Development of reconstruction algorithms for the new TPCs of the upgraded T2K near detector with machine learning techniques

Pre-thesis internship – Oral defense

June 28th, 2023

Anaëlle Chalumeau

Overview

1. Introduction

2. T2K, its near detector and the new TPCs

3. Machine learning: neural network for track reconstruction
    - ▻ How it works
    - ▻ Results
    - ▻ Current challenges and investigations

  - ▻ Internship conclusion

4. Prospects for the PhD

1. Introduction

Neutrinos oscillate:

$$\begin{pmatrix} \nu_e \\ \nu_\mu \\ \nu_\tau \end{pmatrix} = \begin{pmatrix} \text{PNMS} \\ \text{matrix} \end{pmatrix} \begin{pmatrix} \nu_1 \\ \nu_2 \\ \nu_3 \end{pmatrix}$$

1. Introduction

Neutrinos oscillate:

$$\begin{pmatrix}\nu_e\\ \nu_\mu\\ \nu_\tau\end{pmatrix}=\begin{pmatrix}1 & 0 & 0\\ 0 & \cos\theta_{23} & \sin\theta_{23}\\ 0 & -\sin\theta_{23} & \cos\theta_{23}\end{pmatrix}\begin{pmatrix}\cos\theta_{13} & 0 & \sin\theta_{13}e^{-i\delta_{CP}}\\ 0 & 1 & 0\\ -\sin\theta_{13}e^{i\delta_{CP}} & 0 & \cos\theta_{13}\end{pmatrix}\begin{pmatrix}\cos\theta_{12} & \sin\theta_{12} & 0\\ -\sin\theta_{12} & \cos\theta_{12} & 0\\ 0 & 0 & 1\end{pmatrix}D\begin{pmatrix}\nu_1\\ \nu_2\\ \nu_3\end{pmatrix}$$

1. Introduction

Neutrinos oscillate:

$$\begin{pmatrix}\nu_e\\ \nu_\mu\\ \nu_\tau\end{pmatrix} = \begin{pmatrix}1 & 0 & 0\\ 0 & \cos\theta_{23} & \sin\theta_{23}\\ 0 & -\sin\theta_{23} & \cos\theta_{23}\end{pmatrix} \begin{pmatrix}\cos\theta_{13} & 0 & \sin\theta_{13}e^{-i\delta_{CP}}\\ 0 & 1 & 0\\ -\sin\theta_{13}e^{i\delta_{CP}} & 0 & \cos\theta_{13}\end{pmatrix} \begin{pmatrix}\cos\theta_{12} & \sin\theta_{12} & 0\\ -\sin\theta_{12} & \cos\theta_{12} & 0\\ 0 & 0 & 1\end{pmatrix} D \begin{pmatrix}\nu_1\\ \nu_2\\ \nu_3\end{pmatrix}$$

Appearance probability

$$P(\nu_\mu \to \nu_e) = |\langle\nu_e|\nu_\mu(t)\rangle|^2 = P(\theta_{12},\, \theta_{23},\, \theta_{13}\,,\, \delta_{CP}\,,\, \Delta m^2_{21} \text{ and } \Delta m^2_{32})$$

1. Introduction

Neutrinos oscillate:

$$\begin{pmatrix}\nu_e\\ \nu_\mu\\ \nu_\tau\end{pmatrix} = \begin{pmatrix}1 & 0 & 0\\ 0 & \cos\theta_{23} & \sin\theta_{23}\\ 0 & -\sin\theta_{23} & \cos\theta_{23}\end{pmatrix} \begin{pmatrix}\cos\theta_{13} & 0 & \sin\theta_{13}e^{-i\delta_{CP}}\\ 0 & 1 & 0\\ -\sin\theta_{13}e^{i\delta_{CP}} & 0 & \cos\theta_{13}\end{pmatrix} \begin{pmatrix}\cos\theta_{12} & \sin\theta_{12} & 0\\ -\sin\theta_{12} & \cos\theta_{12} & 0\\ 0 & 0 & 1\end{pmatrix} D \begin{pmatrix}\nu_1\\ \nu_2\\ \nu_3\end{pmatrix}$$

Appearance probability

$$P(\nu_\mu \to \nu_e) = |\langle\nu_e|\nu_\mu(t)\rangle|^2 = P(\theta_{12},\, \theta_{23},\, \theta_{13}\,,\, \delta_{CP},\, \Delta m^2_{21} \text{ and } \Delta m^2_{32})$$

Disappearance probability

$$P(\nu_\mu \to \nu_\mu) = |\langle\nu_\mu|\nu_\mu(t)\rangle|^2 = P(\theta_{12},\, \theta_{23},\, \theta_{13}\,,\, \delta_{CP},\, \Delta m^2_{21} \text{ and } \Delta m^2_{32})$$

1. Introduction

Neutrinos oscillate:

$$\begin{pmatrix}\nu_e\\ \nu_\mu\\ \nu_\tau\end{pmatrix} = \begin{pmatrix}1 & 0 & 0\\ 0 & \cos\theta_{23} & \sin\theta_{23}\\ 0 & -\sin\theta_{23} & \cos\theta_{23}\end{pmatrix} \begin{pmatrix}\cos\theta_{13} & 0 & \sin\theta_{13}e^{-i\delta_{CP}}\\ 0 & 1 & 0\\ -\sin\theta_{13}e^{i\delta_{CP}} & 0 & \cos\theta_{13}\end{pmatrix} \begin{pmatrix}\cos\theta_{12} & \sin\theta_{12} & 0\\ -\sin\theta_{12} & \cos\theta_{12} & 0\\ 0 & 0 & 1\end{pmatrix} D \begin{pmatrix}\nu_1\\ \nu_2\\ \nu_3\end{pmatrix}$$

Appearance probability

$$P(\nu_\mu \to \nu_e) = |\langle\nu_e|\nu_\mu(t)\rangle|^2 = P\left(\boxed{\theta_{12},\ \theta_{23},\ \theta_{13},\ \delta_{CP},\ \Delta m^2_{21}\ \text{and}\ \Delta m^2_{32}}\right)$$

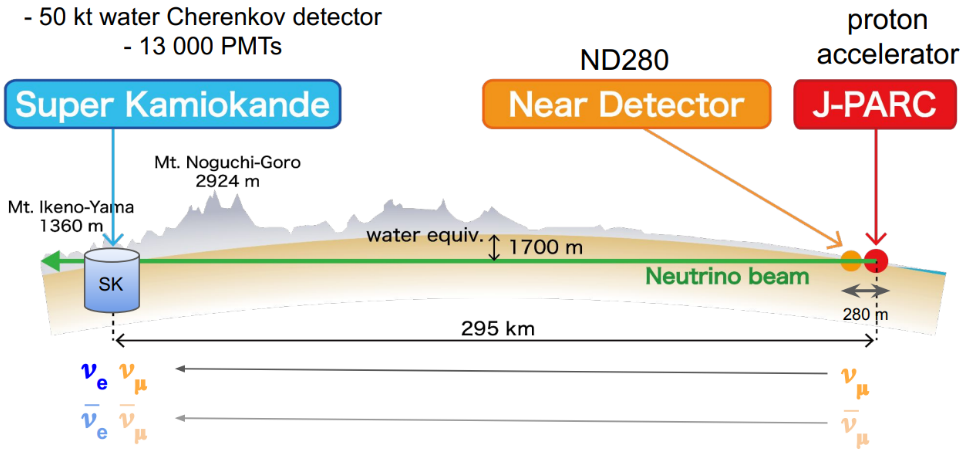
Disappearance probability

$$P(\nu_\mu \to \nu_\mu) = |\langle\nu_\mu|\nu_\mu(t)\rangle|^2 = P\left(\boxed{\theta_{12},\ \theta_{23},\ \theta_{13},\ \delta_{CP},\ \Delta m^2_{21}\ \text{and}\ \Delta m^2_{32}}\right)$$

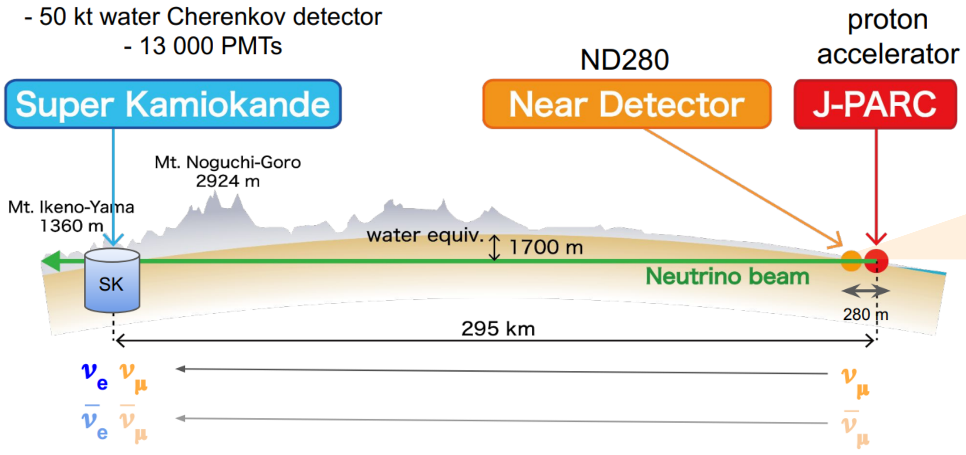
oscillation parameters!

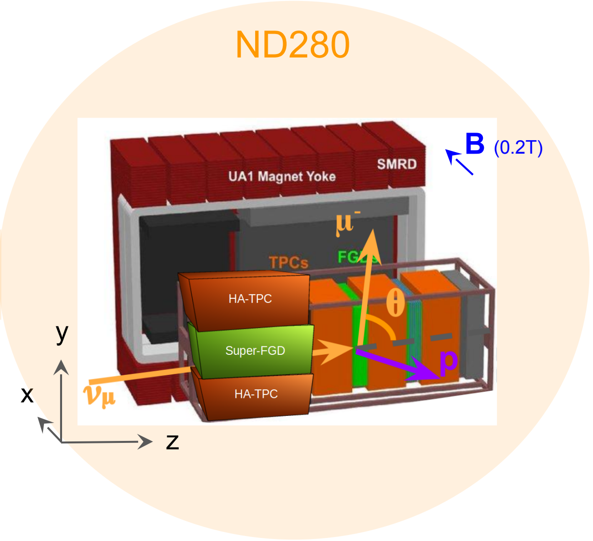
want to know them **precisely**

2. T2K, its near detector and the new TPCs

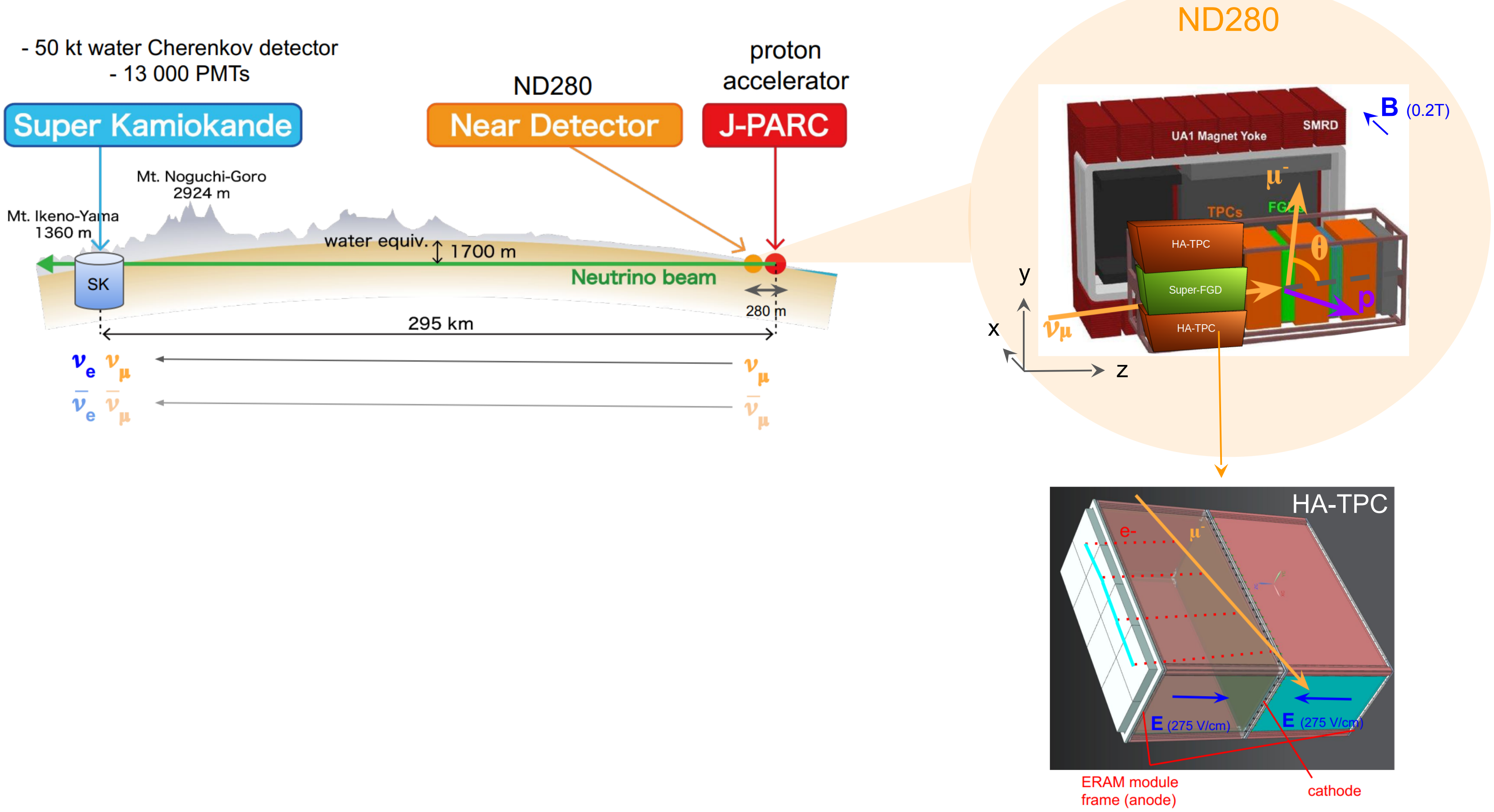


2. T2K, its near detector and the new TPCs





2. T2K, its near detector and the new TPCs



- 50 kt water Cherenkov detector
- 13 000 PMTs

Super Kamiokande

ND280

Near Detector

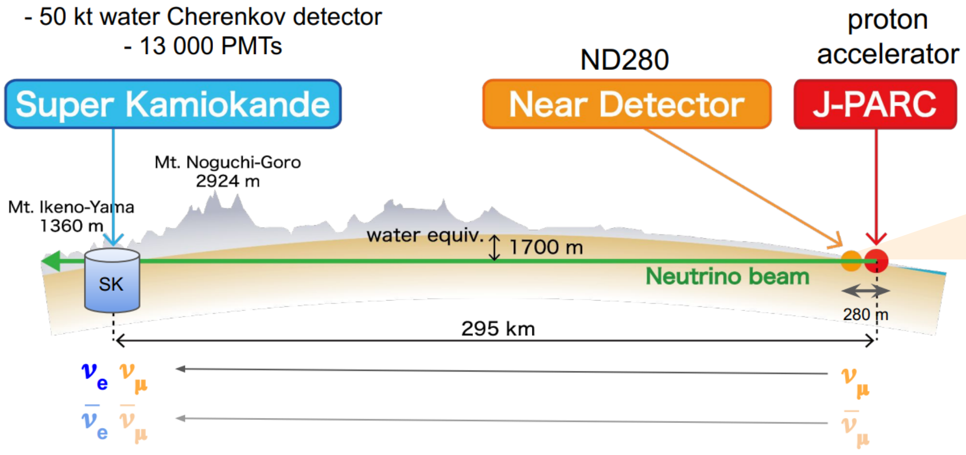
proton accelerator

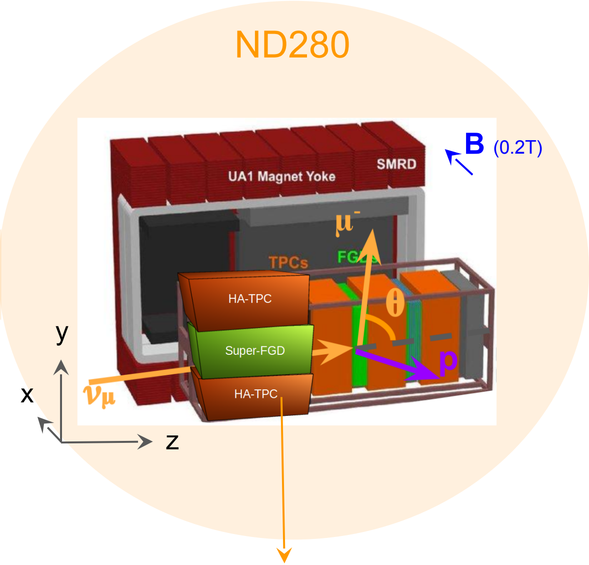
J-PARC

ERAM module frame (anode)

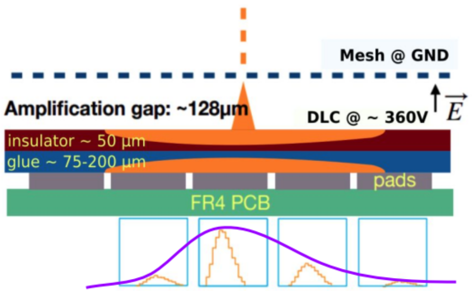
cathode

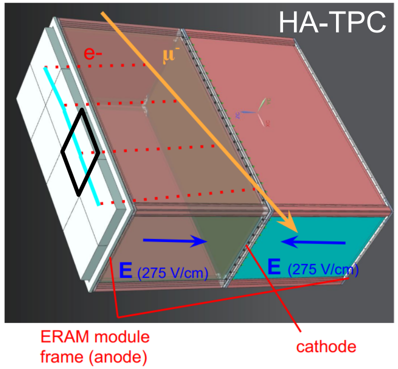
2. T2K, its near detector and the new TPCs



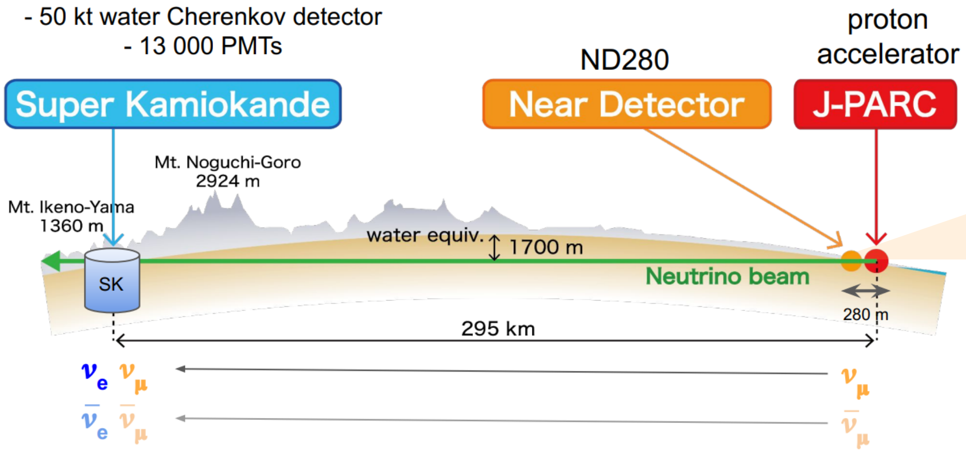


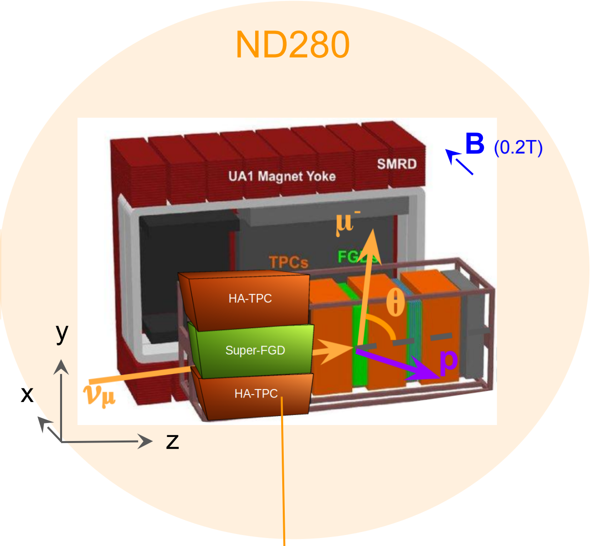
Encapsulated Resistive Anode MicroMegas (ERAM)

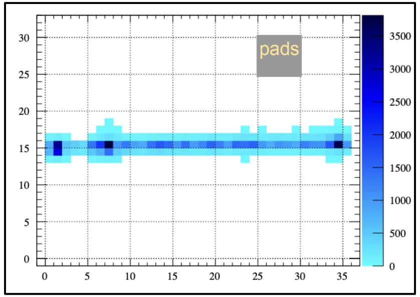




2. T2K, its near detector and the new TPCs

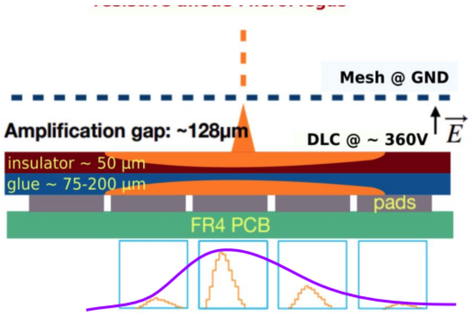


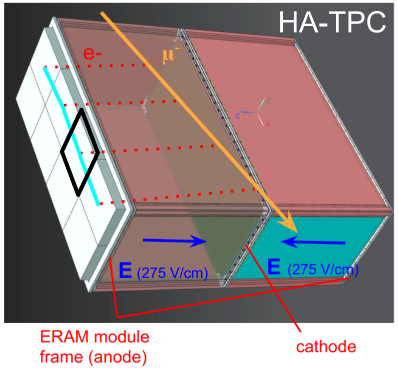




e.g of an event display from a test beam at DESY

Encapsulated Resistive Anode MicroMegas (ERAM)





2. T2K, its near detector and the new TPCs

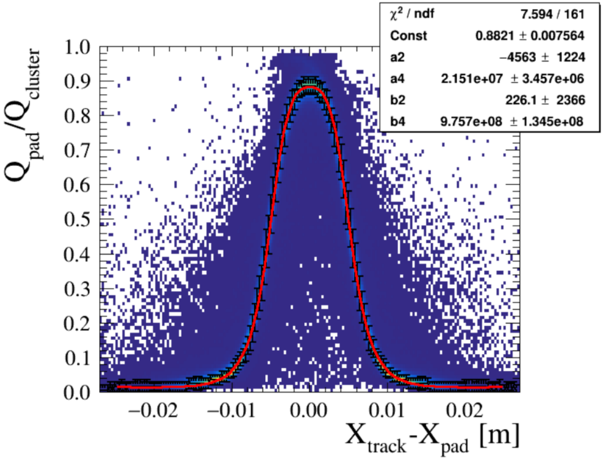
Classical reconstruction in the TPCs

the Pad Response Function method:

2. T2K, its near detector and the new TPCs

Classical reconstruction in the TPCs

the Pad Response Function method:



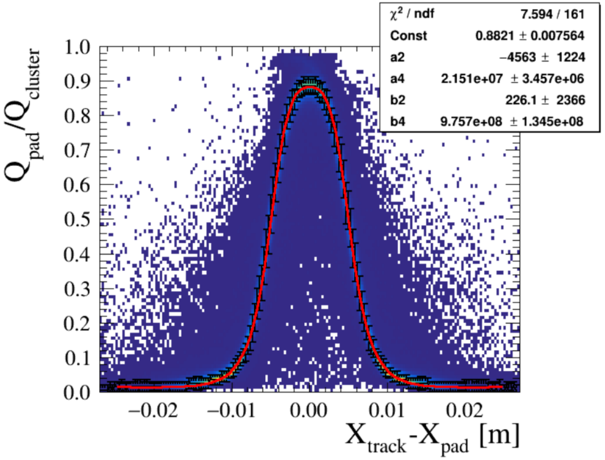
$$PRF(x_{track} - x_{pad}) = \underbrace{Q_{pad}}/Q_{cluster}$$

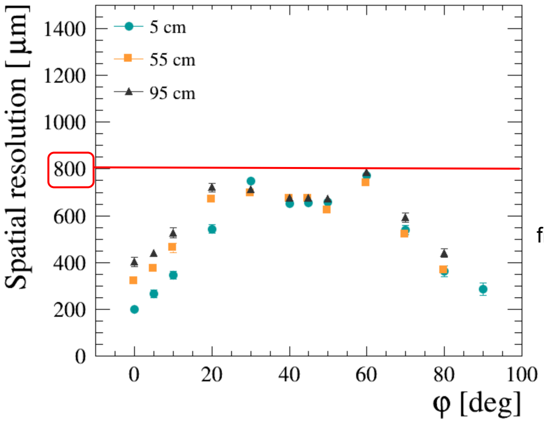
max of the charge in the pad (later called Qmax)

2. T2K, its near detector and the new TPCs

Classical reconstruction in the TPCs

the Pad Response Function method:





from DESY test beam data with a HA-TPC prototype (2022)

$$PRF(x_{track} - x_{pad}) = \underbrace{Q_{pad}}/Q_{cluster}$$

max of the charge in the pad (later called Qmax)

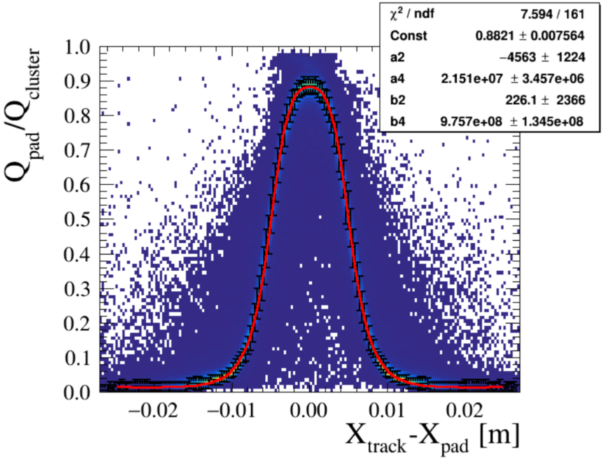
↳ gives spatial resolution <0.8mm

↳ but uses only Qmax of the waveform: limited info!

2. T2K, its near detector and the new TPCs

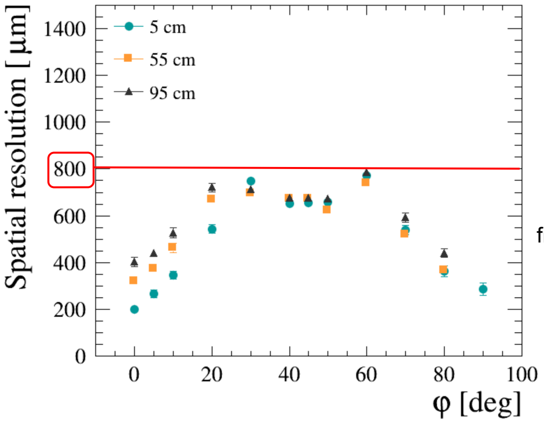
Classical reconstruction in the TPCs

the Pad Response Function method:



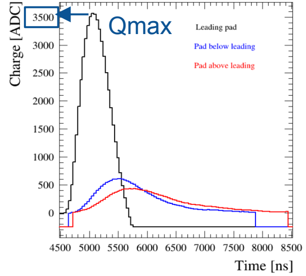
$$PRF(x_{track} - x_{pad}) = \underbrace{Q_{pad}}/Q_{cluster}$$

max of the charge in the pad (later called Qmax)



from DESY test beam data with a HA-TPC prototype (2022)

Waveforms
(charge deposited vs time)



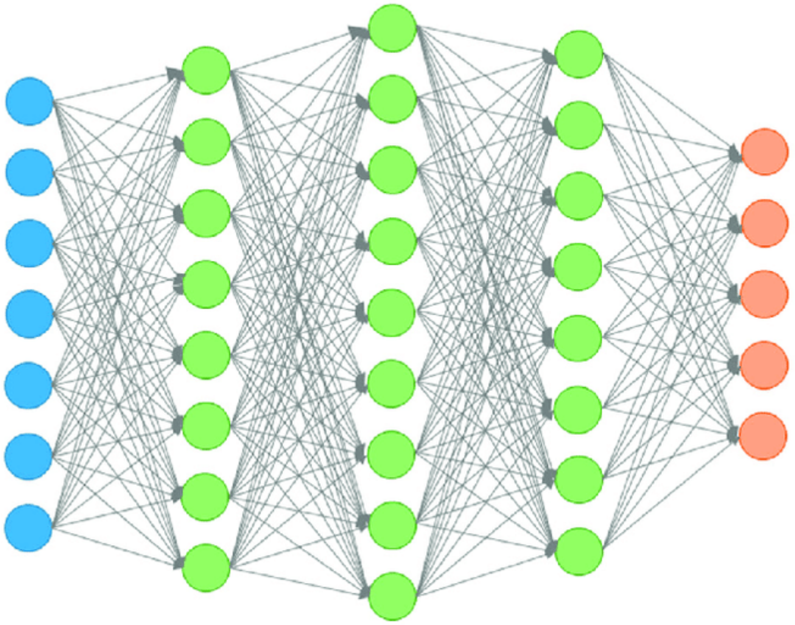
↳ gives spatial resolution <0.8mm

↳ but uses only Qmax of the waveform: limited info!

↳ **could expect better resolutions by using more info from the waveforms**

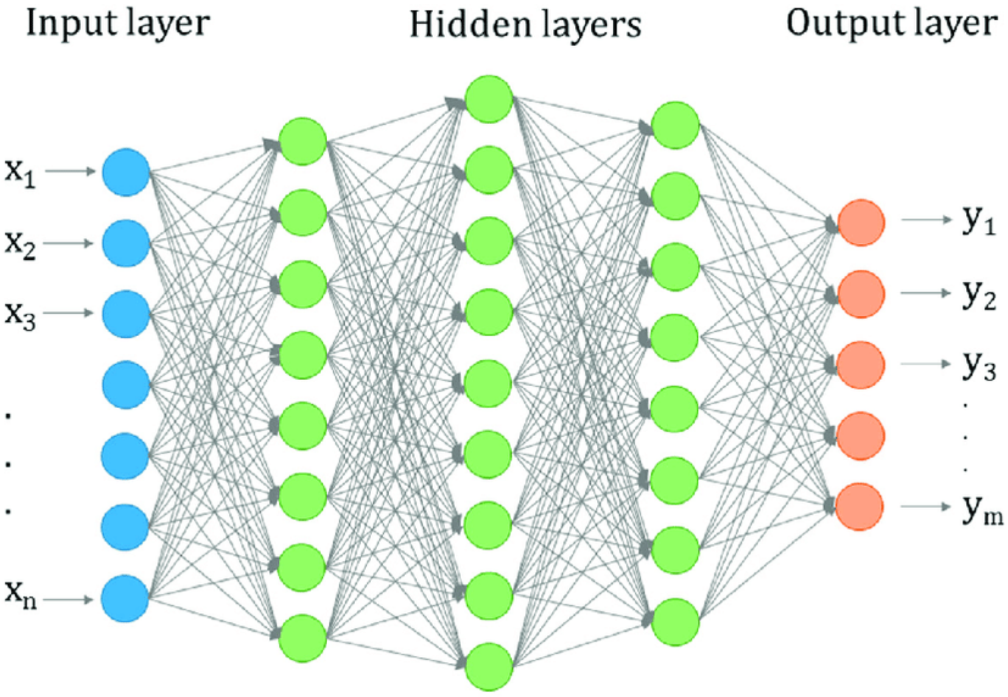
3. Machine learning and deep learning

- ☑ Efficient for large & complexe datasets
- ☑ Power to model non-trivial relationships between inputs/outputs
- ☑ Easily adaptable to various experimental conditions
- ☑ Promising results these past years in particle & neutrino physics
- ☑ Neural networks: very good for image processing



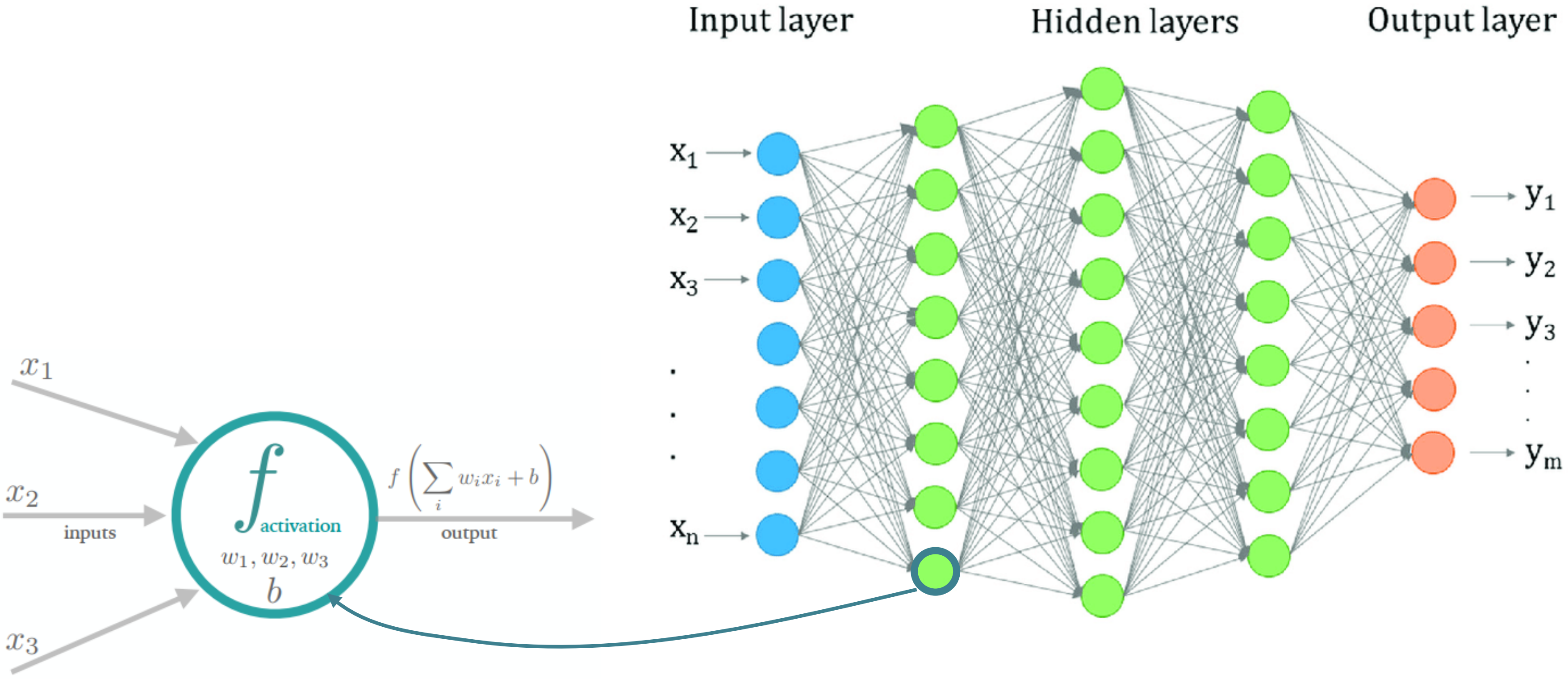
3. Machine learning: neural network for track reconstruction

▻ How it works



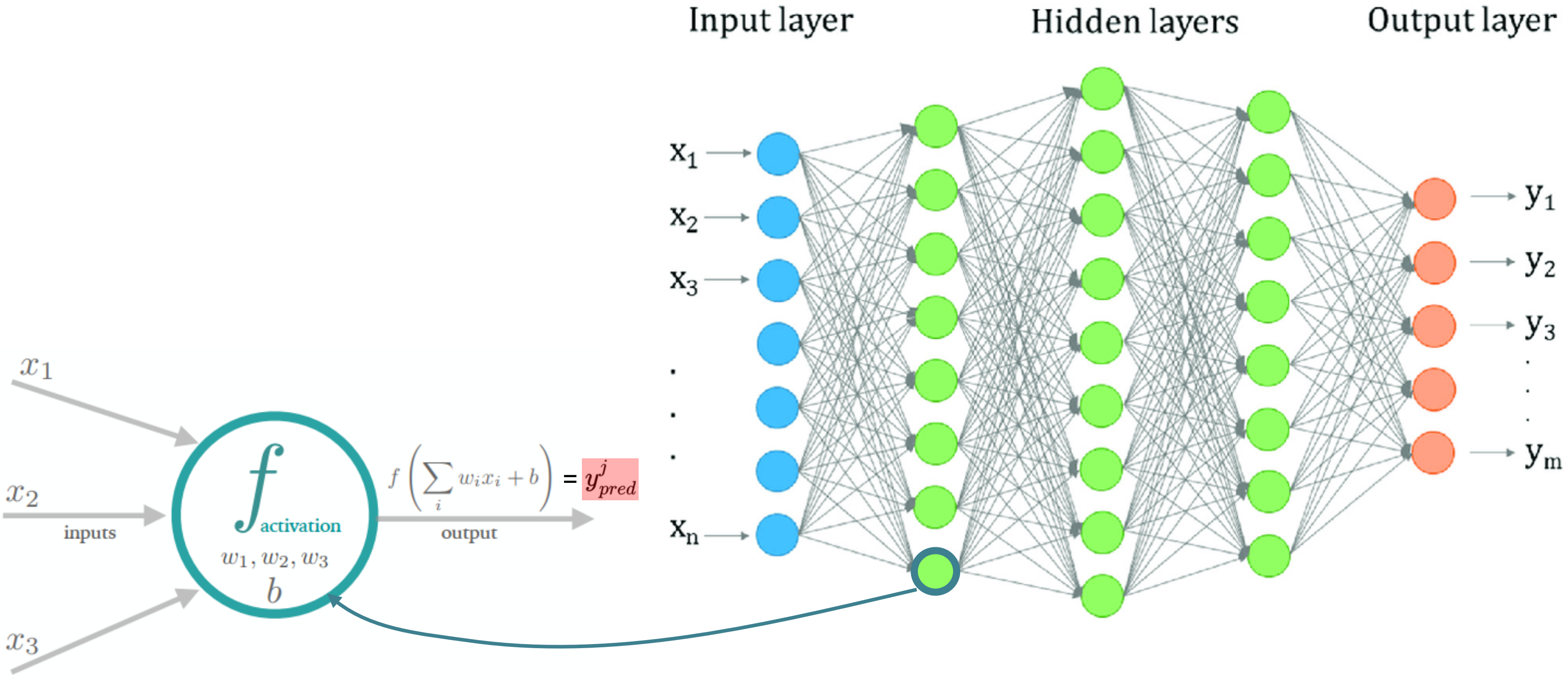
3. Machine learning: neural network for track reconstruction

▻ How it works



3. Machine learning: neural network for track reconstruction

▻ How it works



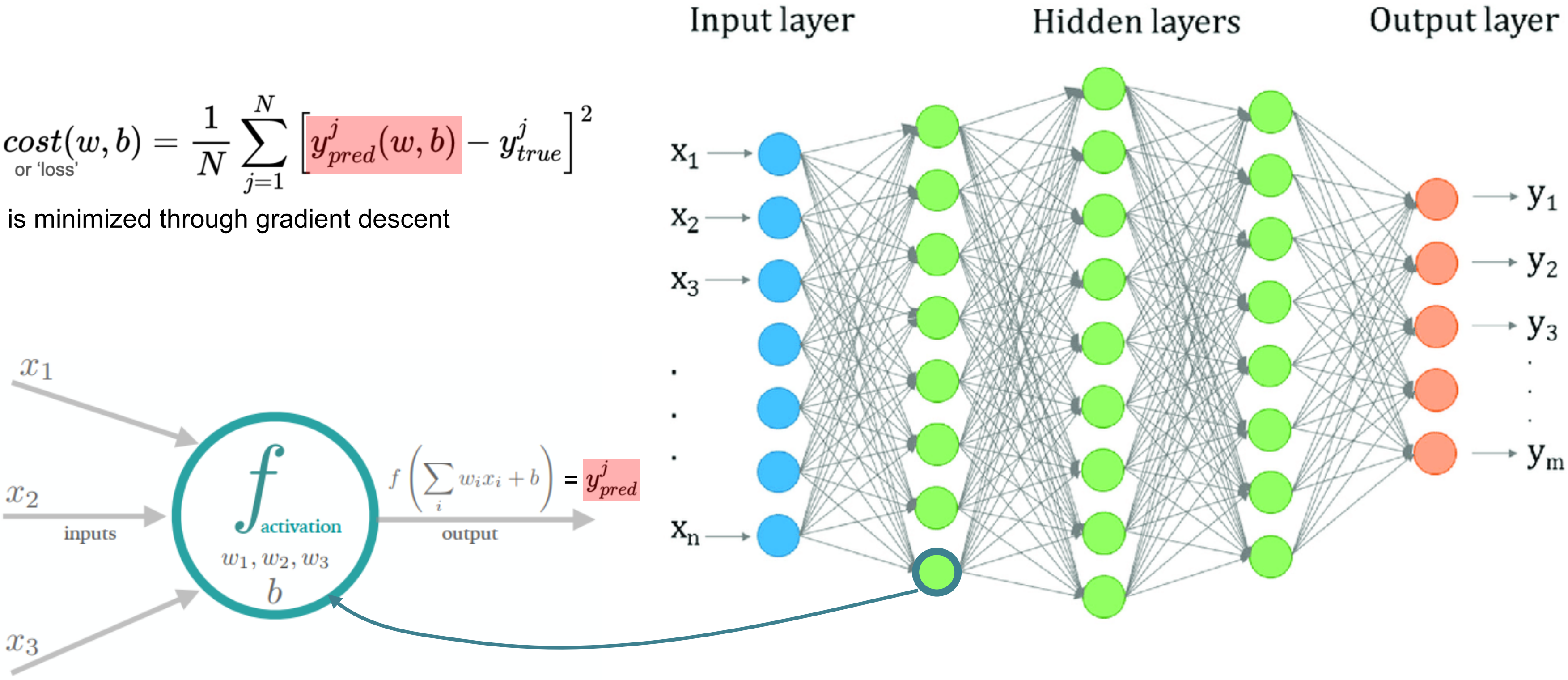
3. Machine learning: neural network for track reconstruction

▻ How it works

$$cost(w,b) = \frac{1}{N}\sum_{j=1}^{N}\left[y^{j}_{pred}(w,b) - y^{j}_{true}\right]^2$$

or 'loss'

is minimized through gradient descent



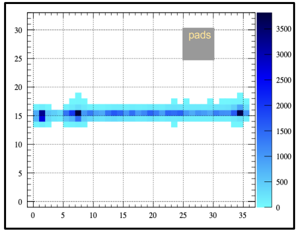
$$f\left(\sum_i w_i x_i + b\right) = y^{j}_{pred}$$

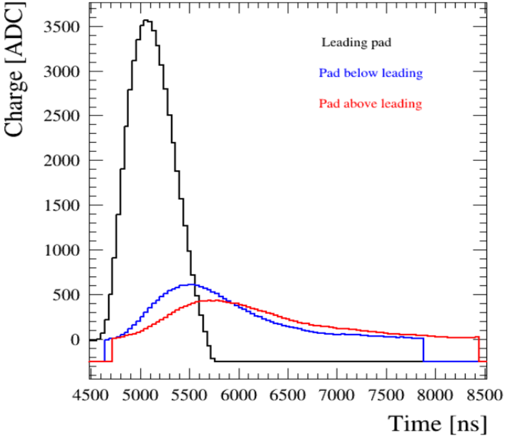
3. Machine learning: neural network for track reconstruction

▻ How it works

Input data

Real data



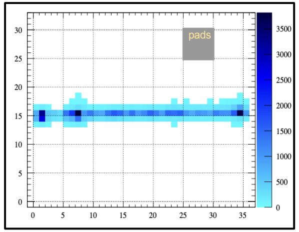


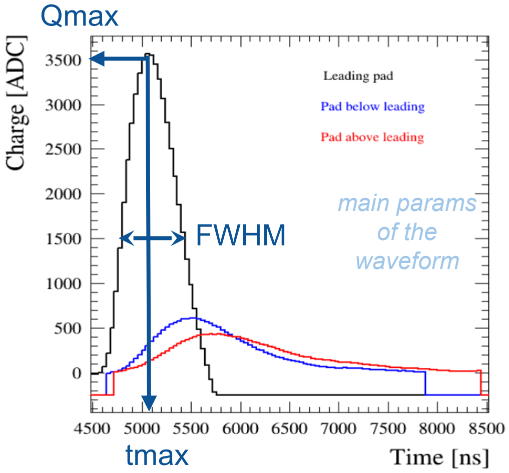
3. Machine learning: neural network for track reconstruction

▻ How it works

Input data

Real data





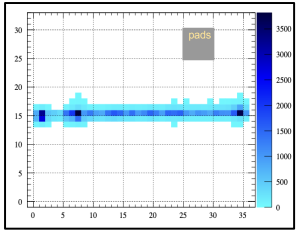
main params of the waveform

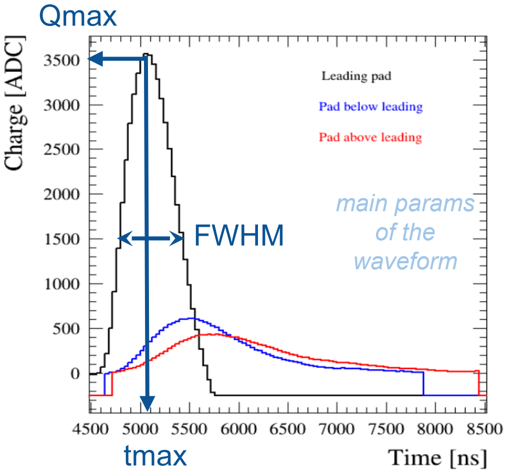
3. Machine learning: neural network for track reconstruction

▻ How it works

Input data

Real data





main params of the waveform

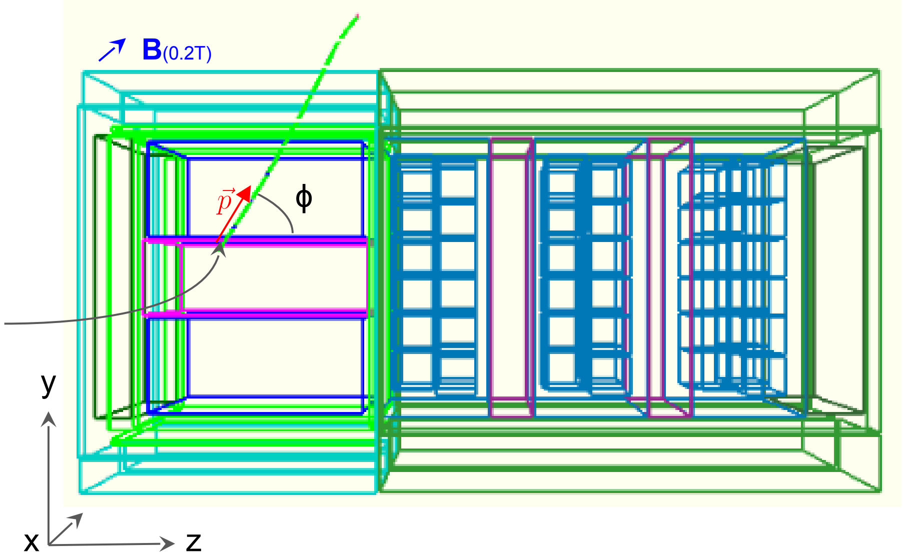
Simulation data ('particle gun')

Simulated data with:

- $p_x = 0$
- $x = 0$

↳ track in (y,z) plane

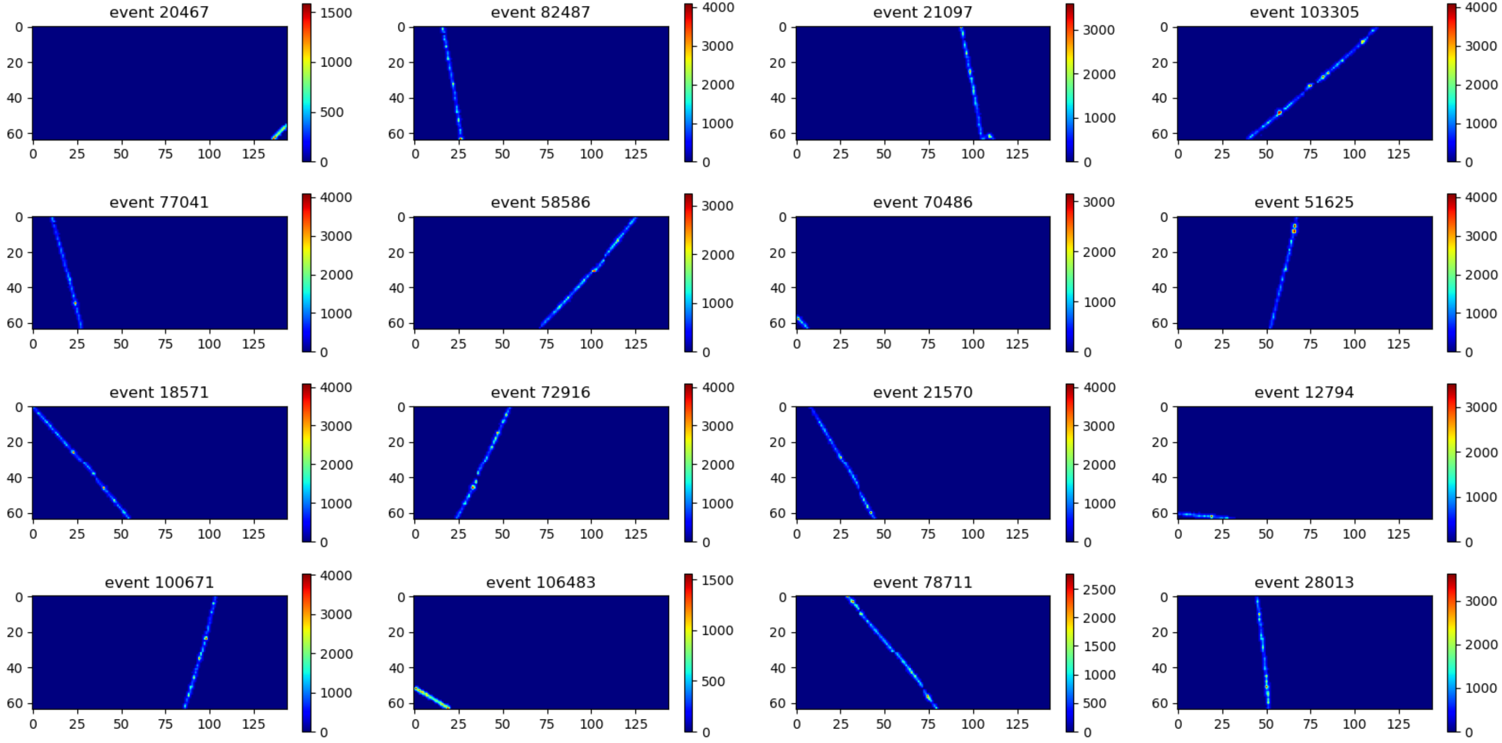
z_{ini} = track entrance in the top HA-TPC



3. Machine learning: neural network for track reconstruction

▻ How it works

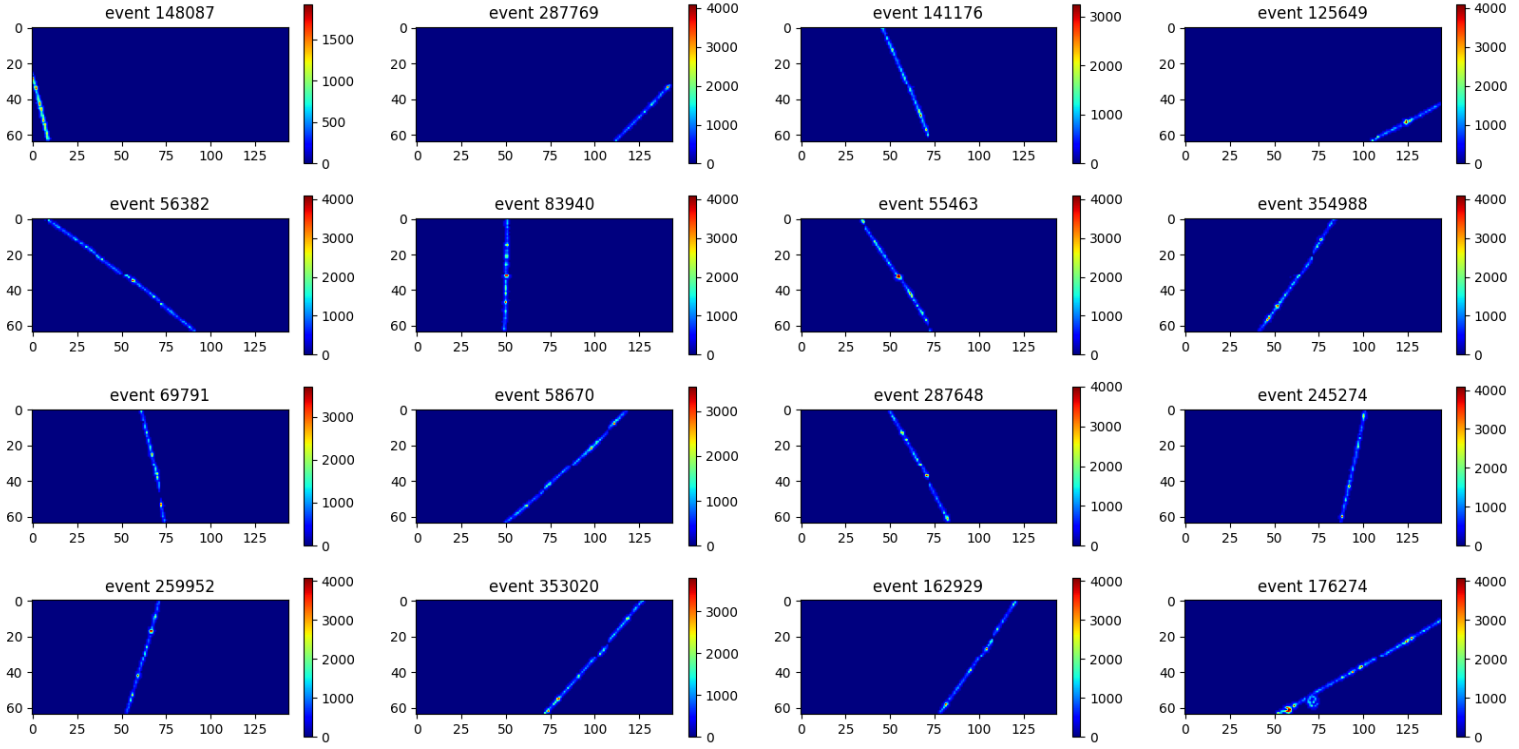
Input data



3. Machine learning: neural network for track reconstruction

▻ How it works

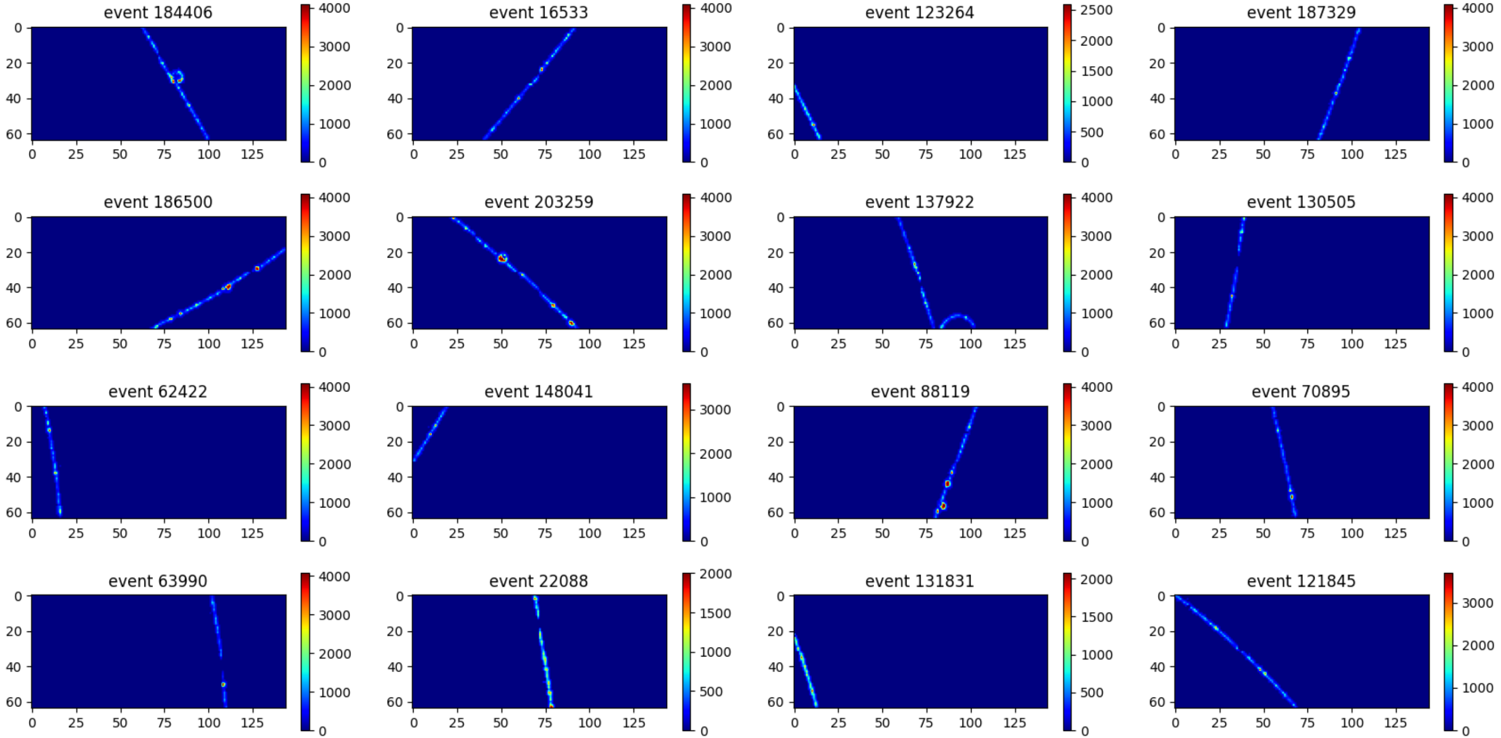
Input data



3. Machine learning: neural network for track reconstruction

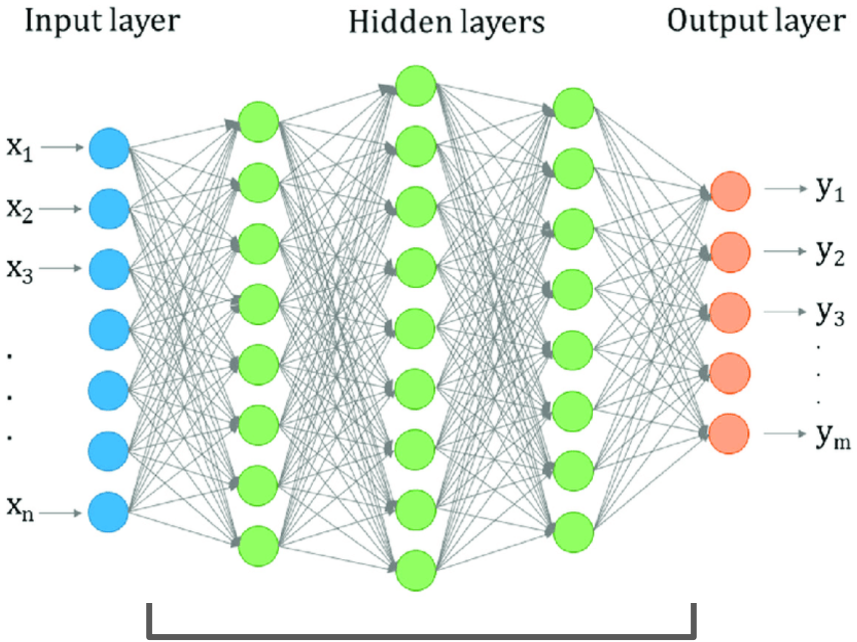
▻ How it works

Input data



3. Machine learning: neural network for track reconstruction

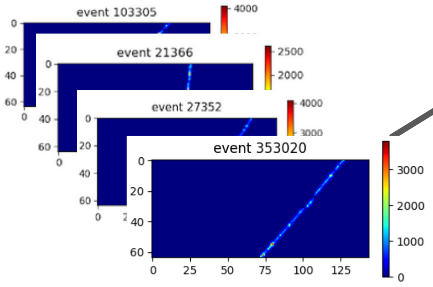
▻ How it works



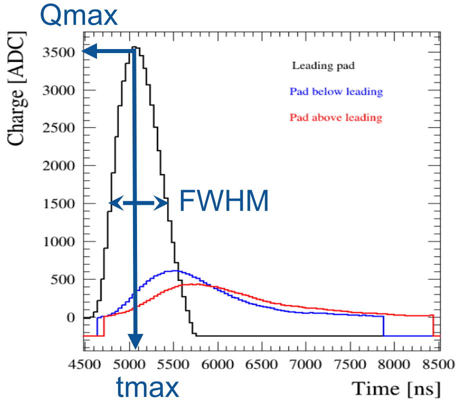
ResNet50 architecture
a convolutional NN

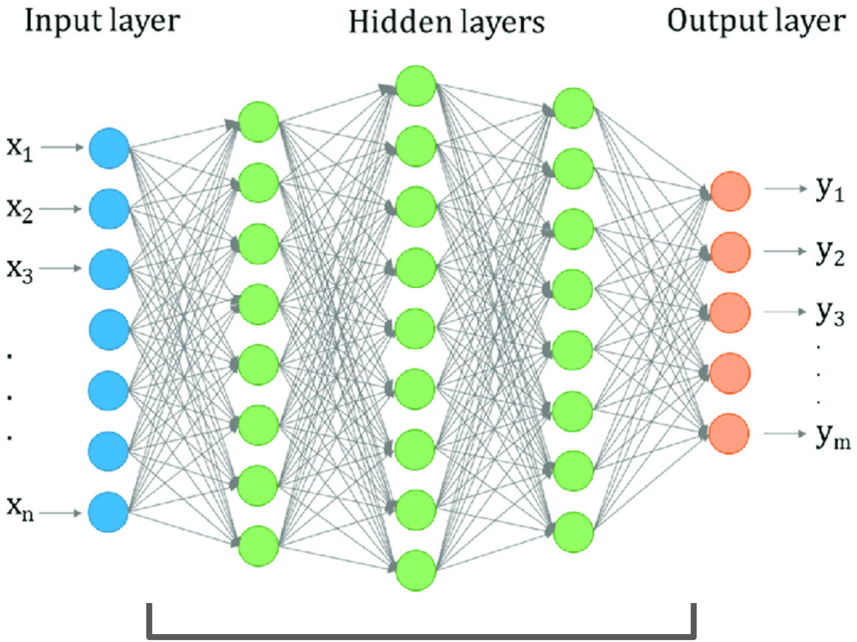
3. Machine learning: neural network for track reconstruction

▻ How it works



only Qmax values at first

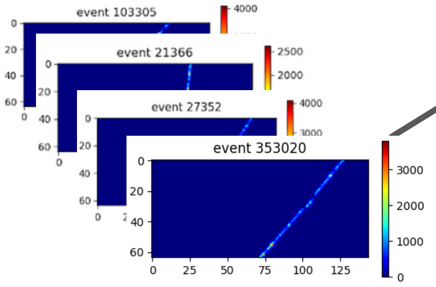




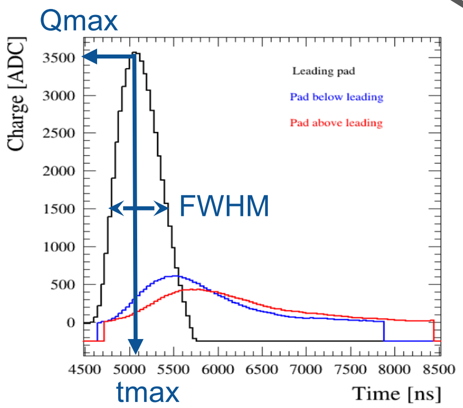
ResNet50 architecture
a convolutional NN

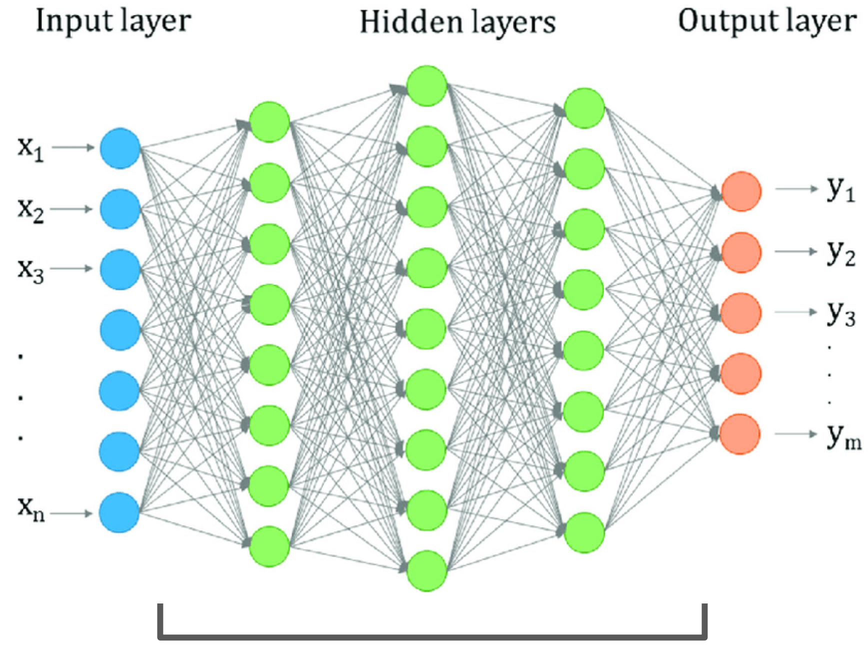
3. Machine learning: neural network for track reconstruction

▻ How it works



only Qmax values at first





[initial positions z_{ini}]

[momenta p_z, p_y, $||\vec{p}||$]

[track angles ϕ]

.
.
.
etc

ResNet50 architecture

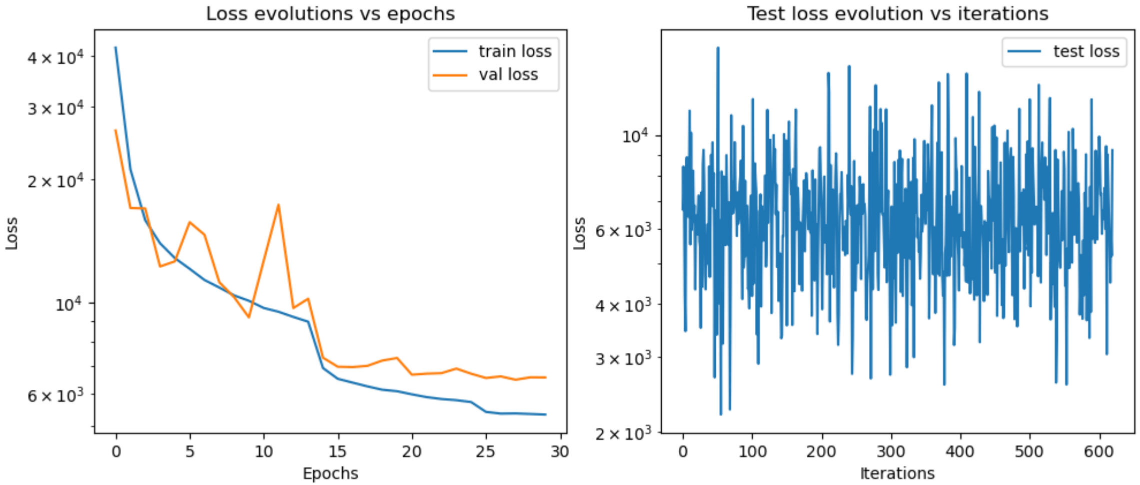
a convolutional NN

3. Machine learning: neural network for track reconstruction

▻ Results

3 predictions: p_y, p_z, p_t
~280 000 events

- Data divided into training/validation/test set (70/15/15%)

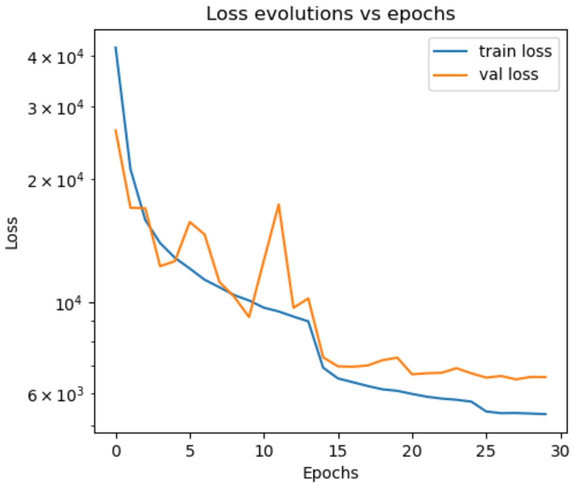


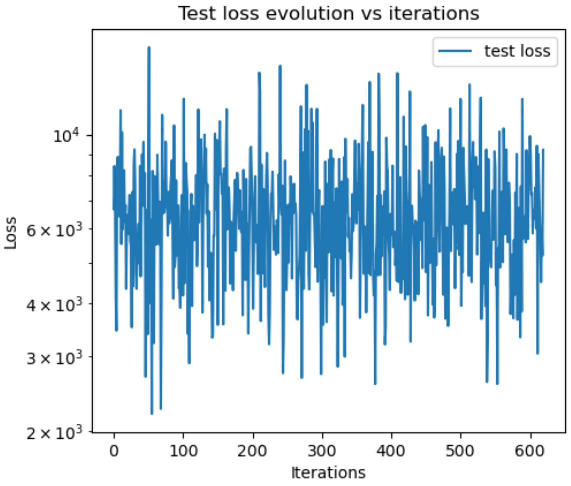
3. Machine learning: neural network for track reconstruction

▻ Results

3 predictions: p_y, p_z, p_t
~280 000 events

- Data divided into training/validation/test set (70/15/15%)





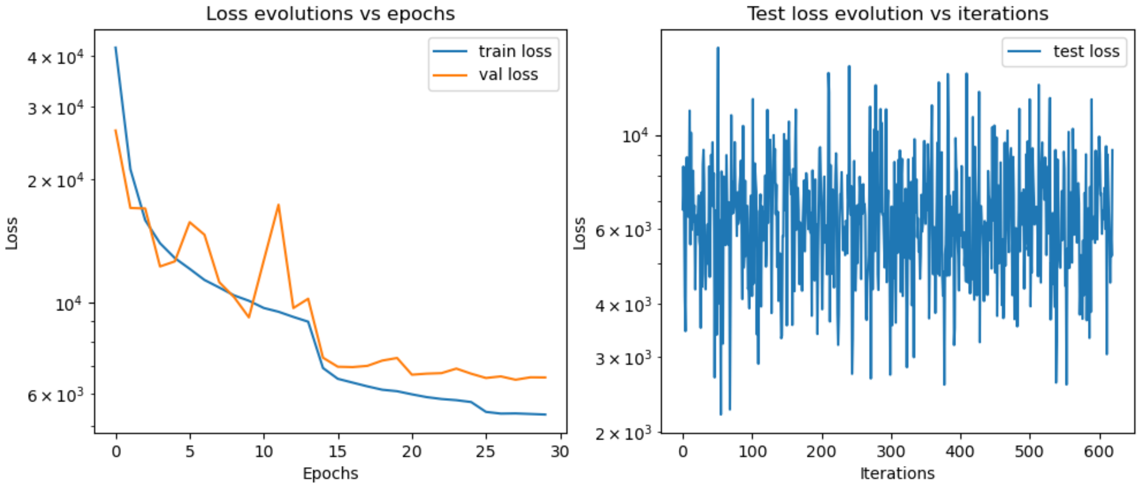
- **Training** set to perform gradient descent and find best parameters (weights & biases)

3. Machine learning: neural network for track reconstruction

▻ Results

3 predictions: p_y, p_z, p_t
~280 000 events

- Data divided into training/validation/test set (70/15/15%)



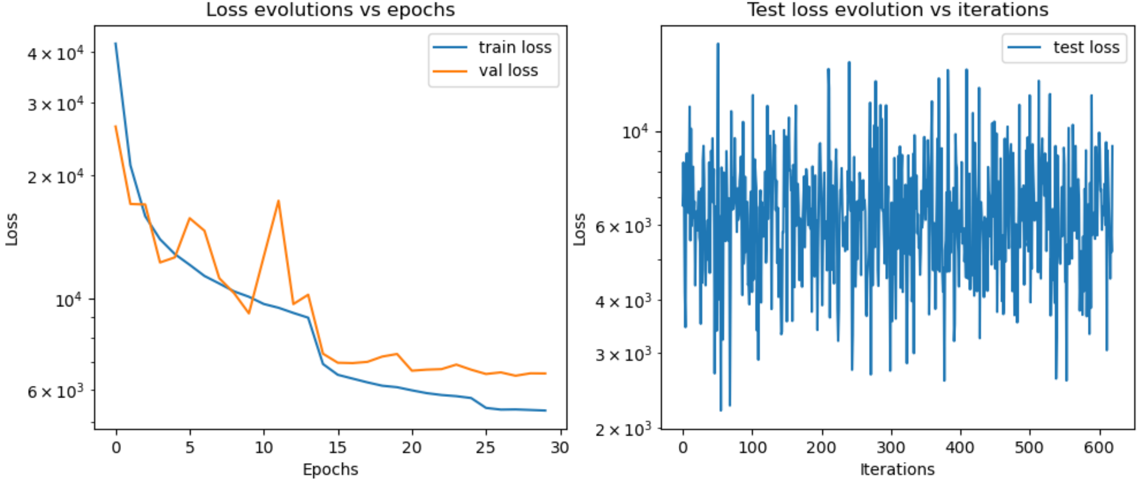
- **Training** set to perform gradient descent and find best parameters (weights & biases)
- **Validation** set to test during training & used to select best parameters

3. Machine learning: neural network for track reconstruction

▻ Results

3 predictions: p_y, p_z, p_t
~280 000 events

- Data divided into training/validation/test set (70/15/15%)



- **Training** set to perform gradient descent and find best parameters (weights & biases)
- **Validation** set to test during training & used to select best parameters
- **Test** set used at the end, after training/validation, to attest the final performance of the model on new/unseen data

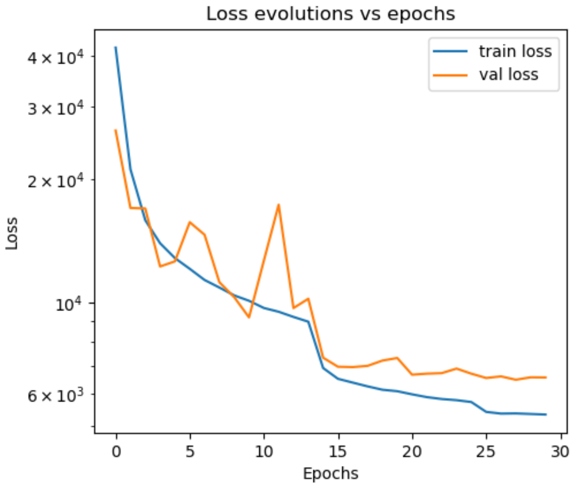
3. Machine learning: neural network for track reconstruction

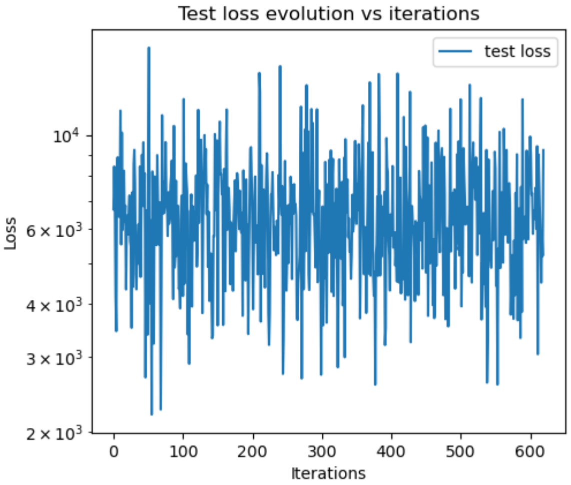
▻ Results

3 predictions: p_y, p_z, p_t
~280 000 events

- ▫ Data divided into training/validation/test set (70/15/15%)

☑ decreases
☑ train/val loss stay close





- ▫ **Training** set to perform gradient descent and find best parameters (weights & biases)
- ▫ **Validation** set to test during training & used to select best parameters
- ▫ **Test** set used at the end, after training/validation, to attest the final performance of the model on new/unseen data

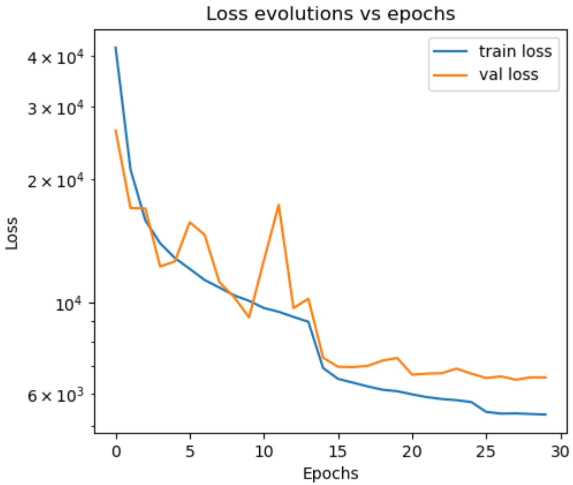
3. Machine learning: neural network for track reconstruction

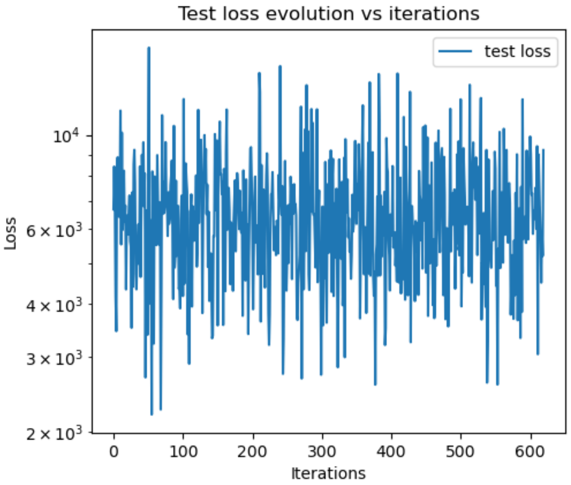
▻ Results

3 predictions: p_y, p_z, p_t
~280 000 events

▫ Data divided into training/validation/test set (70/15/15%)

☑ decreases
☑ train/val loss stay close





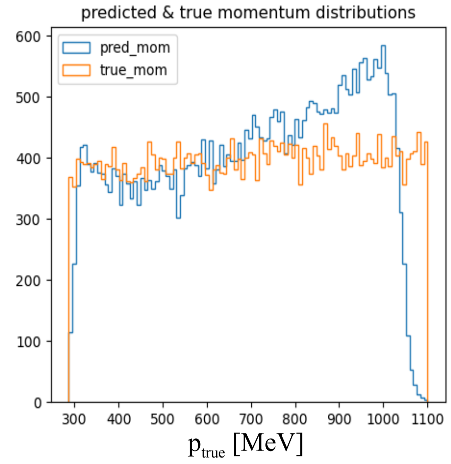
☑ always ~ same OM as final train/val loss

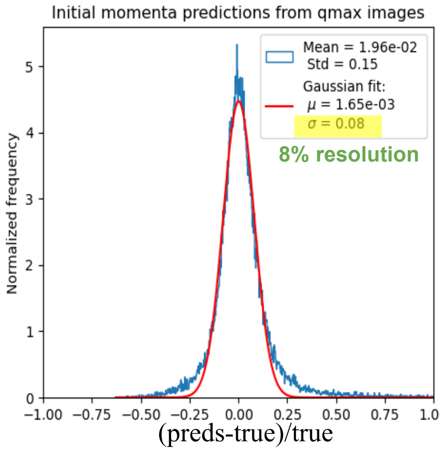
▫ **Training** set to perform gradient descent and find best parameters (weights & biases)
▫ **Validation** set to test during training & used to select best parameters
▫ **Test** set used at the end, after training/validation, to attest the final performance of the model on new/unseen data

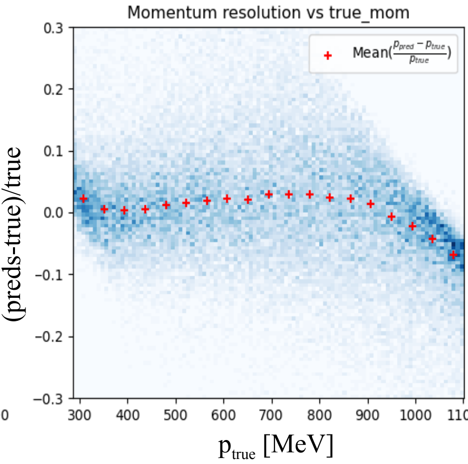
3. Machine learning: neural network for track reconstruction

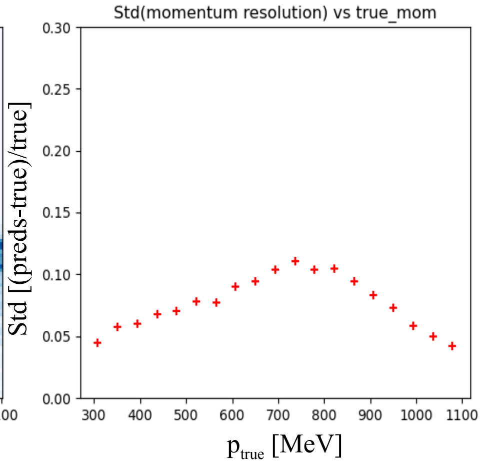
▻ Results

1 prediction: $p_t = \sqrt{p_y^2 + p_z^2}$
~280 000 events


predicted & true momentum distributions


Initial momenta predictions from qmax images

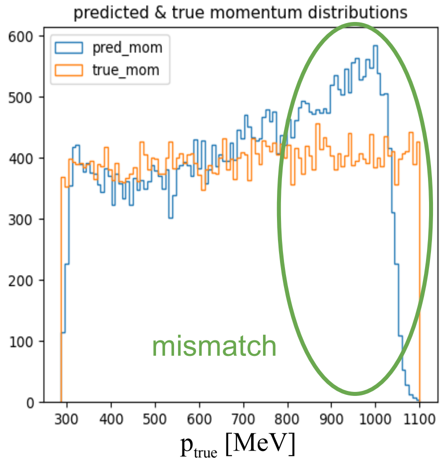

Momentum resolution vs true_mom


Std(momentum resolution) vs true_mom

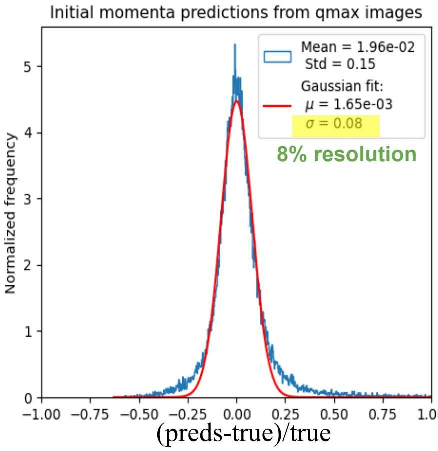
3. Machine learning: neural network for track reconstruction

▻ Results

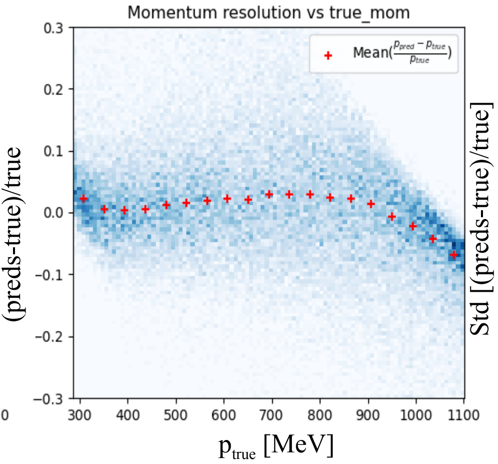
1 prediction: $p_t = \sqrt{p_y^2 + p_z^2}$
~280 000 events

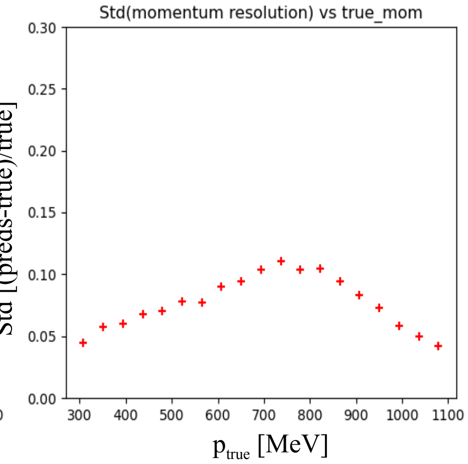


mismatch



8% resolution

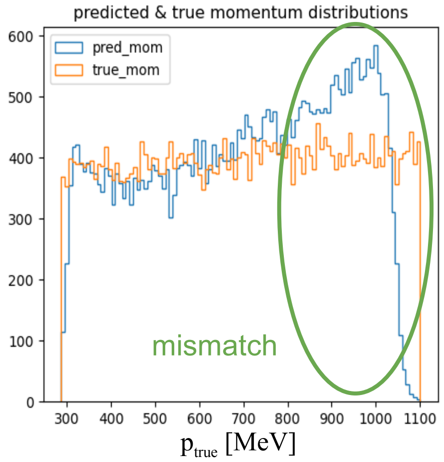




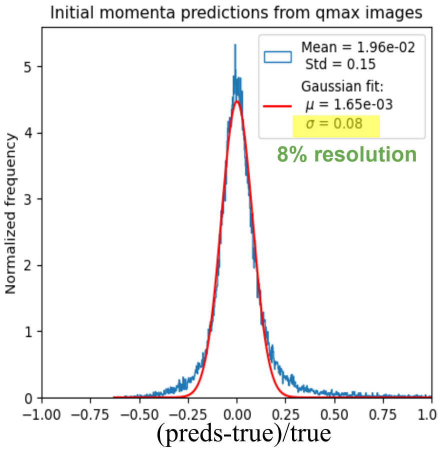
3. Machine learning: neural network for track reconstruction

▻ Results

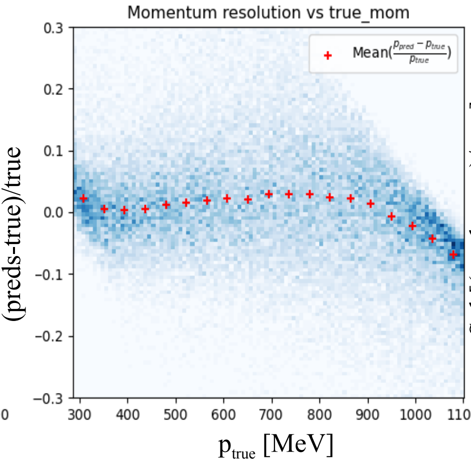
1 prediction: $p_t = \sqrt{p_y^2 + p_z^2}$
~280 000 events

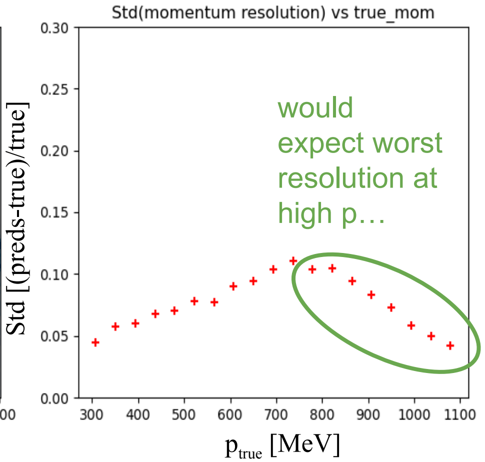


mismatch



8% resolution





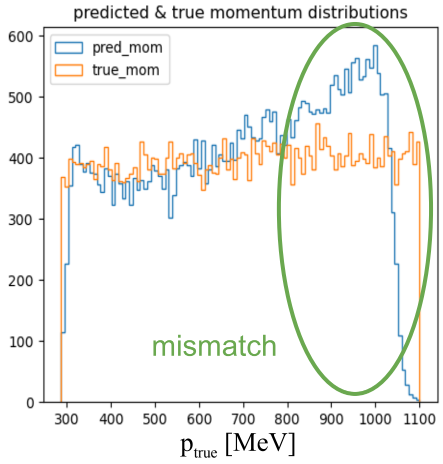
would expect worst resolution at high p…

3. Machine learning: neural network for track reconstruction

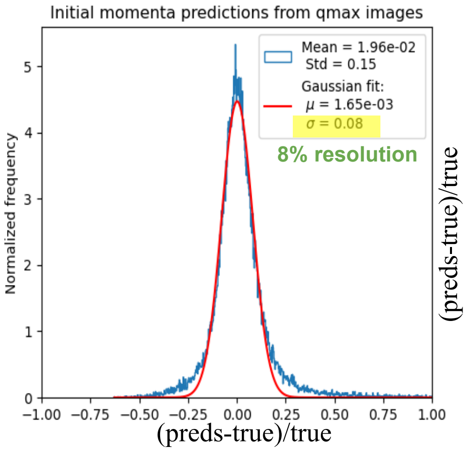
▻ Results

1 prediction: $p_t = \sqrt{p_y^2 + p_z^2}$

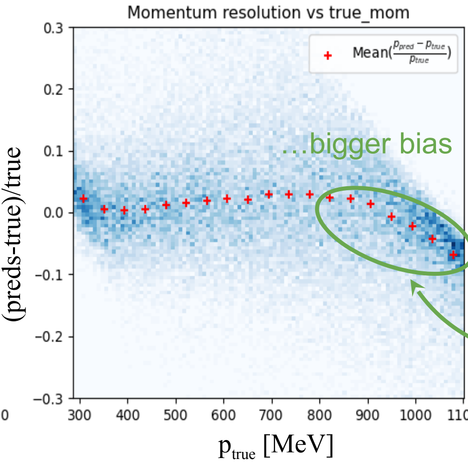
~280 000 events



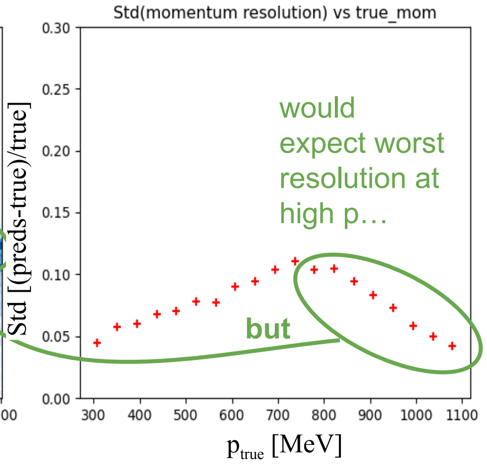
mismatch



8% resolution



…bigger bias



would expect worst resolution at high p…

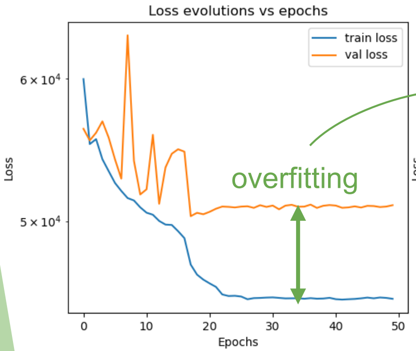
but

3. Machine learning: neural network for track reconstruction

▻ Results

1 prediction: $p_t = \sqrt{p_y^2 + p_z^2}$

~75 000 events



usually because:

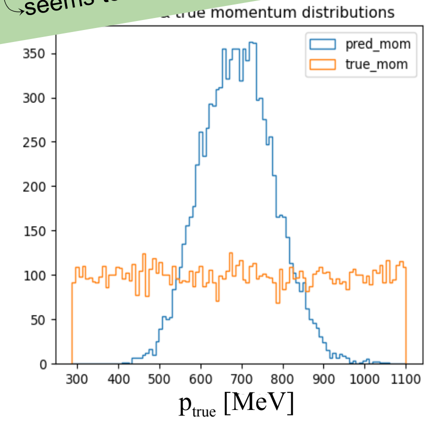
→ not enough data

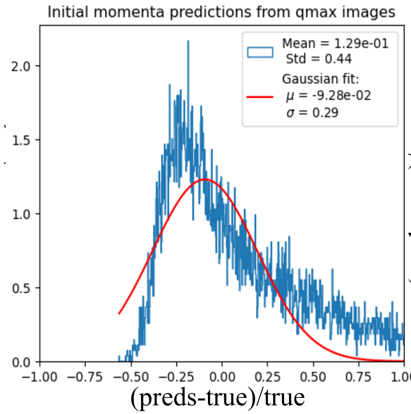
→ complexity of data

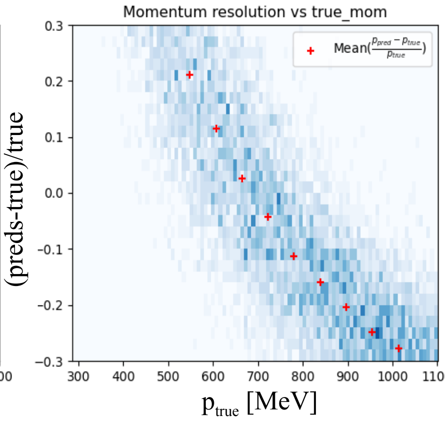
(generalisation is hard)

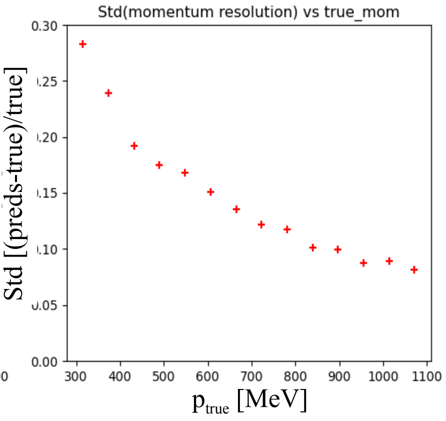
weird behaviour with **3x less events**…

↳seems to **disappear with more data**





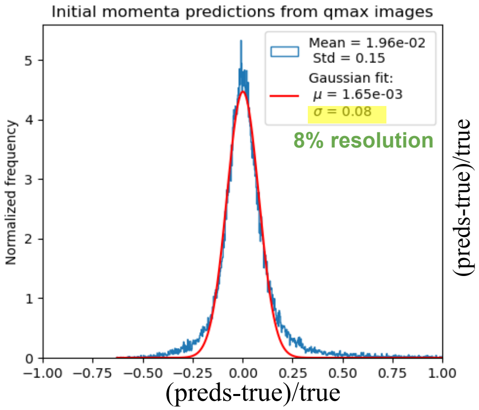




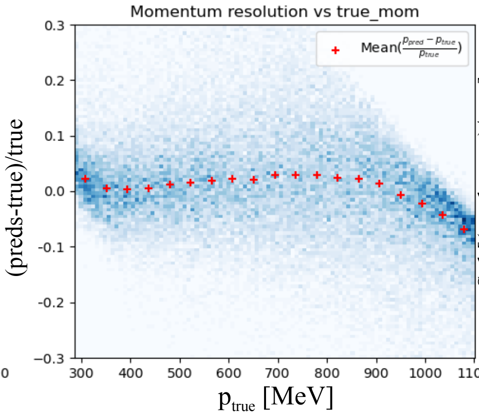
3. Machine learning: neural network for track reconstruction

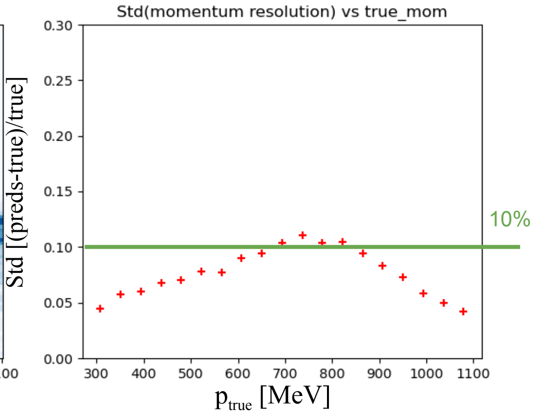
▻ Results

p_t



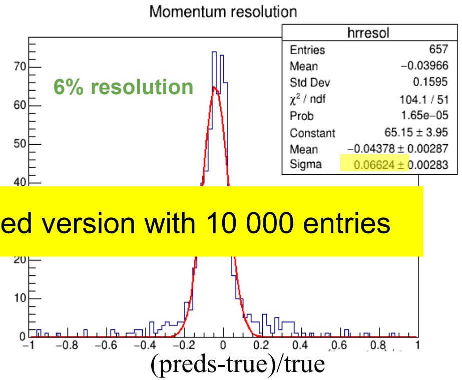
8% resolution





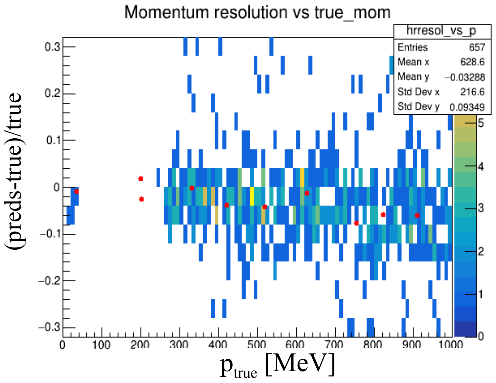
10%

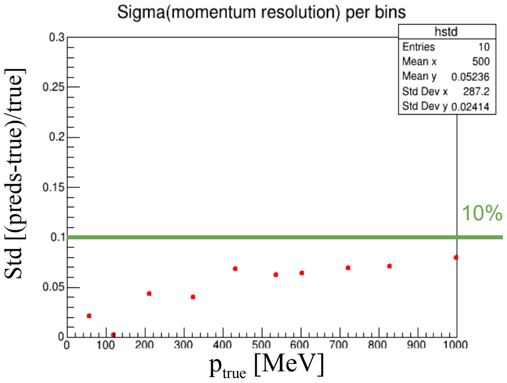
CNN
machine
learning



6% resolution

put updated version with 10 000 entries





10%

‘hatRecon’
classical
reconstruction

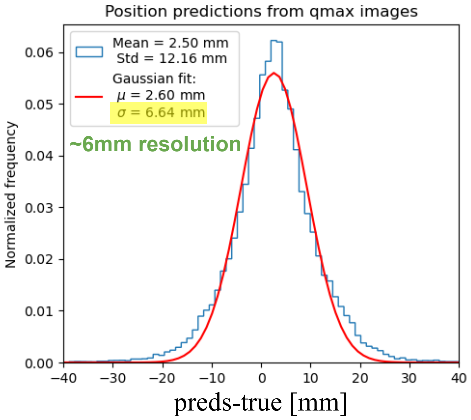
3. Machine learning: neural network for track reconstruction

▻ Results

5 predictions: z_{ini}, p_y, p_z, p_t, ϕ

~280 000 events

z_{ini}

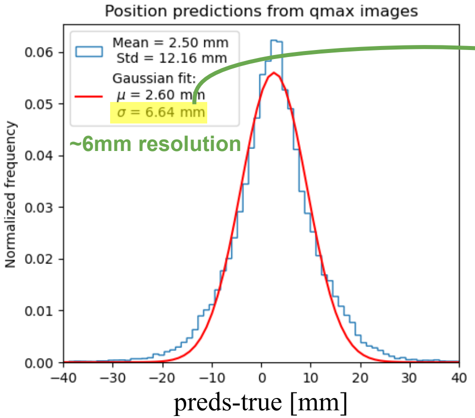


3. Machine learning: neural network for track reconstruction

▻ Results

5 predictions: z_{ini}, p_y, p_z, p_t, ϕ
~280 000 events

z_{ini}



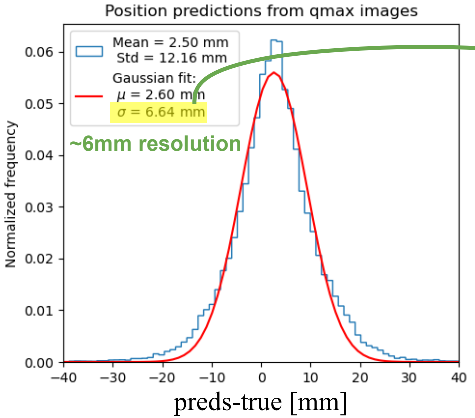
not precise!
1 pad is 10x11mm

3. Machine learning: neural network for track reconstruction

▻ Results

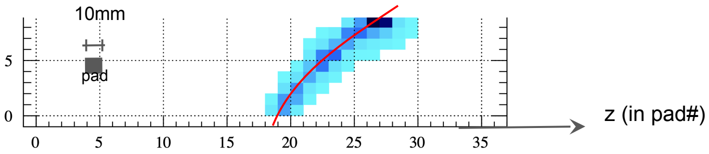
5 predictions: z_{ini}, p_y, p_z, p_t, ϕ
~280 000 events

z_{ini}



~6mm resolution

not precise!
1 pad is 10x11mm

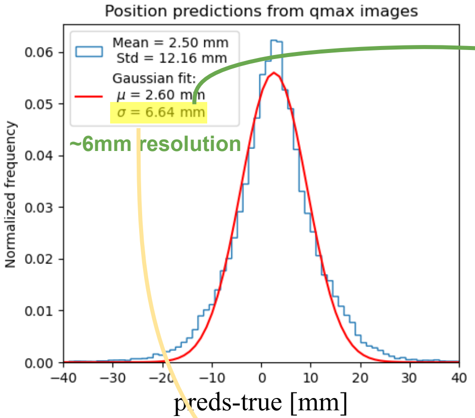


3. Machine learning: neural network for track reconstruction

▻ Results

5 predictions: z_{ini}, p_y, p_z, p_t, ϕ
~280 000 events

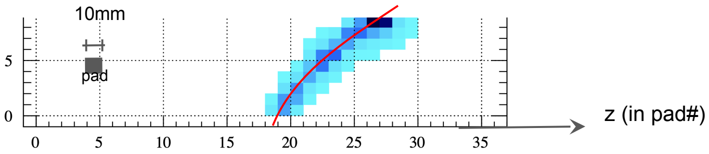
z_{ini}



~6mm resolution

not precise!
1 pad is 10x11mm

vs classical reconstruction: <0.8mm

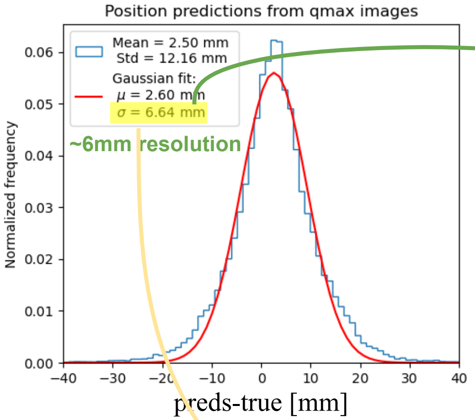


3. Machine learning: neural network for track reconstruction

▻ Results

5 predictions: z_{ini}, p_y, p_z, p_t, ϕ
~280 000 events

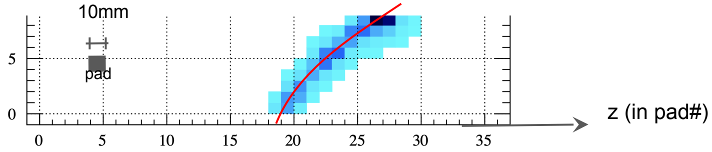
z_{ini}

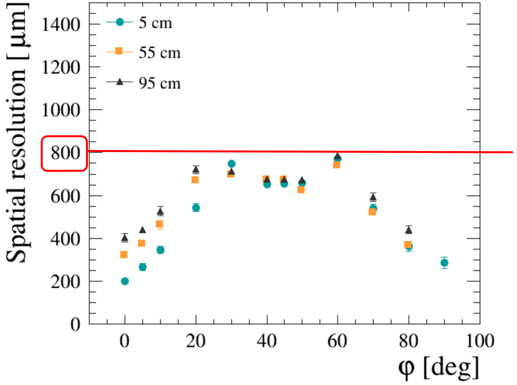


~6mm resolution

not precise!
1 pad is 10x11mm

vs classical reconstruction: <0.8mm





from DESY test beam data with a HA-TPC prototype (2022)

3. Machine learning: neural network for track reconstruction

▻ Current challenges and investigations

❏ Want better spatial resolution

3. Machine learning: neural network for track reconstruction

▻ Current challenges and investigations

❑ Want better spatial resolution:

⤷ understand why it is limited to the resolution of a pad (~10mm)

3. Machine learning: neural network for track reconstruction

▻ Current challenges and investigations

❏ Want better spatial resolution:

⤷ understand why it is limited to the resolution of a pad (~10mm)

☑ try predicting projection of z_{ini} on TPC entrance instead of z_{ini}

☑ generate new simulation data with only vertical tracks at different Δz, drift distance

□ try predicting relative z position wrt pad center instead of absolute position (z_{ini})

3. Machine learning: neural network for track reconstruction

▻ Current challenges and investigations

❑ Want better spatial resolution:

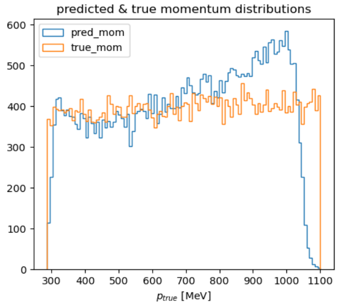
⤷ understand why it is limited to the resolution of a pad (~10mm)

☑ try predicting projection of z_{ini} on TPC entrance instead of z_{ini}

☑ generate new simulation data with only vertical tracks at different Δz, drift distance

□ try predicting relative z position wrt pad center instead of absolute position (z_{ini})

❑ Want better match between distributions of predicted vs true momentum:



3. Machine learning: neural network for track reconstruction

▻ Current challenges and investigations

❏ Want better spatial resolution:

⤷ understand why it is limited to the resolution of a pad (~10mm)

☑ try predicting projection of z_{ini} on TPC entrance instead of z_{ini}

☑ generate new simulation data with only vertical tracks at different Δz, drift distance

□ try predicting relative z position wrt pad center instead of absolute position (z_{ini})

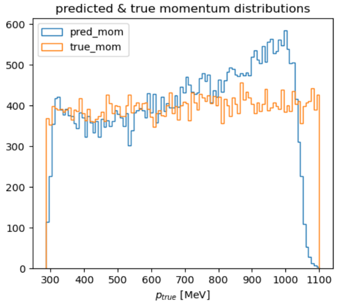
❏ Want better match between distributions of predicted vs true momentum:

☑ standardization of the data: $x_i^{NEW} = \dfrac{x_i - \mu_x}{\sigma_x}$

☑ use more events (75 000 → 280 000)

(☑) border problem? decrease momentum range during test phase (but need more events)

□ generate even more events → ~500 000

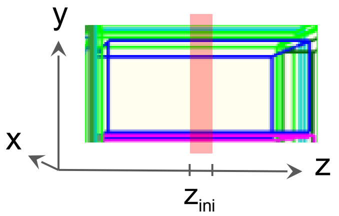


3. Machine learning: neural network for track reconstruction

▻ Current challenges and investigations

Understanding the limitation on z_{ini} resolution

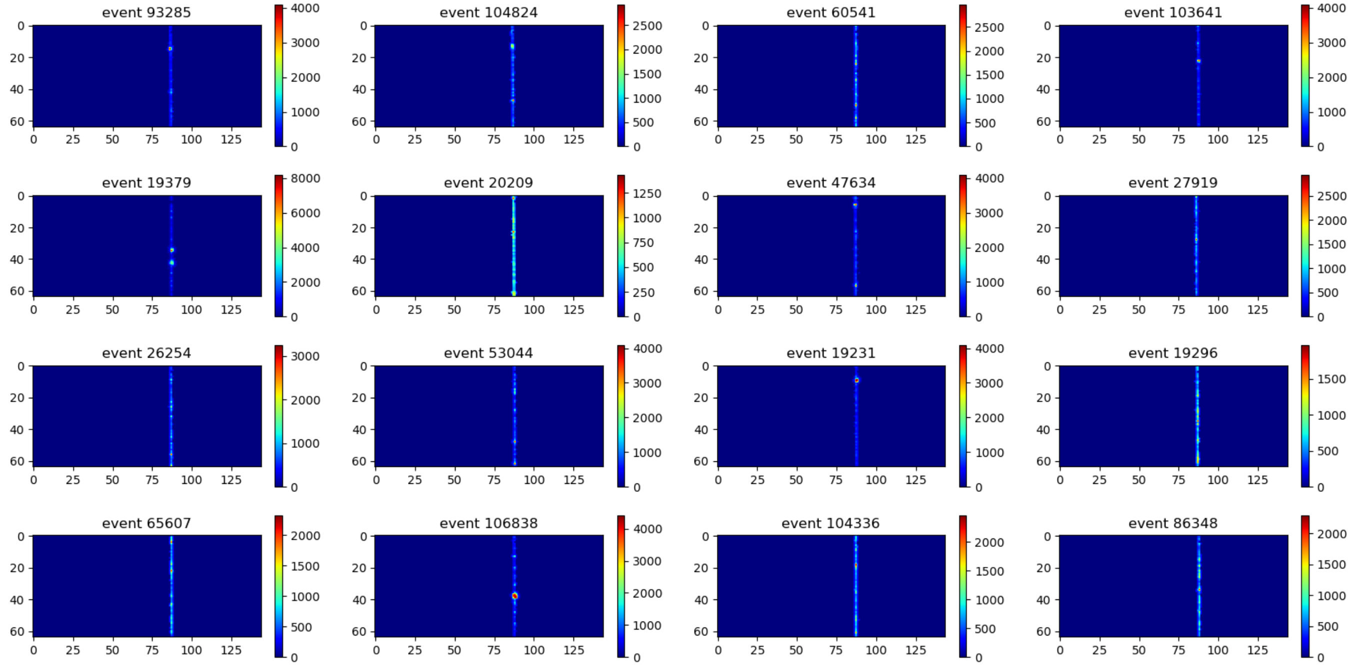
New simu:
vertical tracks
Δz = 1cm (around 1 pad)



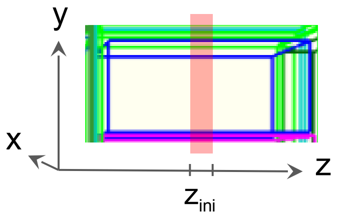
3. Machine learning: neural network for track reconstruction

▻ Current challenges and investigations

Understanding the limitation on z_{ini} resolution



New simu:
vertical tracks
Δz = 1cm (around 1 pad)

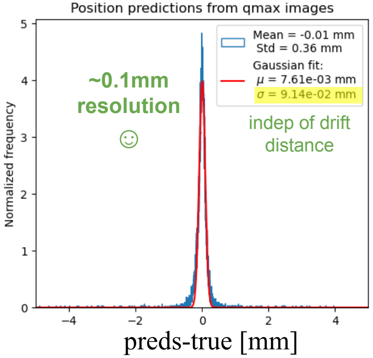


3. Machine learning: neural network for track reconstruction

▻ Current challenges and investigations

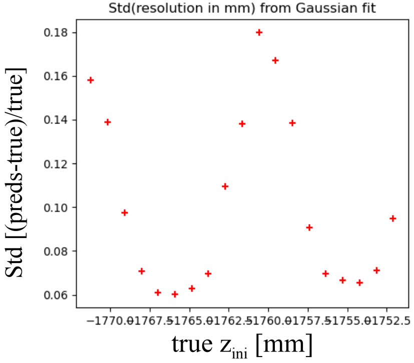
Understanding the limitation on z_{ini} resolution

1 prediction: z_{ini}
~100 000 events

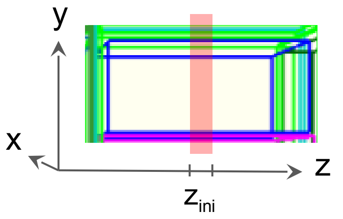


~0.1mm resolution ☺

indep of drift distance



New simu:
vertical tracks
Δz = 1cm (around 1 pad)

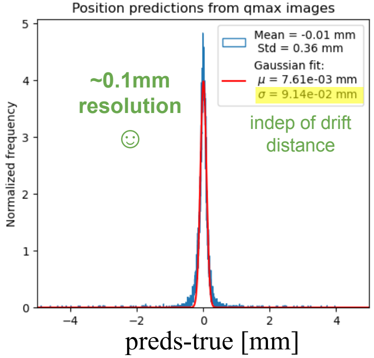


3. Machine learning: neural network for track reconstruction

▻ Current challenges and investigations

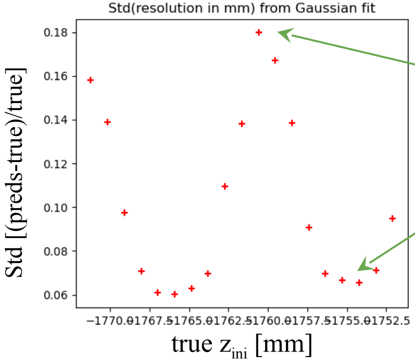
Understanding the limitation on z_{ini} resolution

1 prediction: z_{ini}
~100 000 events



~0.1mm resolution ☺

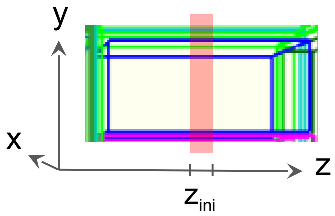
indep of drift distance



worst at pad center

better when between 2 pads

New simu:
vertical tracks
Δz = 1cm (around 1 pad)

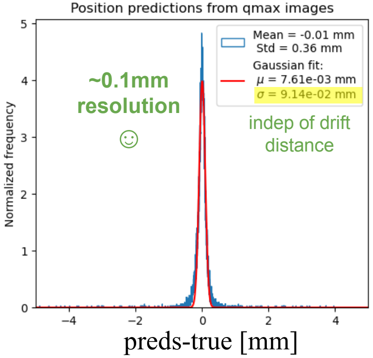


3. Machine learning: neural network for track reconstruction

▻ Current challenges and investigations

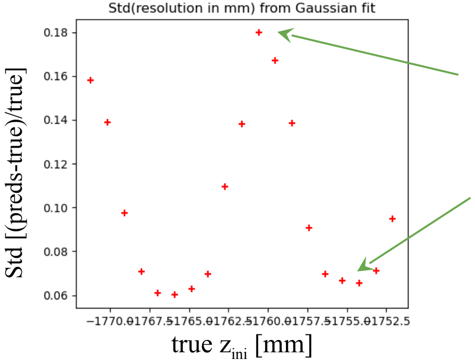
1 prediction: z_{ini}
~100 000 events

Understanding the limitation on z_{ini} resolution



~0.1mm resolution ☺

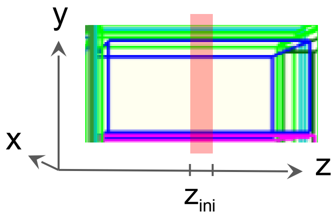
indep of drift distance

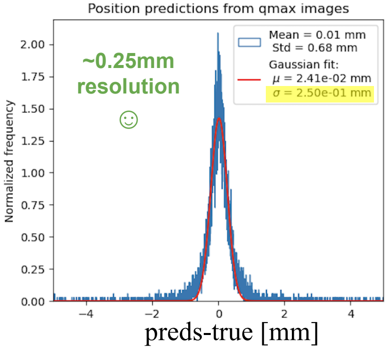


worst at pad center

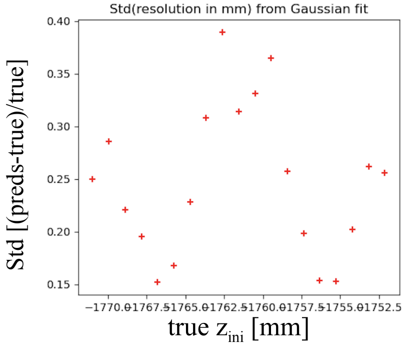
better when between 2 pads

New simu:
vertical tracks
Δz = 1cm (around 1 pad)





~0.25mm resolution ☺



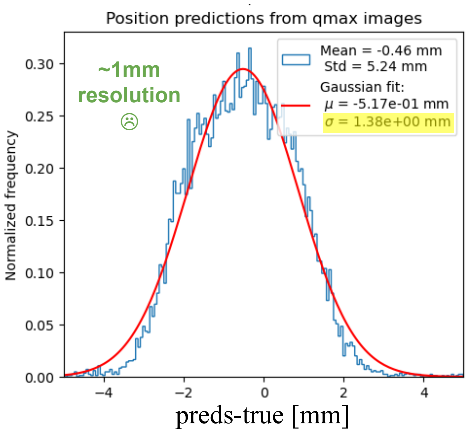
using only the 1rst row of the image, as during test beam

3. Machine learning: neural network for track reconstruction

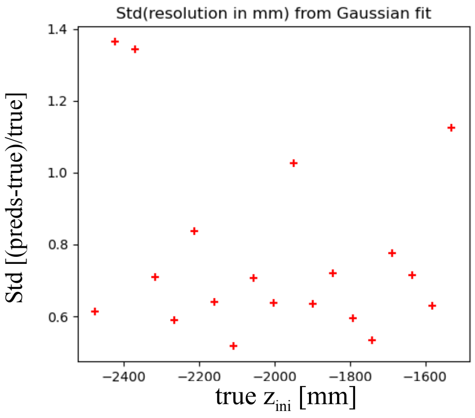
▻ Current challenges and investigations

Understanding the limitation on z_{ini} resolution

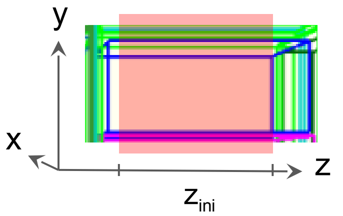
1 prediction: z_{ini}
~100 000 events



~1mm resolution ☹



New simu:
vertical tracks
Δz = 1m (all TPC)



⤷ now try predicting **relative** position
⤷ can also try a simu with intermediate Δz

▻ Internship conclusion

- ★ Development from scratch of a new reconstruction technique with ML
- ★ There remains a lot to understand; biases, resolution vs momentum, initial position resolution
- ★ But gives comparable results to the classical/’official’ reconstruction method

⇢ **Promising! Still work to do & ideas to try!**

4. Prospects for the PhD

Continue the work on this neural network:

Main goal: decrease systematic uncertainties for precise CPV measurements
i.e. need precise track parameters since used in event selection, flux characterizations, cross sections

- aim for same/better resolution on all track parameters (z_{ini}, p_t, ϕ) than with classical reconstruction
- add the FWHM an tmax information to the input images and see how it goes
- try the CNN on test beam data (and later T2K-II data once ND280 upgrade is installed)

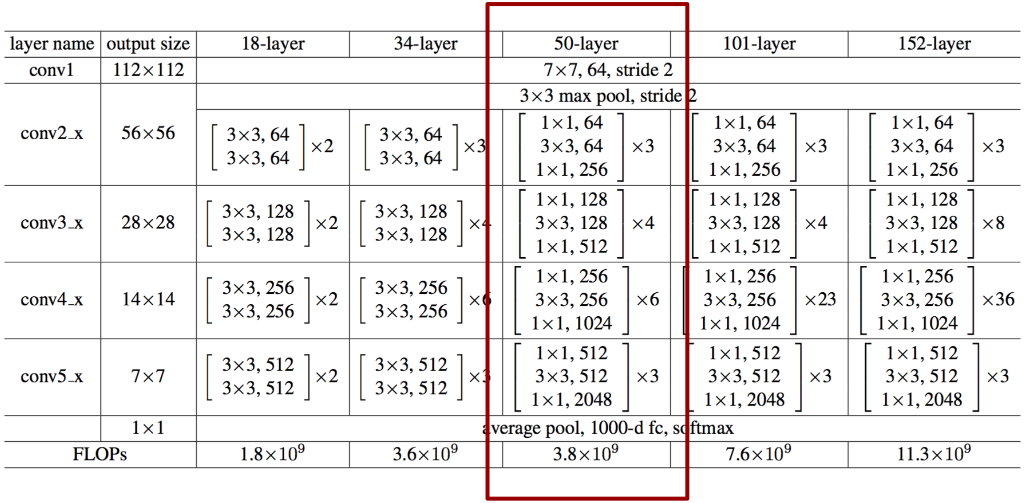
New horizons:

- collaborate more with a T2K ML group (SuperFGD)?
- experiment with another neural network architecture?
- oscillation analysis from ND280 upgrade data: study of ν_e and anti- ν_e interactions

Back-up slides

ML with PyTorch

- ➢ **ResNet50** ⟶



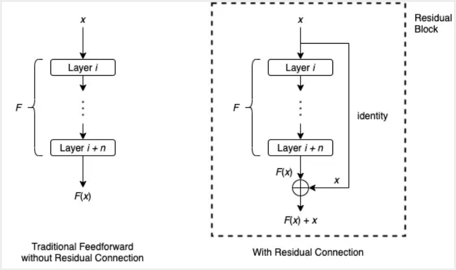
layer name	output size	18-layer	34-layer	50-layer	101-layer	152-layer
conv1	112×112	7×7, 64, stride 2				
conv2_x	56×56	3×3 max pool, stride 2				
		$\begin{bmatrix}3\times3, 64\\3\times3, 64\end{bmatrix}\times2$	$\begin{bmatrix}3\times3, 64\\3\times3, 64\end{bmatrix}\times3$	$\begin{bmatrix}1\times1, 64\\3\times3, 64\\1\times1, 256\end{bmatrix}\times3$	$\begin{bmatrix}1\times1, 64\\3\times3, 64\\1\times1, 256\end{bmatrix}\times3$	$\begin{bmatrix}1\times1, 64\\3\times3, 64\\1\times1, 256\end{bmatrix}\times3$
conv3_x	28×28	$\begin{bmatrix}3\times3, 128\\3\times3, 128\end{bmatrix}\times2$	$\begin{bmatrix}3\times3, 128\\3\times3, 128\end{bmatrix}\times4$	$\begin{bmatrix}1\times1, 128\\3\times3, 128\\1\times1, 512\end{bmatrix}\times4$	$\begin{bmatrix}1\times1, 128\\3\times3, 128\\1\times1, 512\end{bmatrix}\times4$	$\begin{bmatrix}1\times1, 128\\3\times3, 128\\1\times1, 512\end{bmatrix}\times8$
conv4_x	14×14	$\begin{bmatrix}3\times3, 256\\3\times3, 256\end{bmatrix}\times2$	$\begin{bmatrix}3\times3, 256\\3\times3, 256\end{bmatrix}\times6$	$\begin{bmatrix}1\times1, 256\\3\times3, 256\\1\times1, 1024\end{bmatrix}\times6$	$\begin{bmatrix}1\times1, 256\\3\times3, 256\\1\times1, 1024\end{bmatrix}\times23$	$\begin{bmatrix}1\times1, 256\\3\times3, 256\\1\times1, 1024\end{bmatrix}\times36$
conv5_x	7×7	$\begin{bmatrix}3\times3, 512\\3\times3, 512\end{bmatrix}\times2$	$\begin{bmatrix}3\times3, 512\\3\times3, 512\end{bmatrix}\times3$	$\begin{bmatrix}1\times1, 512\\3\times3, 512\\1\times1, 2048\end{bmatrix}\times3$	$\begin{bmatrix}1\times1, 512\\3\times3, 512\\1\times1, 2048\end{bmatrix}\times3$	$\begin{bmatrix}1\times1, 512\\3\times3, 512\\1\times1, 2048\end{bmatrix}\times3$
	1×1	average pool, 1000-d fc, softmax				
FLOPs		1.8×10^9	3.6×10^9	3.8×10^9	7.6×10^9	11.3×10^9

- ➢ **Loss**: $\mathrm{MSE} = \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2$

- ➢ **Optimizer** (how to update weights): extension of stochastic gradient descent

- ➢ **Hyperparameters:**
  - ▫ training/validation/test set splitting (now: 70/15/15)
  - ▫ **batch size**: commonly used: 64, also tried 1, 16, 32, 128
  - ▫ **learning rate**: initially at 0.01 + scheduler to decrease it dynamically
  - ▫ **patience:** how many epochs before decreasing LR
  - ▫ **epoch size** (1, 10, 30, 50, 100) -> implement dynamical epoch size
  - ▫ ‘model choice’: ResNet50, 101, 152 for now but mostly ResNet50

Residual NN

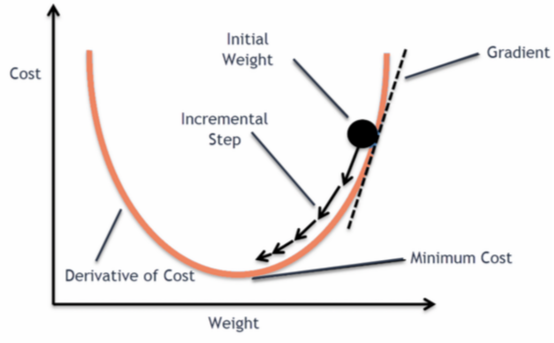


Traditional Feedforward without Residual Connection

With Residual Connection

ML with PyTorch

NNlearning process:



1/ Measures how wrong the NN predictions are:

$$\underset{\text{or 'loss'}}{cost}(w, b) = \frac{1}{N} \sum_{j=1}^{N} \left[y^{j}_{pred}(w, b) - y^{j}_{true} \right]^{2}$$

2/ Performs a **gradient descent** algo to find the weight/bias values which best minimize the cost

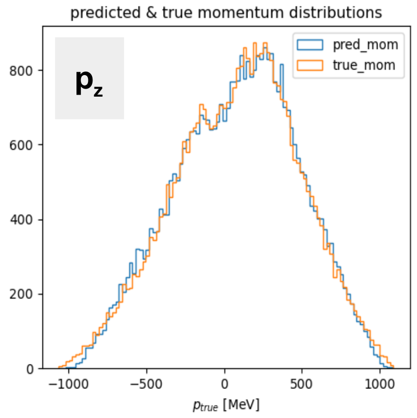
This is done by looping several times over the all dataset (1 loop = 1 'epoch')

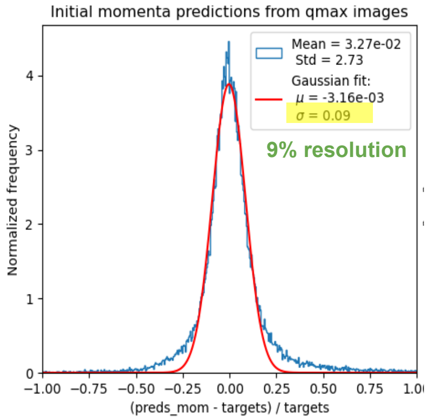
$$\theta_j = \theta_j - \alpha \frac{\partial}{\partial \theta_j} J(\theta)$$

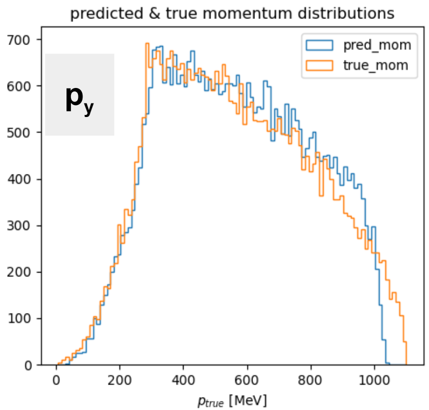
params (w,b) learning rate cost function

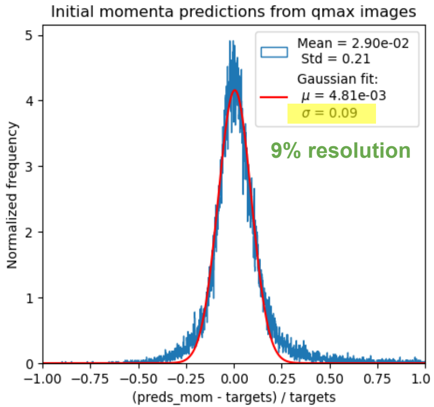
Other results

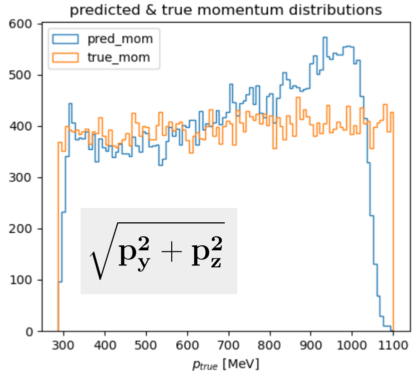
3 predictions: p_y, p_z, p_t
~280 000 events

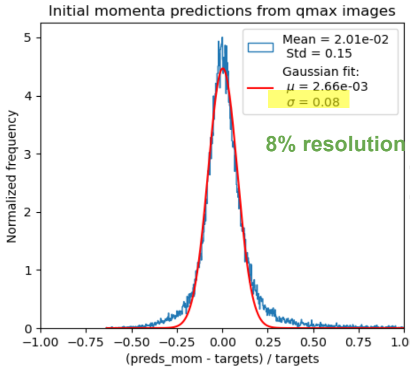


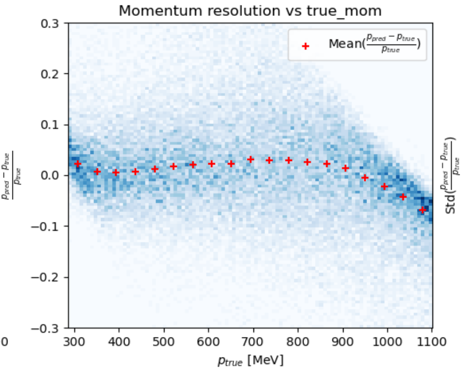


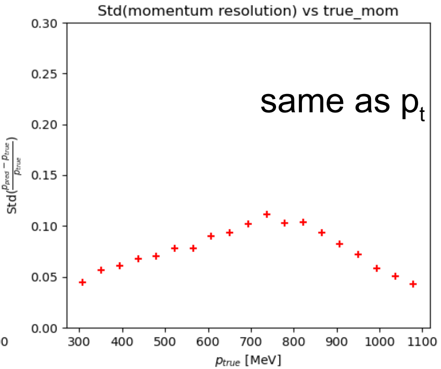




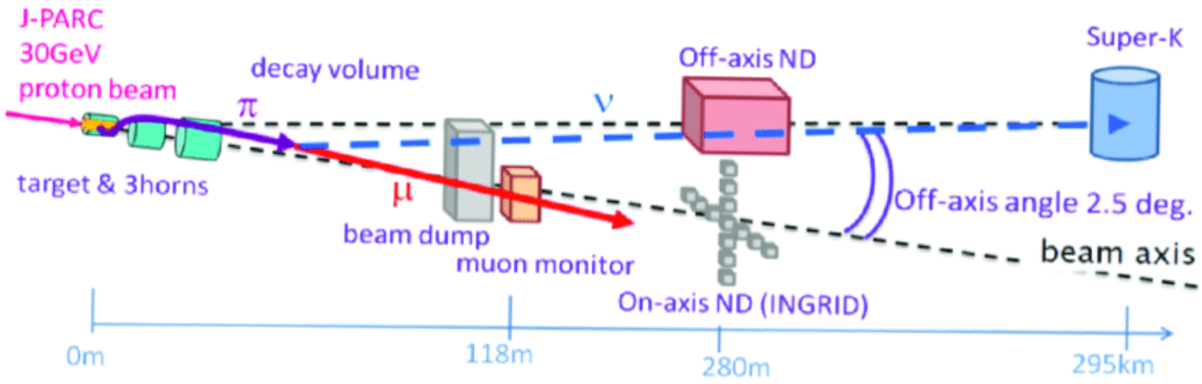






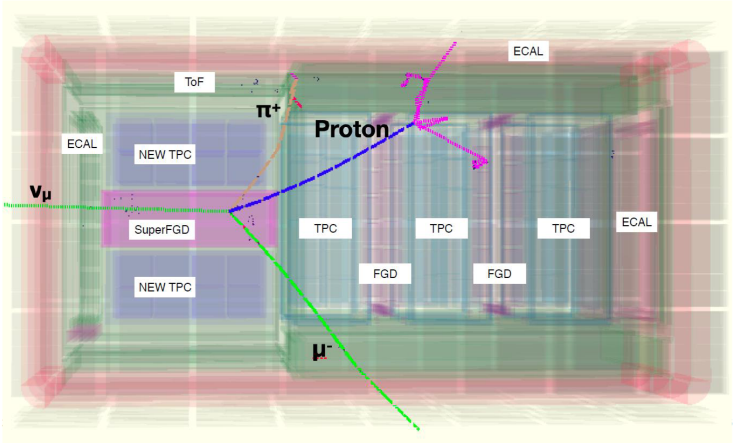


T2K beam production

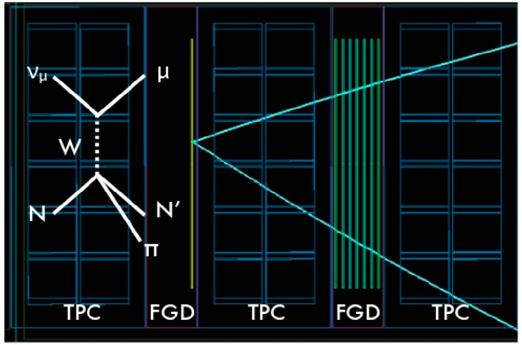


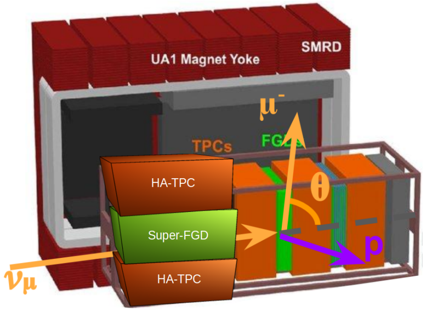
Event displays before/after ND280 upgrade

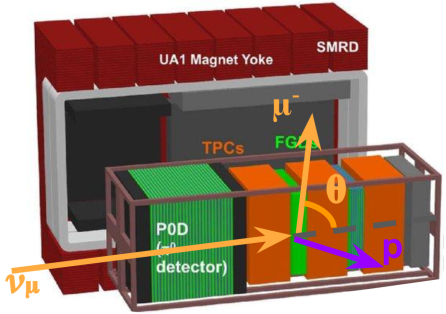
event simulated with ND280 upgrade configuration



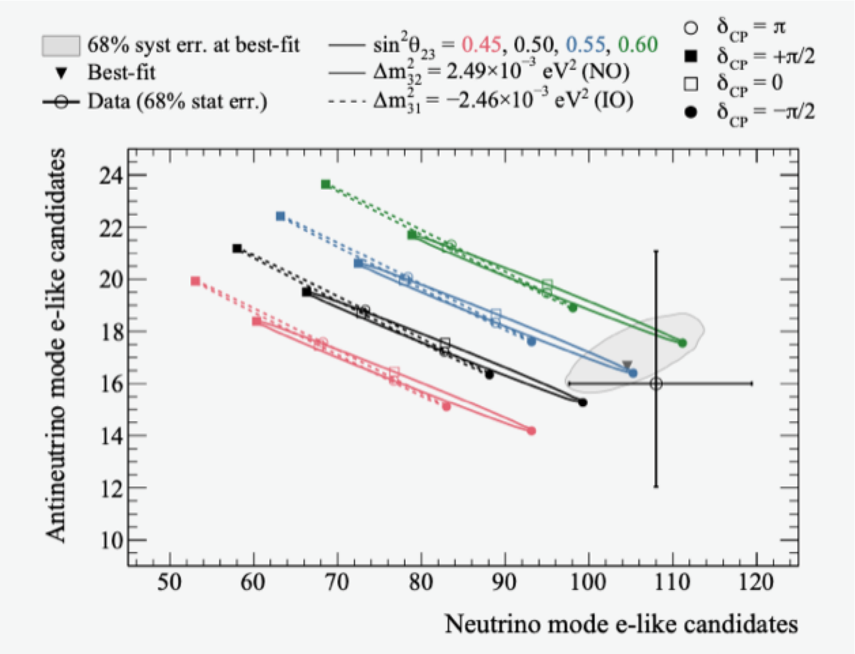
event display in old configuration







δ_{CP} measurements with T2K



Apparition number of anti-ν_e vs apparition number of ν_e at the far detector SK