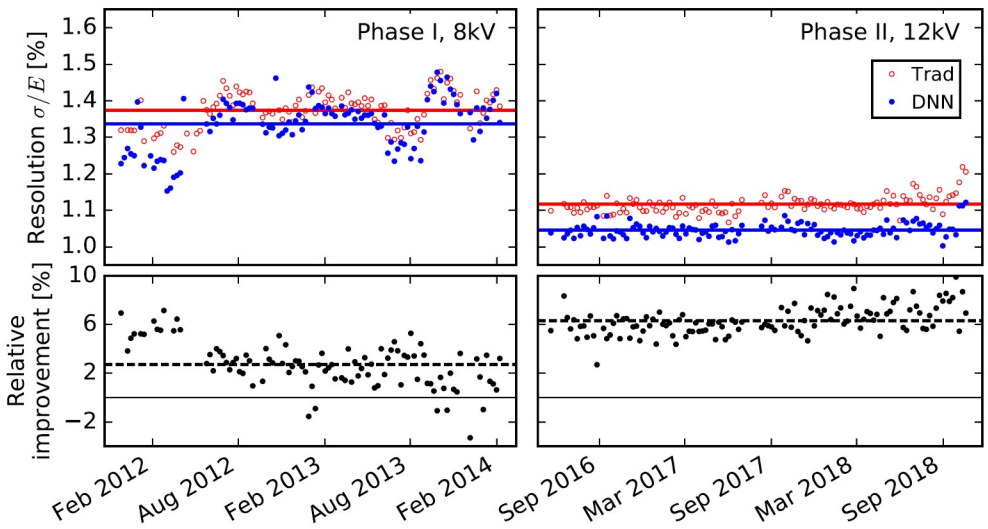
First application: charge energy reconstruction

- Reconstruction works on MC over the energy range under study
- Resolution (σ) at the ^{208}Tl full absorption peak (2615 keV):
  - **DNN: 1.21% (SS: 0.73%)**
  - **EXO Recon: 1.35% (SS: 0.93%)**
- Network outperforms in disentangling mixed induction and collection signals
  - See valley before ^{208}Tl peak, right in 0ν ROI!



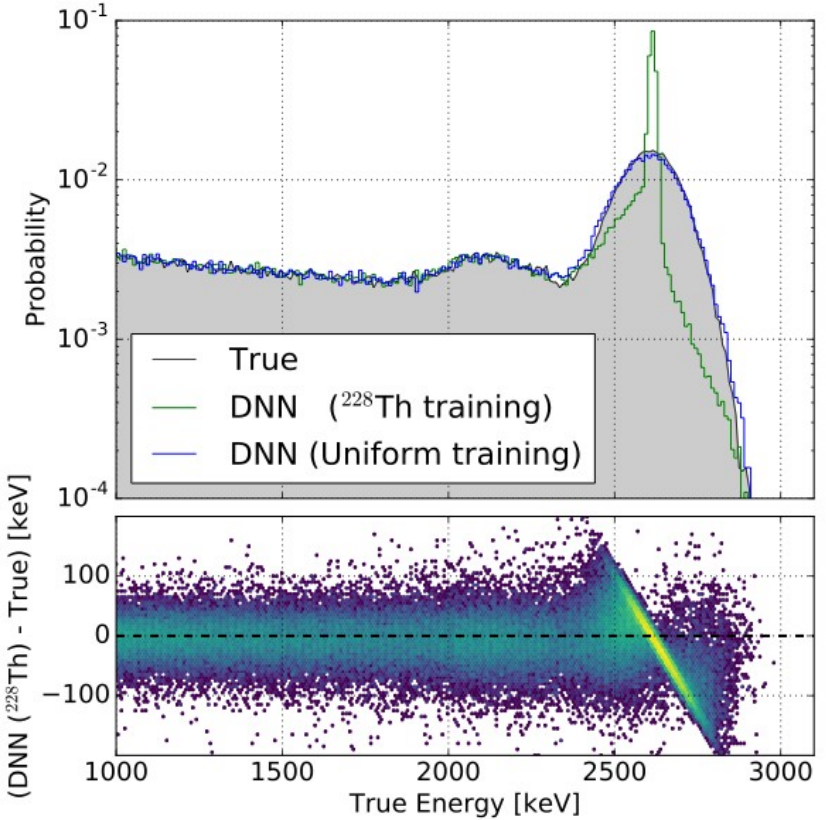
First application: charge energy reconstruction

- Applied to data and anti-corrleated with scintillation, the DNN based „rotated“ resolution outperforms EXO by 2-6%, depending on the week
- The better performance of the DNN alerted that something was lacking in the traditional approach and triggered improvements in EXO-recon
- While the cause is now largely understood (handling of mixed induction and collection signals), the developed traditional solution in EXO-recon is still outperformed by the DNN



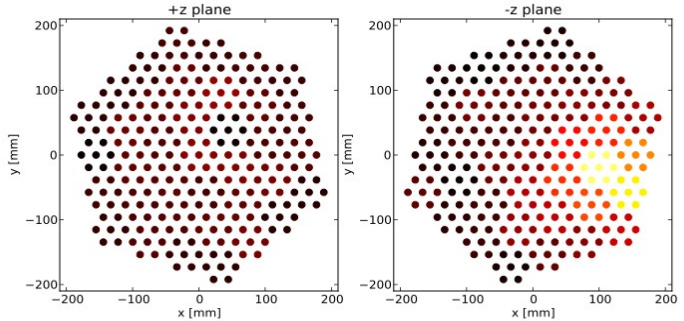
First application: Pitfalls of DNNs

- Potential danger of DNN is that they learn to reproduce the training data well but perform poorly on real data.
  - **Validation on real data is critical**
- We saw this in EXO-200:
  - DNN over-trains on sharp MC training peaks and shuffles independent validation events towards the sharp peaks → resolution too good to be true
  - Mitigated by using training events with uniform energy distribution



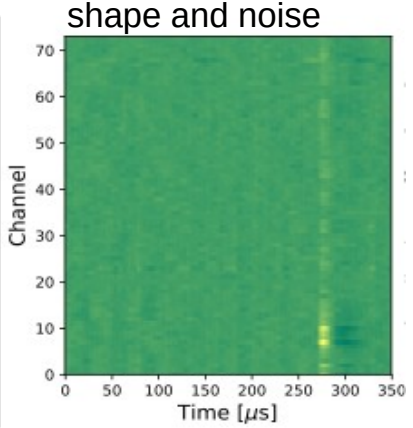
Second application: light position reconstruction

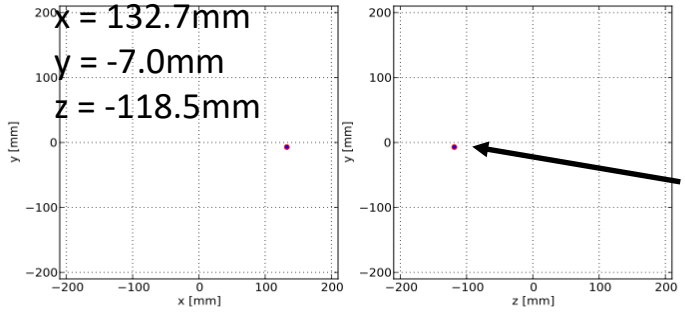
- Event position reconstruction from scintillation light
  - Truth label provided by ionization information of real data
  - Input are all 74 raw APD **real data** waveforms cropped to 350 µs



Event position is encoded in APD pattern

The time dimension adds information on waveform shape and noise



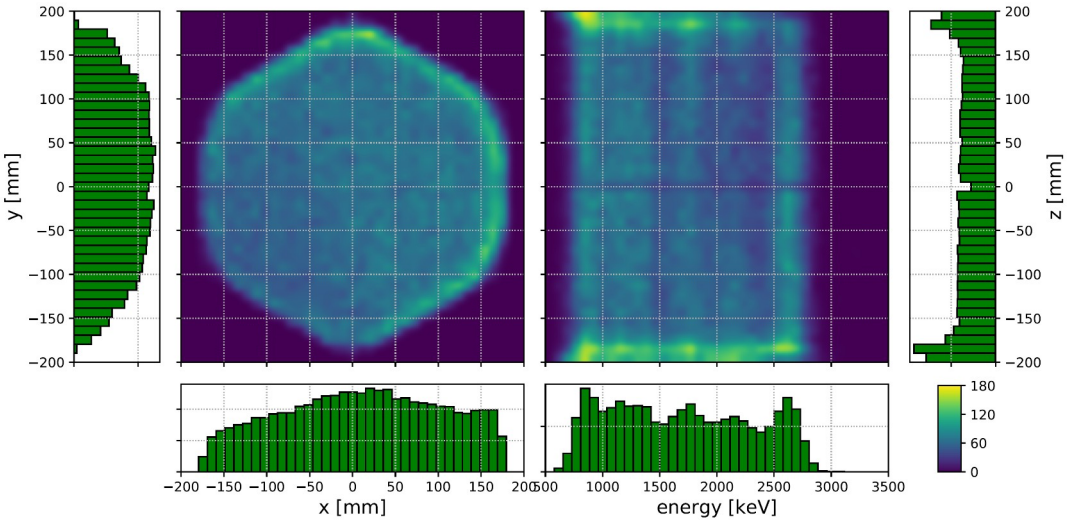


Truth information extracted from ionization signal

Light reconstruction details

- Waveform image is fed to CNN consisting of 4 convolutional and 3 fully connected layers
- Output has three units corresponding to event x-, y-, z-coordinates
- Loss function is Euclidean loss with L2 regularization

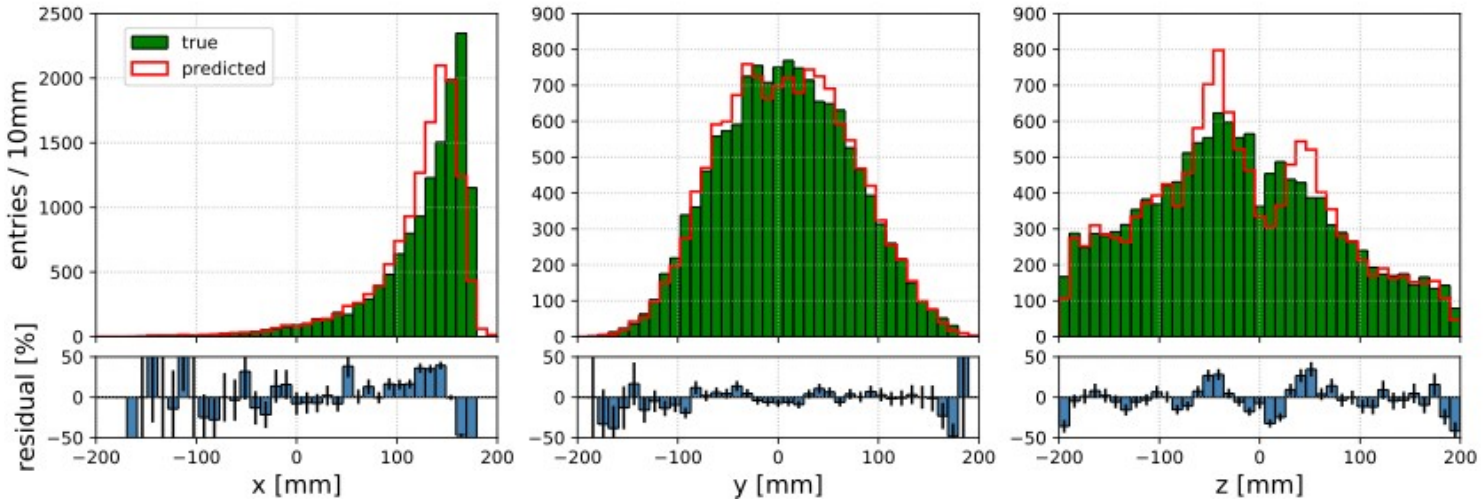
- Training is done on **real** calibration data with uniform distribution in space and energy



Second application: light position reconstruction

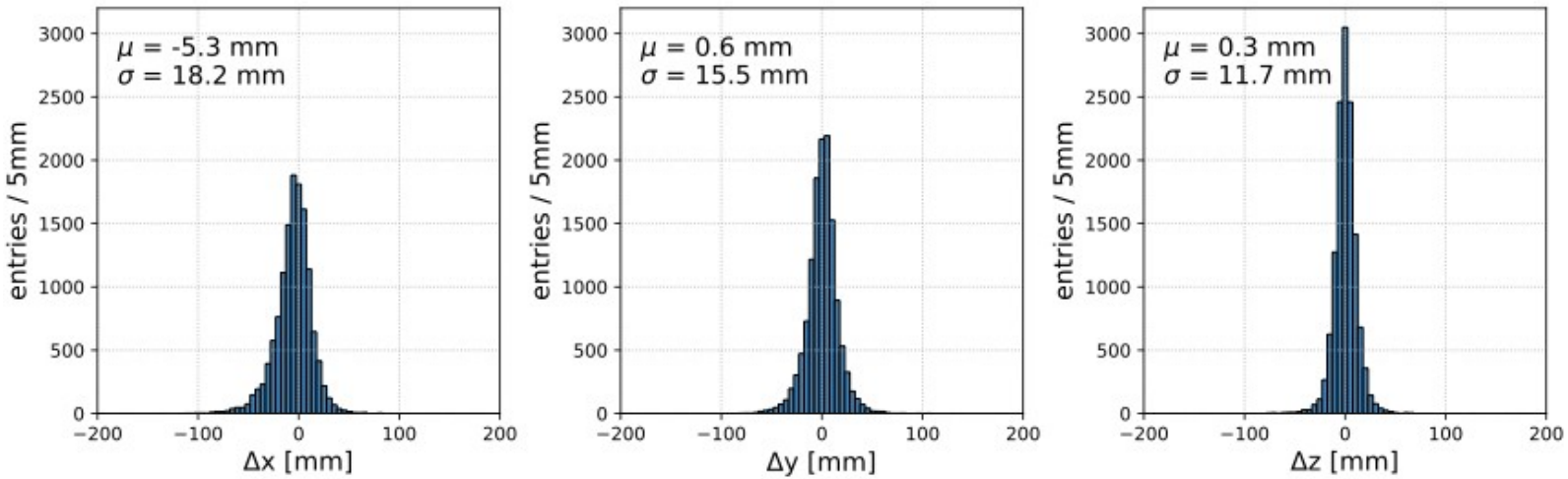
- Loss function reaches 200 mm^2 after training the DNN for 200 epochs
- The corresponding resolution in 3D is 25 mm
- The model is tested on different types of source data at different locations
- No alternative light position reconstruction in standard analysis, so uncontested

Accuracy: 22.5mm (d_x = 13.6mm, d_y = 11.3mm, d_z = 8.1mm) corresponding to R^2 = 0.99



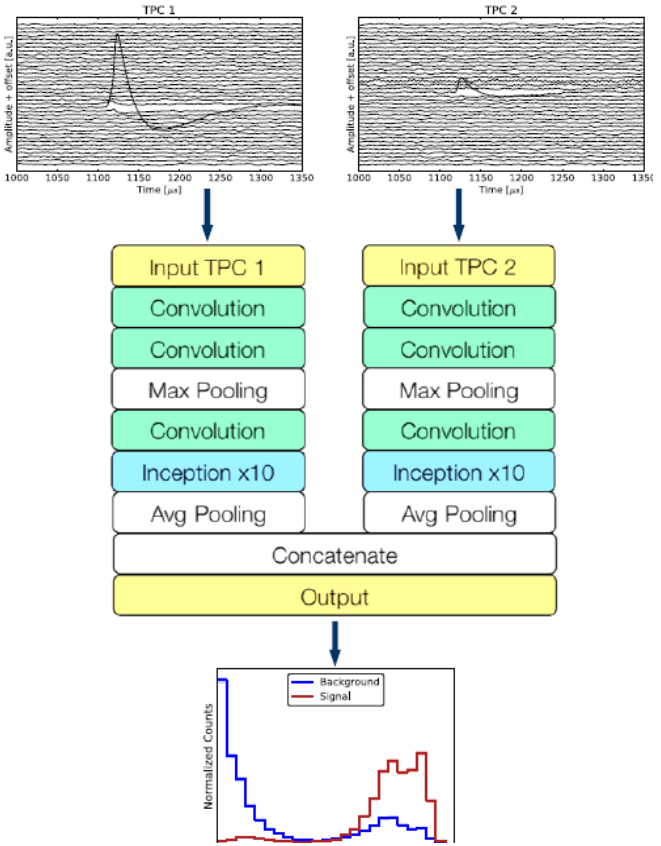
Second application: light position reconstruction

- Loss function reaches 200 mm^2 after training the DNN for 200 epochs
- The corresponding resolution in 3D is 25 mm
- The model is tested on different types of source data at different locations
- No alternative light position reconstruction in standard analysis, so uncontested



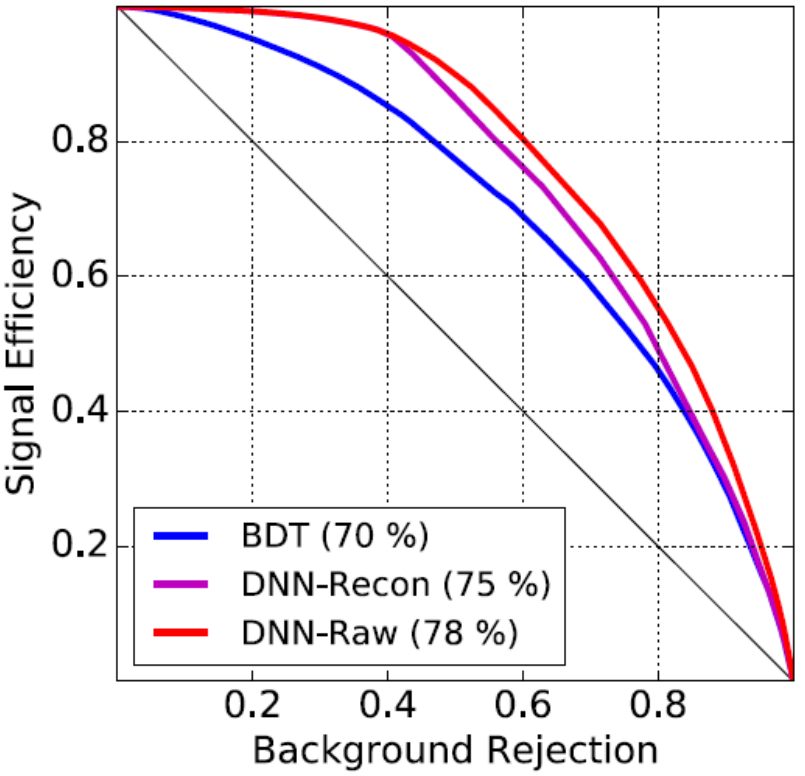
Third application: Signal/Background Discrimination

- Binary (ββ vs γ) DNN based discriminator as an additional variable to the “traditional” ML fit
- DNN trained on waveforms re-generated from EXO recon’d signals (not on raw waveforms)
- Shared weights on the two TPC-halves branches
- Training events with uniform energy distribution, 50/50 signal/background



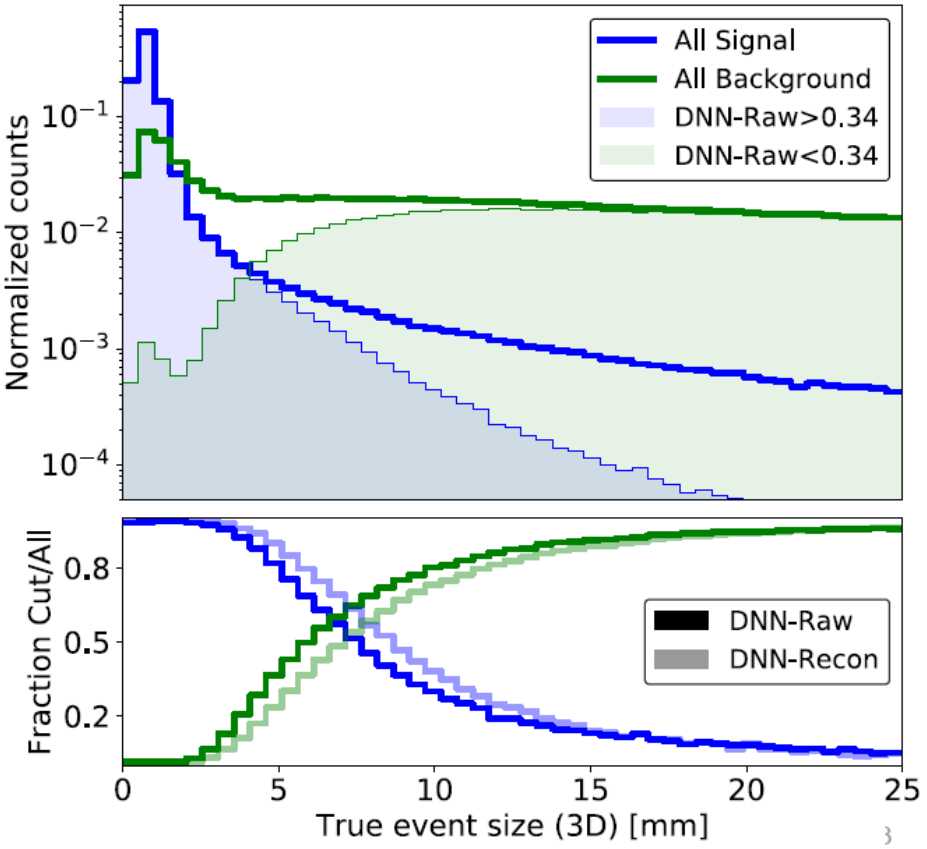
Third application: Signal/Background Discrimination

- DNN outperforms previously used BDT discriminator
- Overall, a 25% sensitivity improvement, compared to non-ML based analysis
  - *Phys. Rev. Lett. **123**, 2019, 161802*
  - Kudos to grad. students who made this happen (Tobias Ziegler&Mike Jewel most of all)



Third application: Signal/Background Discrimination

- ββ events are more localized than γ
- DNN efficiency demonstrates correlation with the true event size in the MC
- Indicates that the DNN picks up correct features of the waveform when reconstructing events
- Data/MC agreement of the “DNN variable” validated with real calibration data
  - Agreement not perfect, but comparable to other “shape” errors.

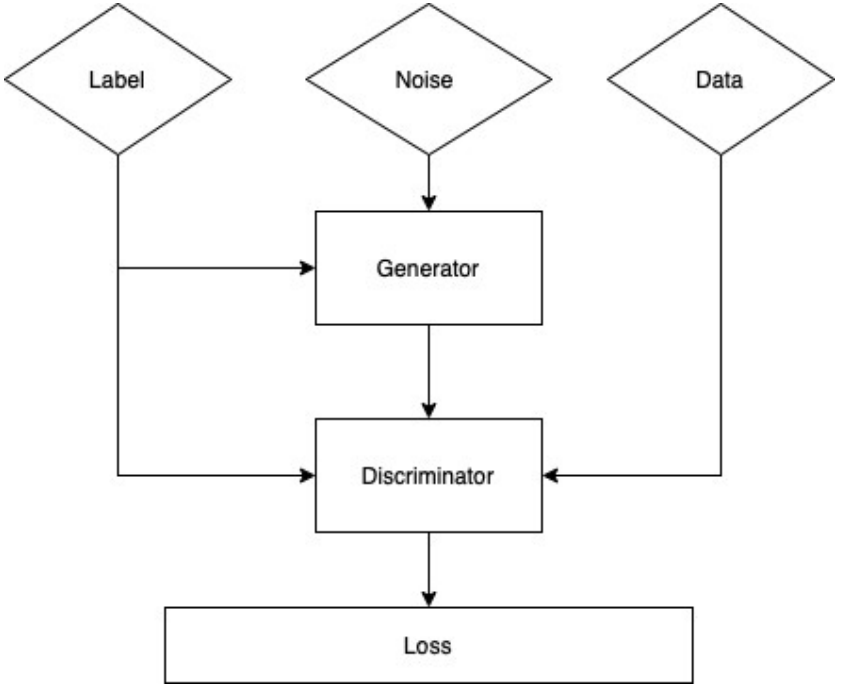


More recent: MC with GANs

- EXO-200's earlier attempts to develop a detailed photon-tracking MC did not succeed
  - Poor agreement with data, possibly due to imperfect knowledge of optical properties or shortcuts in geometry implementation
  - It was also very resource-consuming to track photons
  - A simple parametric simulation of the overall light yield per one array of APDs was used instead, only for limited purposes
- We showed that one can train a GAN network directly with waveforms from calibration data, bypassing the needs for detailed knowledge of optical properties and detector geometry
  - *Importantly, we compared the output at all levels – from raw waveforms to signal amplitude and its position dependency, to reconstructed energy spectra*
  - *JINST 18 P06005 (2023)*. Led by UCSD grad. student Shaolei Li

More recent: MC with GANs

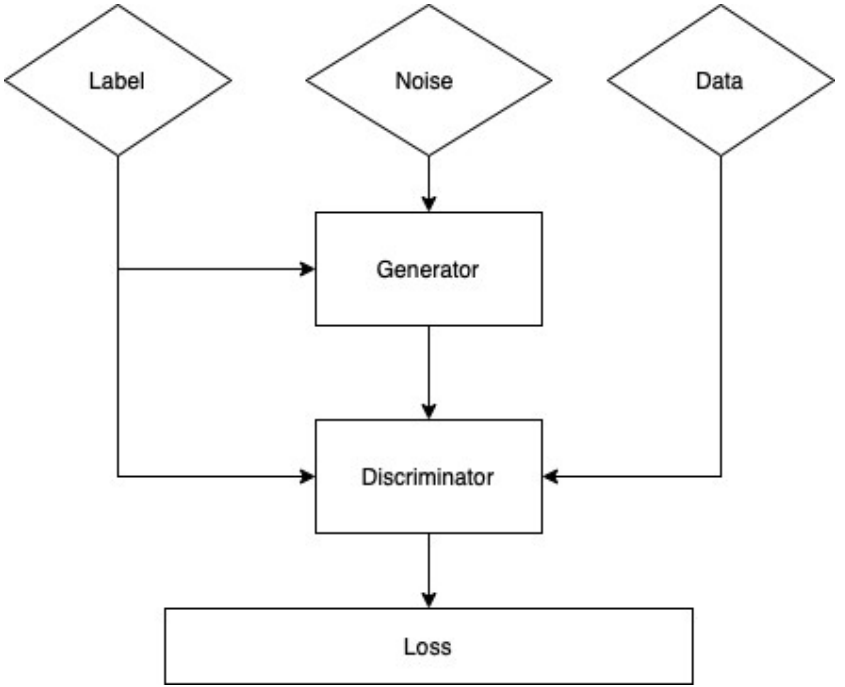
- Generator starts from white noise and label with requested position, energy
- Critic (discriminator) compares the generated waveform to data sample
- Wasserstein (Васерштейн) distance, aka Kantorovich distance, as a metric for comparison (more stable than standard GAN, outputs a continuous variable, instead of classification real/fake)
- Constrainer: supervises training and ensures the generated waveform conforms to the requested label



$$L = \underbrace{\mathop{\mathbb{E}}_{\tilde{x}\sim P_g}[D(\tilde{x})] - \mathop{\mathbb{E}}_{x\sim P_r}[D(x)]}_{\text{Wasserstein distance}} + \underbrace{\lambda \mathop{\mathbb{E}}_{\hat{x}\sim P_{\hat{x}}}[(\|\nabla D(\hat{x})\|_2 - 1)^2]}_{\text{gradient penalty}}$$

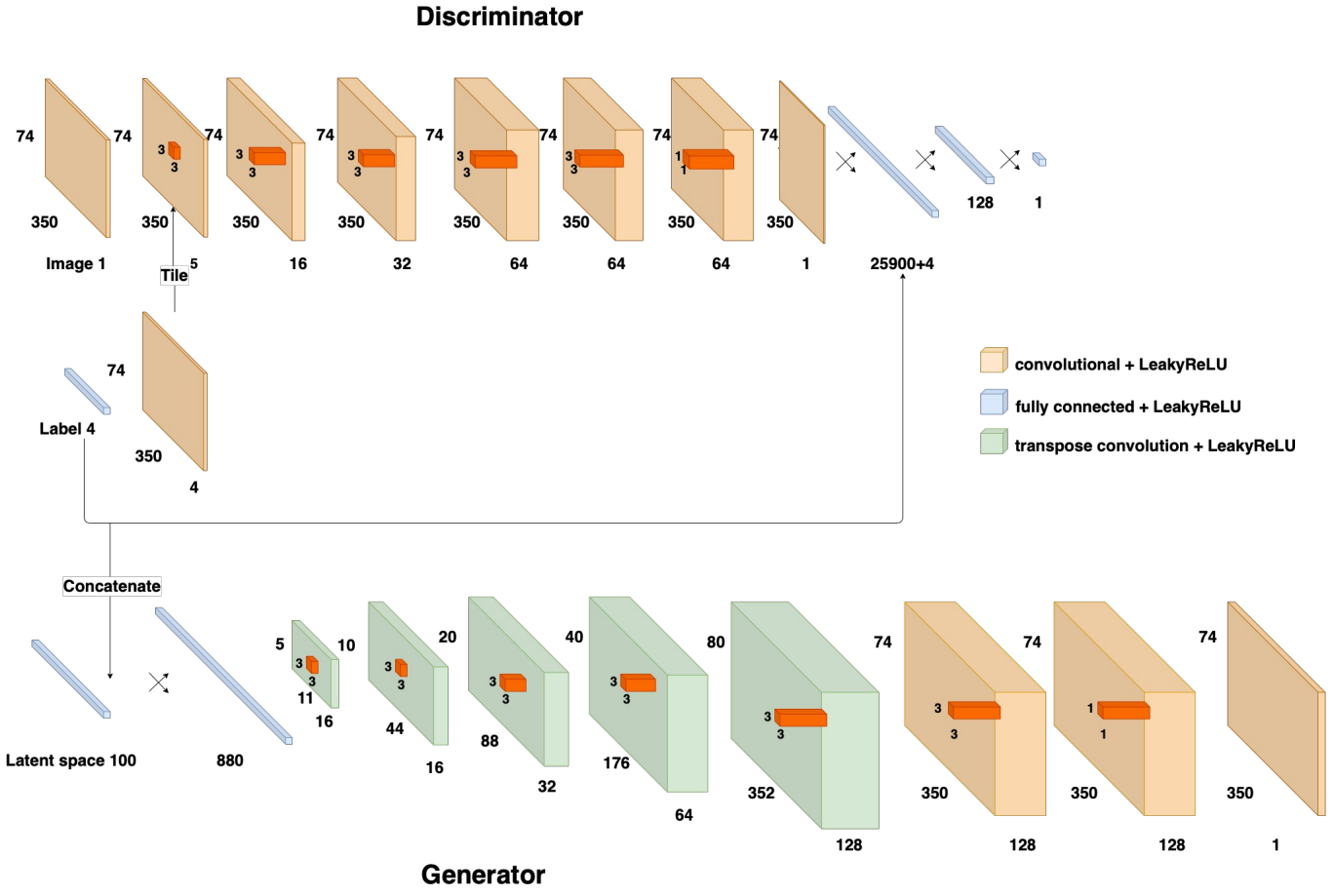
More recent: MC with GANs

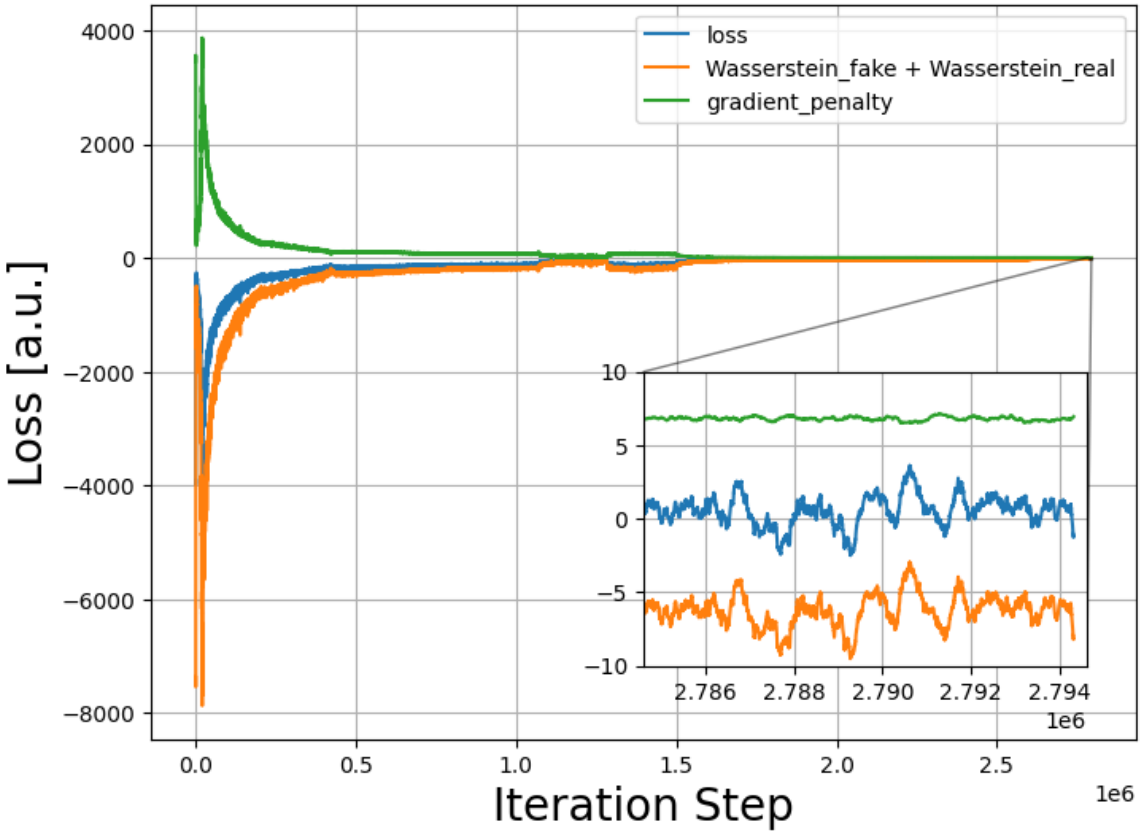
- Wasserstein (Васерштейн) distance, aka Kantorovich distance, aka Earth's Mover Distance as a metric for comparison (loss)
  - The "distance" the generator must "move" the fake data to match the real one
- In practice, this is realized by two ingredients:
  - Output neuron of the discriminator is linear (not sigmoid)
  - The output is made into a ~1-Lipschitz function by constraining the norm of the critic's gradient (wrt interpolated input) to 1
  - When trained under these conditions, the difference between (averaged over batch) the descriminator's outputs for fake and real images is a direct estimate of the Mover's distance between them



$$L = \underbrace{\mathbb{E}_{\tilde{x}\sim P_g}[D(\tilde{x})] - \mathbb{E}_{x\sim P_r}[D(x)]}_{\text{Wasserstein distance}} + \underbrace{\lambda \mathbb{E}_{\hat{x}\sim P_{\hat{x}}}[(\|\nabla D(\hat{x})\|_2 - 1)^2]}_{\text{gradient penalty}}$$

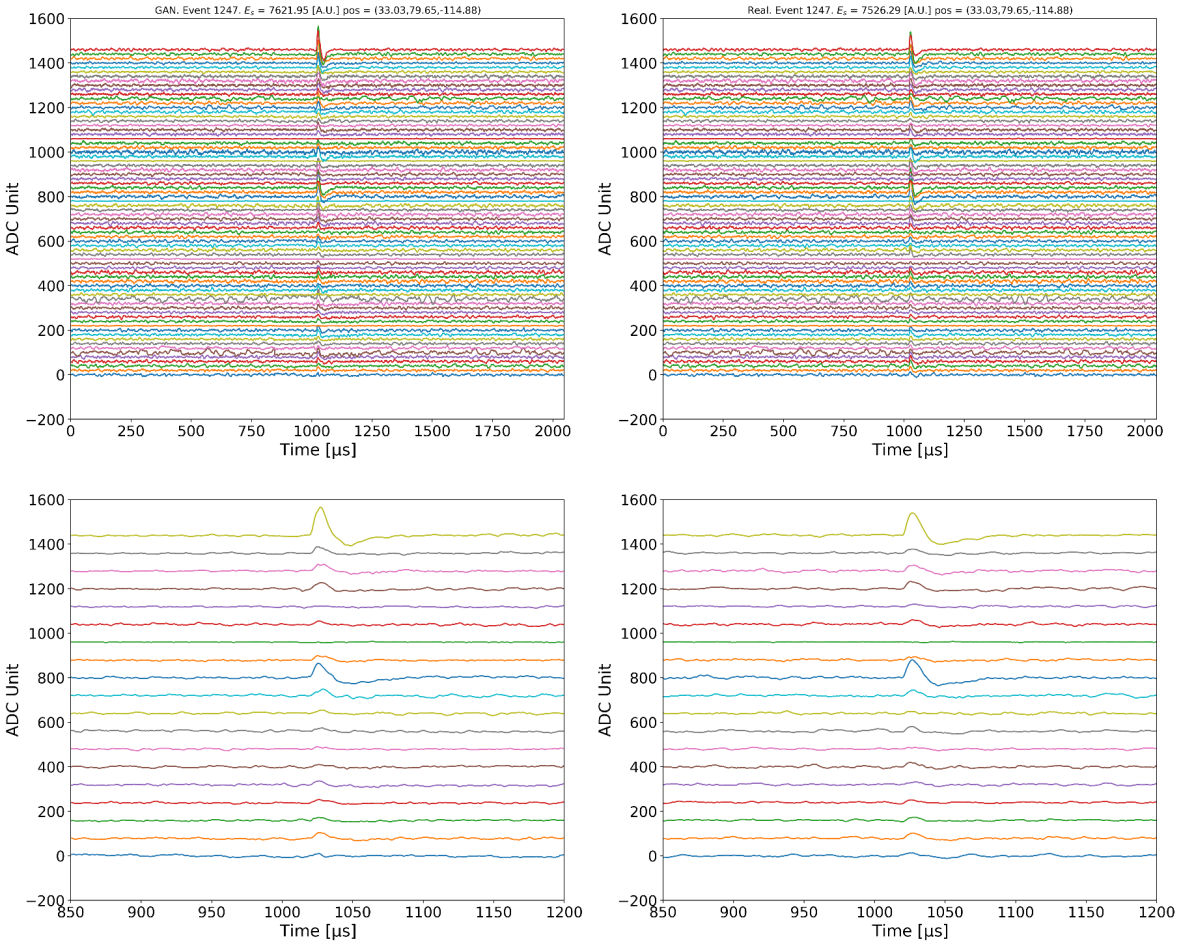
generated real interpolated


Discriminator
74
74
350
Image 1
3
3
74
350
Tile
5
3
3
74
350
16
3
3
74
350
32
3
3
74
350
64
3
3
74
350
64
1
1
74
350
64
74
350
1
25900+4
128
1
74
Label 4
350
4
convolutional + LeakyReLU
fully connected + LeakyReLU
transpose convolution + LeakyReLU
Concatenate
Latent space 100
880
5
10
3
3
11
16
20
3
3
44
16
40
3
3
88
32
80
3
3
176
64
74
3
3
352
128
74
3
3
350
128
74
1
1
350
128
74
350
1
Generator


loss
Wasserstein_fake + Wasserstein_real
gradient_penalty
Loss [a.u.]
Iteration Step
1e6
4000
2000
0
−2000
−4000
−6000
−8000
0.0
0.5
1.0
1.5
2.0
2.5
10
5
0
−5
−10
2.786
2.788
2.790
2.792
2.794
1e6

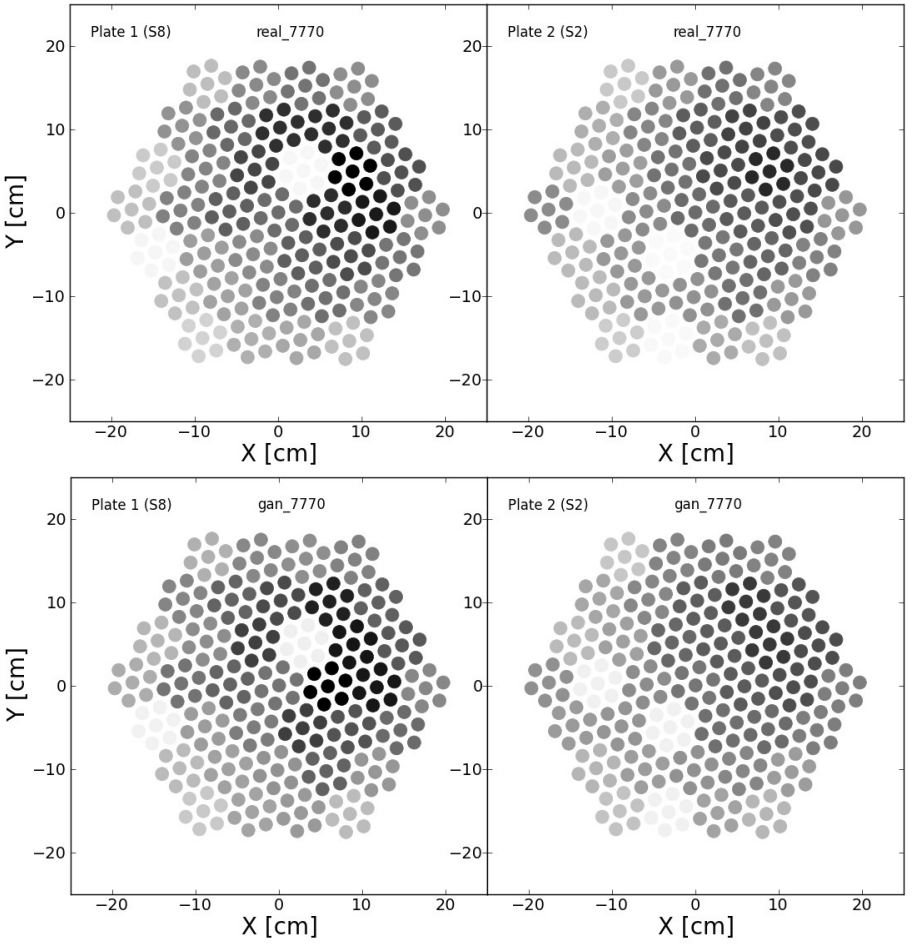
More recent: MC with GANs

- Raw waveform comparison
- GAN generates waveforms more than an order of magnitude faster than the standard EXO approach
  - that does not even include photon tracking



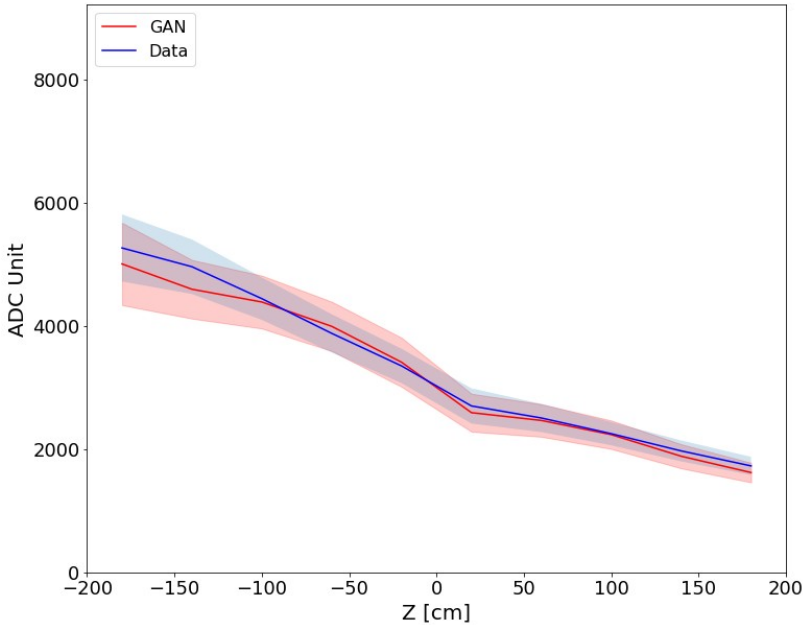
More recent: MC with GANs

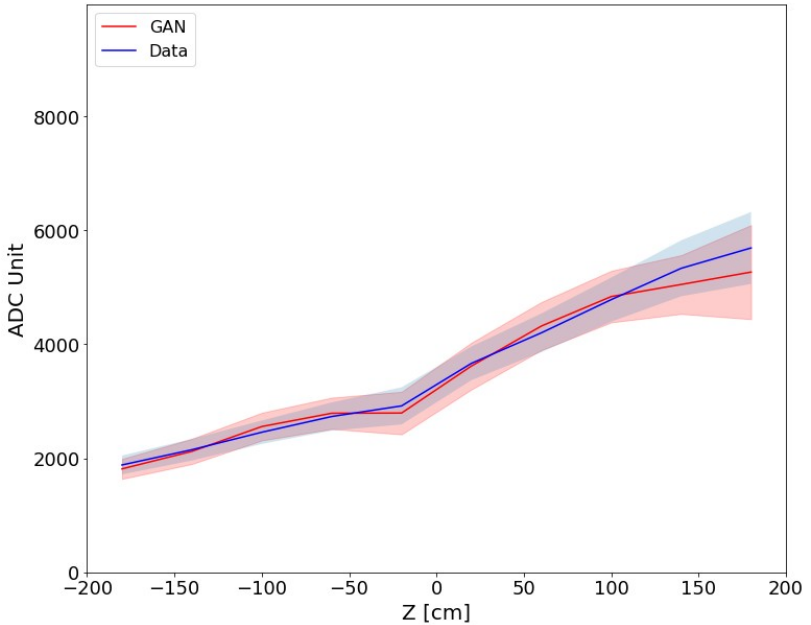
- Summed amplitude per APD gang
  - GAN reproduces the dead channels



More recent: MC with GANs

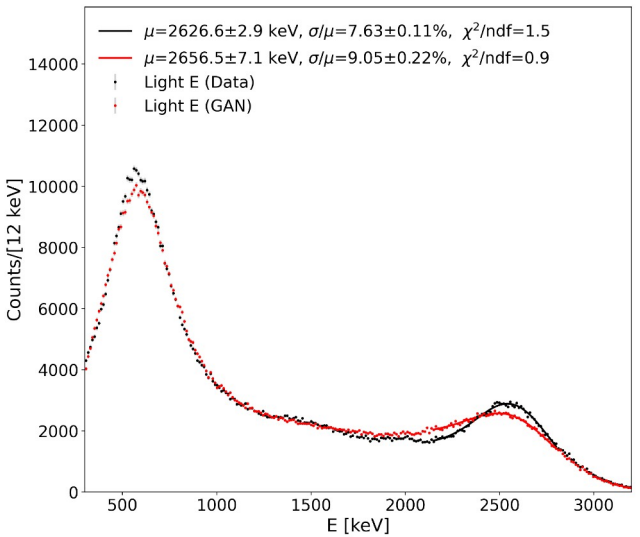
- Position dependence of light response reproduced

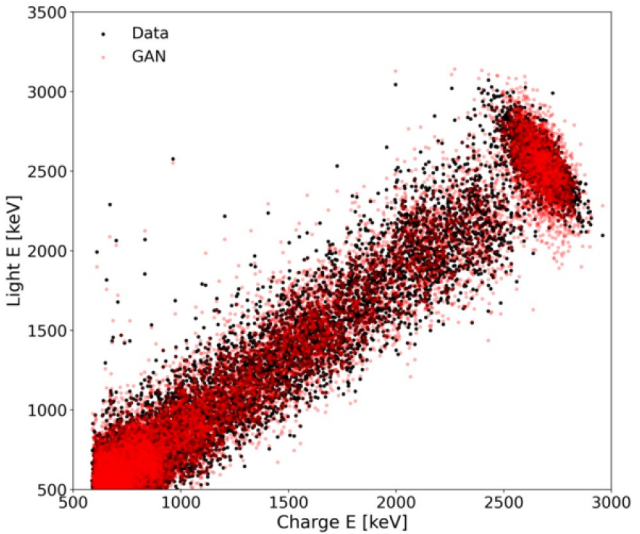




More recent: MC with GANs

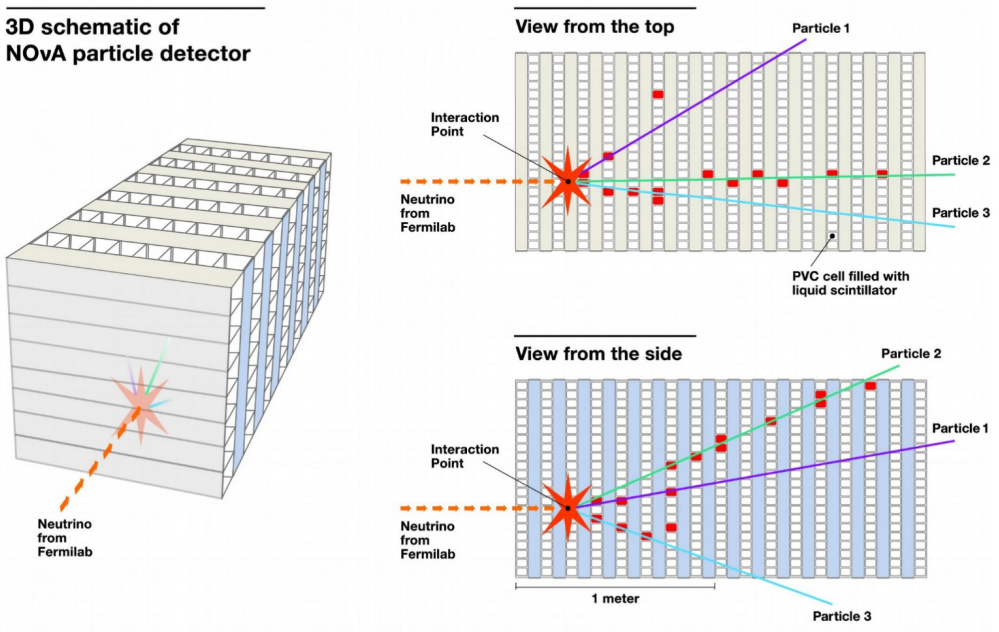
- Anti-correlation between charge and light signals reproduced
  - Optimal angle is slightly different
- Light-only energy spectrum looks good but does not reproduce the resolution exactly
  - Consistent with the extra uncertainty added by imperfect truth labels. Experiments that could train on calibration data with more precise labels can do better





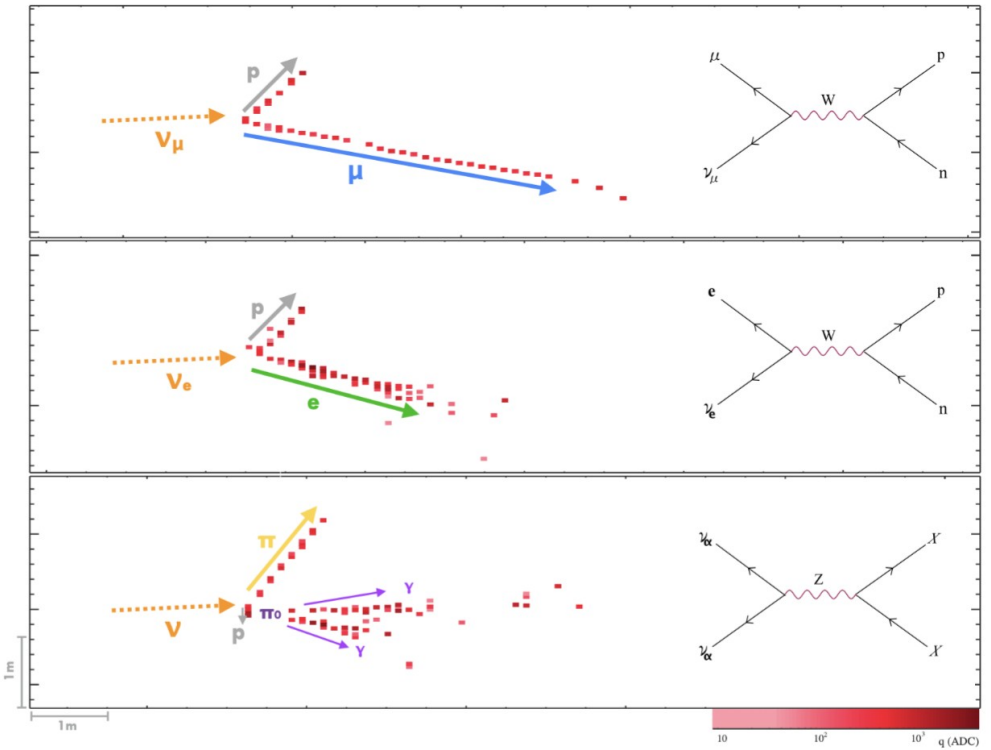
NOvA

- Long-baseline neutrino oscillation experiment at Fermilab's NuMI beam
- Two functionally identical detectors 809 km apart
  - Plastic extrusions filled with a liquid scintillator
  - Light routed by WLS fibers to APDs
- Far detector consists of **896** alternating horizontal and vertical planes
  - Each plane contains **384** 4 cm x 6 cm x 155 cm cells for a total of **344,064 cells**



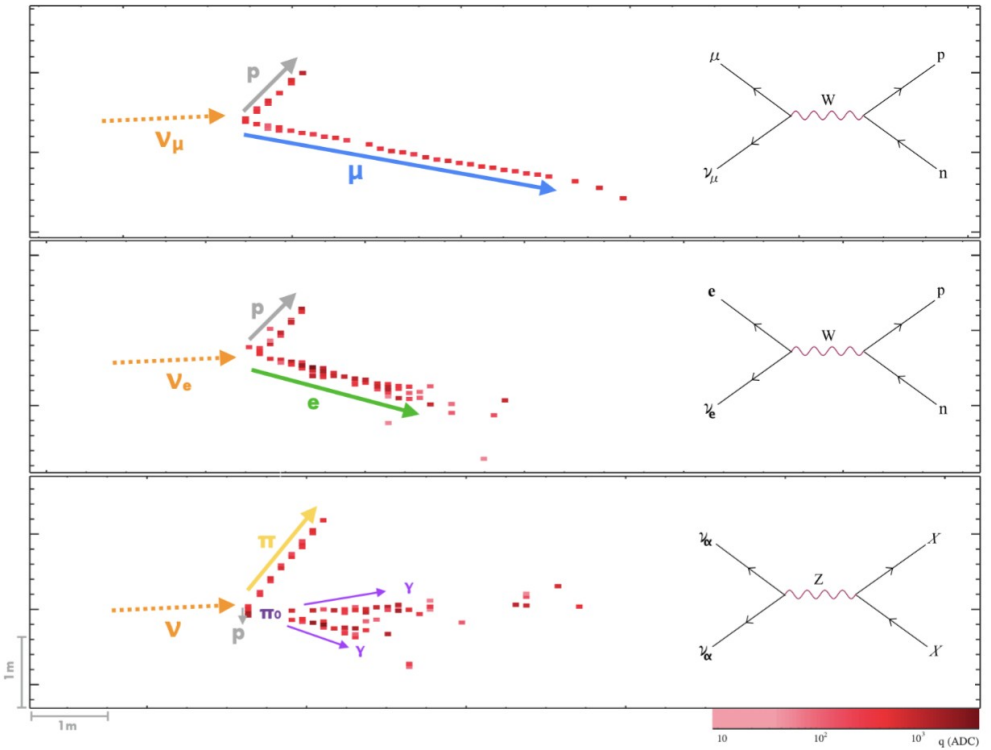
NOvA

- Reconstruction of the ν energy and flavor state at the detector is essential to oscillation measurements
- The flavor state can be determined in charged-current (CC) interactions which leave a charged lepton in the final state
- Neutral-current (NC) interactions bear no signature of the flavor of the interacting neutrino and are thus a background for the charged-current analyses



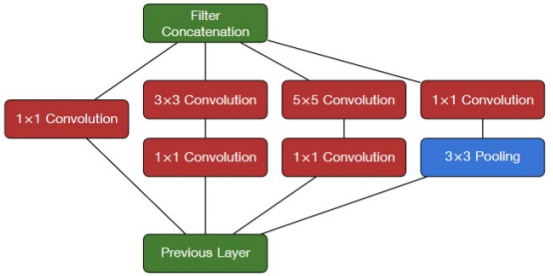
NOvA: First result, CVN

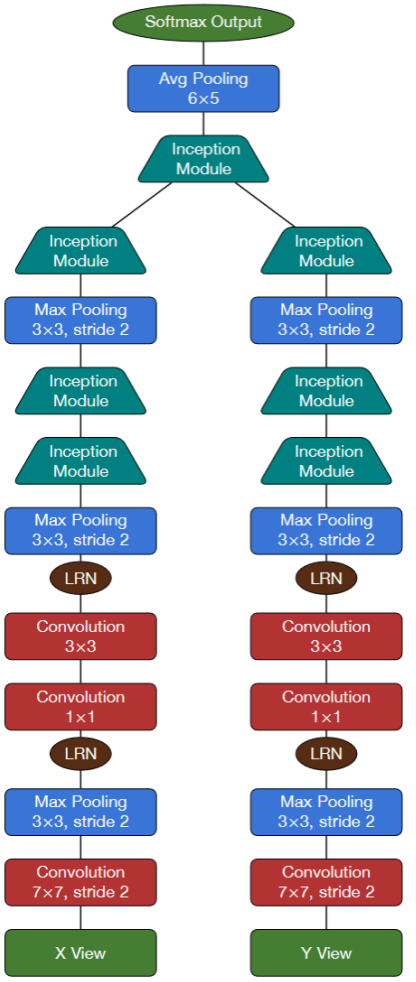
- A convolutional neural network to classify events into:
  - ν_μ CC (long, low dE/dx track)
  - ν_e CC (wide shower, rather than a track)
  - ν_τ CC (varying τ decay final states)
  - ν NC (flavor impossible to identify)
- Directly from pixel maps
  - 100 (planes) x 80 (cells) image
  - Two views (x-z and y-z) for each event
  - Pixel intensity = calibrated energy
    - 8-bit representation used for an **8-fold savings over floating point**, w/o substantial reduction in performance



2016 JINST 11 P09001

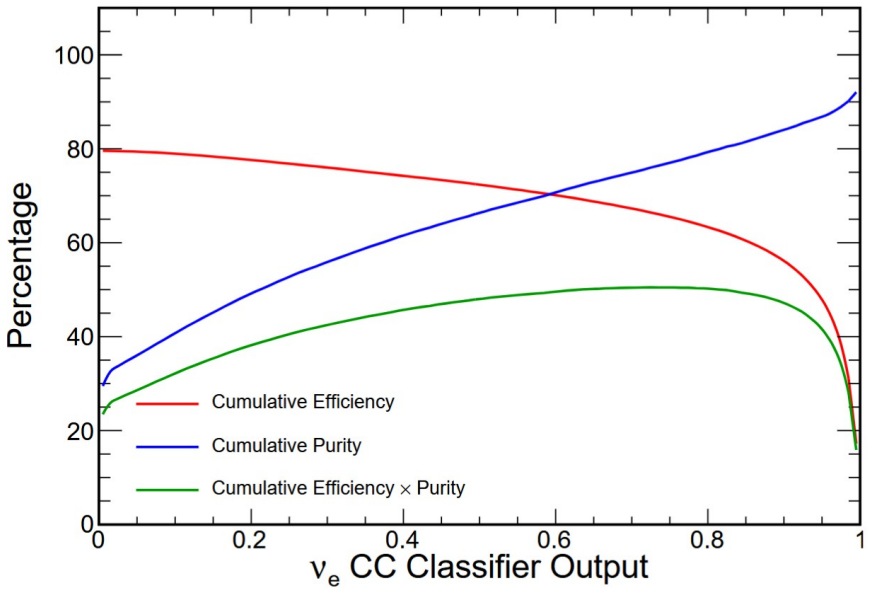
NOvA: First result, CVN

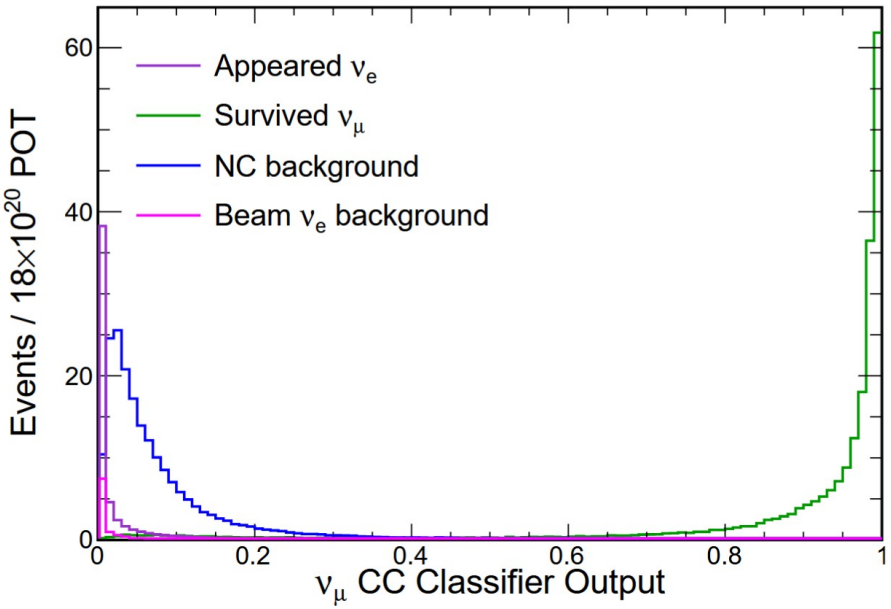




- Two-path CNNs, with each branch based on GoogLeNet
  - "Inception modules" distribute input from preceding layers into branches with filters of different scales
  - "Local response normalization" normalizes the response of a given cell in a kernel map relative to the activity of adjacent kernel maps. This helps the network avoid local minima and to converge to a more optimal set of weights
- Training details:
  - ~10^5 training events per class
  - 32 events per batch
  - Regularization by L2, dropout, and data increase by augmentation

NOvA: First result, CVN

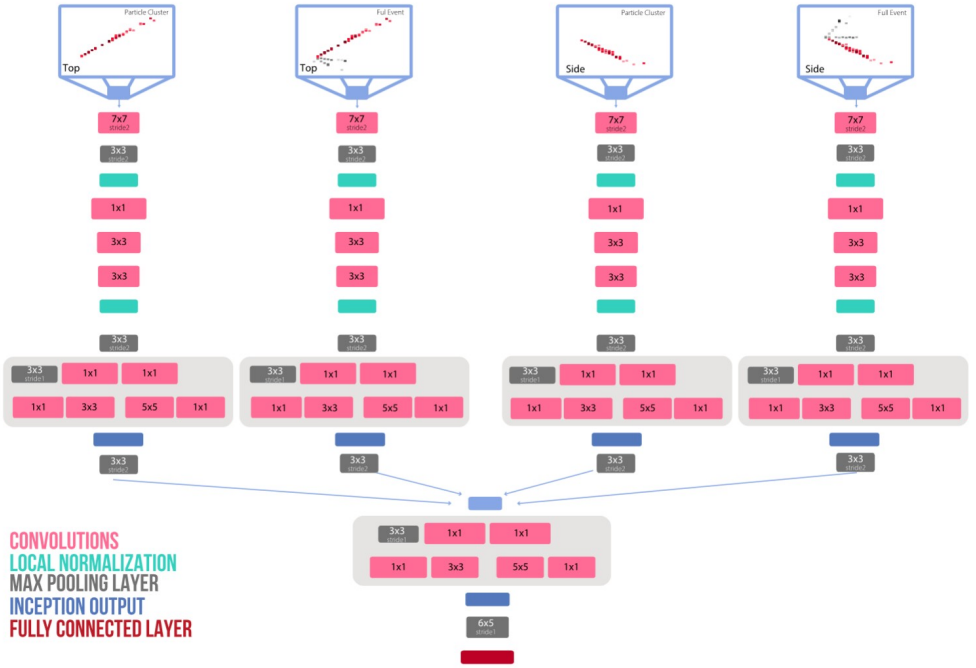




- Modest improvement (58% vs 57%) for recon efficiency of ν_μ CC interactions, cf. traditional approach
- Bigger gain (49% vs 35%) for ν_e. Significant as ν_e-appearance measurements (θ_{13}, mass hierarchy) are stat-limited in NOvA
- Similar sensitivity to the systematic uncertainty of the inputs

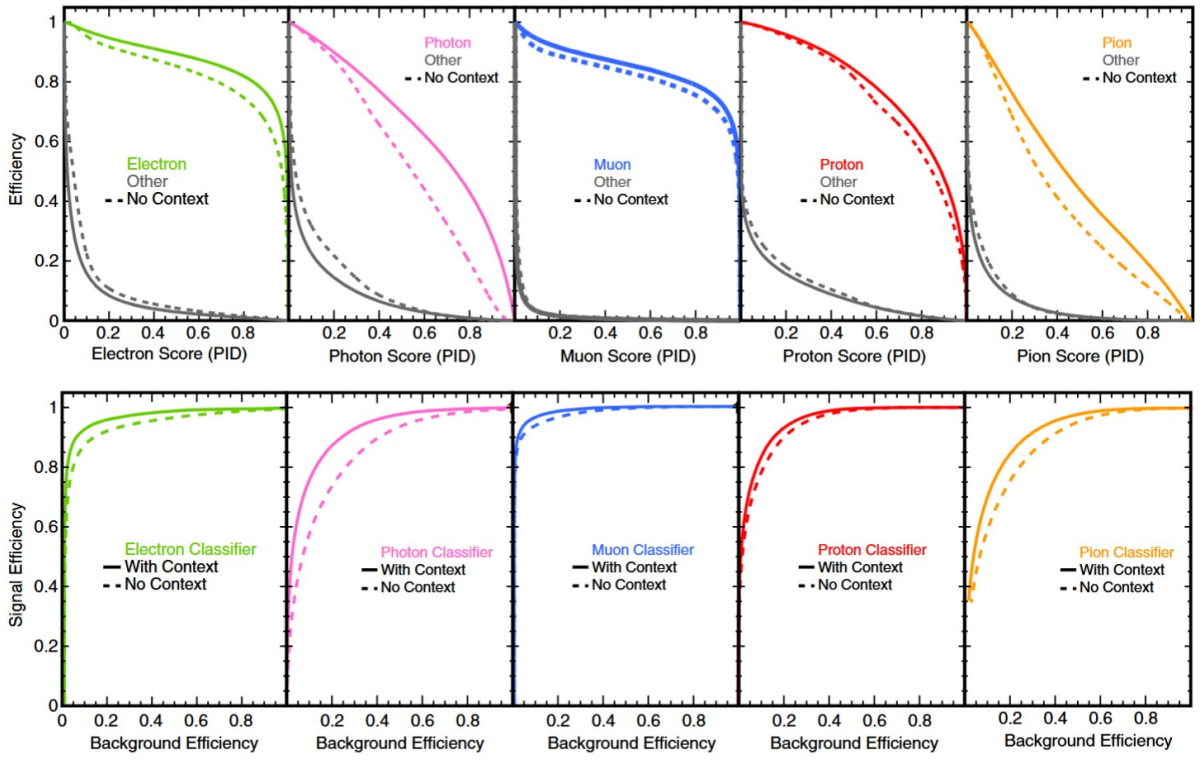
NOvA: Second result, ProngCVN

- Extension of the initial approach to classify *individual particles* (cf., events) using both views of the particle and both views of the entire event
- This gives the network ***contextual information*** about individual particles



Physical Review D 100, 073005 (2019)

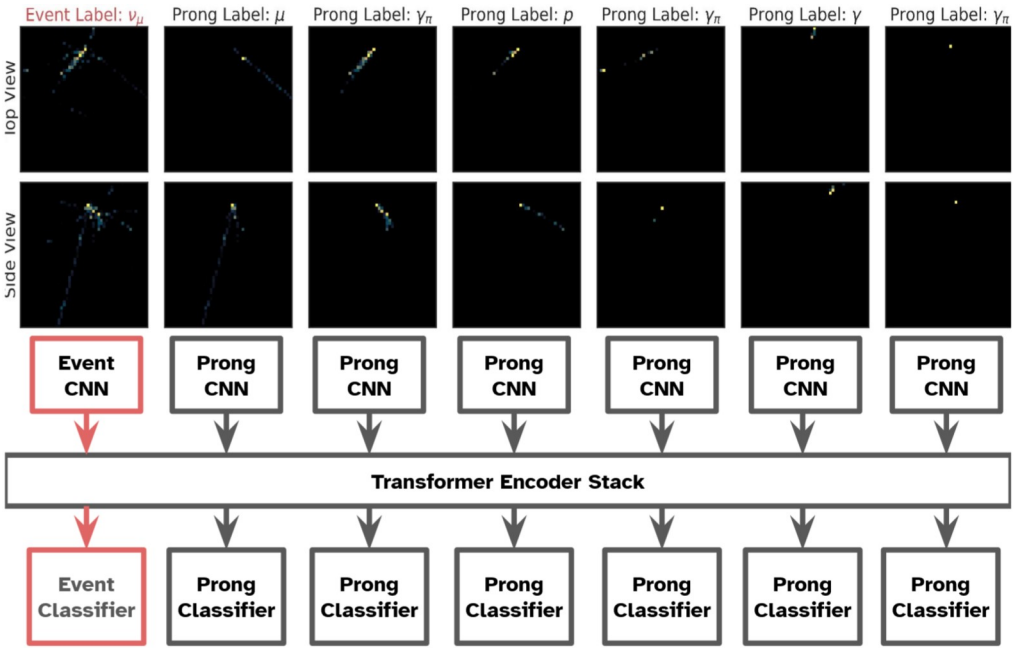
NOvA: Second result, ProngCVN



- Improvements for all particle types compared to particle-only network
- In particular, 10% improvement in efficiency of selecting γs and πs

NOvA: More recent, Joint particle/event classification

- NOvA events leave >99% of pixels empty
  - SparseCNN concept applies convolution only in regions where data exist, saving on resources
  - Replaces traditional convolution and pooling with Minkowski sparse convolution
- Number of prongs varies between events, requiring to classify variable-length sets
  - Transformer architecture, commonly used for language processing, is well-suited for this
  - Transformer is attention-based, focusing on regions with high importance, further reducing computation burden; Provides means to study interpretability

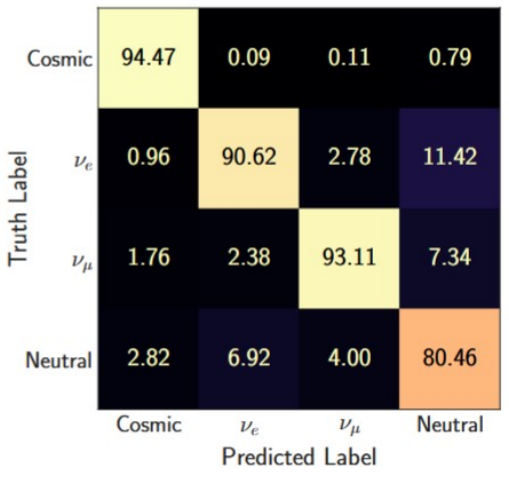


arXiv:2303.06201v1

NOvA: More recent, Joint particle/event classification

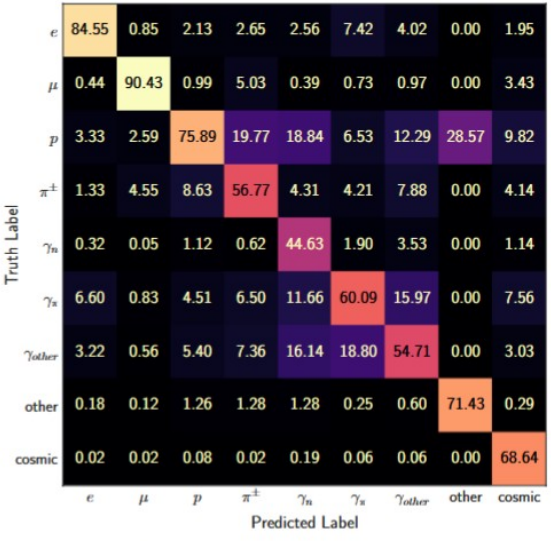
- Comparable to the earlier CVN for event reconstruction
- Few percent improvement in particle reconstruction
- Personal note: this is not particularly impressive. Could probably be achieved with old architecture and much larger training dataset size, hyperparameter optimization

Metric	Transformer CVN	Event CVN
Accuracy	0.894	0.897
Precision	0.894	0.908
Recall	0.894	0.897
ROC AUC	0.982	0.984



(a) Event Reconstruction

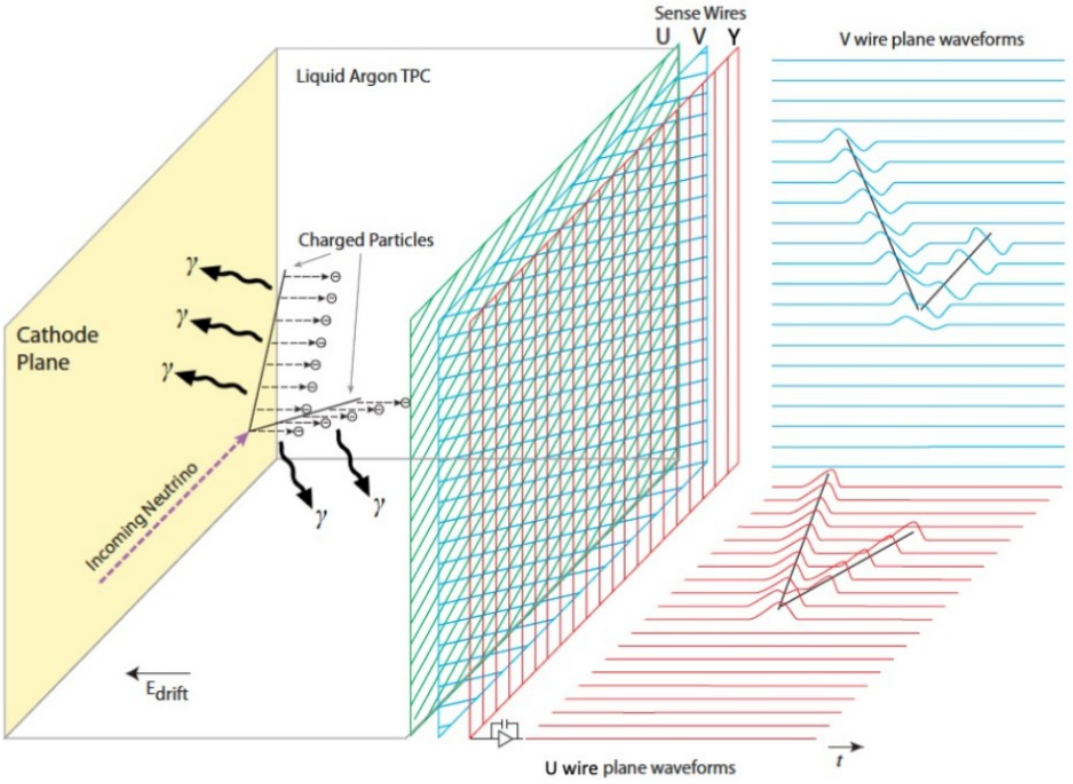
Metric	Transformer CVN	Prong CVN
Accuracy	0.783	0.726
Precision	0.783	0.760
Recall	0.783	0.726
ROC AUC	0.951	0.932



(b) Prong Reconstruction

SBND: Cosmogenic background removal

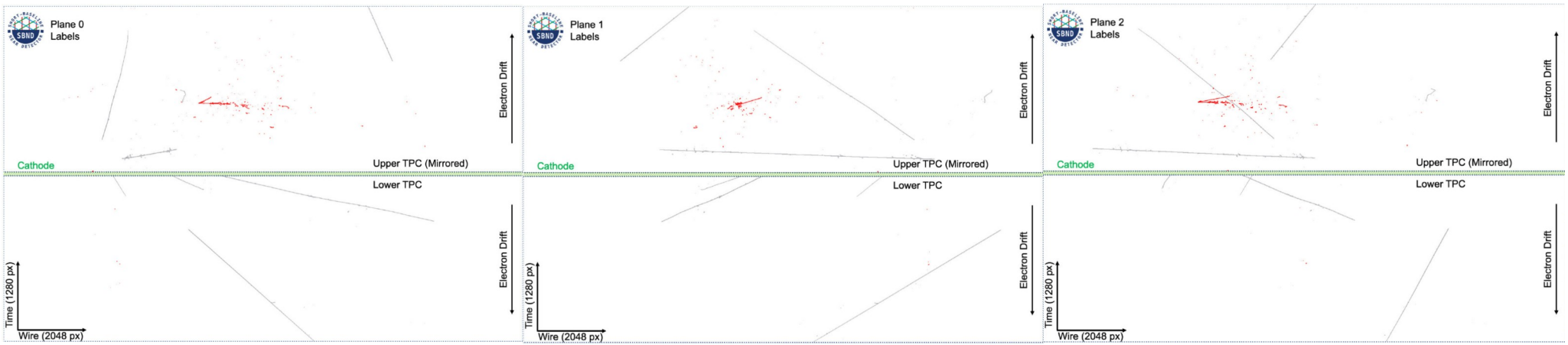
- SBND is a large LAr TPC searching for sterile νs
- Anode Plane Assemblies per for charge readout each half-TPC
  - 3 wire planes per assembly
  - ~11k wire channels total
- Scintillation light readout and veto used to remove backgrounds, but not sufficient for cosmic ray contamination (shallow location)
  - Pattern recognition applied to TPC data needed to discern cosmic- from ν-induced activity



Front. Artif. Intell. 4(555), 370–385 (2021)

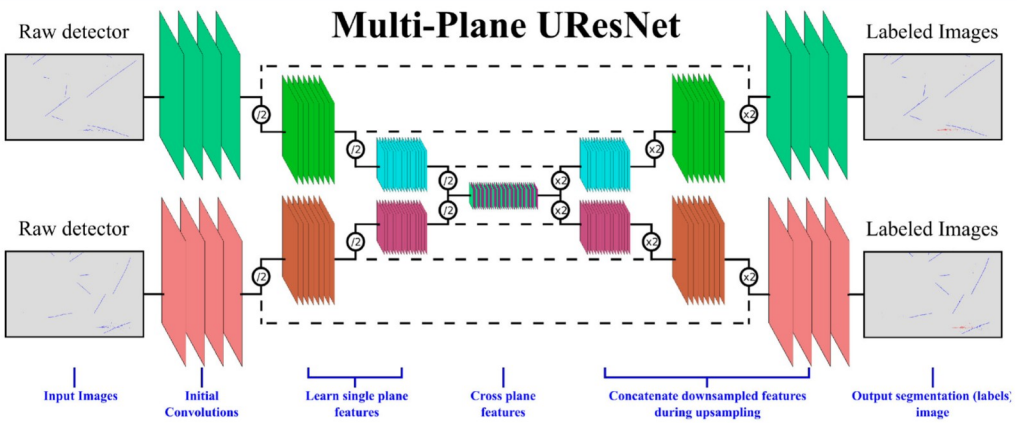
SBND: Cosmogenic background removal

- Tagging the raw TPC data as cosmic- or ν-induced on a pixel-by-pixel basis
- Original GENIE+CORSIKA MC dataset with event images as a combination of three planes with 1280 (time) x 2048 (wire) pixels
  - Too large for GPU memory available at the time. Decreased resolution to 640 x 1024. Plans to rerun with full resolution once better hardware is available
- Pixel-level Truth information (red - neutrinos, grey - cosmics, white - background)



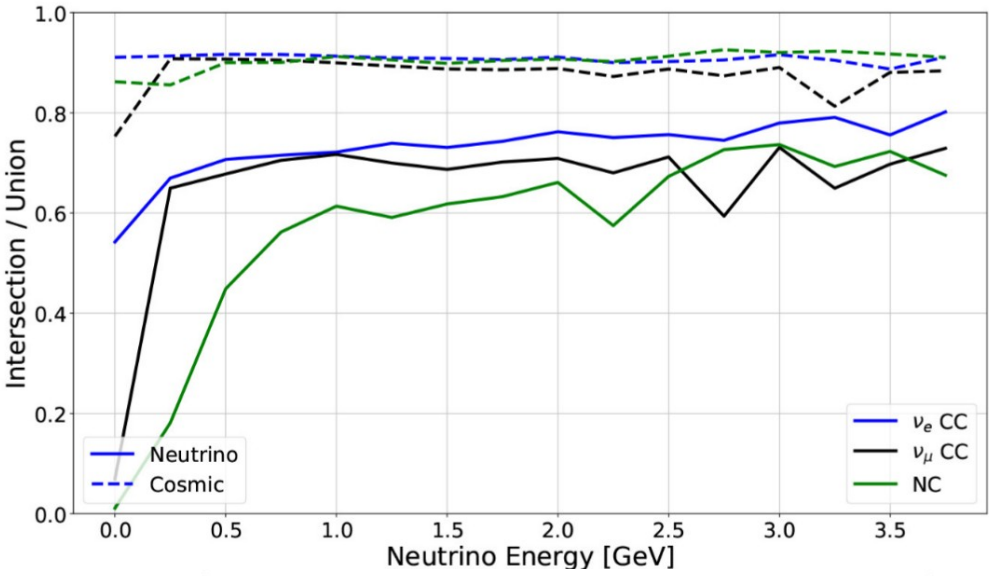
SBND: Cosmogenic background removal

- “Semantic segmentation” - associating parts of the image (pixels) with distinct classes based on UNet architecture
  - Encoder downsamples input using residual convolutional units (sums convolutional/nonlinearity output with input) to produce compressed representation in the bottleneck
  - Decoder upsamples the image back to input resolution
  - Shortcut connections between the two improve localization, mitigate gradient issues
- Addition - per-plane filters are concatenated together into one set of convolutional filters and proceed through convolutions together
  - Learns cross-plane geometrical features
- Note: three planes are used as input, only two are shown on the right for simplicity
- Convolutional weights are shared across all three planes for up-sampling and down-sampling of the network



SBND: Cosmogenic background removal

- IoU metric - overlap between pixels' true and predicted labels
  - Computed as the ratio of intersection (common pixels in both sets) to union (all pixels in either set)
- Hard to directly compare to traditional background rejection algorithms
- But estimated to be an improvement



Category	Efficiency (%)
ν_e CC	91.5
ν_μ CC	78.6
NC	37.3
Cosmics	91.1 cosmic-only event rejection

Finishing remarks

- MLP and CNN are here to stay
  - Remember they are “universal function approximators”, so be creative about inputs (e.g., “image” can be amplitude vs time, recon’d energy vs PMT index, or whatever). They can figure it out
  - Keep in mind the “relational inductive bias” table when picking the best architecture
- More complicated architectures used in generative networks, semantic segmentation, et cetera, often still use these as building blocks, so learn them well
  - Don’t just let PyTorch do all the job for you, invest some effort to understand a bit better what’s going on “under the hood”. It will help avoid silly mistakes
- Critically assess the new fancy architectures before investing time in them. Make sure the improvement is clearly demonstrated

Now, let’s start Tutorial

- Jupyter notebook and some waveform data
- Password: cUaw (please don’t share further)