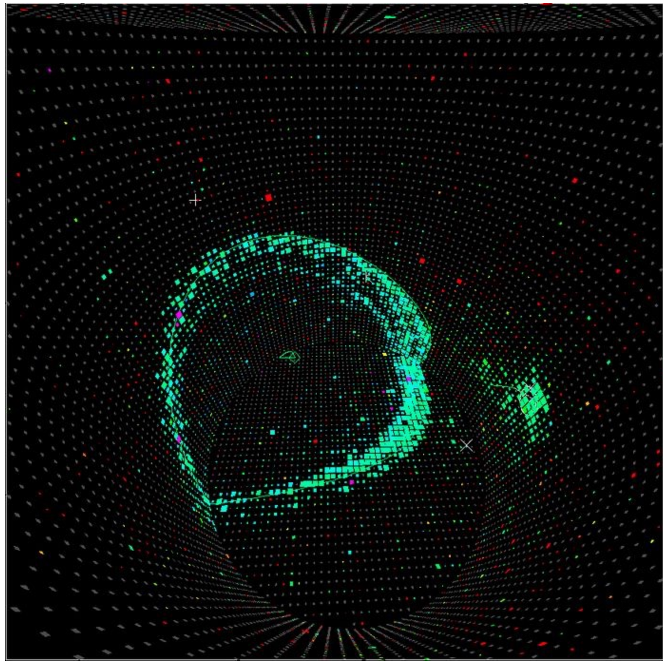
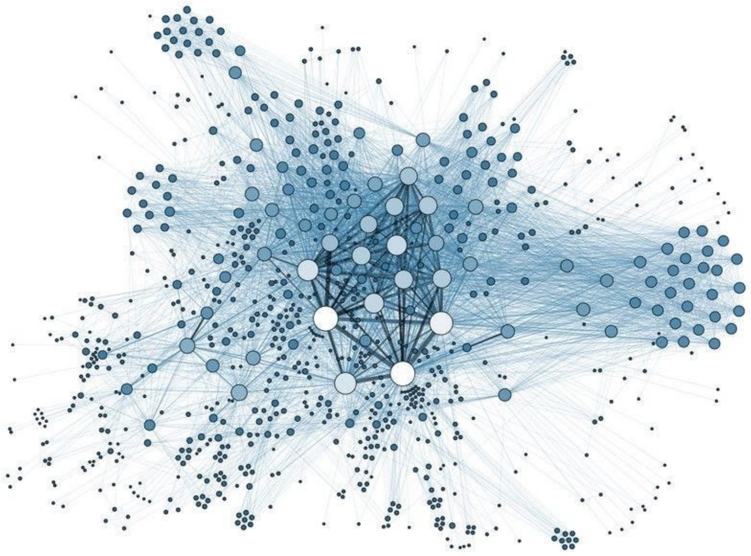
ILΛNCE

International Laboratory for Astrophysics, Neutrino and Cosmology Experiments

Machine learning for low energy event reconstruction in HyperKamiokande





Summary

Introduction

- Why low energy electrons ?
- Notions of machine learning

Our work

Anthi

Investigating variables for energy reconstruction

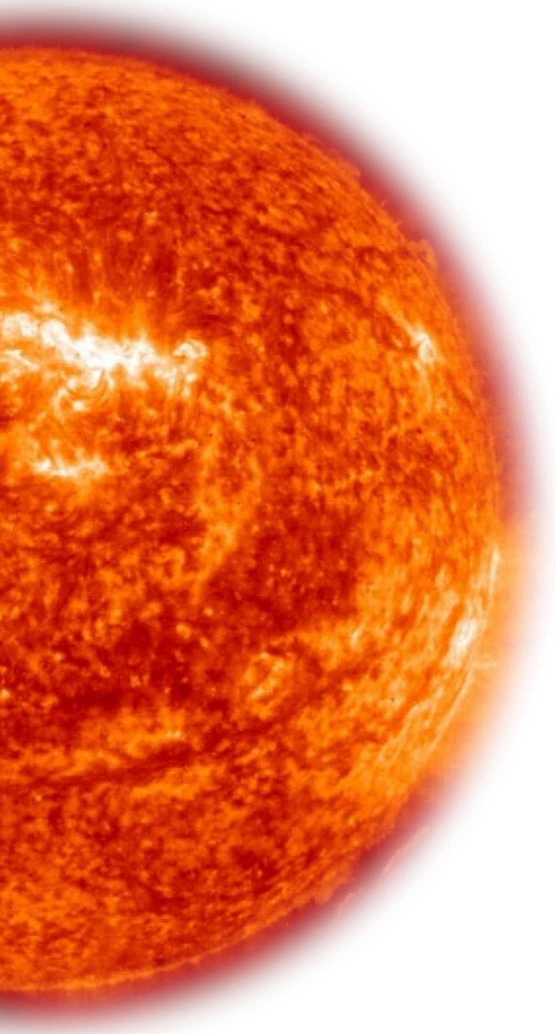
- Bayesian neural network
- Our simulation
- Variables and models
- Results
- Persectives

Clément

Graphic neural network for vertex reconstruction

- Graphic neural network
- Models
- Hyperparameters optimization
- Results and perspectives

Why low energy electrons ?

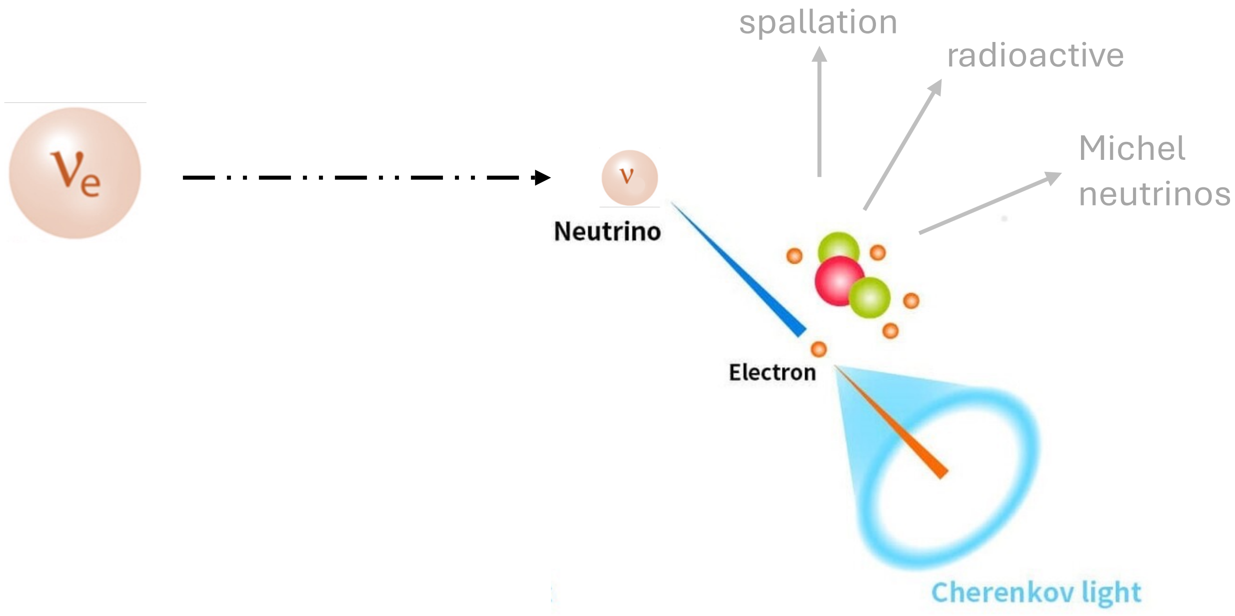


$$^3\mathrm{He} + p \rightarrow {}^4\mathrm{He} + e^+ + \nu_e$$

$$(hep)$$

$$^8\mathrm{B} \rightarrow {}^8\mathrm{Be}^* + e^+ + \nu_e$$

$$(^8B)$$

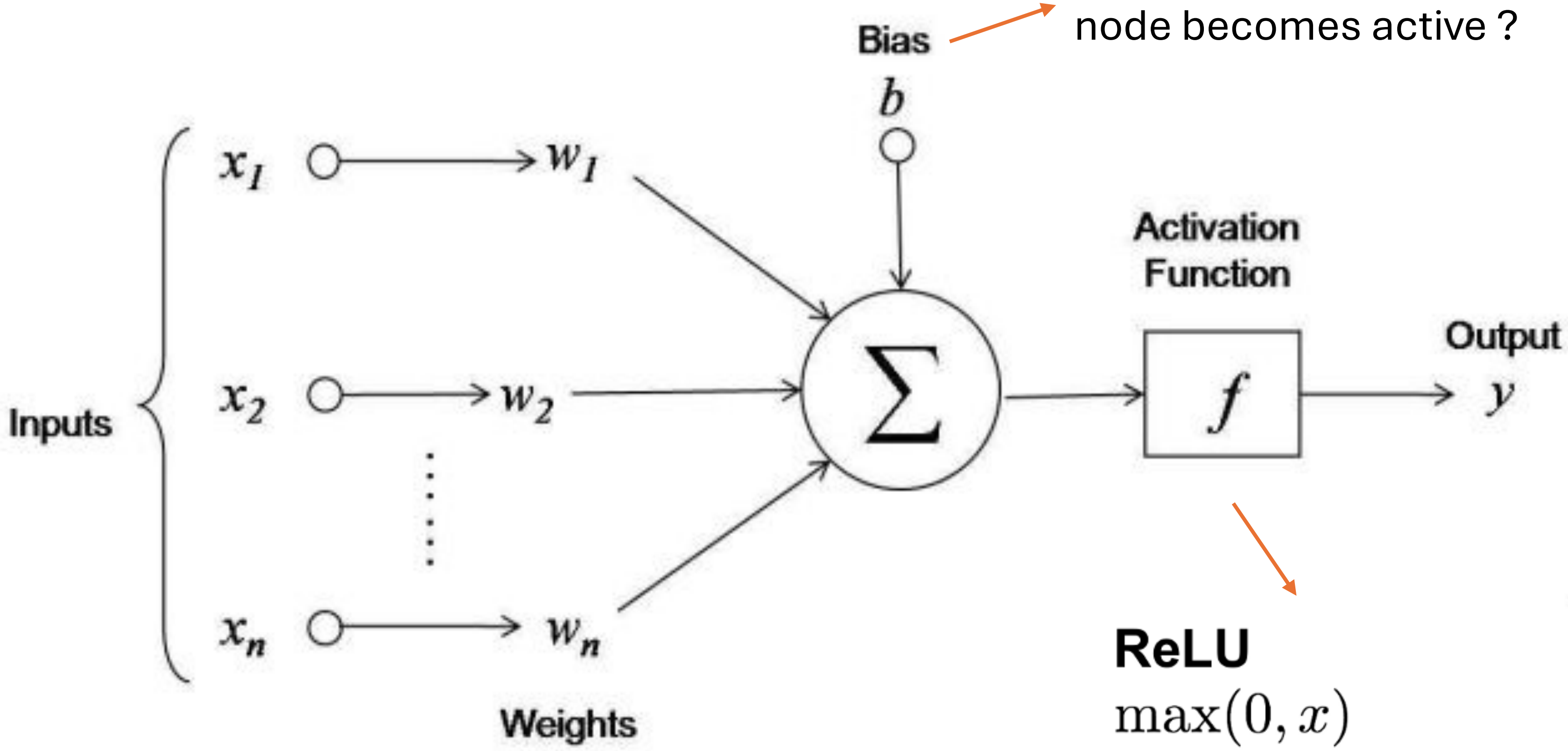


<u>Our goal :</u> improve statistics by better reconstruction

Notions of machine learning

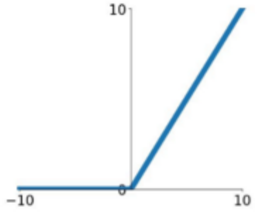
Perceptron model :

How big this sum needs to be before the node becomes active ?



ReLU

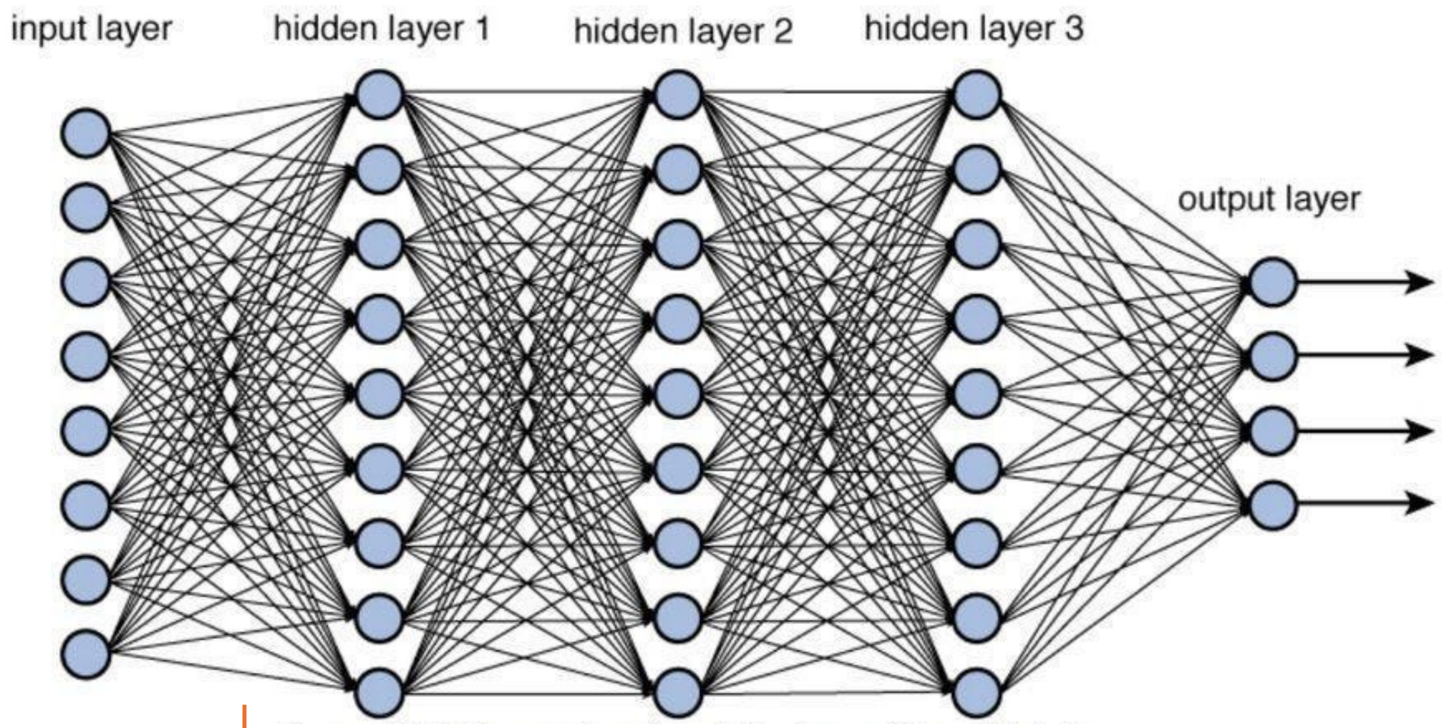
$\max(0, x)$



Which x_i is important here ?

Notions of machine learning

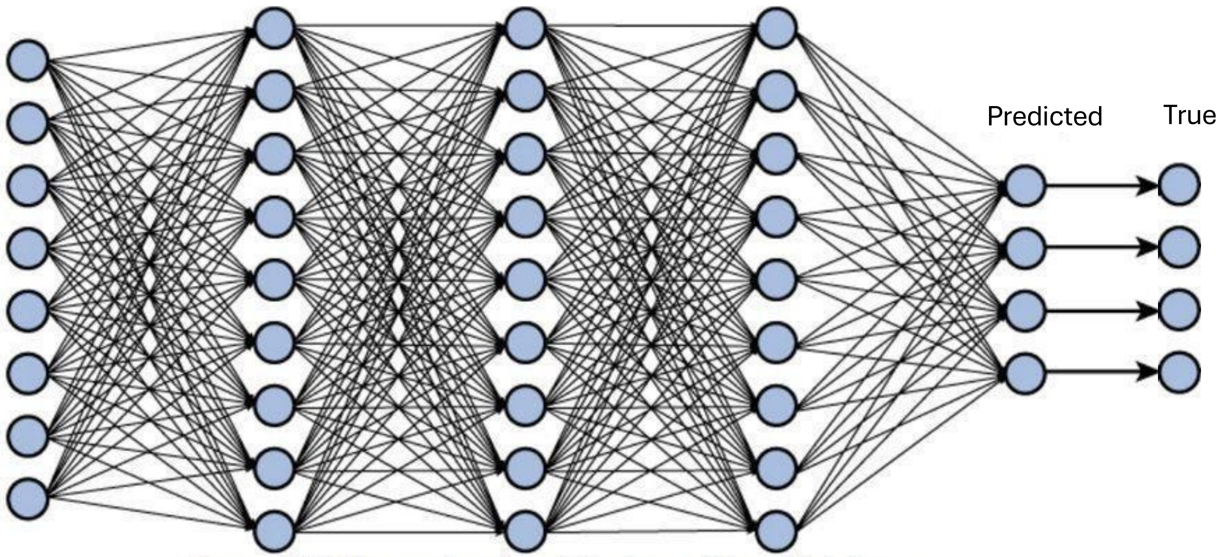
<u>Multi perceptron model or Neural Network:</u>



$$\begin{bmatrix} w_{0,0} & w_{0,1} & \cdots & w_{0,n} \\ w_{1,0} & w_{1,1} & \cdots & w_{1,n} \\ \vdots & \vdots & \ddots & \vdots \\ w_{k,0} & w_{k,1} & \cdots & w_{k,n} \end{bmatrix} \begin{bmatrix} a_0^{(0)} \\ a_1^{(0)} \\ \vdots \\ a_n^{(0)} \end{bmatrix} + \begin{bmatrix} b_0 \\ b_1 \\ \vdots \\ b_k \end{bmatrix}$$

Notions of machine learning

Multi perceptron model or Neural Network:

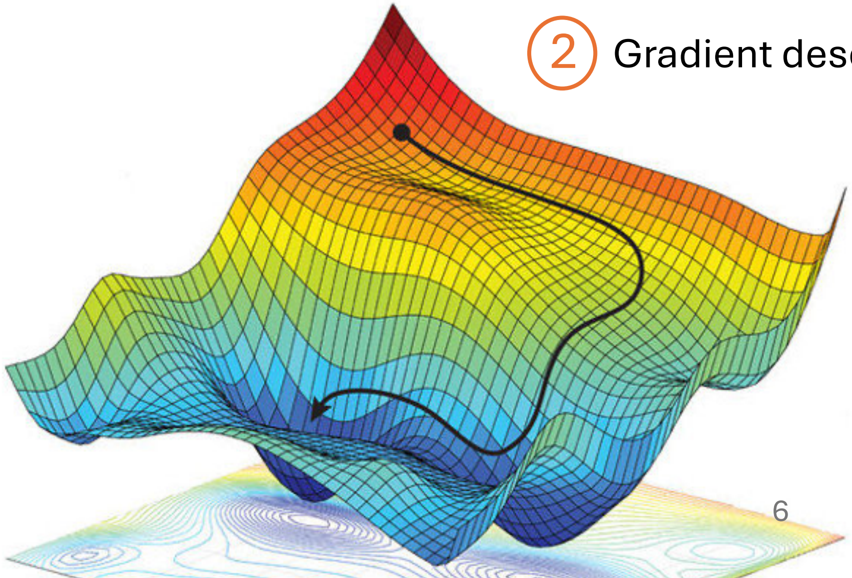


1 Loss

$$L = \text{MSE} = \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2$$

$\delta^{(L)}$

2 Gradient descent

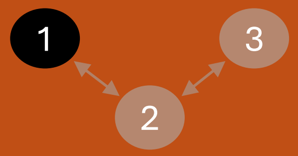


3 Backpropagation

$$\mathbf{W}^{(l)} \leftarrow \mathbf{W}^{(l)} - \eta\frac{\partial L}{\partial \mathbf{W}^{(l)}} \dashrightarrow \frac{\partial L}{\partial \mathbf{W}^{(l)}} = \delta^{(l)}\mathbf{a}^{(l-1)T}$$

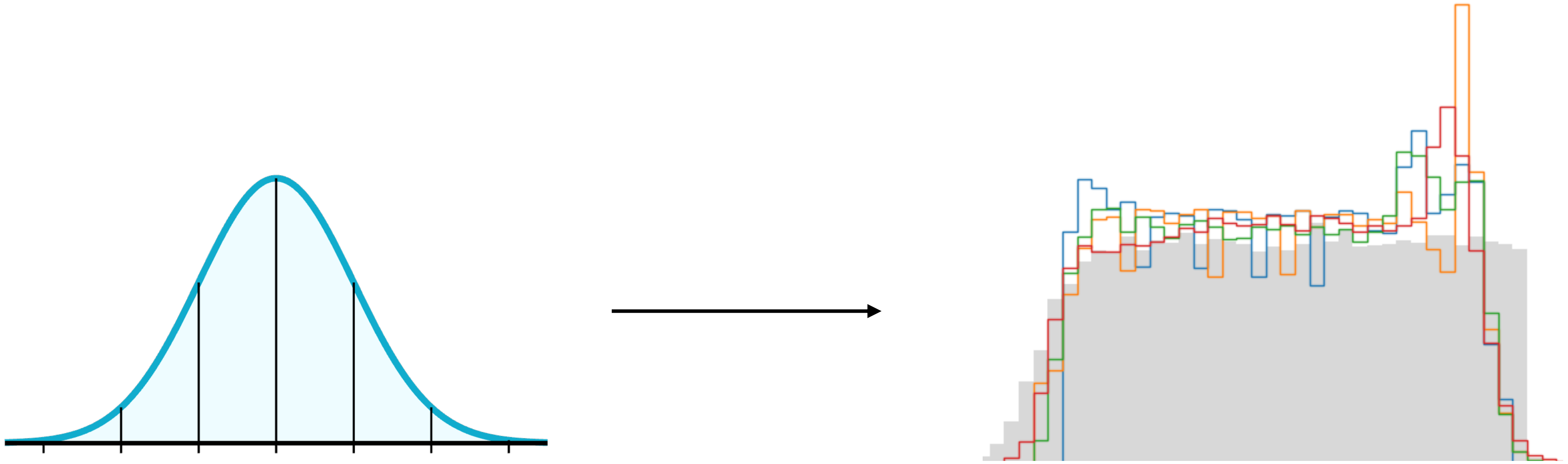
$$\mathbf{b}^{(l)} \leftarrow \mathbf{b}^{(l)} - \eta\frac{\partial L}{\partial \mathbf{b}^{(l)}} \dashrightarrow \frac{\partial L}{\partial \mathbf{b}^{(l)}} = \delta^{(l)}$$

learning rate



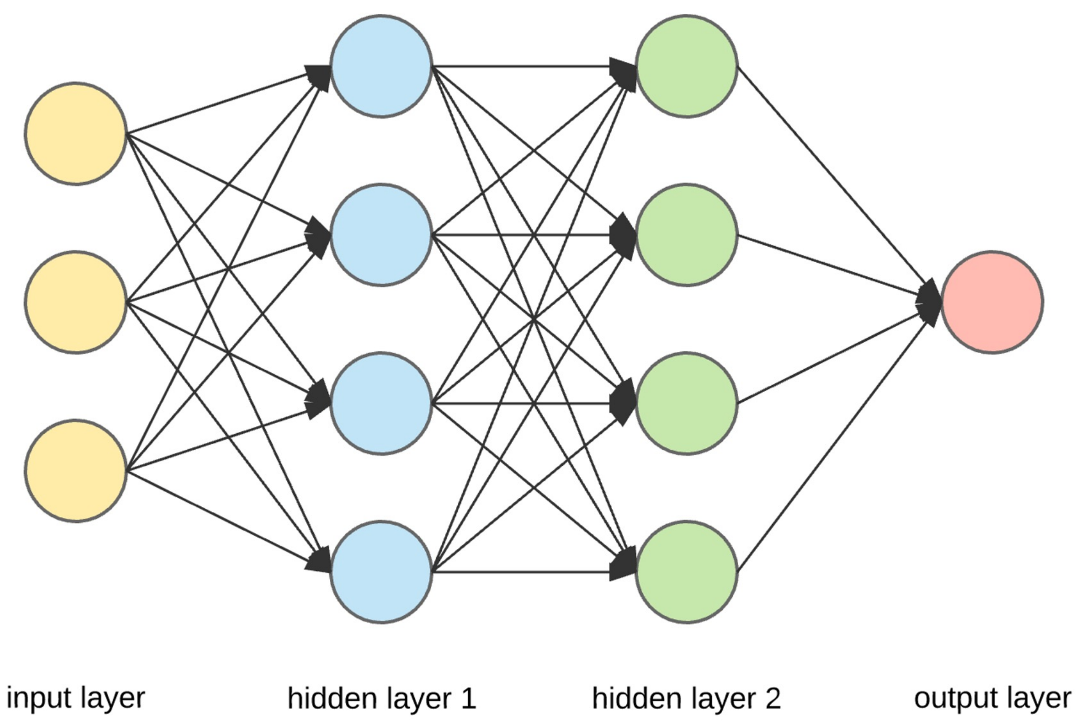
Anthi :

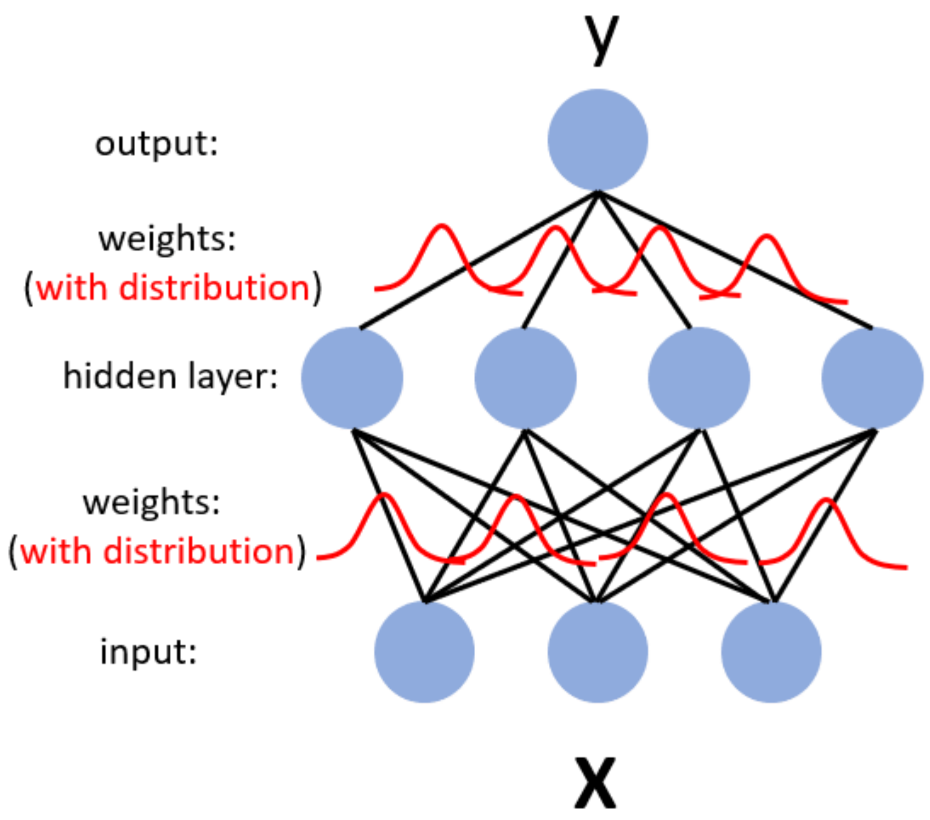
Investigating variables for energy reconstruction



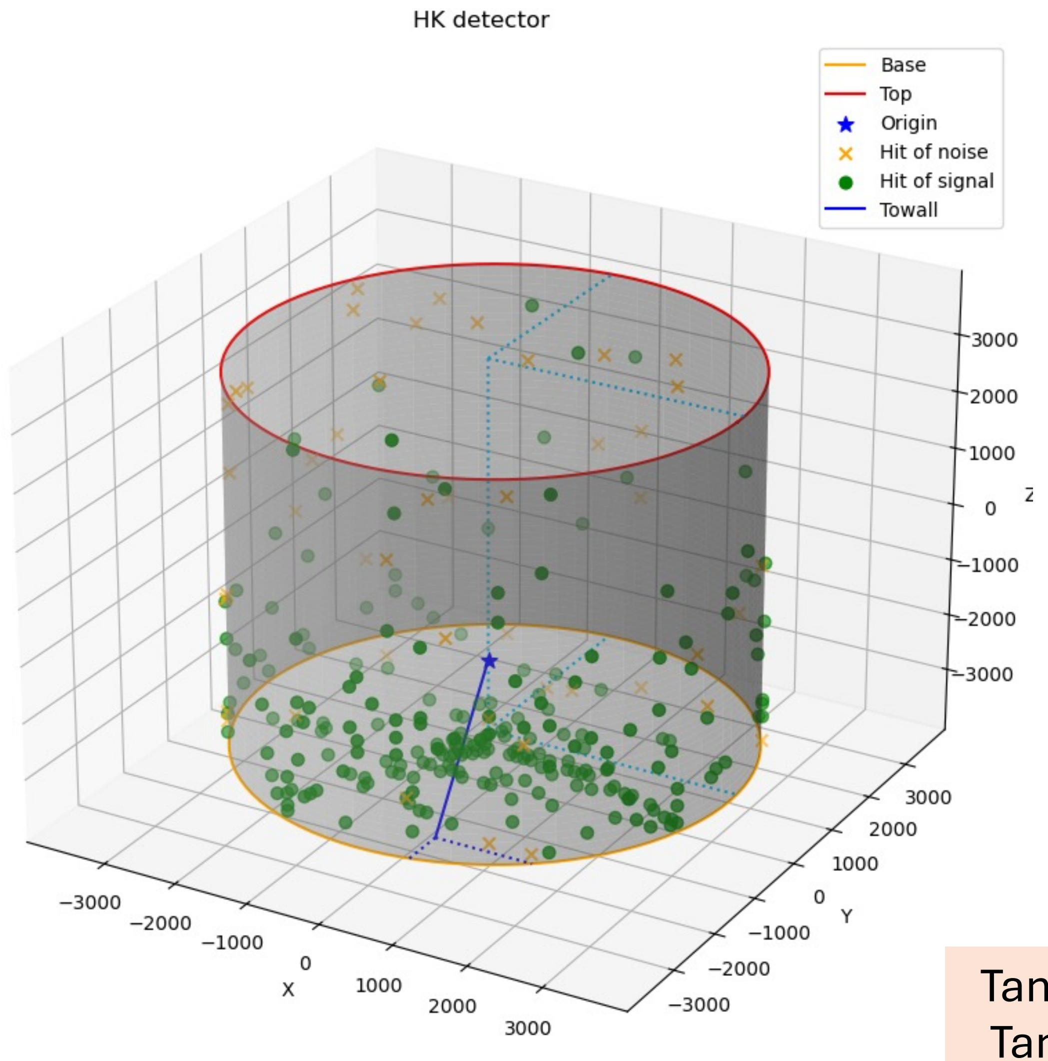
Bayesian Neural Network

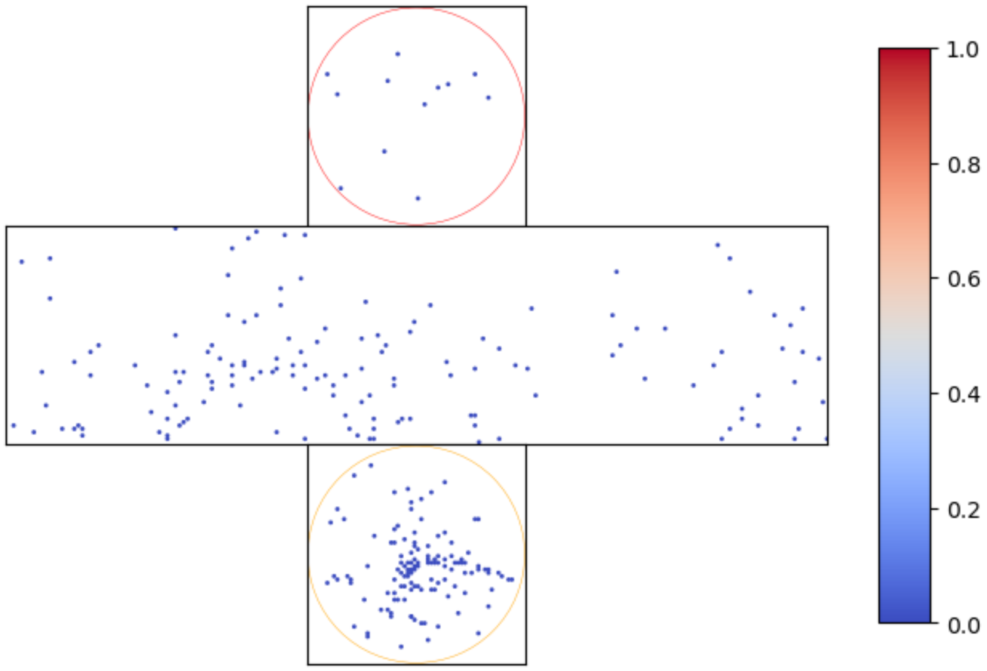
$$y = f(\beta_0 + \beta_1 x_1 + \beta_2 x_2 ... + \beta_n x_n)$$
$$\beta_i \sim \mathcal{N}(0, 1^2)$$





Our simulation





Tank diameter = 6480cm
Tank height = 6575.1cm

The variables and the models

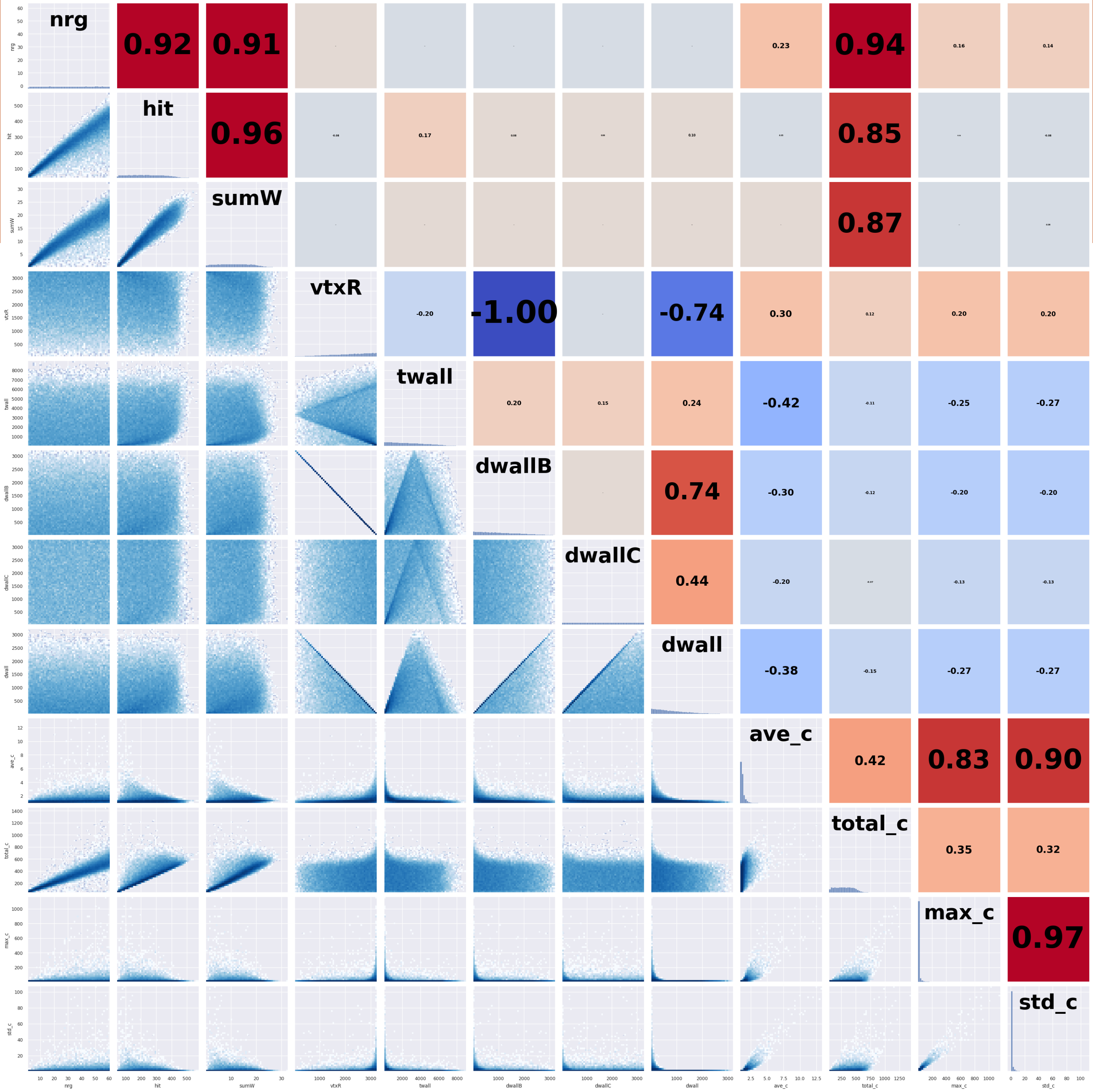
3 Layer

6 Layer

More Neurons

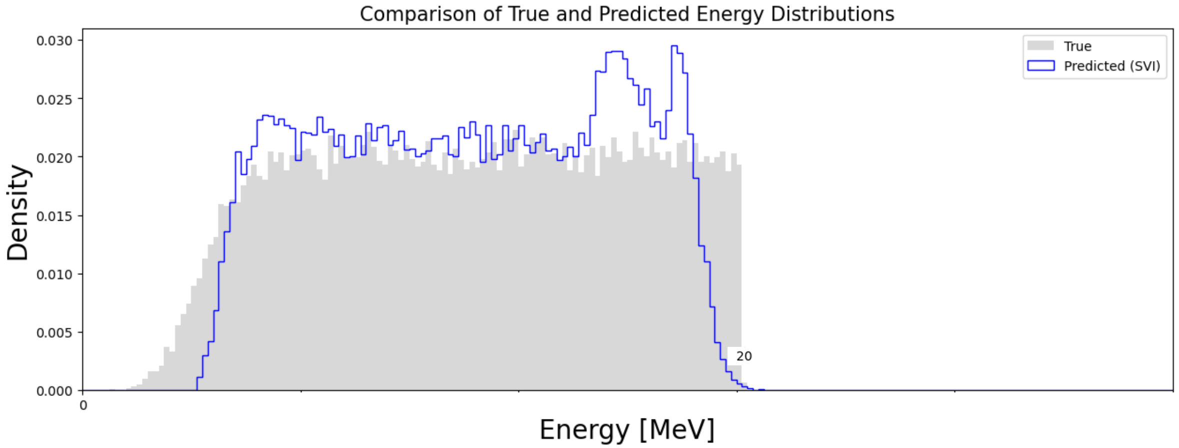
Last Layer

Dropout

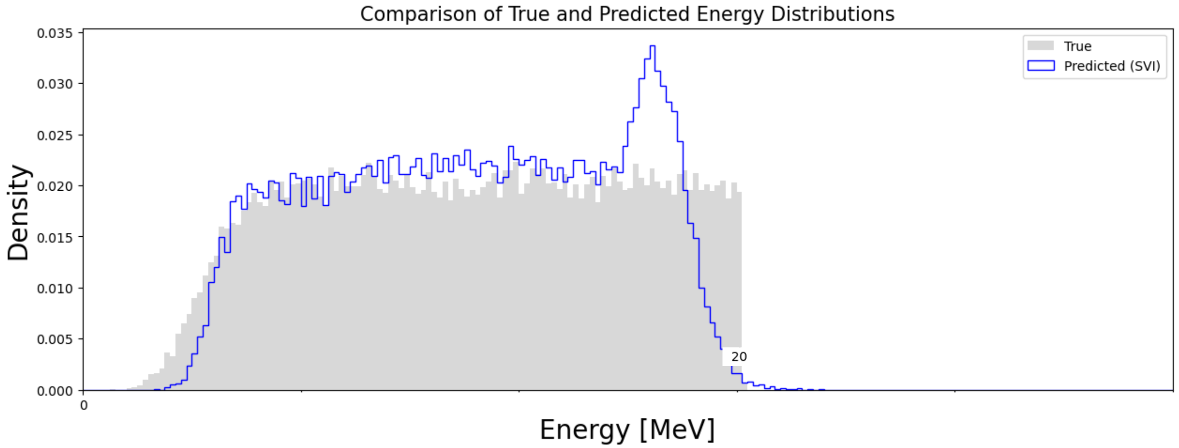


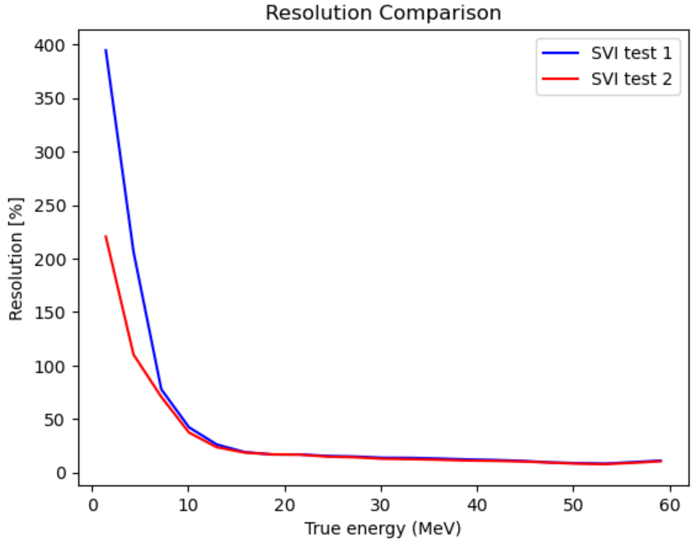
Example Results (input variables)

1 **nhits + sumW**

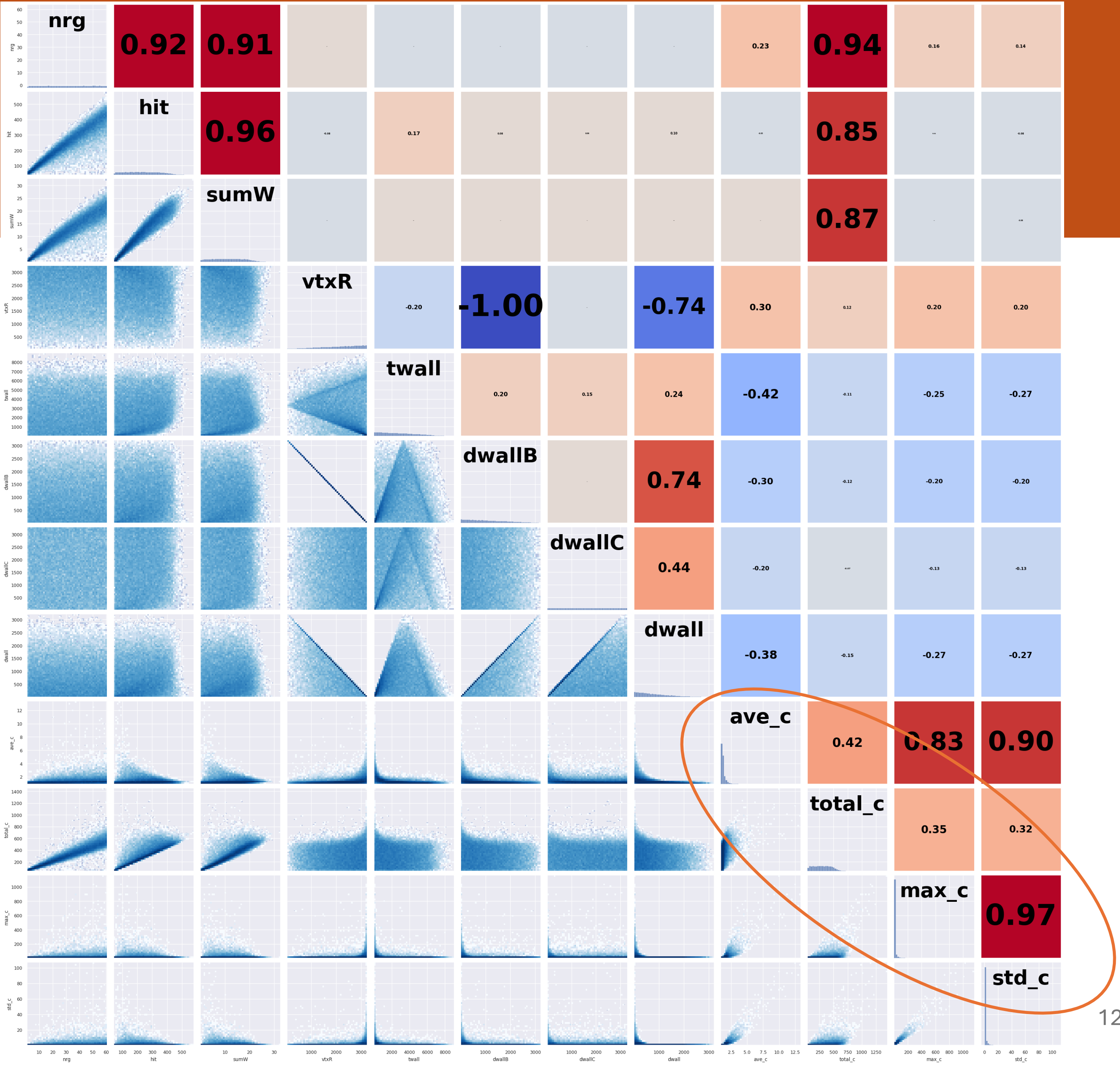


2 **nhits + sumW + position variables**



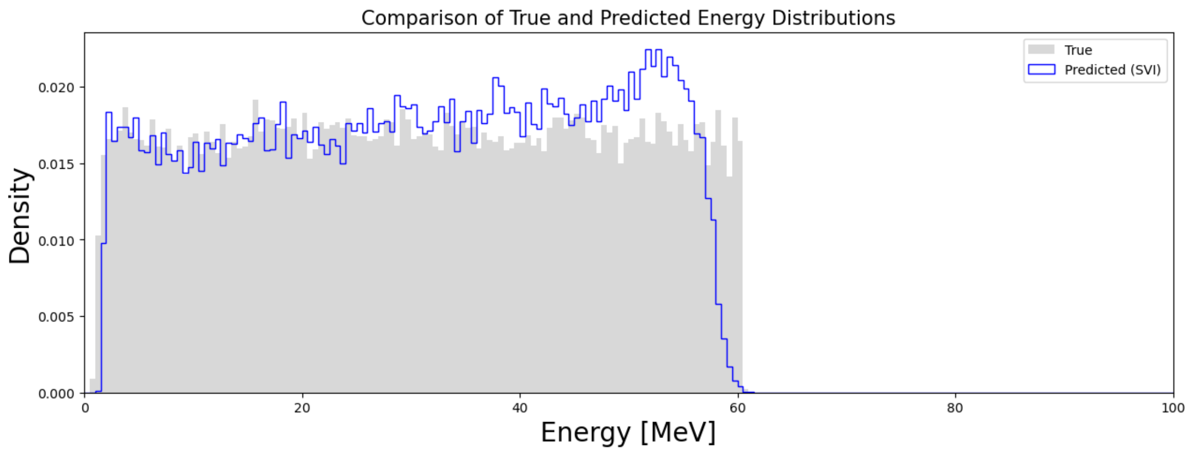


Adding input variables

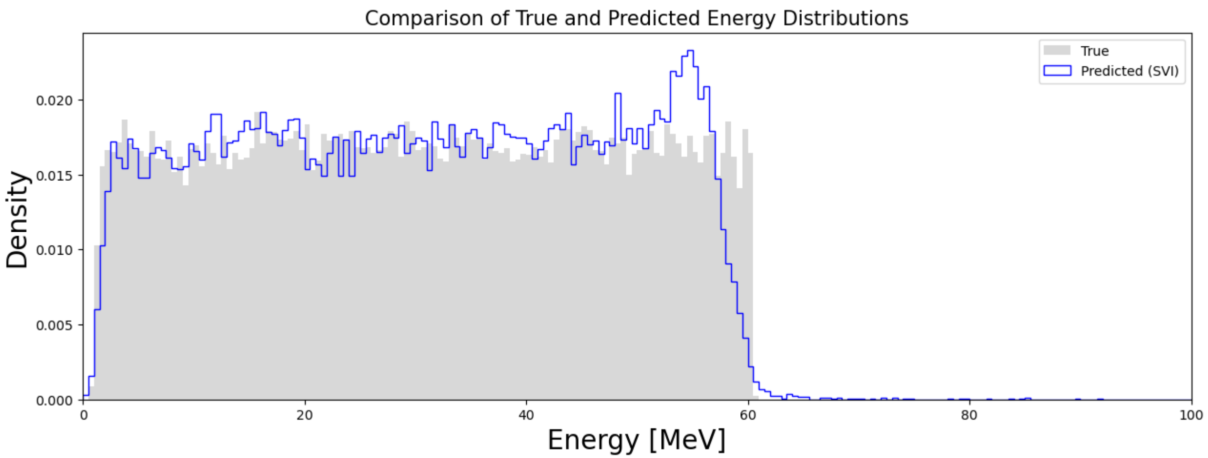


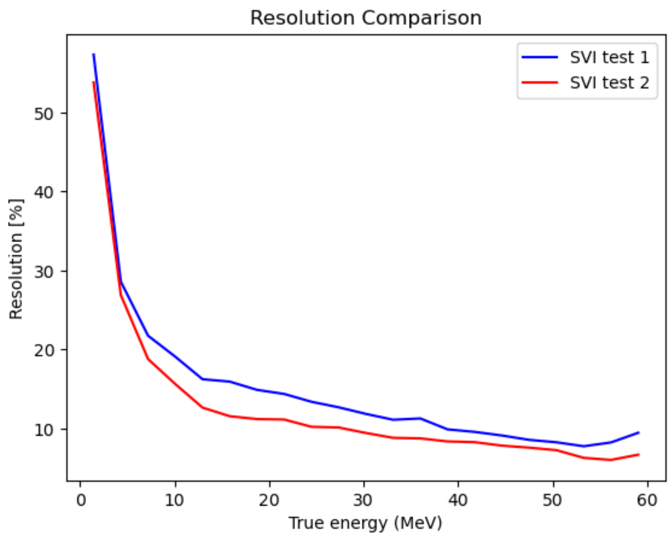
Example Results (set4/set5)

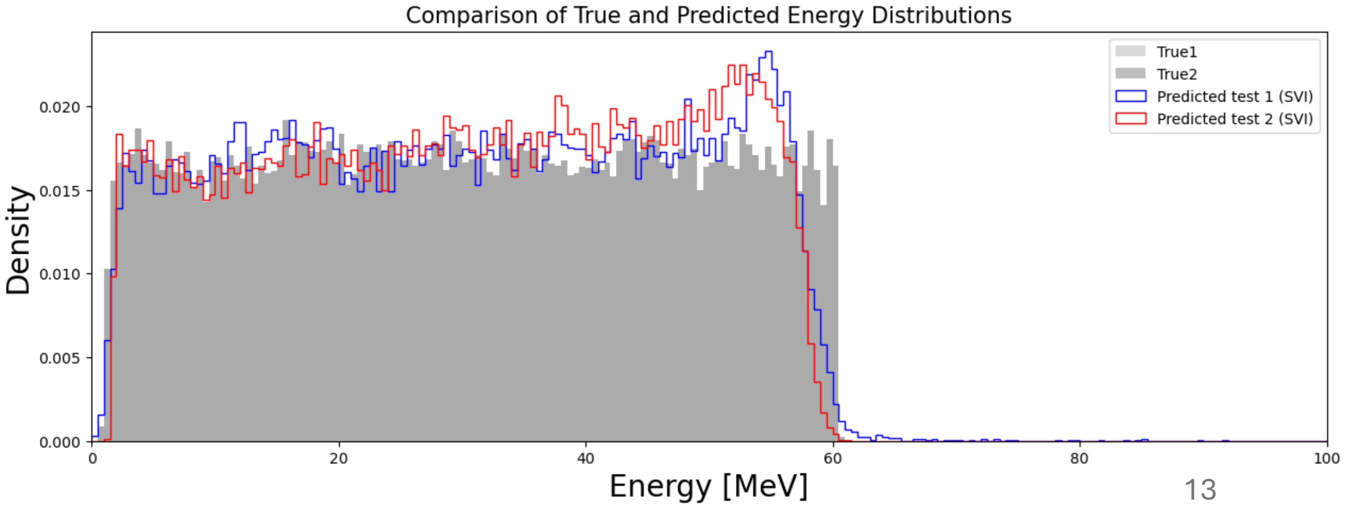
1 **nhits + sumW +position variables**



2 **nhits + sumW + position variables + charge variables**



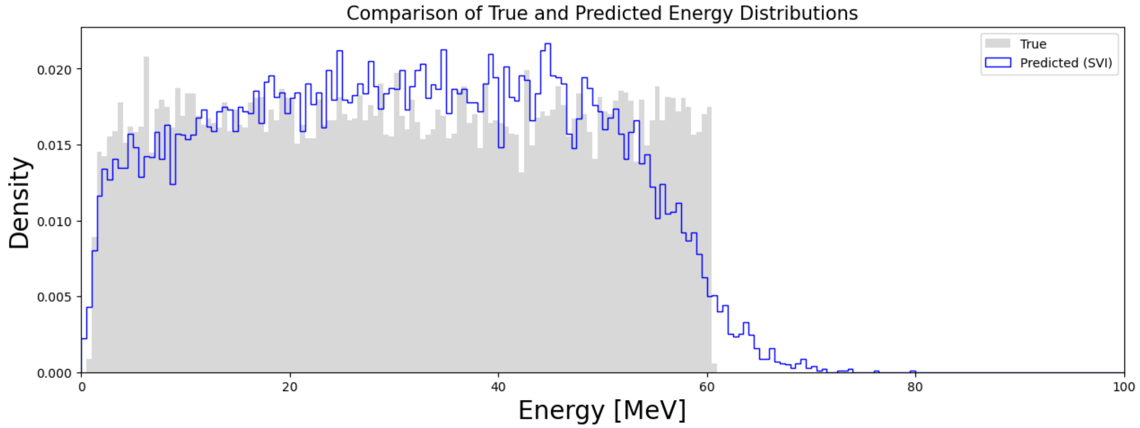




Example Results (input events)

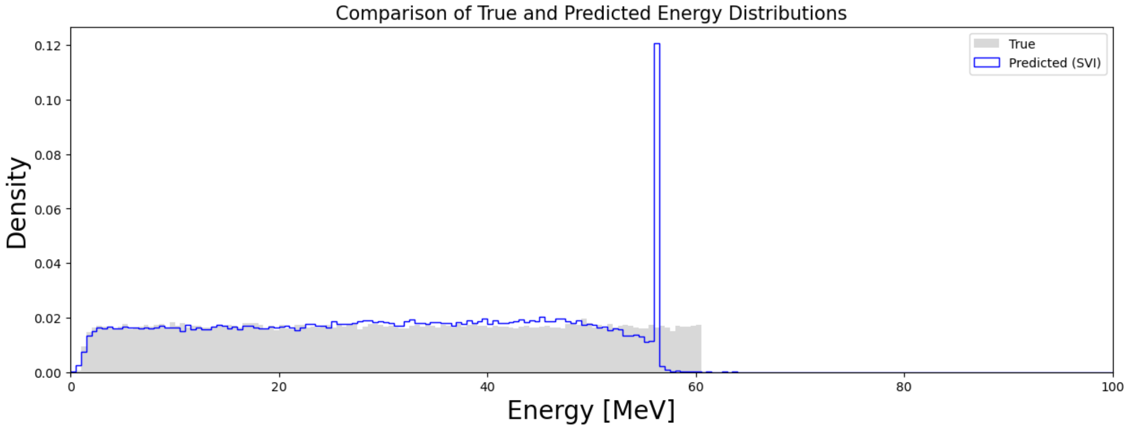
1

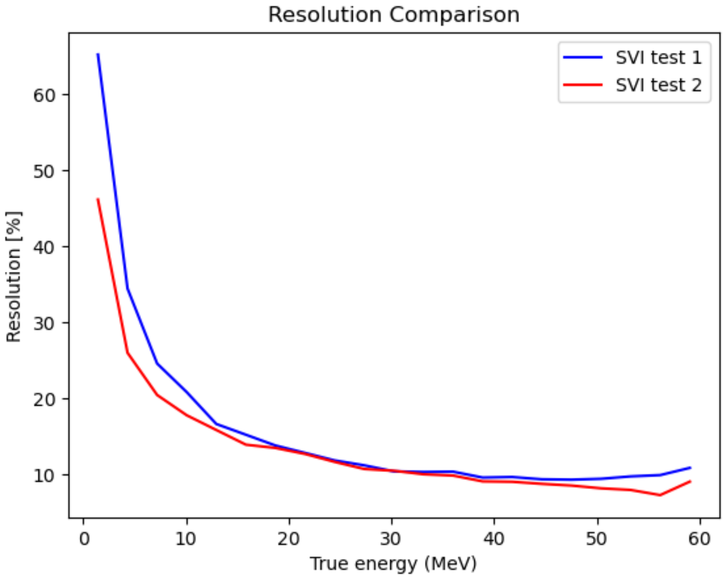
Input events=20480

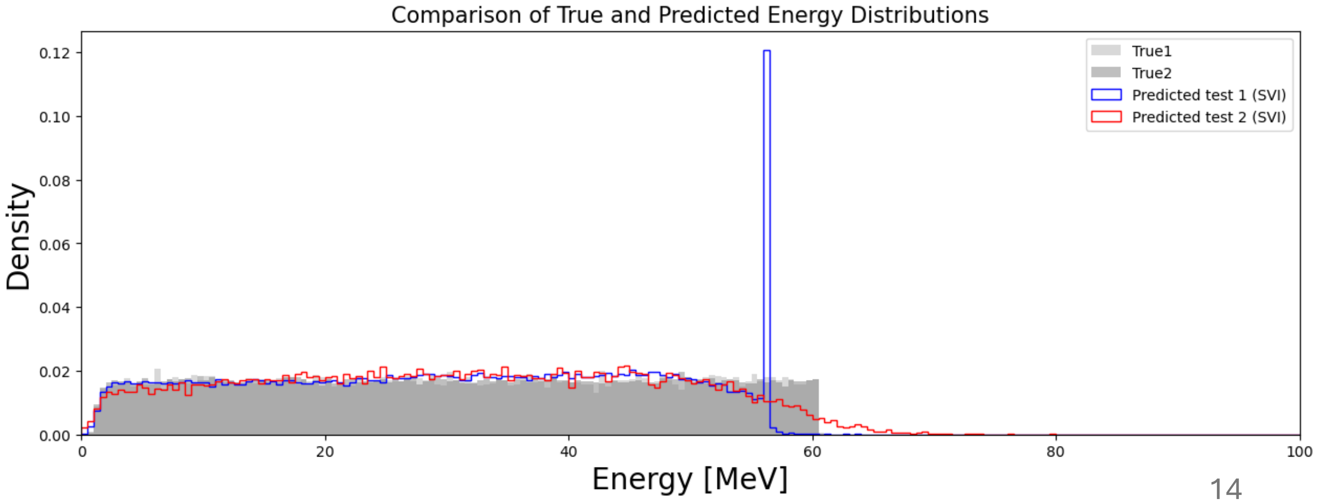


2

Input events=81920



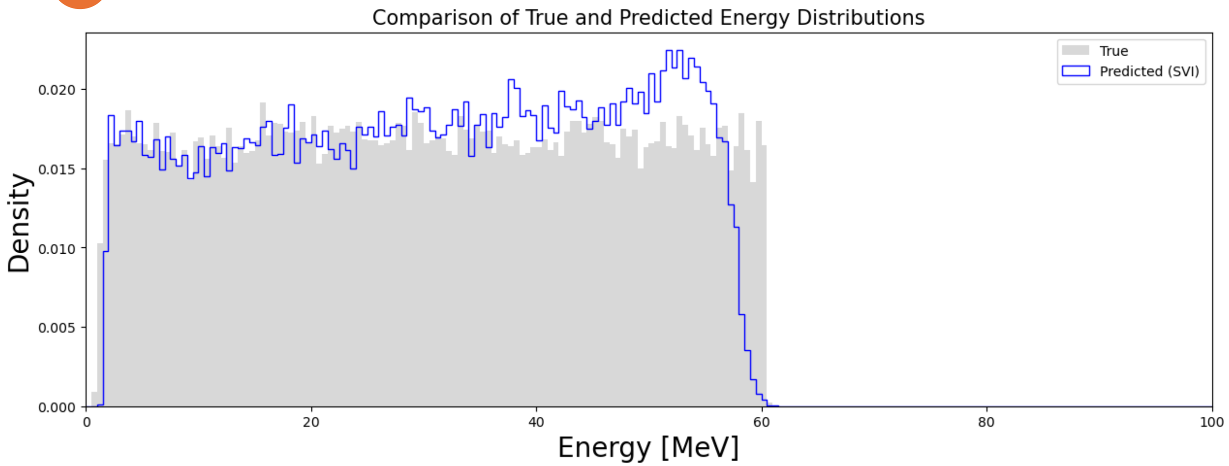




Example Results (neurons of the model)

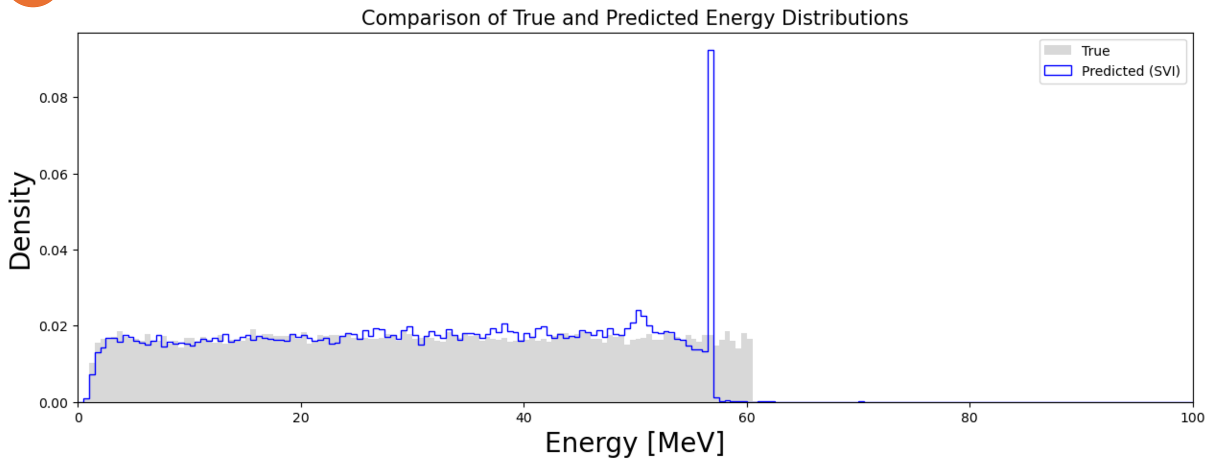
1

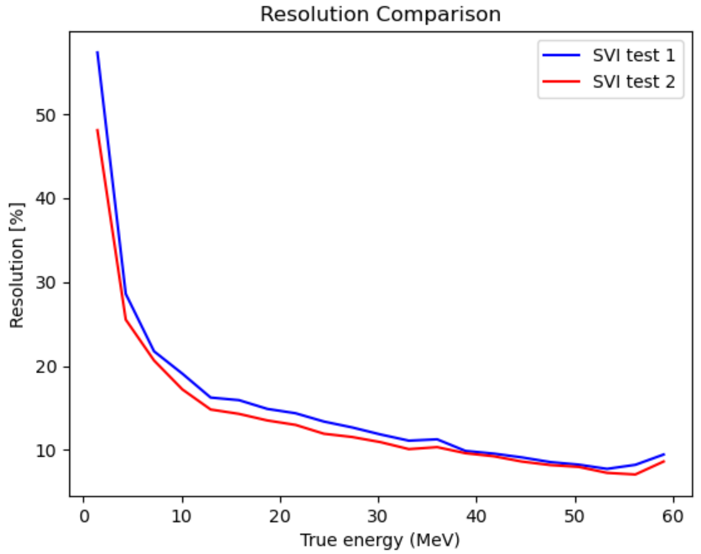
8 neurons



2

128 neurons

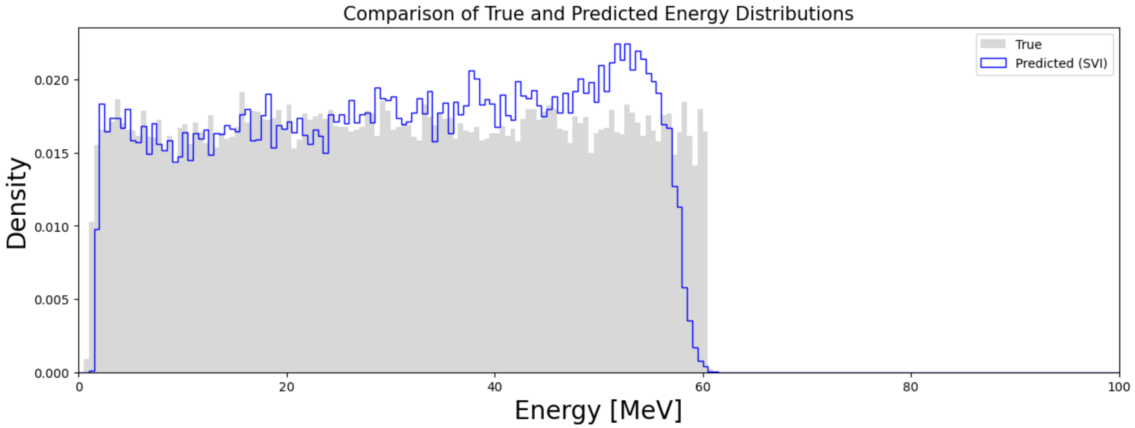




Example Results (All BNN/Last Layer)

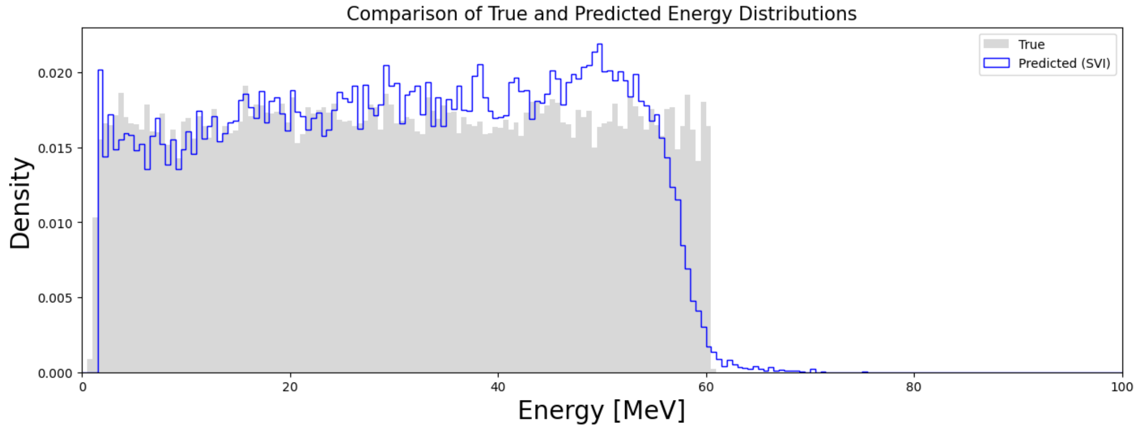
1

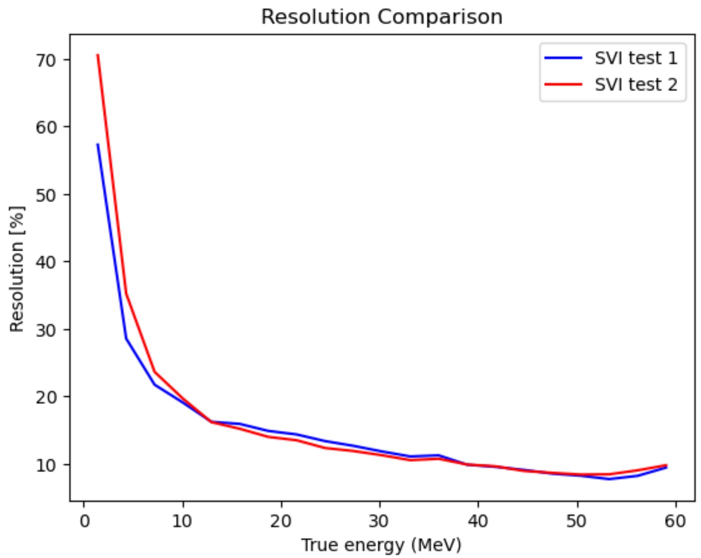
All layers are BNN

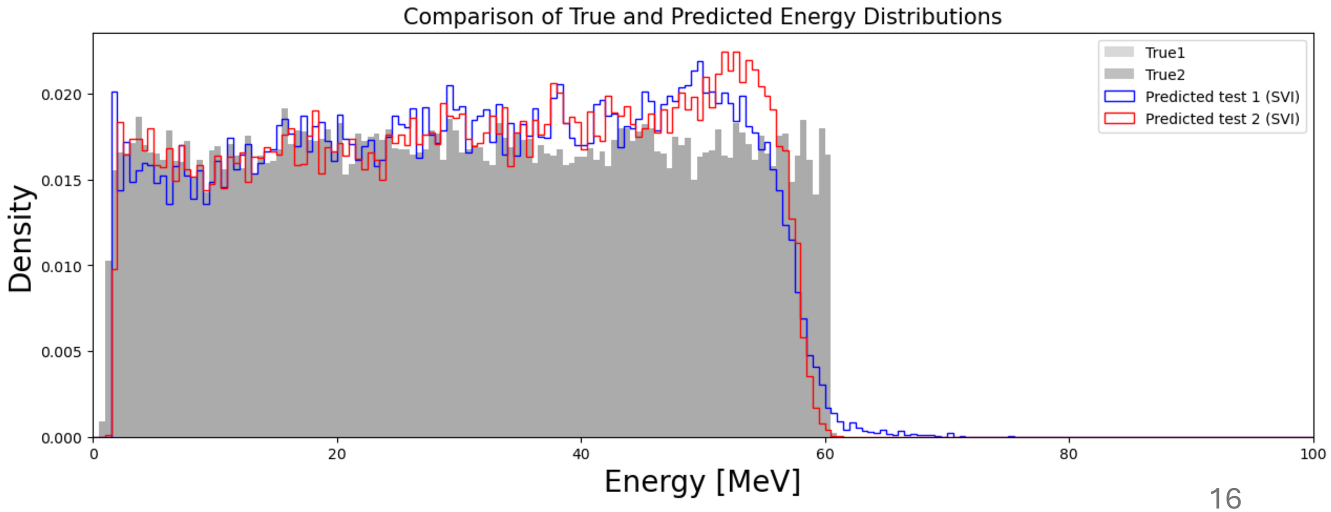


2

Only last layer BNN



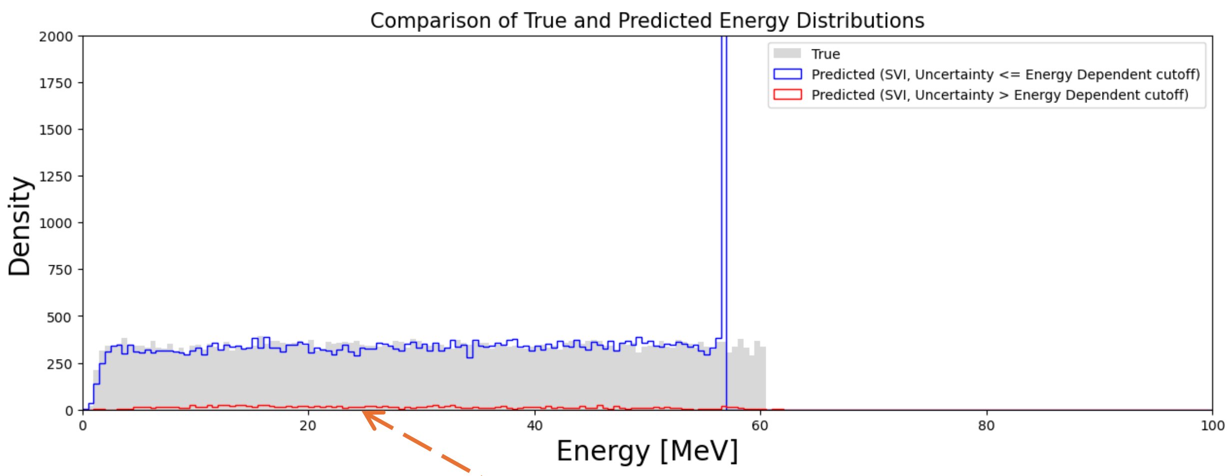




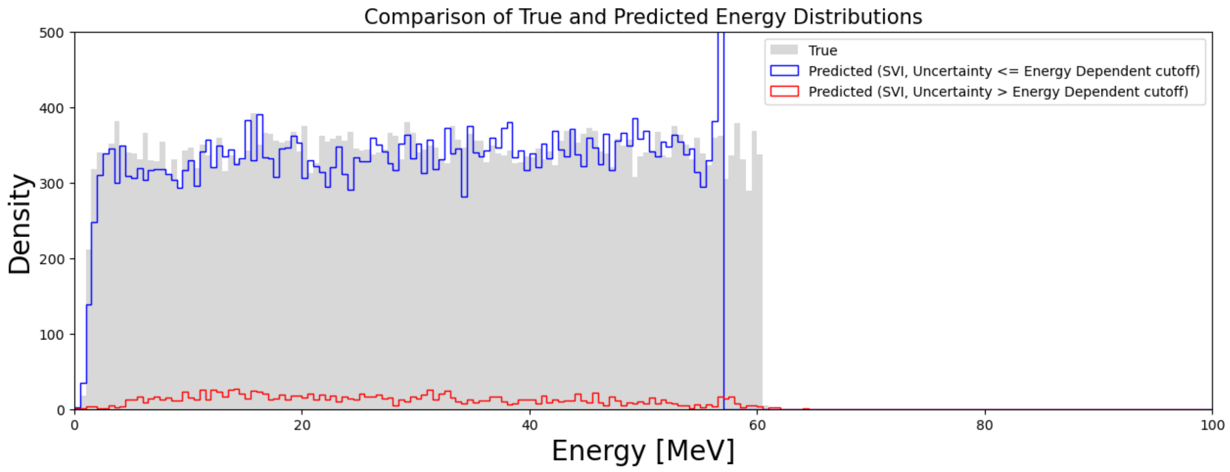
Best Results (More Neurons, set5, up to 60MeV)

1

All energy range

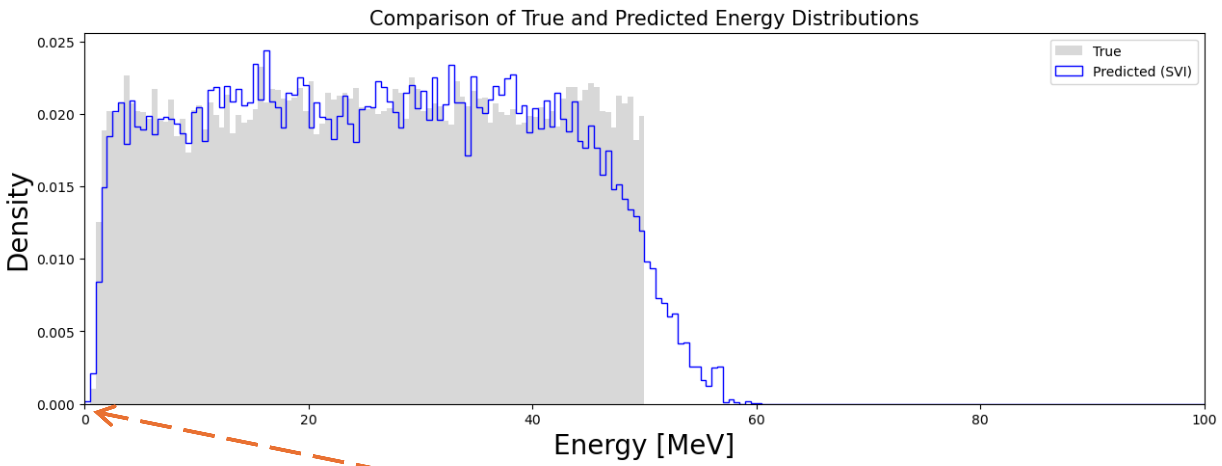


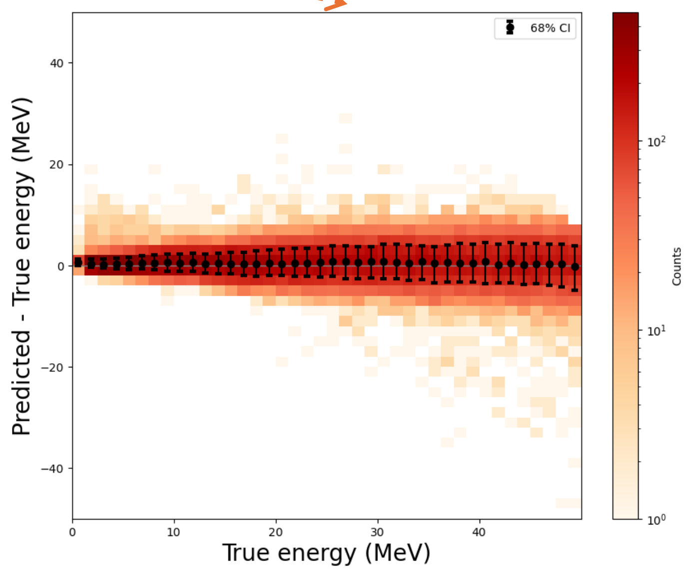
zoom



2

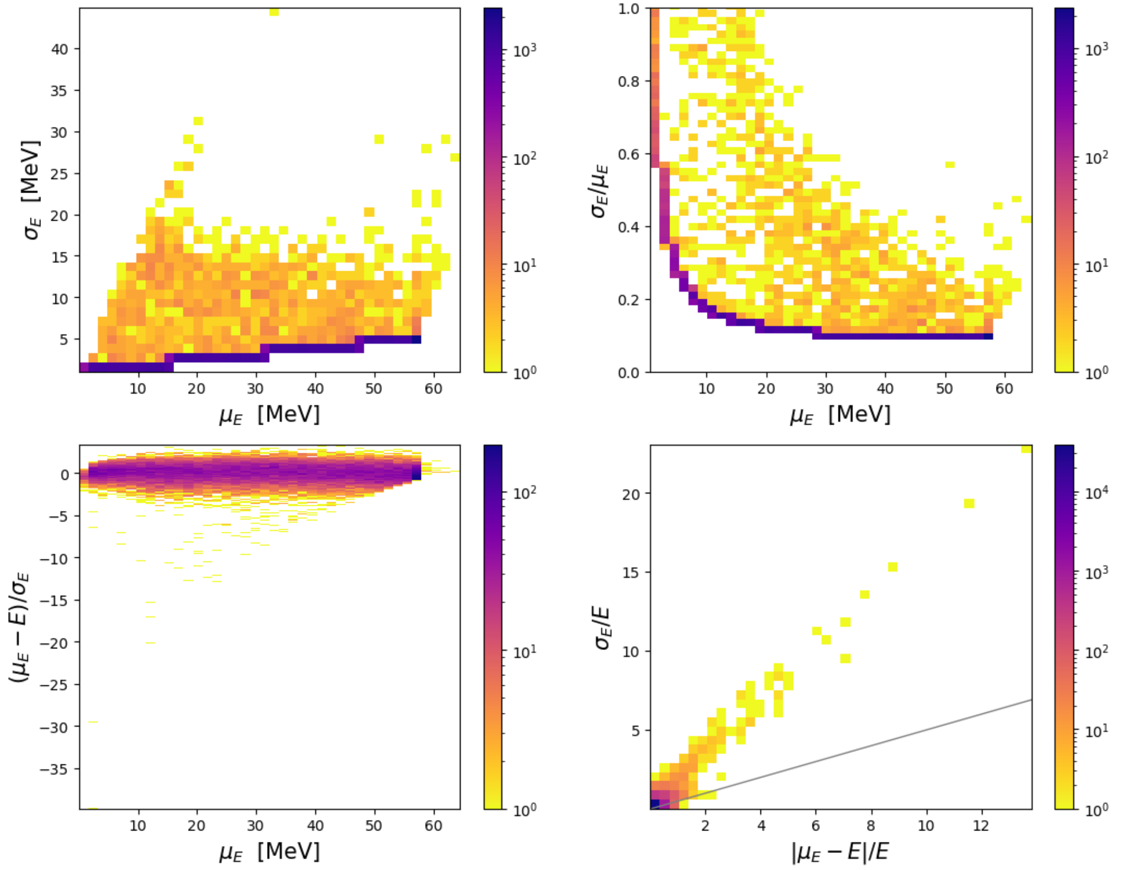
Cut energy range

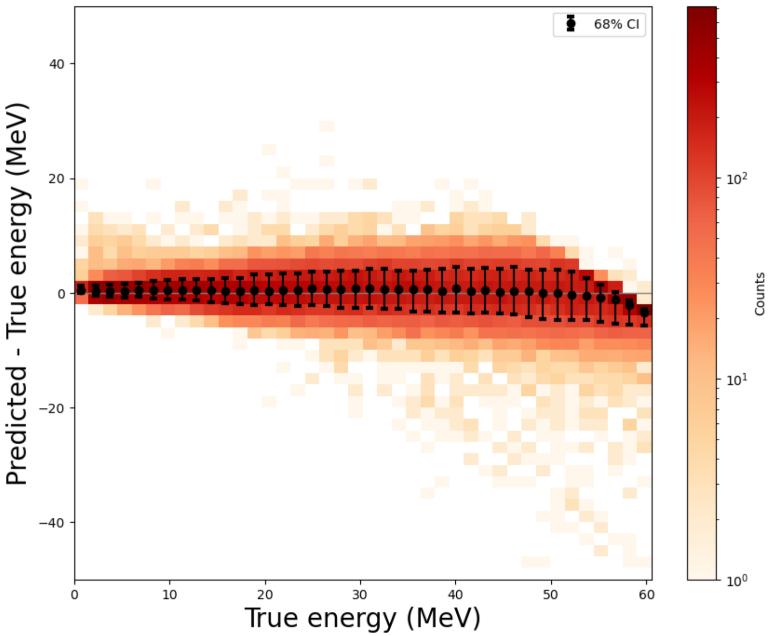


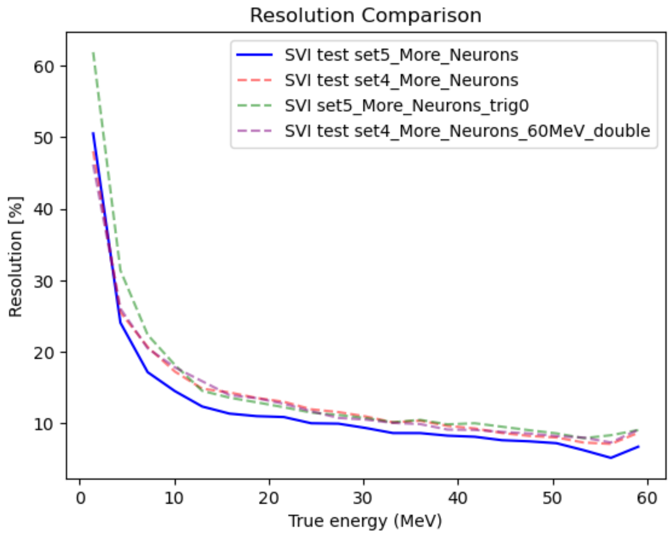


Best Results (More Neurons, set5, up to 60MeV)

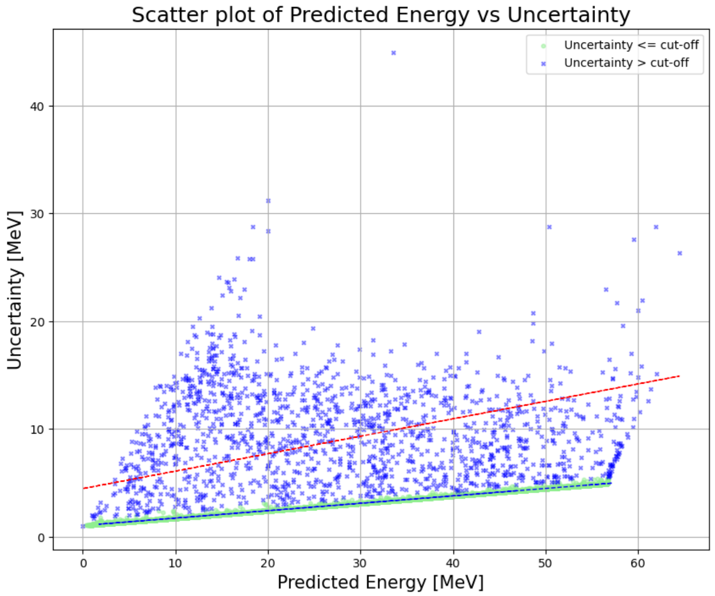
Heatmaps of Statistical Metrics for Energy Predictions

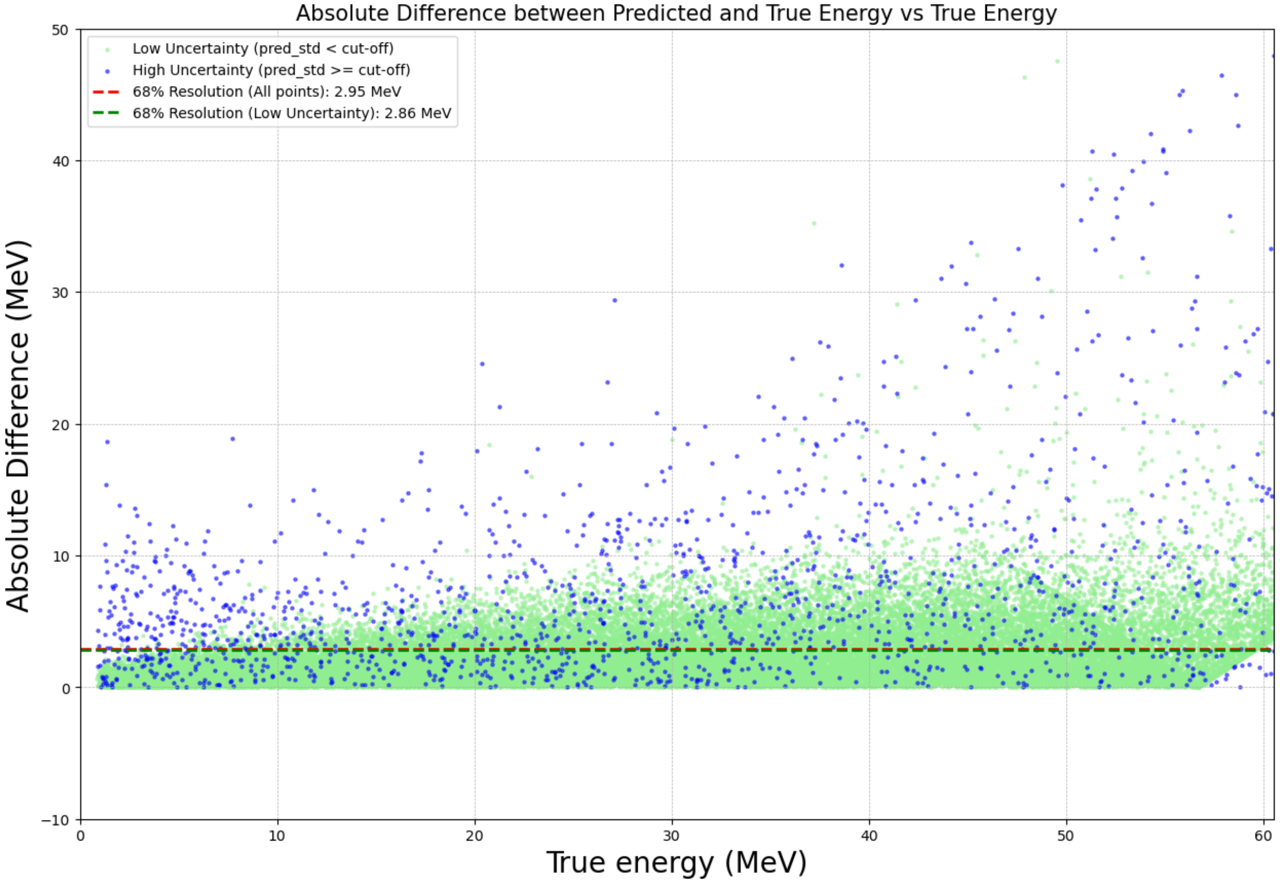


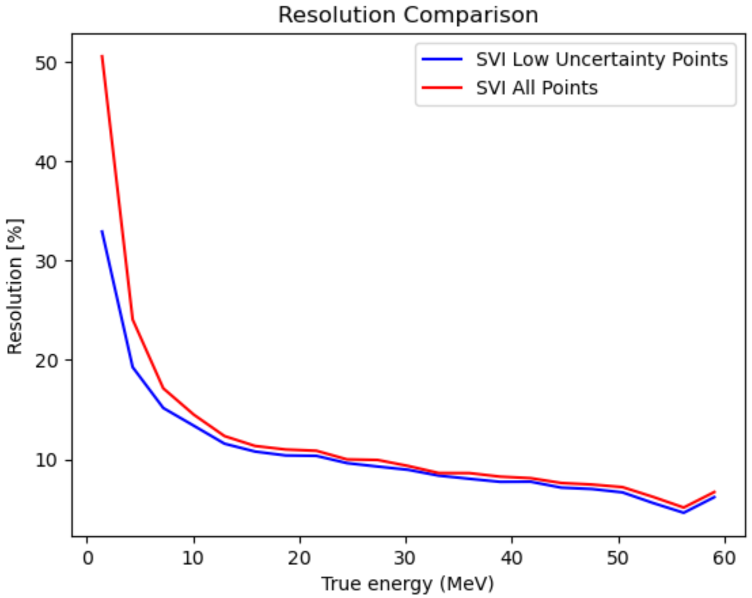




Best Results/uncertainty

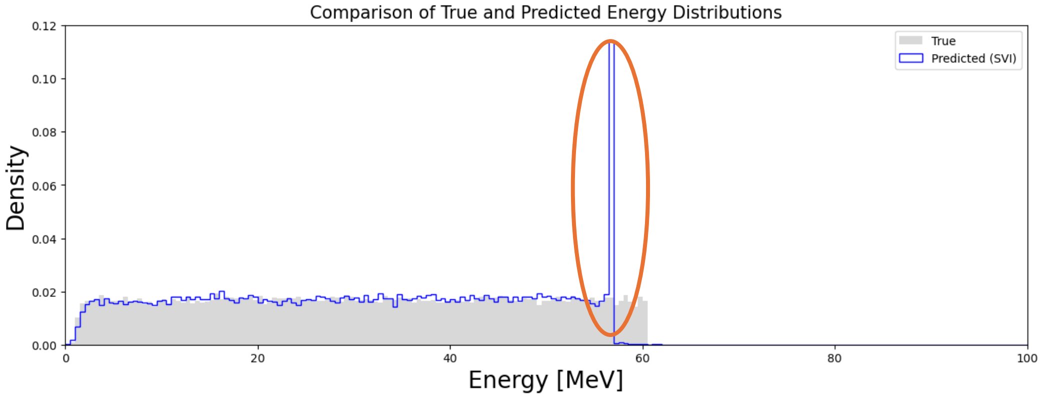




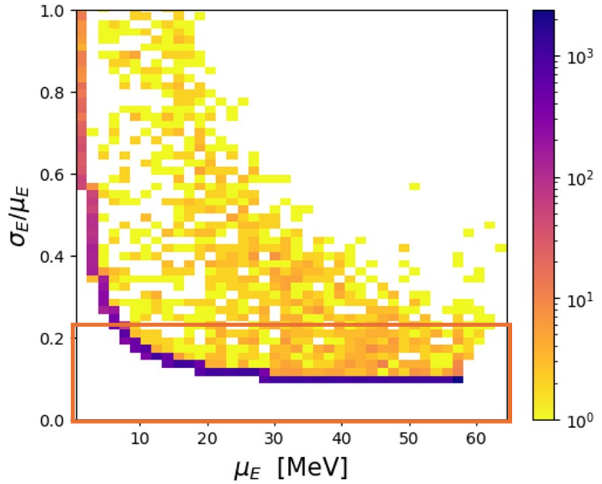


Perspectives

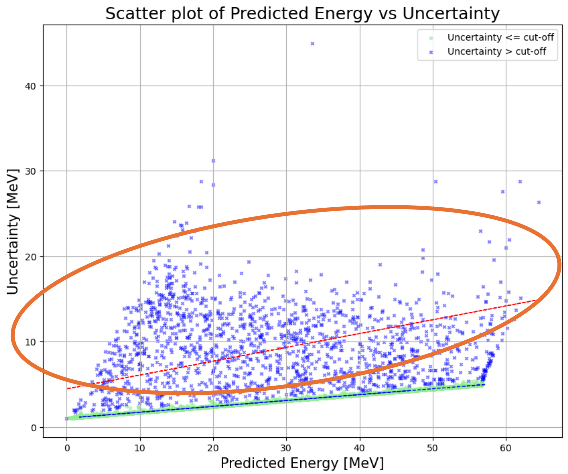
1 Investigate the peak



2 Make sure that the uncertainty reflects the difference true-pred



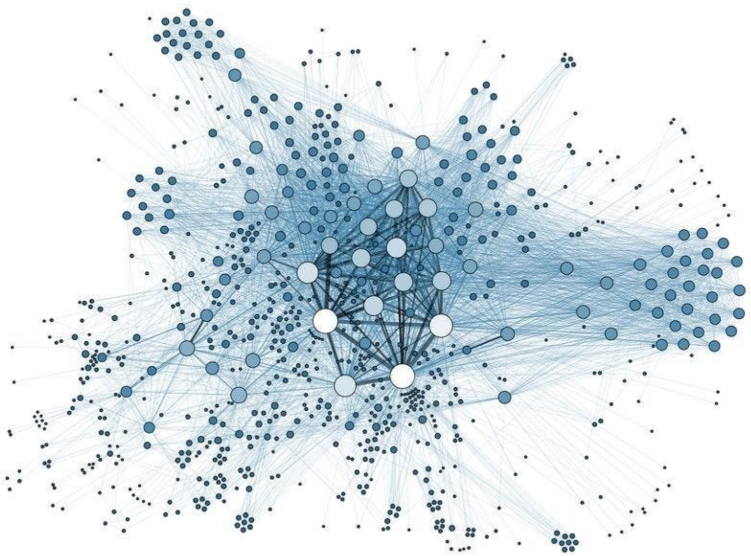
3 Investigate the badly reconstructed events



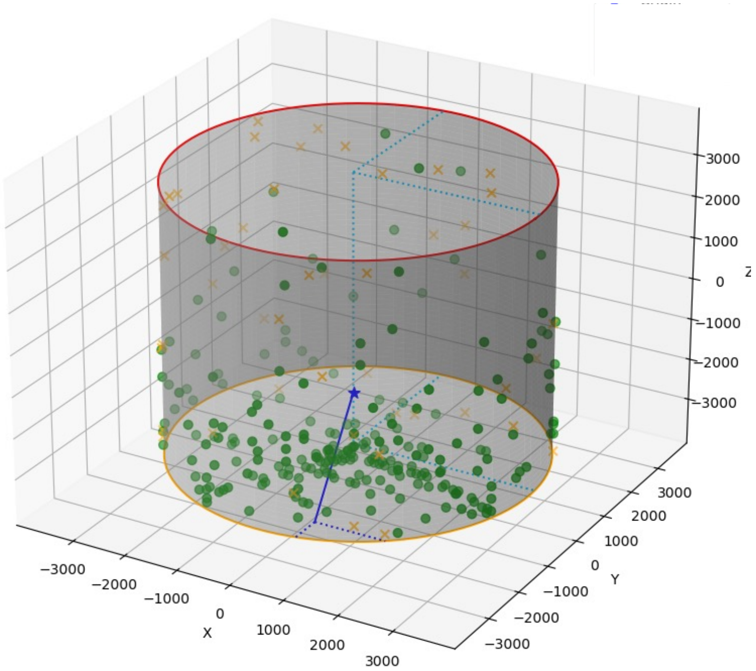
4 Add the n_eff variable

Clément :

Graphic neural network for vertex reconstruction

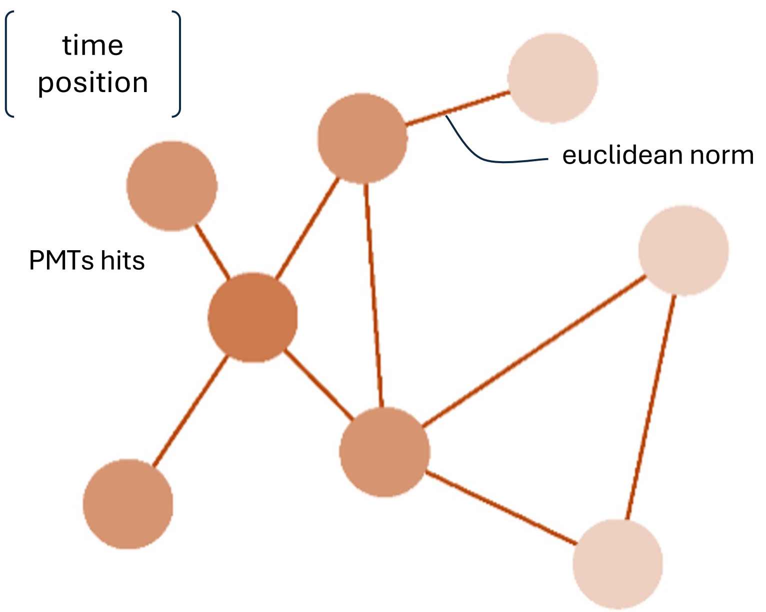


→

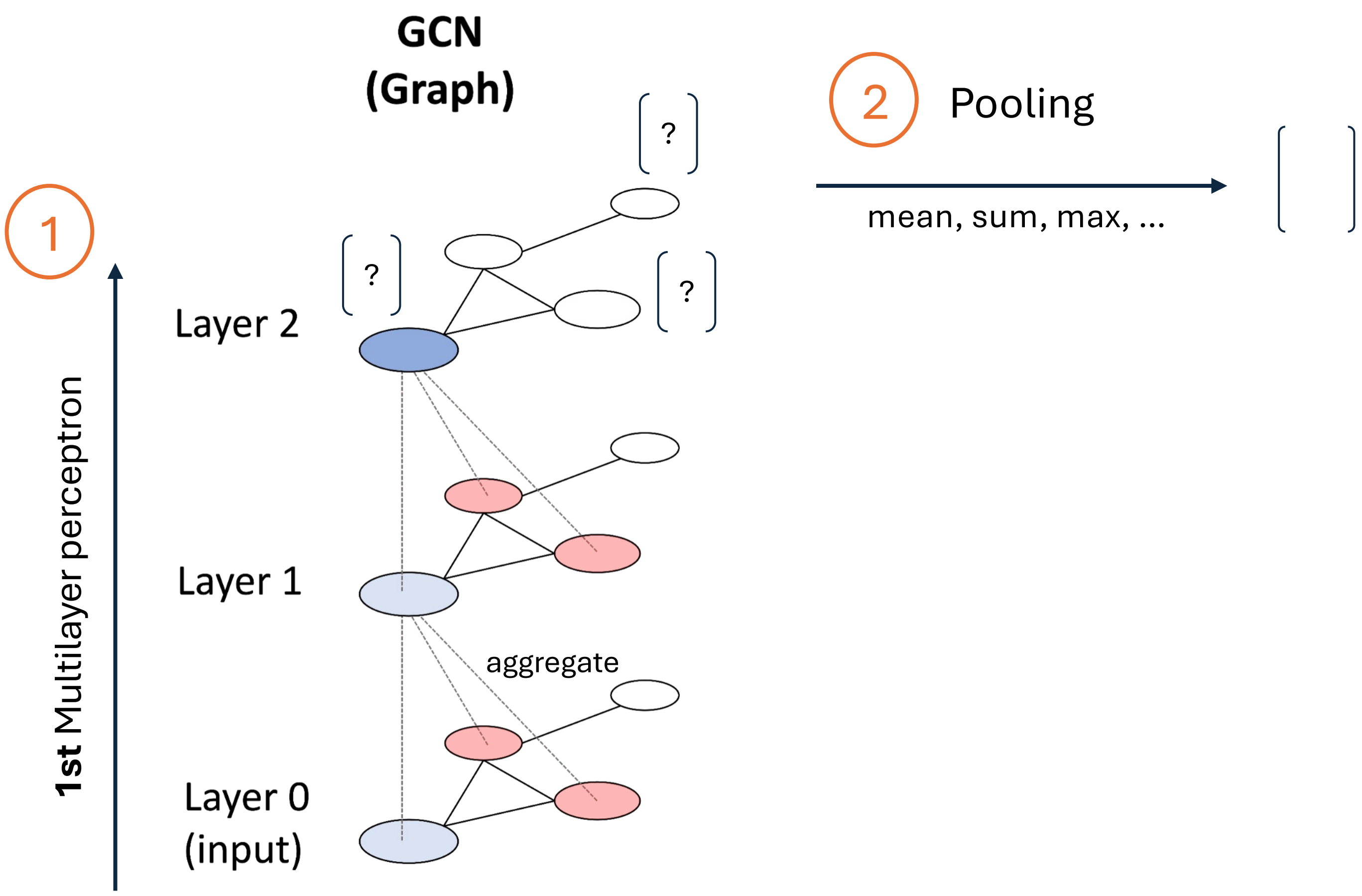


Graphic neural network

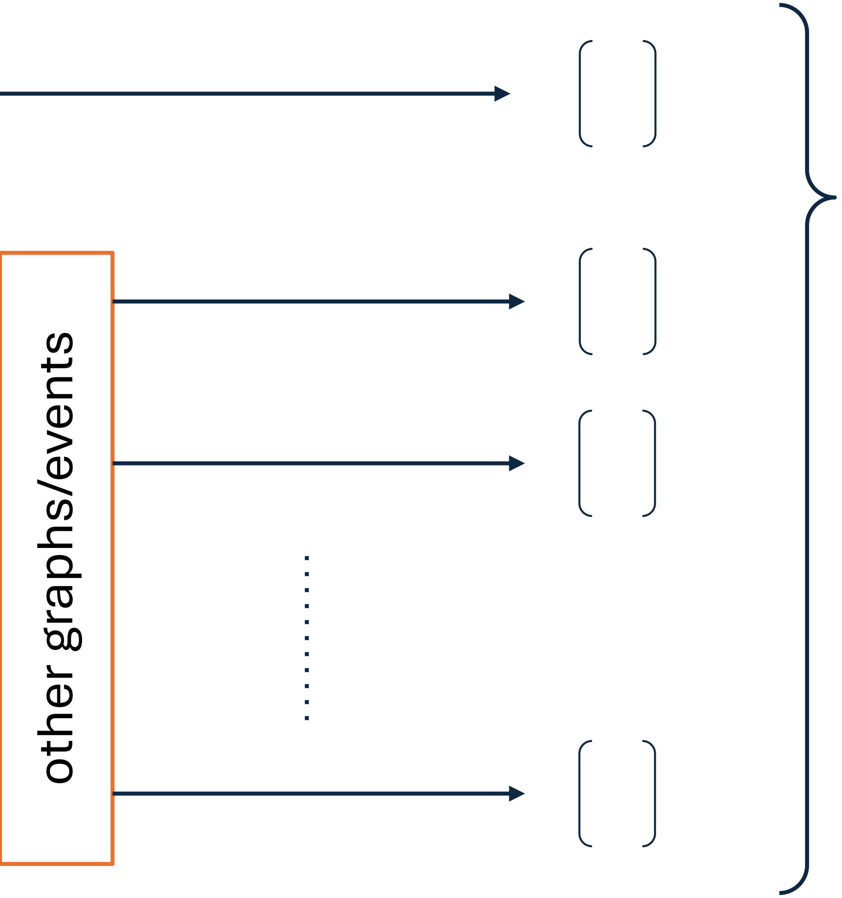
1 graph = 1 event



Graph data



Graphic neural network



3 **2nd** Multilayer perceptron $\xrightarrow{\text{predict}}$ Energy or **vertex**

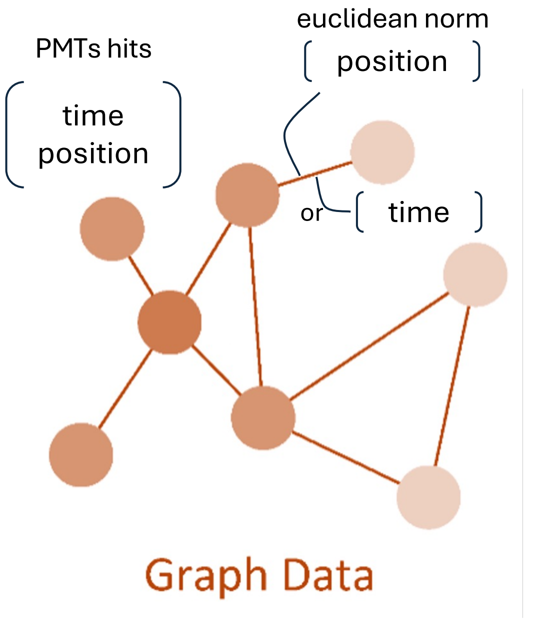
1 + 2 + 3 Training

4 Validation

5 Test

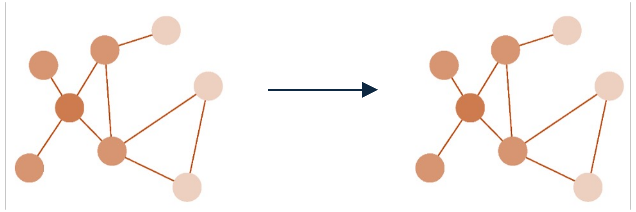
Model for vertex reconstruction (WatChMaL)

1 graph = 1 event



ResGatedConv layers

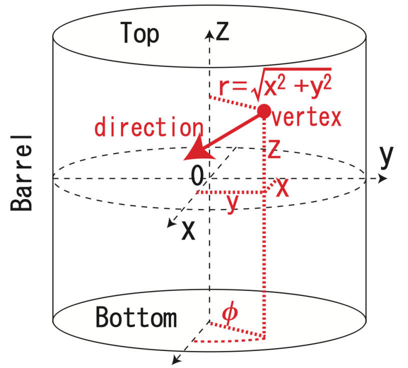
```
conv_in_channels: [32, 64]
```



32 features per nodes

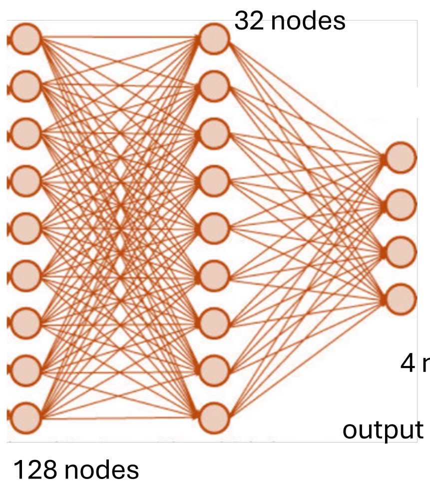
64 features per nodes

Pooling



Linear layers

```
linear_out_features: [128, 32, 4]
```



vertex coord

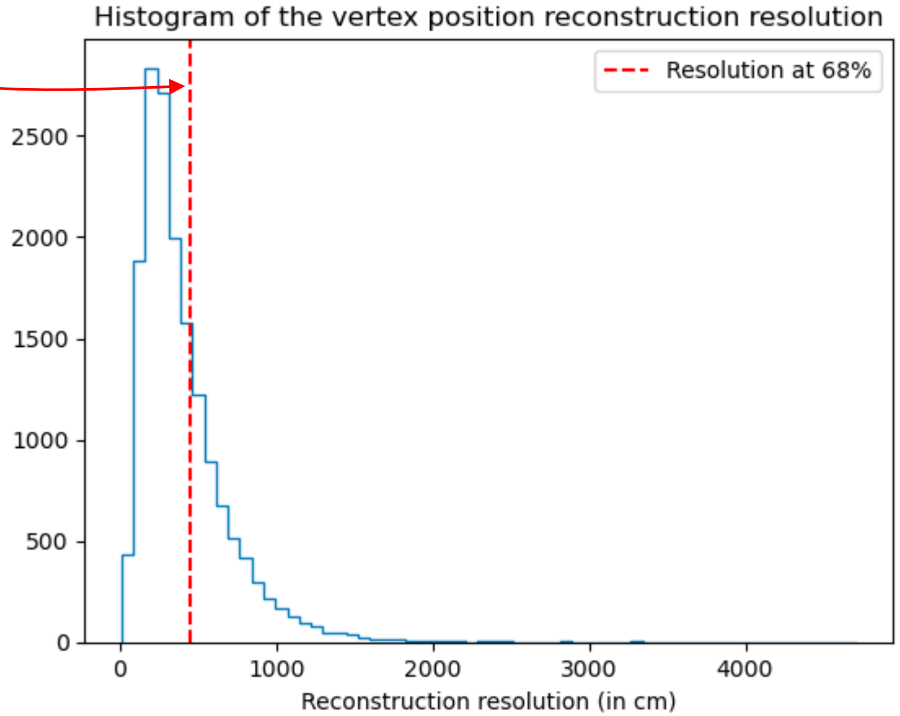
4 nodes = output prediction

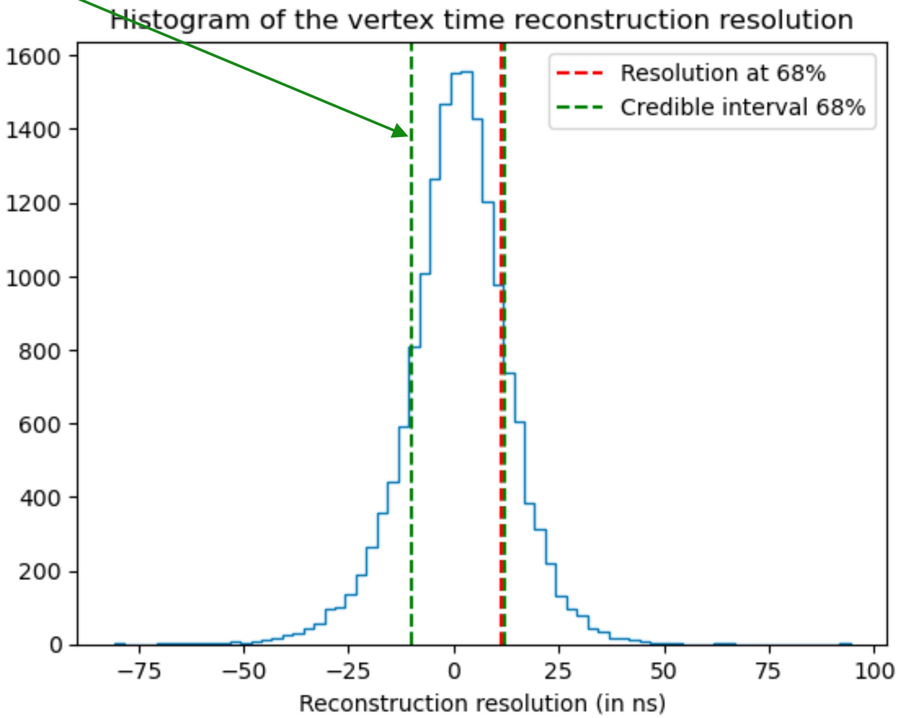
Results

Resolution at 68% for:

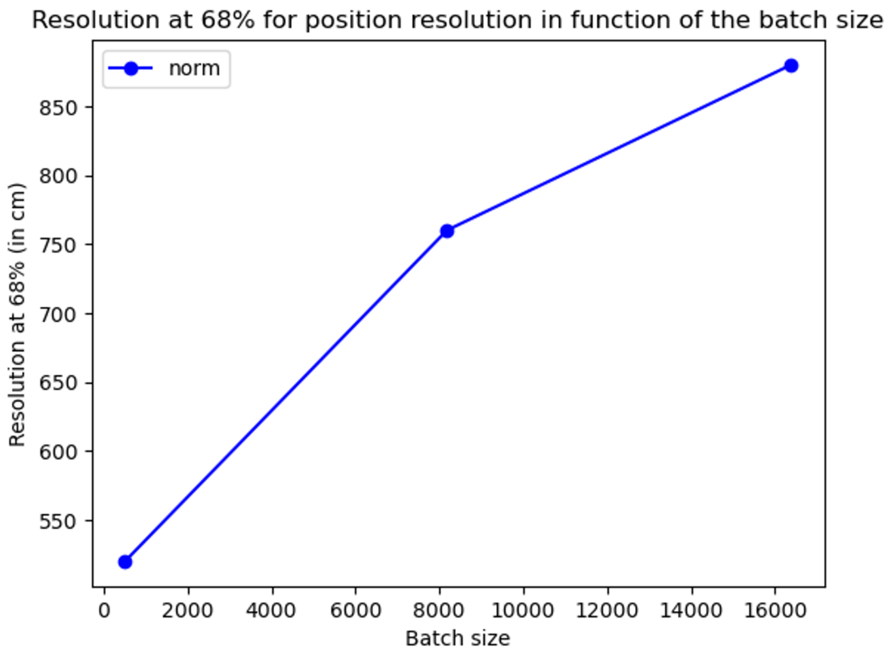
- on the position norm: 446.74 cm
- on the direction projection: 242.53 cm
- orthogonal to the direction projection: 334.34 cm
- on time: 11.14 ns

Credible interval on time at 68%: -9.99 ns to 12.01 ns





Parameters exploration



<u>Nohitc VS hitc :</u>

608 cm VS 571cm

Charge VS Max Charge VS Log Charge :

783cm/16ns VS 781cm/63ns VS 1124cm/23ns

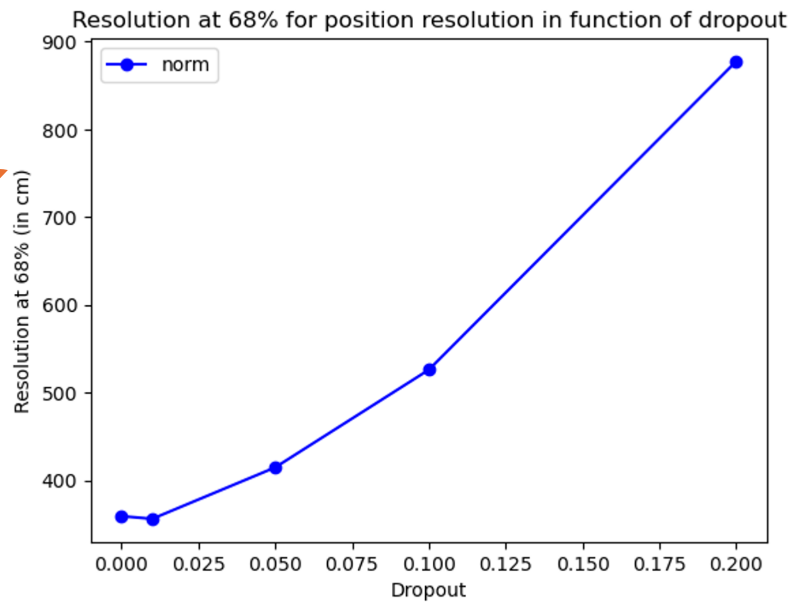
No noise VS noise :

357cm VS 672cm

only 50k events

Dropout 0 VS 0.1 :

608 cm VS 619 cm



<u>40 epochs VS 100 epochs :</u>

608 cm VS 495 cm

ResgatedConv VS Conv :

783 cm VS 825 cm

MSE VS Weighted MSE :

452 cm/15ns VS 447 cm/11ns

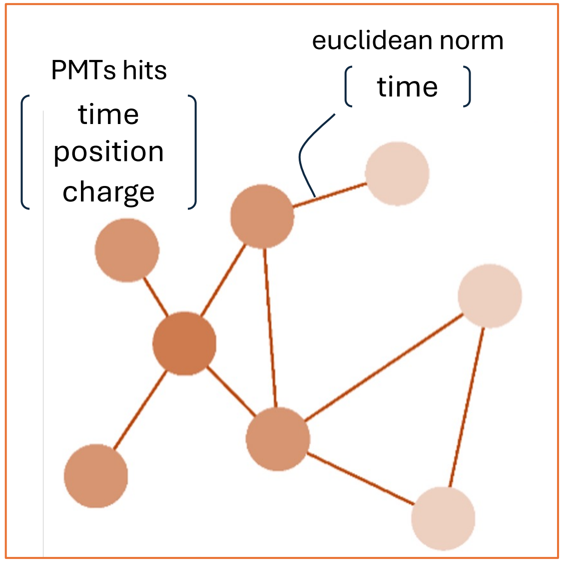
$$\begin{pmatrix}1\\1\\1\\\text{weight}\end{pmatrix} \times \begin{pmatrix}\text{hitx}\\\text{hity}\\\text{hitz}\\\text{hitt}\end{pmatrix}$$

$$\downarrow$$

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

Results

300k events



Configuration

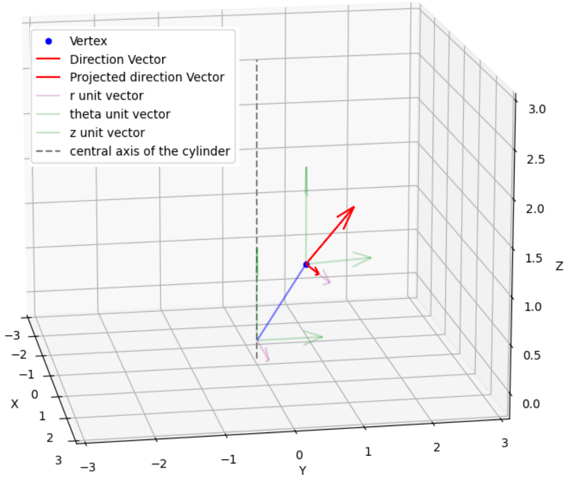
Model : same than before
Epochs : 40
Batchsize : 516
Loss : MSE
Dropout : 0

Best results so far

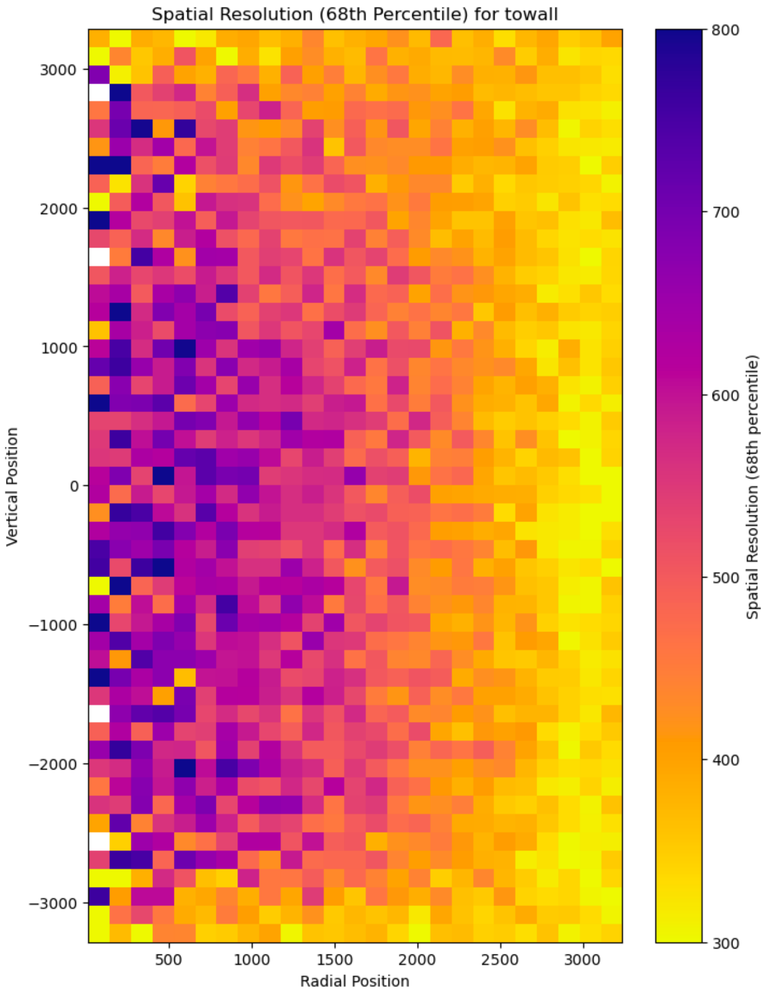
Resolution at 68% for:
- on the position norm: 428.81 cm
- on the direction projection: 252.55 cm
- orthogonal to the direction projection: 324.52 cm
- on time: 11.77 ns

Credible interval on time at 68%: -12.63 ns to 10.86 ns

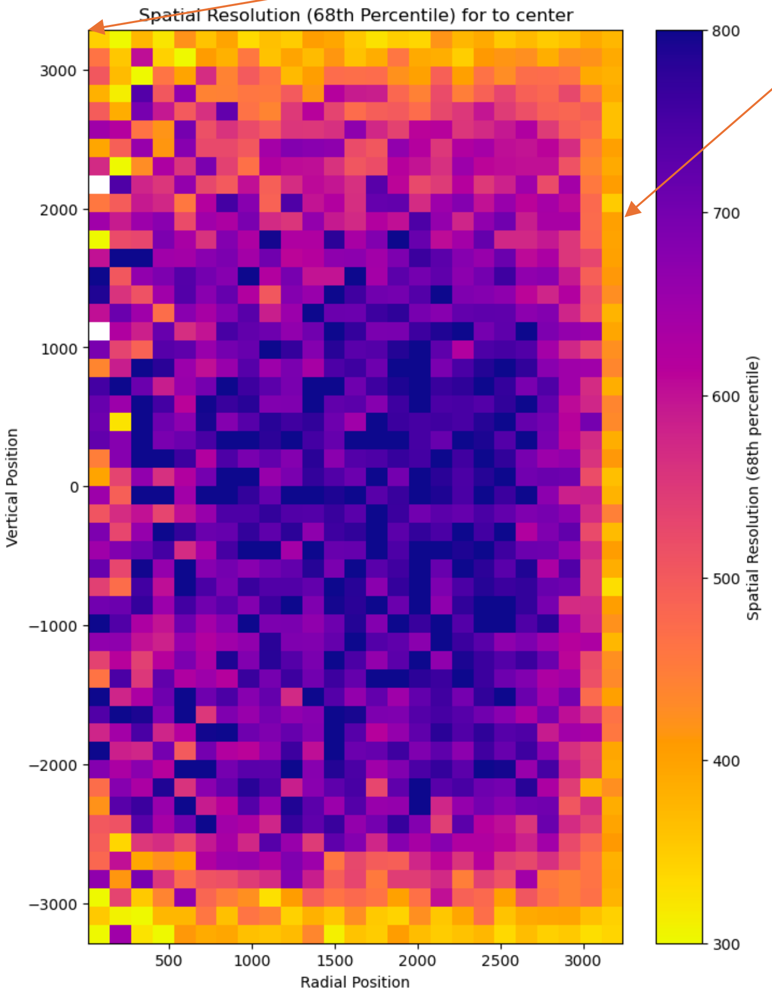
Results



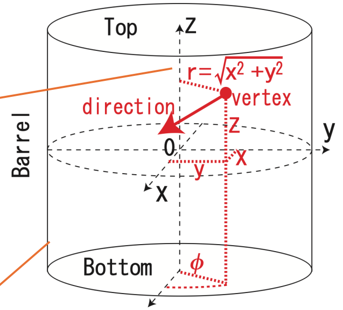
Vertex
Direction Vector
Projected direction Vector
r unit vector
theta unit vector
z unit vector
central axis of the cylinder
X
Y
Z



Spatial Resolution (68th Percentile) for towall
Vertical Position
Radial Position
Spatial Resolution (68th percentile)

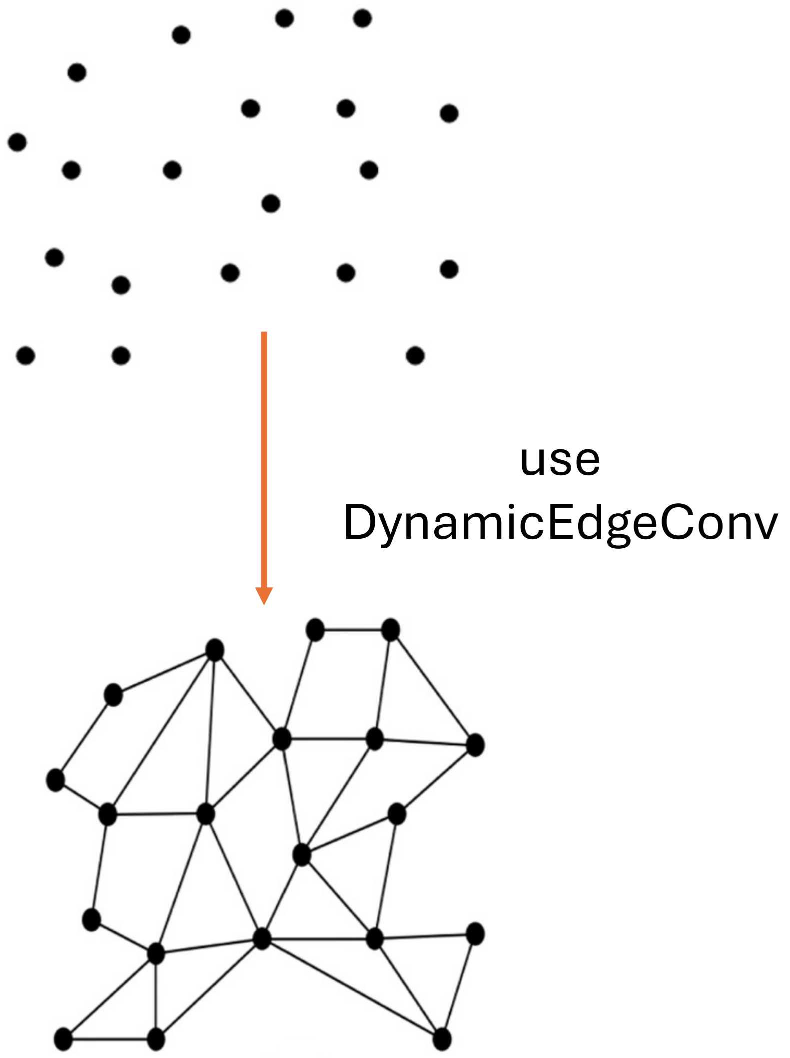


Spatial Resolution (68th Percentile) for to center
Vertical Position
Radial Position
Spatial Resolution (68th percentile)



Top
Z
$r=\sqrt{x^2+y^2}$
vertex
direction
Barrel
0
y
x
Bottom
ϕ

Perspectives



→ investigate different models and hyperparameters to improve upon traditional algorithm

→ implement Bayesian neural network

→ merge Anthi's work with mine to have a fully reconstructed vertex

Thank you for your attention !