UNIVERSITÀ DEGLI STUDI
DI GENOVA

Istituto Nazionale di Fisica Nucleare
Sezione di Genova

CaloMan:
Fast generation of calorimeter showers with density estimation on learned manifolds

Humberto Reyes-González
University of Genoa (DiFi UniGe)

In collab with:
M. Letizia (MaLGa UniGe), J. Cresswell, B. Ross, G. Loaiza-Ganem, A. Caterini (Layer 6 AI)

Calorimeter Challenge workshop
Frascati, 30/5/2023

Introduction.

- CHALLENGE: Develop fast and accurate ML models for high dimensional calorimeter shower generation.
- OUR APPROACH: Density estimation on learned Manifolds with CaloMAN arXiv:2211.15380.
- WE PRESENT RESULTS ON: **Photons1** and **Pions1** datasets.

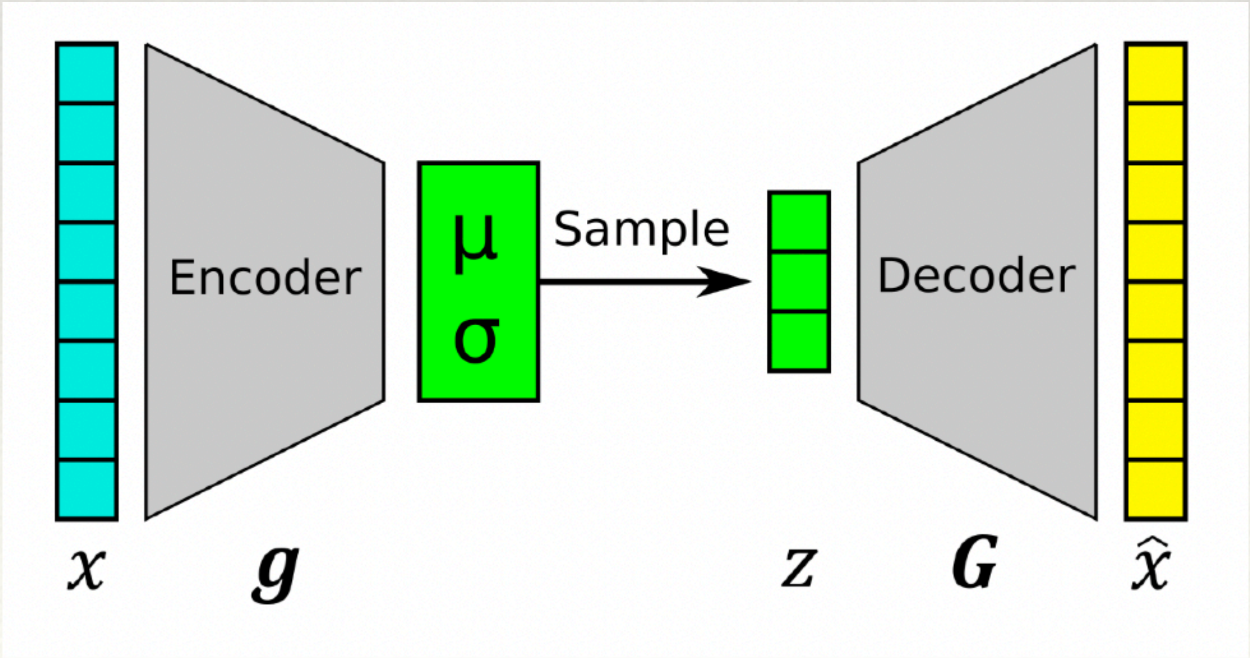
Generative Networks.

Generalized auto encoders (GAE)

- Implicitly learns PDF
- Useful only for sampling
- Map data to a latent space
- Not so stable training
- Efficient sampling in high dimension

Examples:

- Autoencoders (AE)
- Variational Autoencoders (VAE)
- Generative Adversarial Networks (GAN)
- …

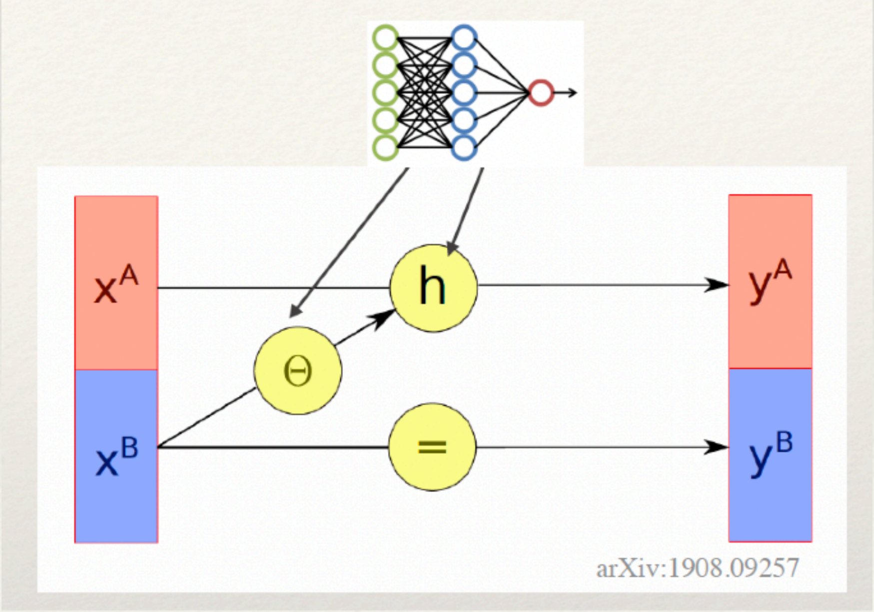


Density estimators (DE)

- Explicitly learns PDF
- Mappings can be bijective
- Usually, more stable learning
- Learnable, but very heavy models in HD.

Examples:

- Normalizing Flows (NF)
- Score-based models
- Diffusion models
- …



CAN WE TAKE THE BEST OF EACH OF THEM?

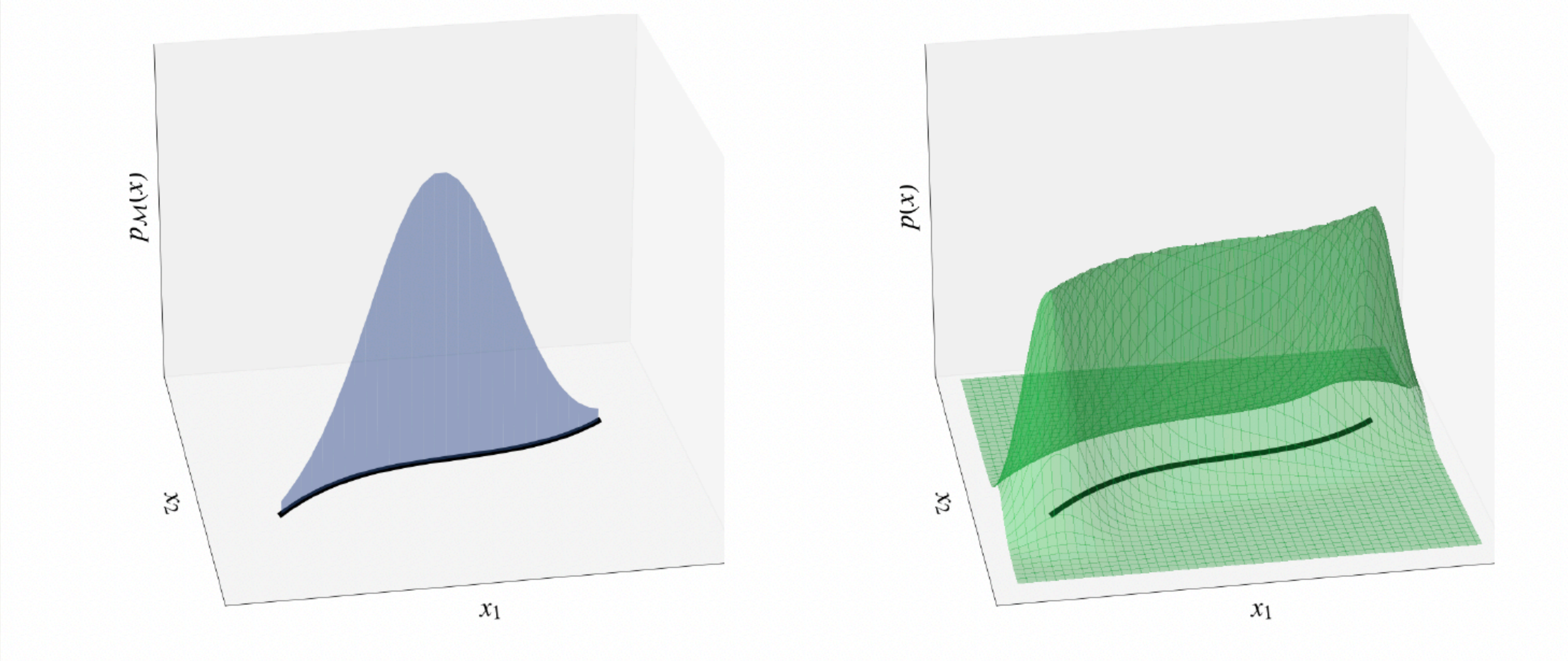
Manifold Hypothesis

Manifold Hypothesis states that high dimensional real-world data is supported in a low dimensional sub-manifold $\mathscr{M} \subset \mathbb{R}^D$

In principle, calorimeter showers are governed by simple laws of physics and must have a much lower dimensional structure.

Manifold overfitting arXiv:2204.0717 :
When trying to model a target distribution $\mathbf{T}$, suported on $\mathscr{M}$, with a DE that learns $p_\theta(x)$ on $\mathbb{R}^D$, MLE can fail when the dimensions of $\mathbf{T}$ and $p_\theta(x)$ differ.

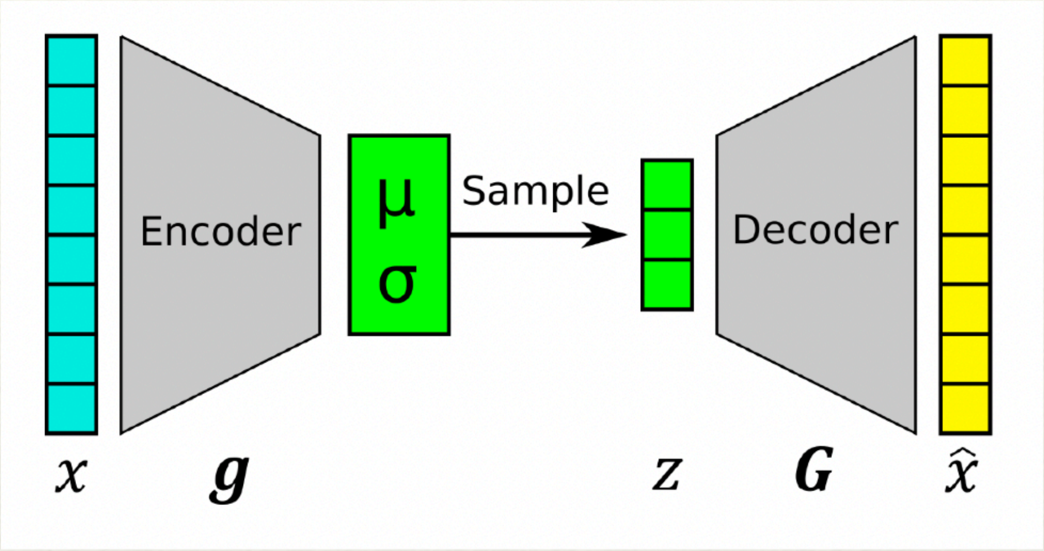
Solution:
First learn the data $\mathscr{M}$ and estimate the distribution on $\mathscr{M}$



Two-step models

STEP 1: Learn $\mathscr{M}$ with a generalized autoencoder.
This may be an **AE**, **VAE**, GAN, Wasserstein AE, bi-GAN, etc.

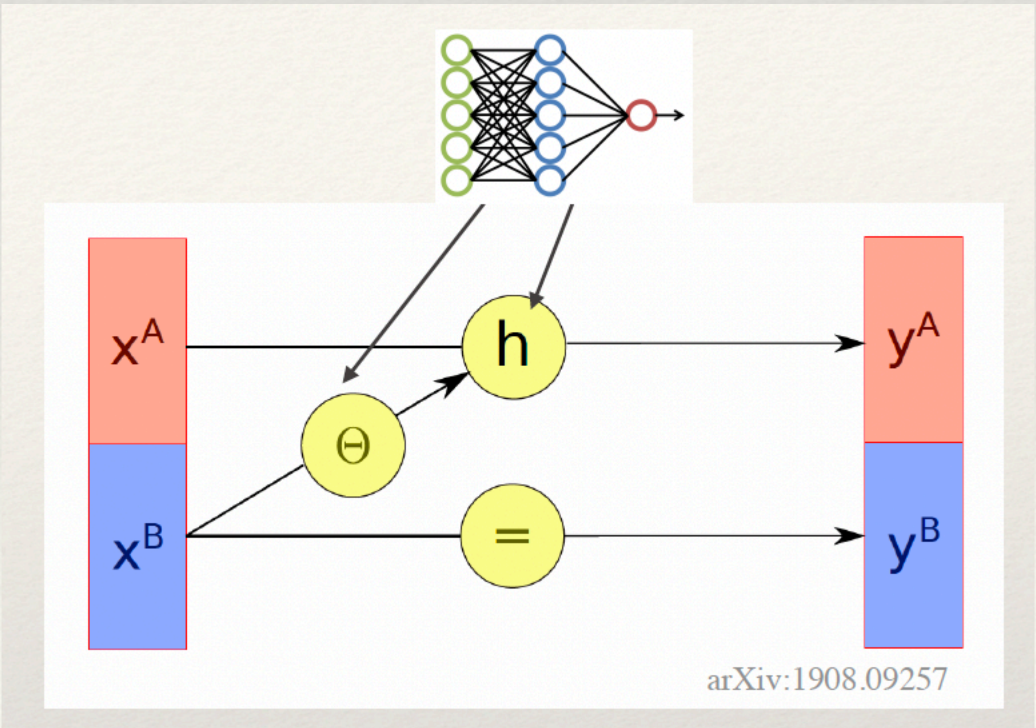
Here we use



+

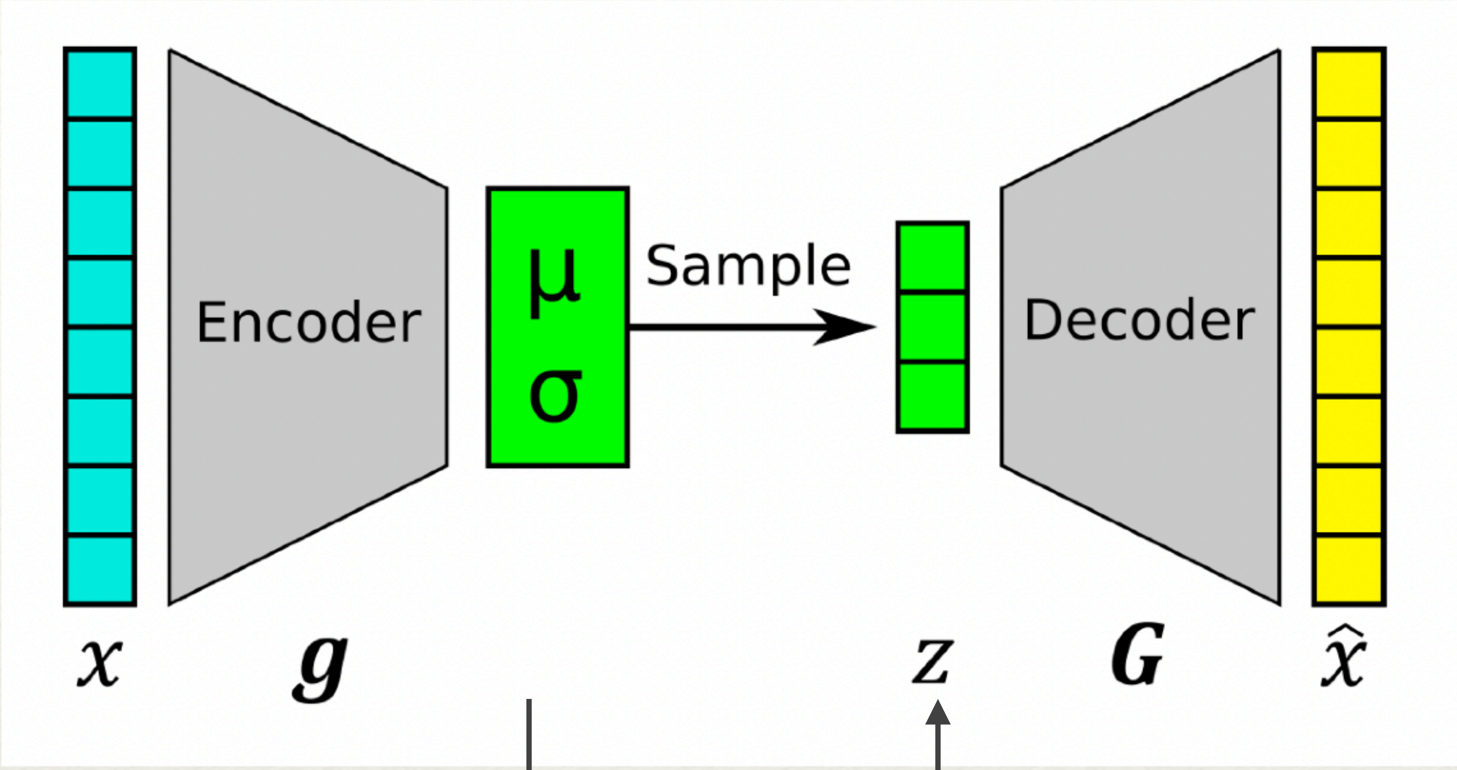
STEP 2: Perform density estimation on the manifold, with **NFs**, autoregressive, score-based, diffusion models

Here we use (Coupling) NFs



Two-step models: Chosen scheme

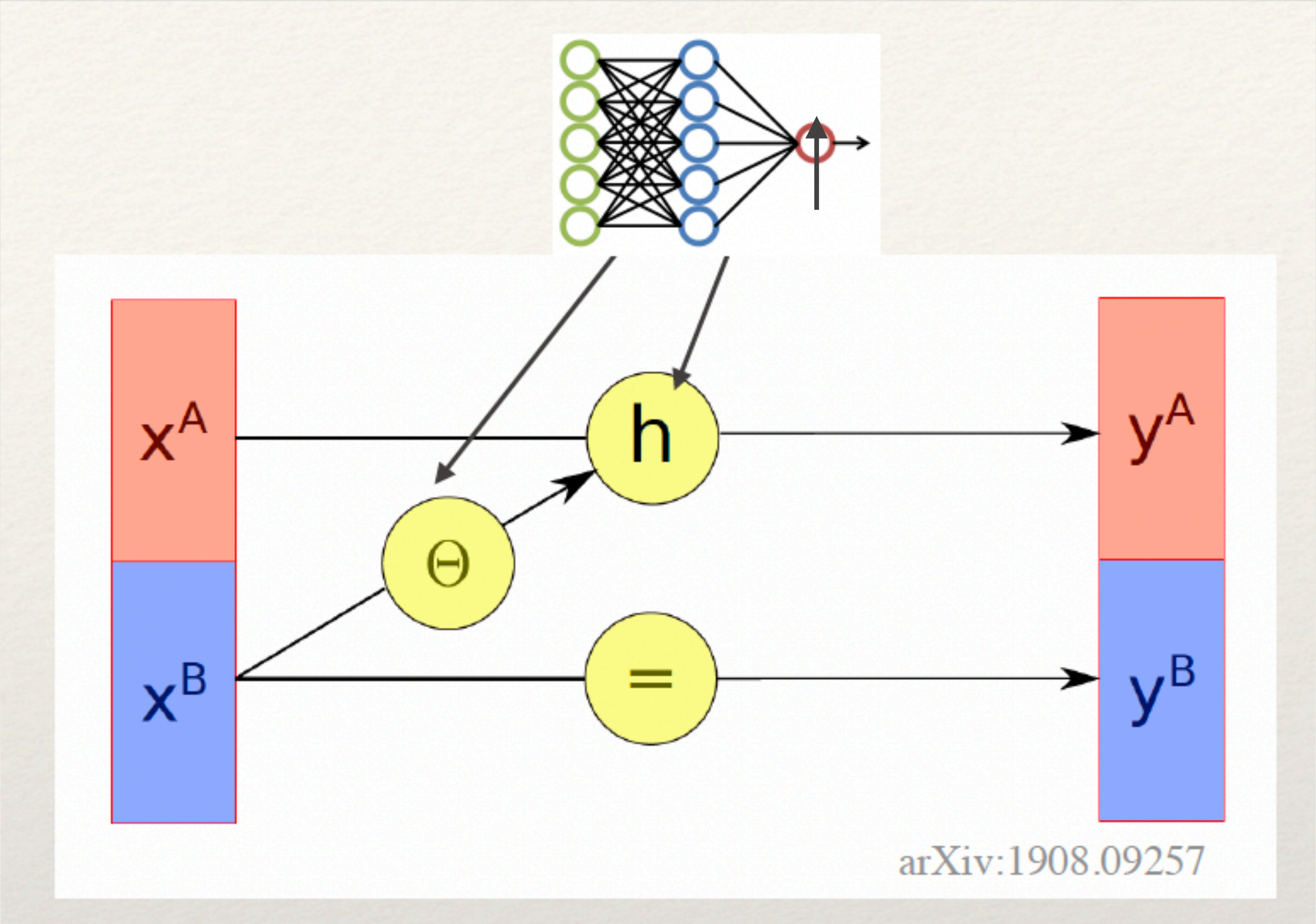
INPUT



OUTPUT

VAE or AE

CONDITIONER.



Coupling Flow.

Estimating latent space dimensionality

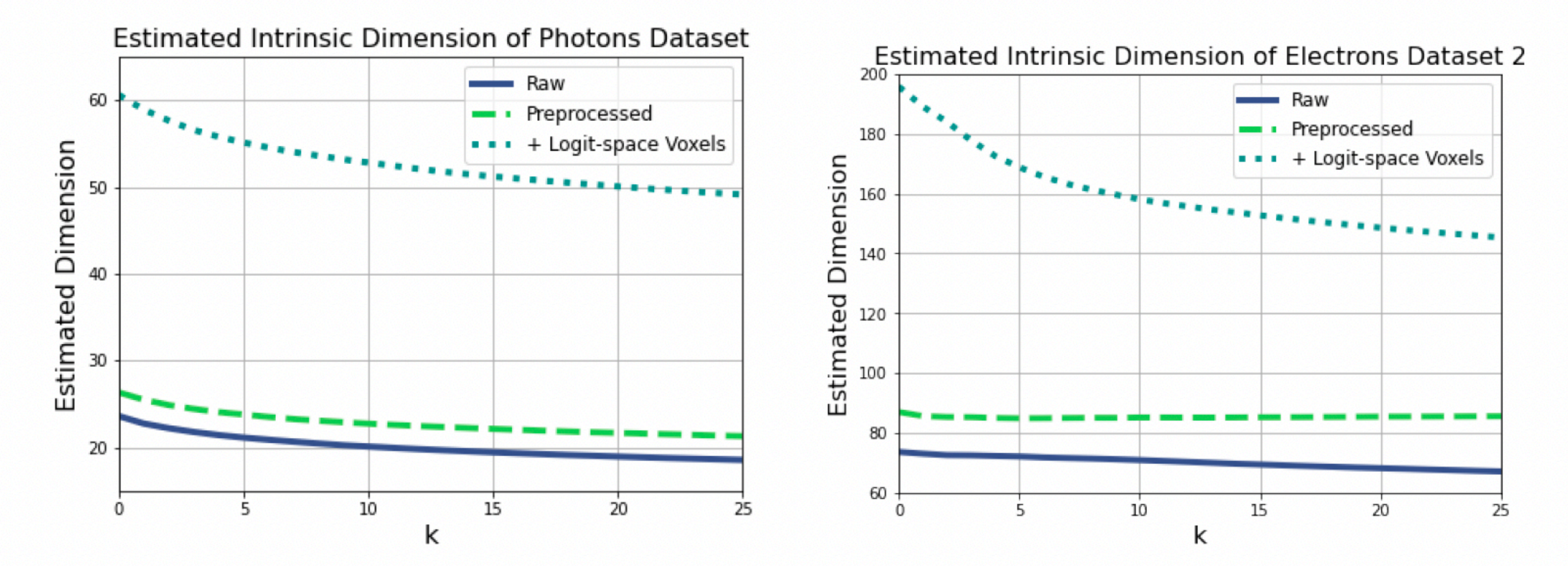
A GAE will learn a manifold with fixed dimensionality d.
Estimating d is up to us!

We used the Levina-Bickel statistical intrinsic estimator derived from the expected number of neighbours per unit volume as dimension increases:

$$\hat{d}_k = \left(\frac{1}{n(k-1)} \sum_{i=1}^{n} \sum_{j=1}^{k-1} \log \frac{T_k(x_i)}{T_j(x_i)} \right)^{-1}$$

Where $T_k(x_i)$ is the Euclidean distance between x_i and its kth nearest neighbour.

Estimating latent space dimensionality



	Raw	Preprocess	Preprocess +log
Photons1	20	23	55
Electrons2	70	82	160

Attempt 1: Conditioning on E_{inc}

PREPROCESSING STRATEGY (A LA CALOSCORE arXiv:2106.05285):

Energy per voxel:

$$E'_{vox} = \frac{E_{vox}}{fE_{inc}} \qquad f = 3.1$$

Incident energy (conditioner)

$$E'_{inc} = \frac{E_{inc} - E_{min}}{E_{max} - E_{min}}$$

Attempt 1: Conditioning on E_{inc}

ARCHITECTURE

VAE

Encoder: [512,512,512]
Decoder: [512,512,512]
Learning Rate: .001
Max epochs: 200
LR scheduler:
Early stopping: None

COUPLING FLOW

Bijector: Rational Quadratic Spline (RQS)
N bins: 8
Tail bound: 1
NF layers: 4
(Residual) hidden layers: [256,256,256]
Learning rate: .001
LR scheduler: None
Early stopping: mean histogram difference
Max epochs: 200

LATENT SPACE DIMENSIONS: 20

Attempt 1: Conditioning on E_{inc}

RESULTS

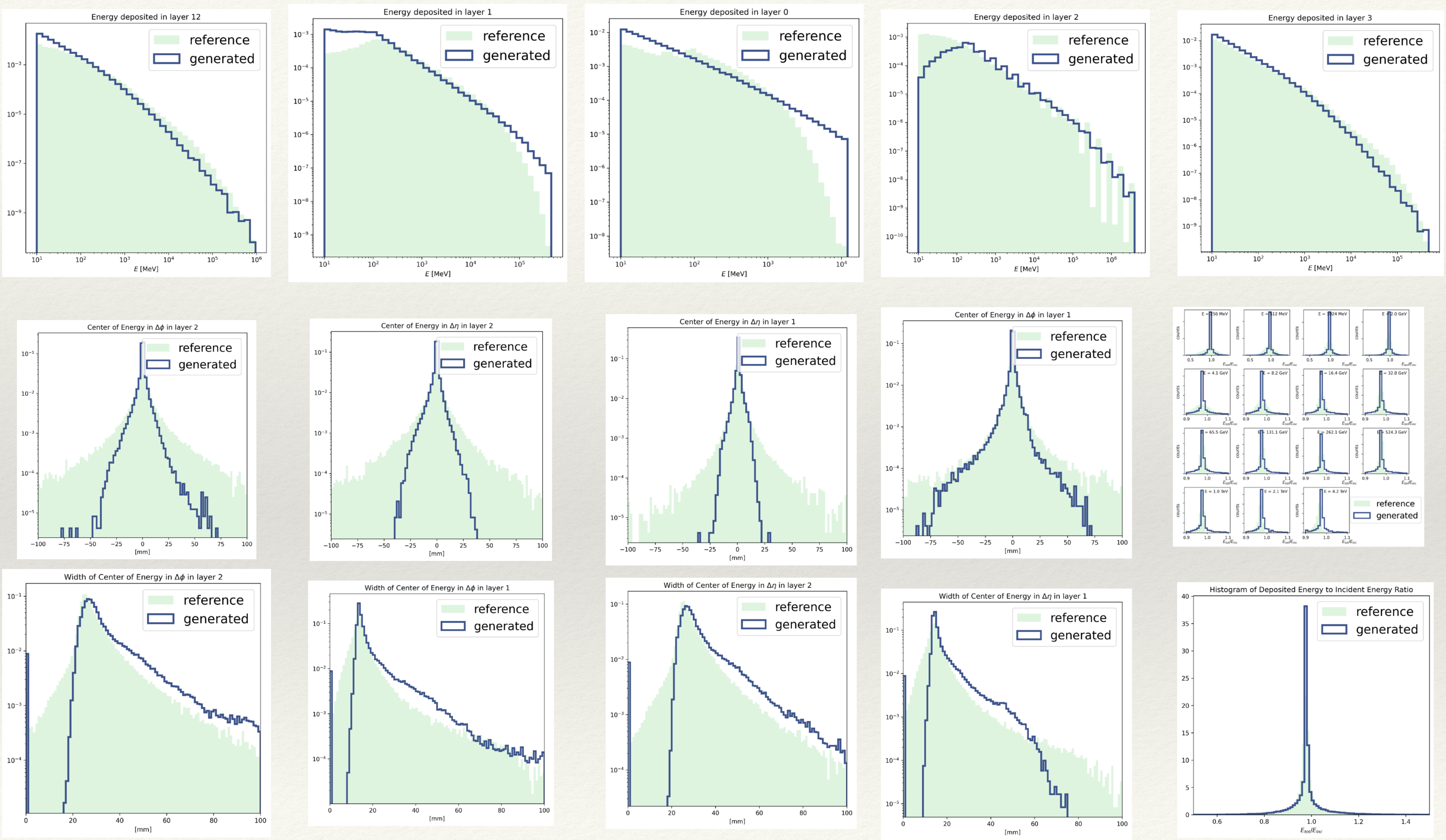
Separation power:

E_{tot}/E_{inc}: 0.0483
E_{layers}: 0.023
EC_{η}: 0.0323
EC_{ϕ}: 0.0227
$Width_{\eta}$: 0.1043
$Width_{\phi}$: 0.09277
Average : 0.0539

Time:

batch_size:500, num_samples:500: 0.2208s,
batch_size:500, num_samples:100000: 0.4056s
batch_size:1000, num_samples:1000: 0.22418s
batch_size:1000, num_samples:100000: 0.2973s
batch_size:5000, num_samples:5000: 0.202905s
batch_size:5000, num_samples:100000: 0.22263s
batch_size:10000, num_samples:10000: 0.20490s
batch_size:10000, num_samples:100000: 0.21063s
batch_size:50000, num_samples:50000: 0.221610s
batch_size:50000, num_samples:100000: 0.2080s

Attempt 1: Conditioning on E_{inc}



Attempt 2 : Conditioning on E_{inc} and E_{layers}

PREPROCESSING STRATEGY (A LA CALOFLOW arXiv:2206.11898):

STEP 1: Learning E_{layer}

Conditioners:

$$u_0 = \frac{\sum E_{layer}}{E_{inc}}, \quad u_1 = \frac{E_0}{\sum E_{layer}}, u_2 = \frac{E_1}{\sum E_{layer} - E_0}, \ldots$$

$$E_{inc} = \log_{10}(\frac{E_{inc}}{33.3\text{GeV}})$$

$$u = \log \frac{x}{1-x}; \quad x = \alpha + (1+2\alpha)u; \quad \alpha = 10^{-6}$$

STEP 2:

$$E'^{i}_{vox} = \frac{E^i_{vox}}{E^i_{layer}} \qquad E_{vox} = \log \frac{x}{1-x}; x = \alpha + (1-2\alpha)E_{vox}$$

(a) (b)

$$E_{inc} = \log_{10}(\frac{E_{inc}}{33.3\text{GeV}})$$

$$E'_{layer} = \log_{10}(\frac{E_{layer} + 1\text{keV}}{100\text{GeV}})$$

↑

Uploaded version

Attempt 2 (a): Conditioning on E_{inc} and E_{layers}

ARCHITECTURE.

STEP 2:

AE

Encoder: [512,512,512]
Decoder: [512,512,512]
Learning Rate: .001
Max epochs: 200
LR scheduler:
Early stopping: None

COUPLING FLOW

Bijector: RQS
N bins: 8
Tail bound: 1
NF layers: 4
(Residual) hidden layers: [128,128,128]
Learning rate: .001
LR scheduler: None
Early stopping: -log(L)
Max epochs: 200

STEP 1:

COUPLING FLOW

Bijector: RQS
N bins: 8
Tail bound: 1
NF layers: 4
(Residual) hidden layers: [128,128,128]
Learning rate: .001
LR scheduler: None
Early stopping: -log(L)
Max epochs: 200

LATENT SPACE DIMENSIONS: 30

Attempt 2 (a): Conditioning on E_{inc} and E_{layers}

RESULTS

STEP 1:

E_{tot}/E_{inc}: 0.00248
E_{layers}: 0.000214
Average : 0.00135

STEP 2:

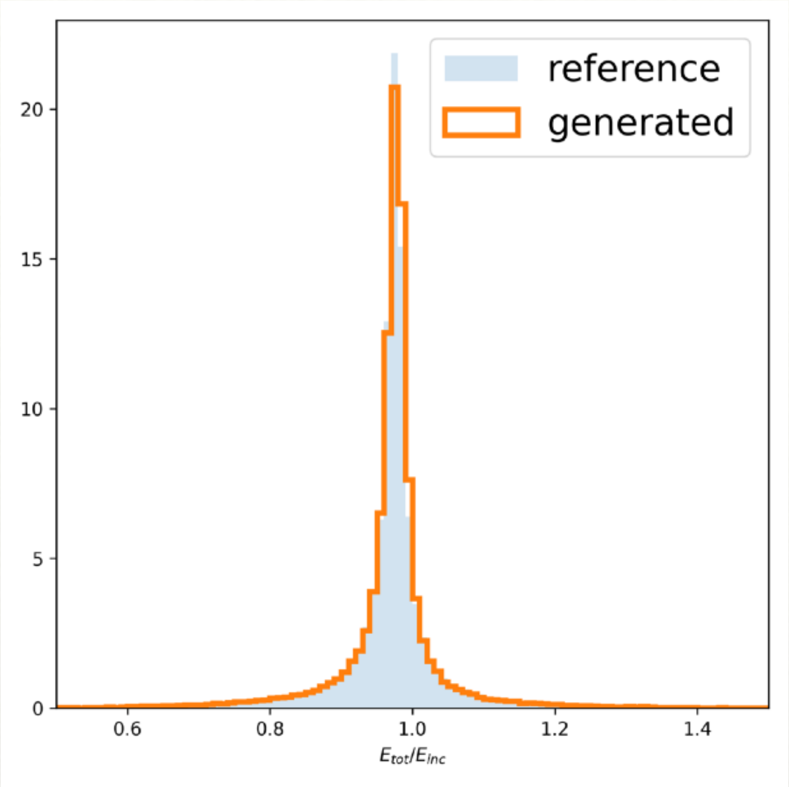
E_{tot}/E_{inc}: 0.1112
E_{layers}: 0.002
EC_{η}: 0.0302
EC_{ϕ}: 0.01211
$Width_{\eta}$: 0.13938
$Width_{\phi}$: 0.246974
Average : 0.09033

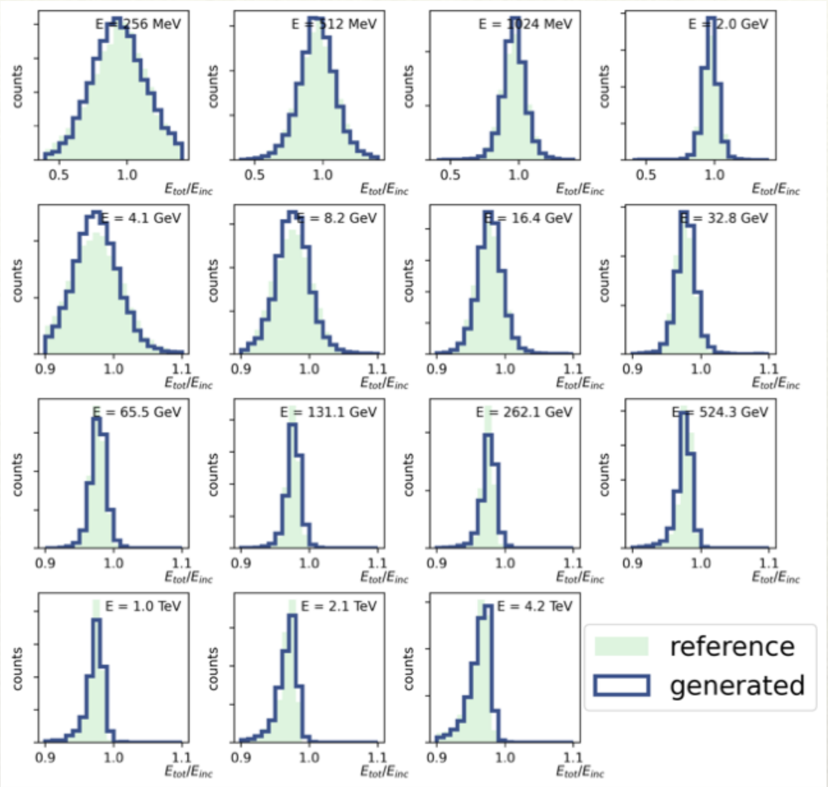
121000 E_{layer} samples generated in 1.59596 seconds

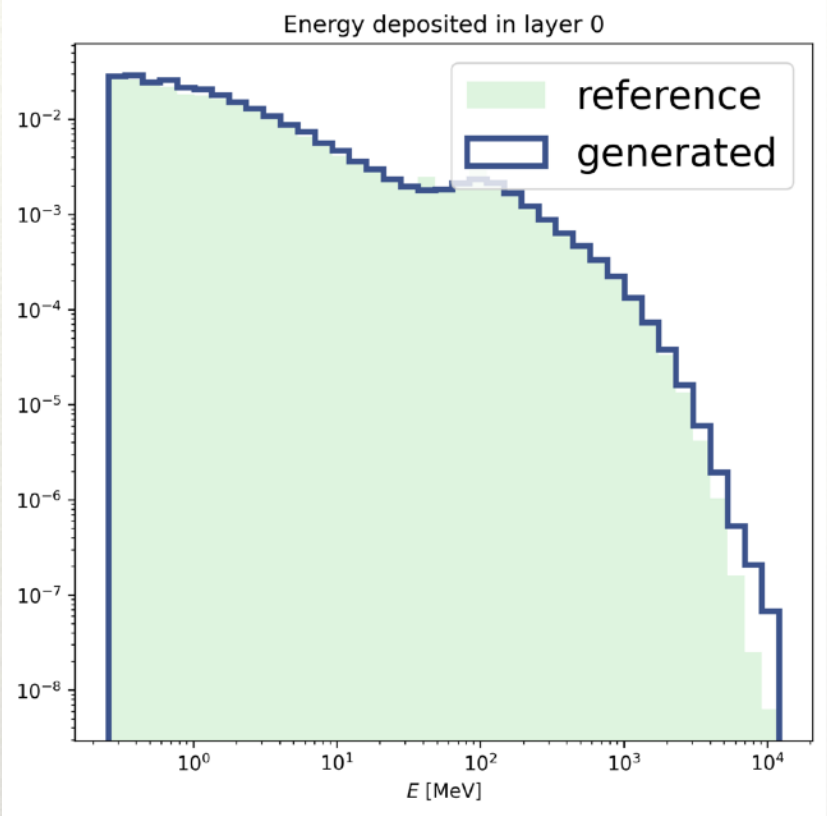
121000 shower samples generated in 3.7 seconds

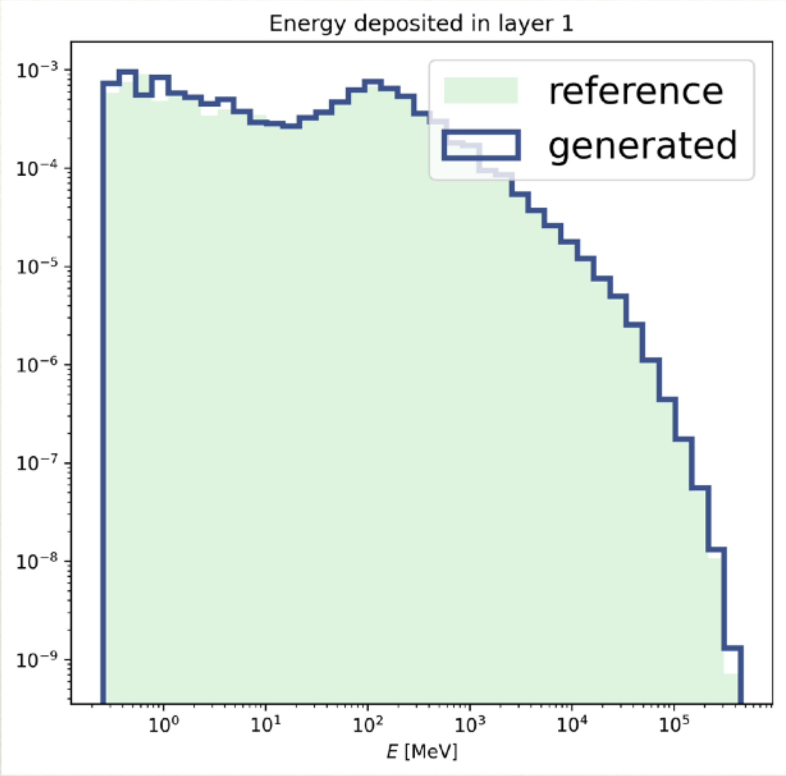
Attempt 2 (a): Conditioning on E_{inc} and E_{layers}

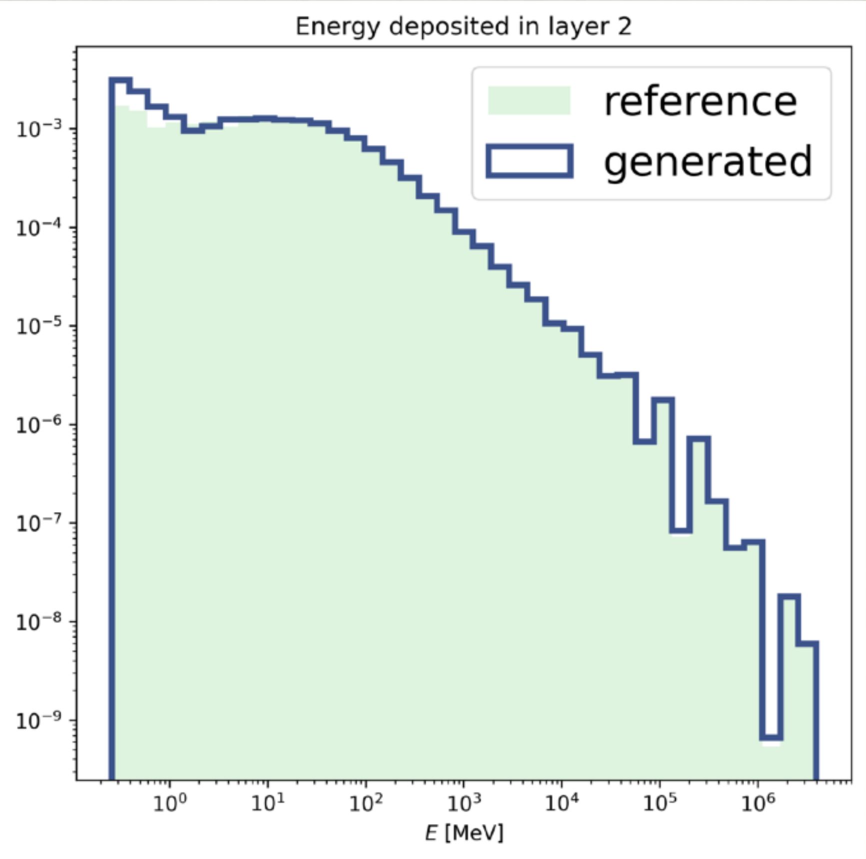
STEP 1:

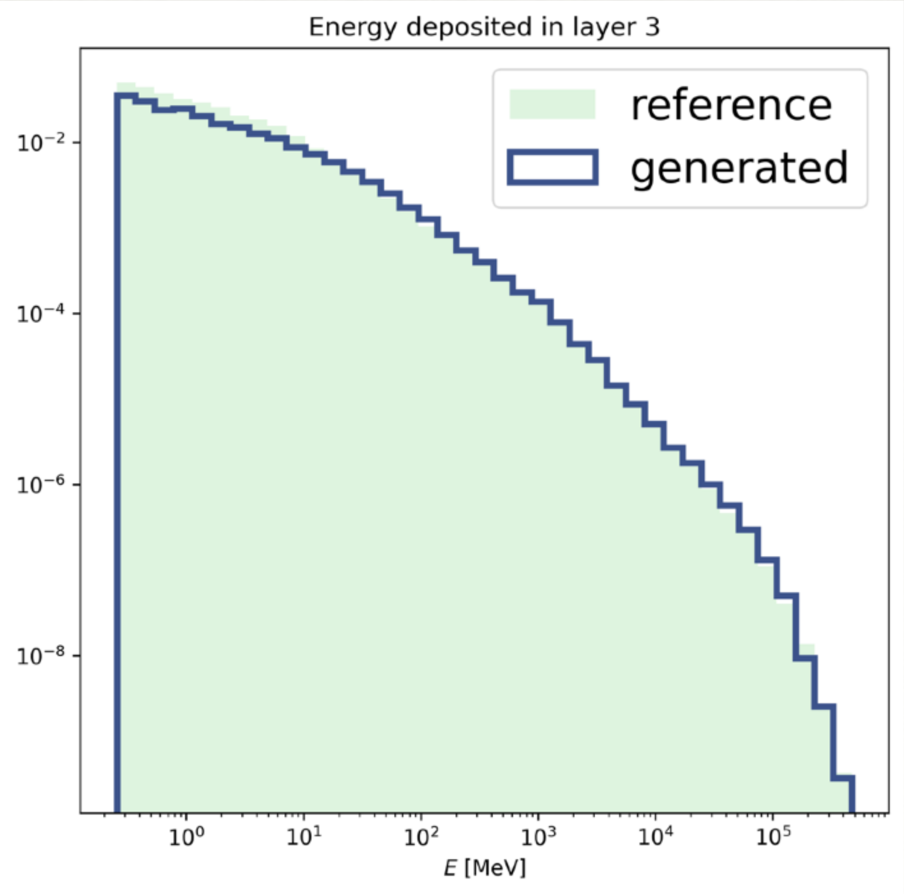


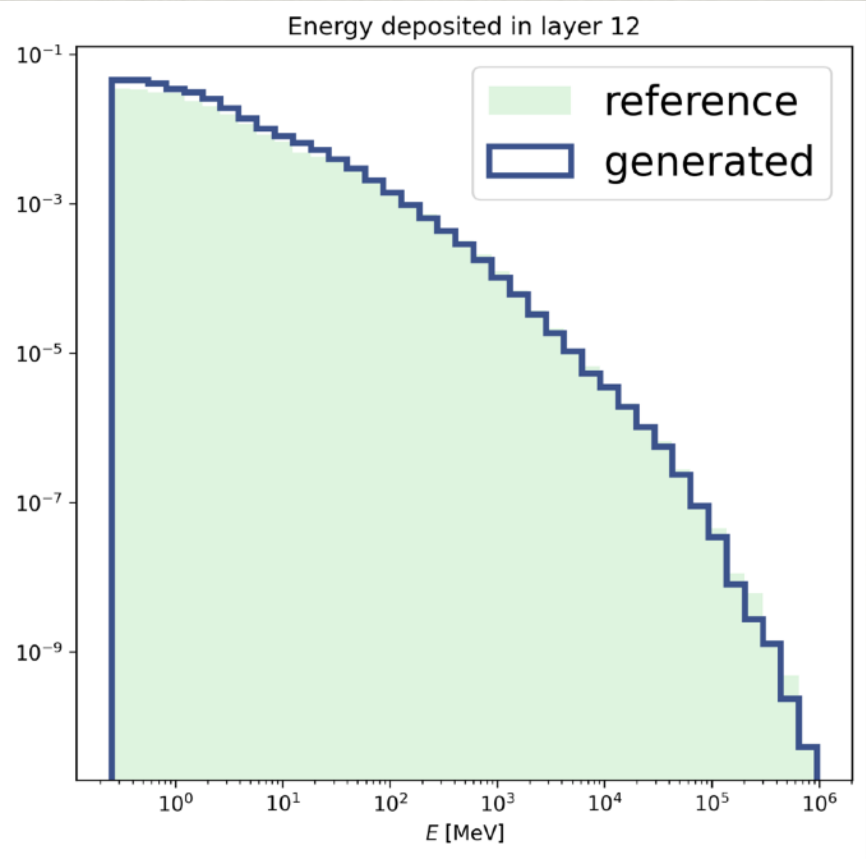






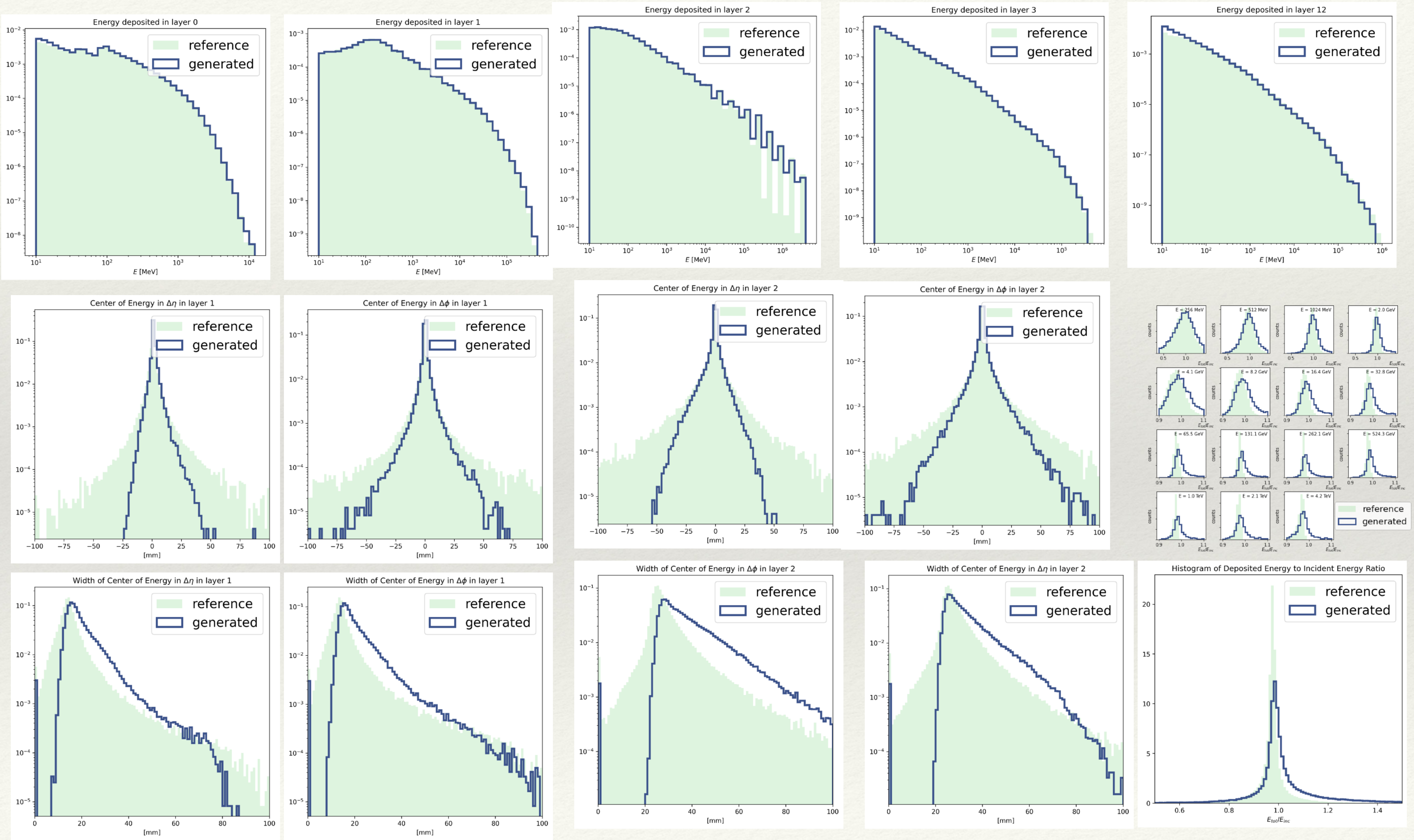






Attempt 2 (a): Conditioning on E_{inc} and E_{layers}

STEP 2:



PIONS

PIONS: Conditioning on E_{inc} and E_{layers}

Preprocess: A la CaloFlow, no log

STEP 2:

AE

Encoder: [512,512,512]
Decoder: [512,512,512]
Learning Rate: .001
Max epochs: 200
LR scheduler:
Early stopping: L2 error

COUPLING FLOW

Bijector: RQS
N bins: 8
Tail bound: 1
NF layers: 4
(Residual) hidden layers: [64,64,64]
Learning rate: .001
LR scheduler: None
Early stopping: -log(L)
Max epochs: 200

STEP 1:

AUTOREGRESSIVE FLOW

Bijector: RQS
N bins: 8
Tail bound: 1
NF layers: 8
(Residual) hidden layers: [128,128,128]
Learning rate: .0001
LR scheduler: None
Early stopping: -log(L)
Max epochs: 200

LATENT SPACE DIMENSIONS: 20 (Estimated: 12)

PIONS: Conditioning on E_{inc} and E_{layers}

RESULTS

STEP 1:

E_{tot}/E_{inc}: 0.00187
E_{layers}: 0.00028
Average : 0.00107

STEP 2:

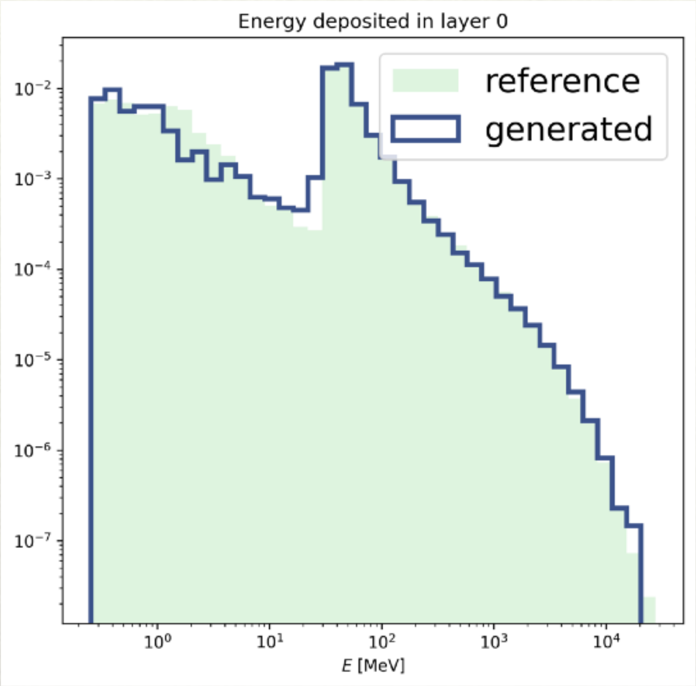
E_{tot}/E_{inc}: 0.03681
E_{layers}: 0.00056
EC_{η}: 0.03688
EC_{ϕ}: 0.0367
$Width_{\eta}$: 0.229
$Width_{\phi}$: 0.2317
Average : 0.09534

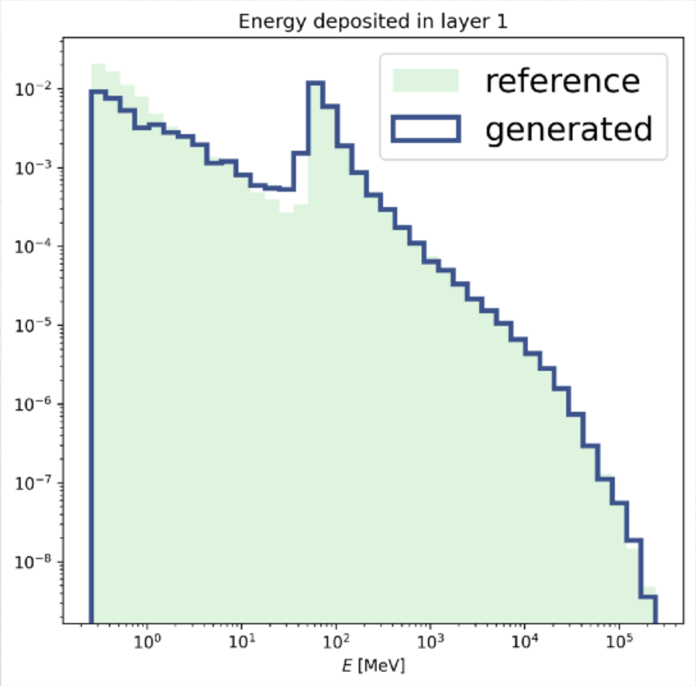
24046 E_{layer} samples generated in 9.71327 seconds

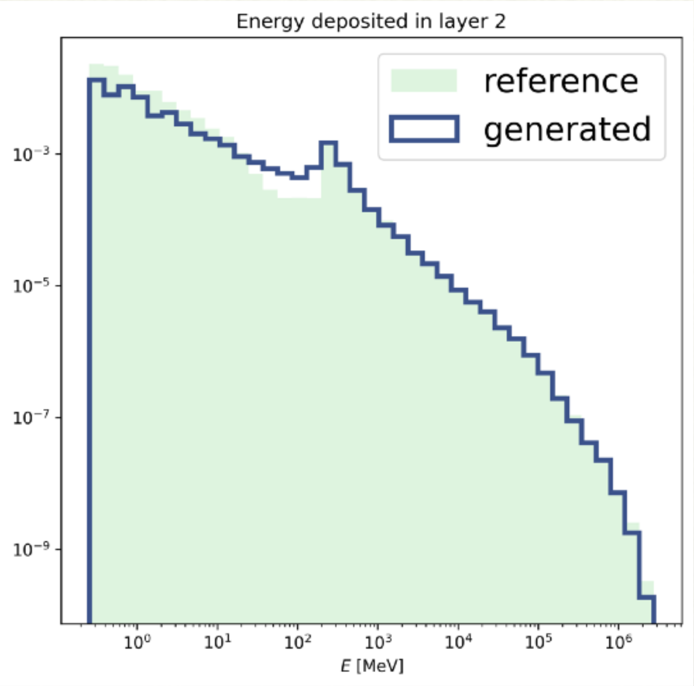
120230 shower samples generated in 3.53365 seconds

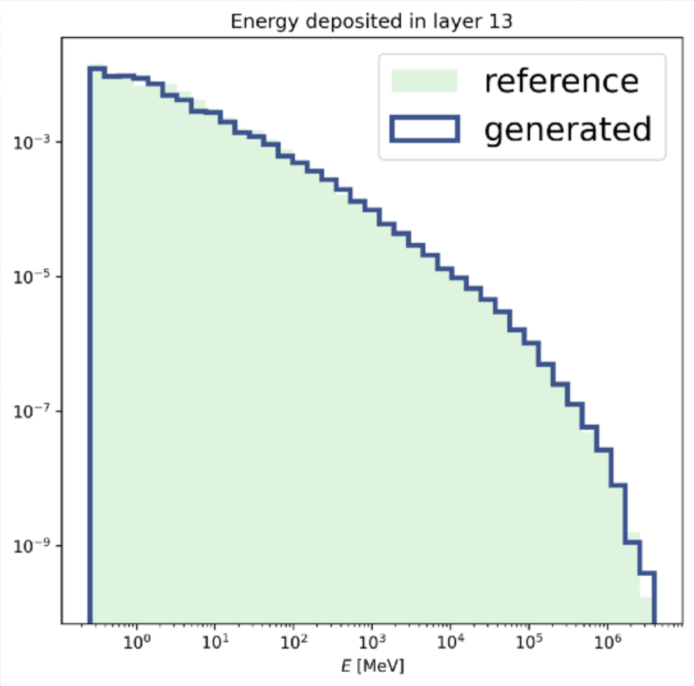
PIONS: Conditioning on E_{inc} and E_{layers}

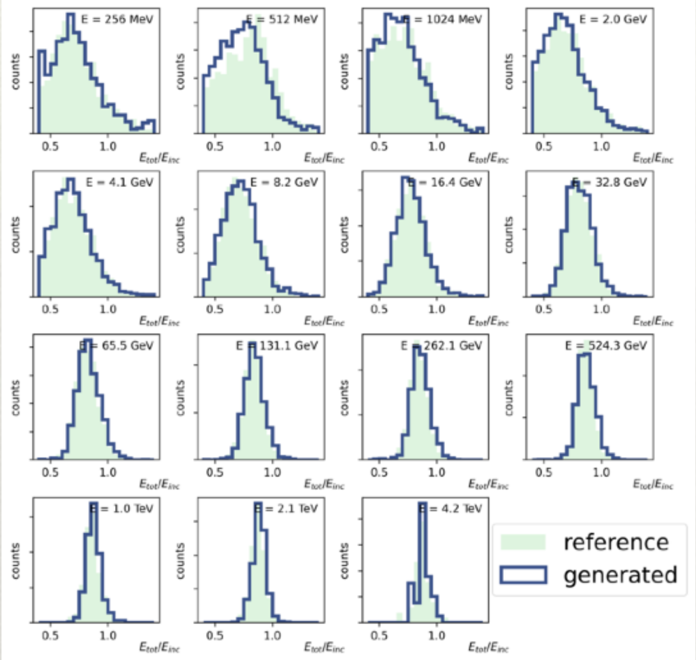
STEP 1:

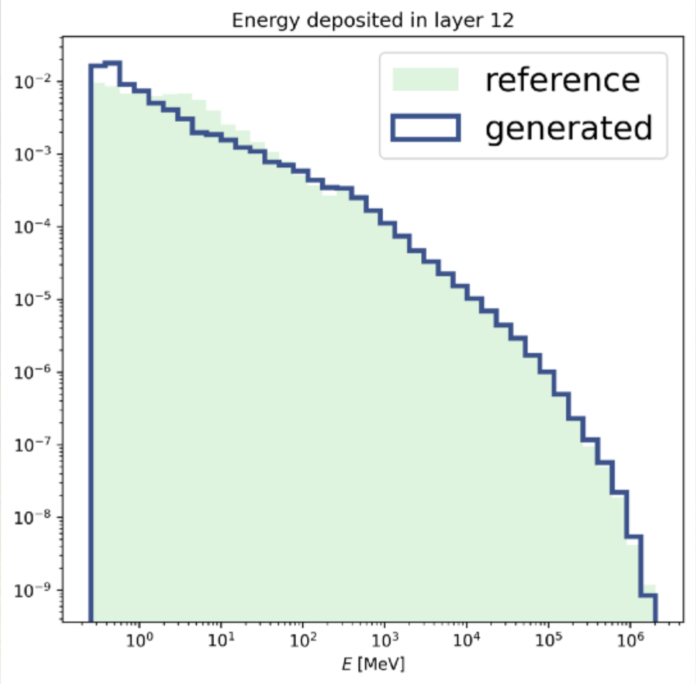


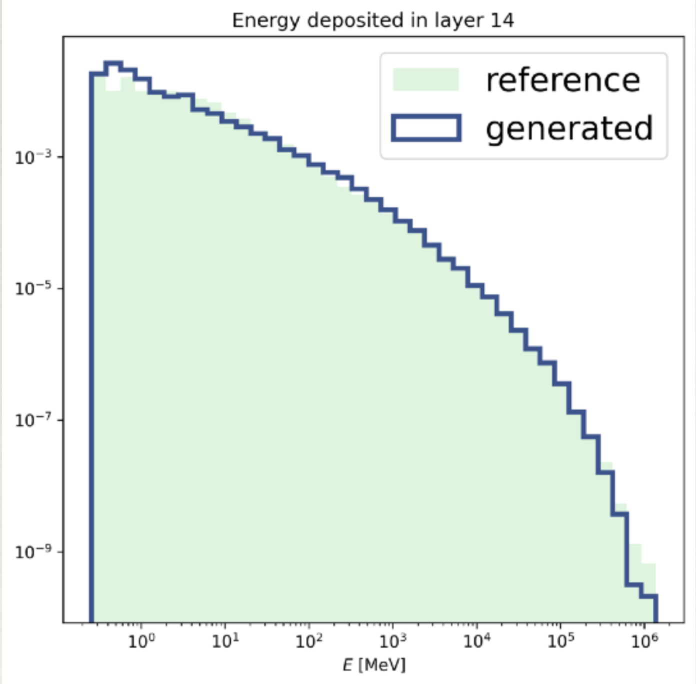


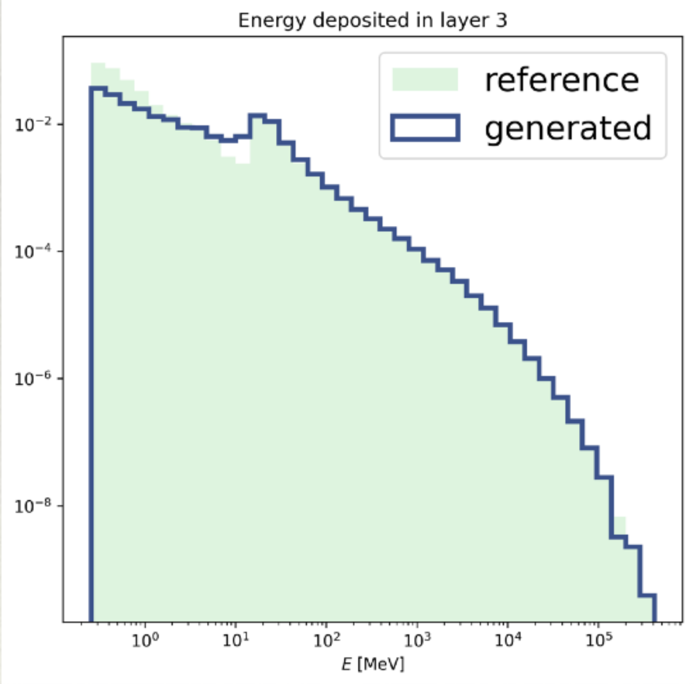


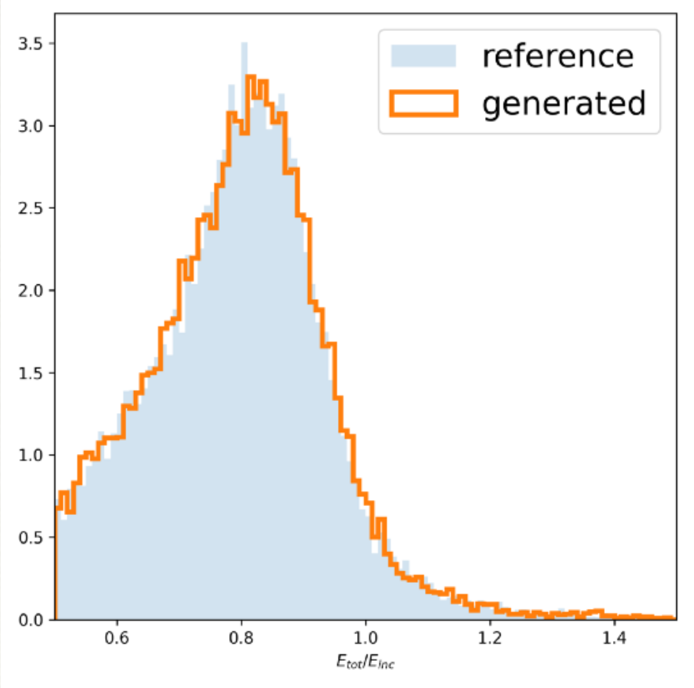




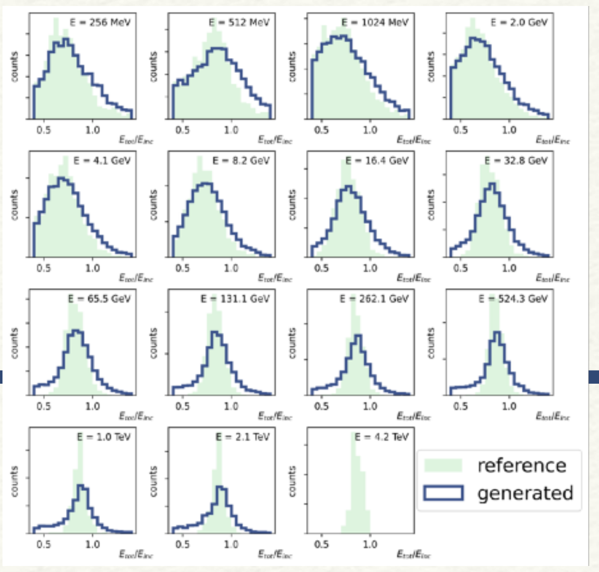


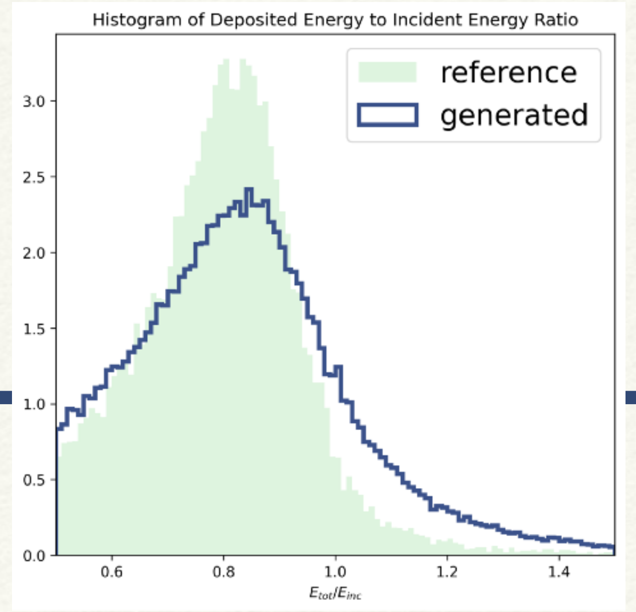






PIONS: Conditioning on E_{inc} and E_{layers}





STEP 2:

Many things left to try:

- To log or not to log
- Fine tuning the preprocessing strategy
- Regularization.
- Divide and conquer
- Many architectures/hyperparameters to explore
- **We are going slowly… but we will get there**.

Conclusions

- Density estimation of latent space is a very promising approach:
- We see potential to accurately describe high dimensional Calorimeter showers
- As a highlight we obtain compact and very fast generative network systems.
- We plan to test it scalability tackling the rest of the datasets.
- Very much worth exploring more applications in HEP.
- EXTRA: Interesting to harvest Academia+Private sector synergies.

BACK UP

Attempt 2 (b)

STEP 2:

AE

Encoder: [256,256,256]
Decoder: [256,256,256]
Learning Rate: .001
Max epochs: 200
LR scheduler:
Early stopping: None

COUPLING FLOW

Bijector: RQS
N bins: 8
Tail bound: 1
NF layers: 4
(Residual) hidden layers: [256,256,256]
Learning rate: .001
LR scheduler:
Early stopping: -log(L)
Max epochs: 1000

LATENT SPACE DIMENSIONS: 80

Attempt 2 (b)

RESULTS*

Separation power:

E_{tot}/E_{inc}: 0.2525
E_{layers}: 0.0037
EC_{η}: 0.0137
EC_{ϕ}: 0.01407
$Width_{\eta}$: 0.1969
$Width_{\phi}$: 0.15099
Average : 0.10533

Timing:

batch_size:500, num_samples:500: 0.10954332s,
batch_size:500, num_samples:100000: 0.138106s
batch_size:1000, num_samples:1000:0.0443694s
batch_size:1000, num_samples:100000: 0.08901s
batch_size:5000, num_samples:5000: 0.0365597s
batch_size:5000, num_samples:100000: 0.04129s
batch_size:10000, num_samples:10000: 0.033784s
batch_size:10000, num_samples:100000:0.03680s
batch_size:50000, num_samples:50000: 0.033s
batch_size:50000, num_samples:100000: 0.03157s

*Tested with true E_{layers}

Attempt 2 (b)

RESULTS