Deep generative models for fast shower simulation in ATLAS

Dalila Salamani

University of Geneva

Second topical meeting of the HSF Detector Simulation Working Group

06/03/2019

Outline

- Context and motivation
- Rise of generative models
  - Generative models for HEP
  - Model architecture : VAE
  - Validation of generation performance
- Latent space analysis
- Conclusion

Motivation

- Successful physics program in ATLAS depends on the availability of high statistics from Monte Carlo simulated events.
- Currently **~40 % of ATLAS computing time** is in simulation, of which shower simulation is a large part.
- The LHC will collect more and more events in the future, including **HL-LHC**.

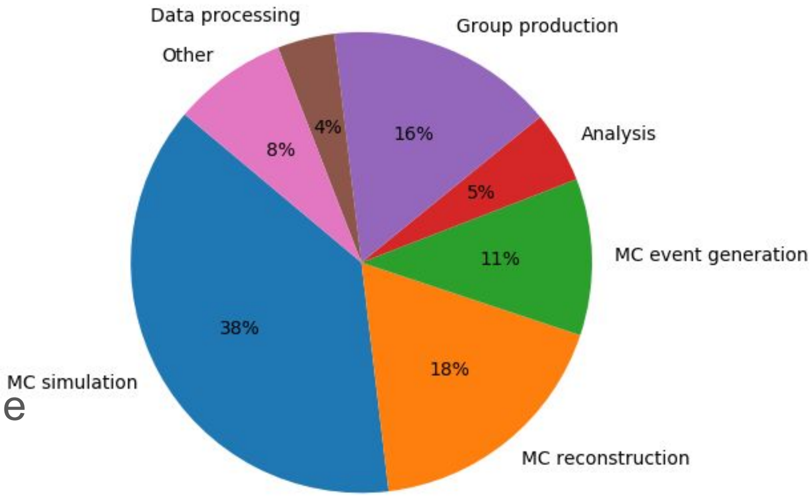


Figure from D.Costanzo, J.Catmore, LHCC meeting

Motivation

- Successful physics program in ATLAS depends on the availability of high statistics from Monte Carlo simulated events.
- Currently **~40 % of ATLAS computing time** is in simulation, of which shower simulation is a large part.
- The LHC will collect more and more events in the future, including **HL-LHC**.
- **Challenge**: Develop fast shower simulation framework.
    - Fast simulation (AF2)
    - **Fast simulation using generative models**

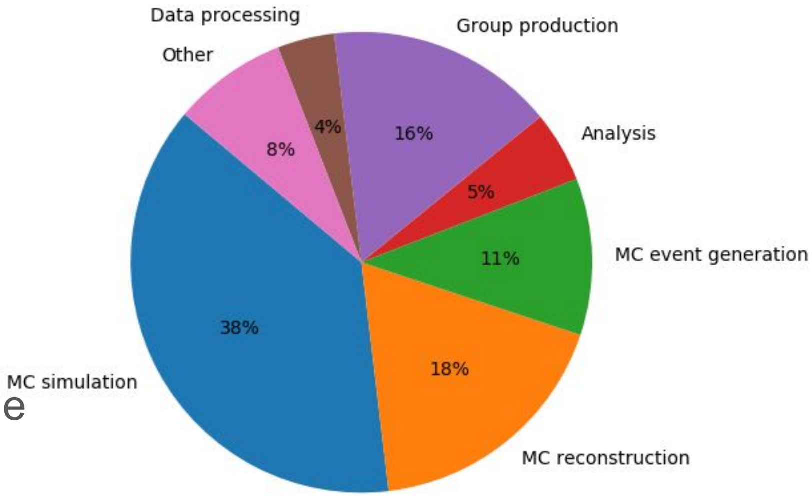
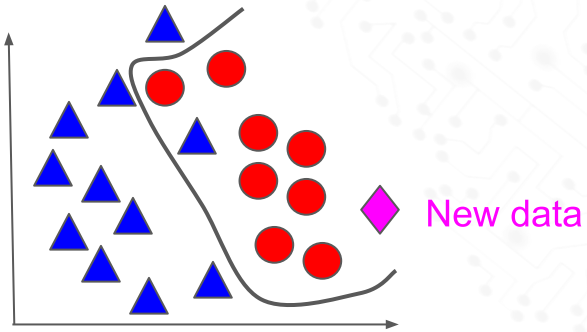


Figure from D.Costanzo, J.Catmore, LHCC meeting

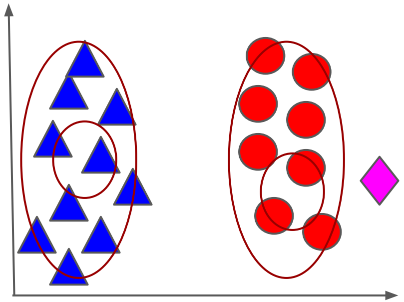
What is AI about ?

Decision Making
$p(y|x)$



High probability to get **red** label

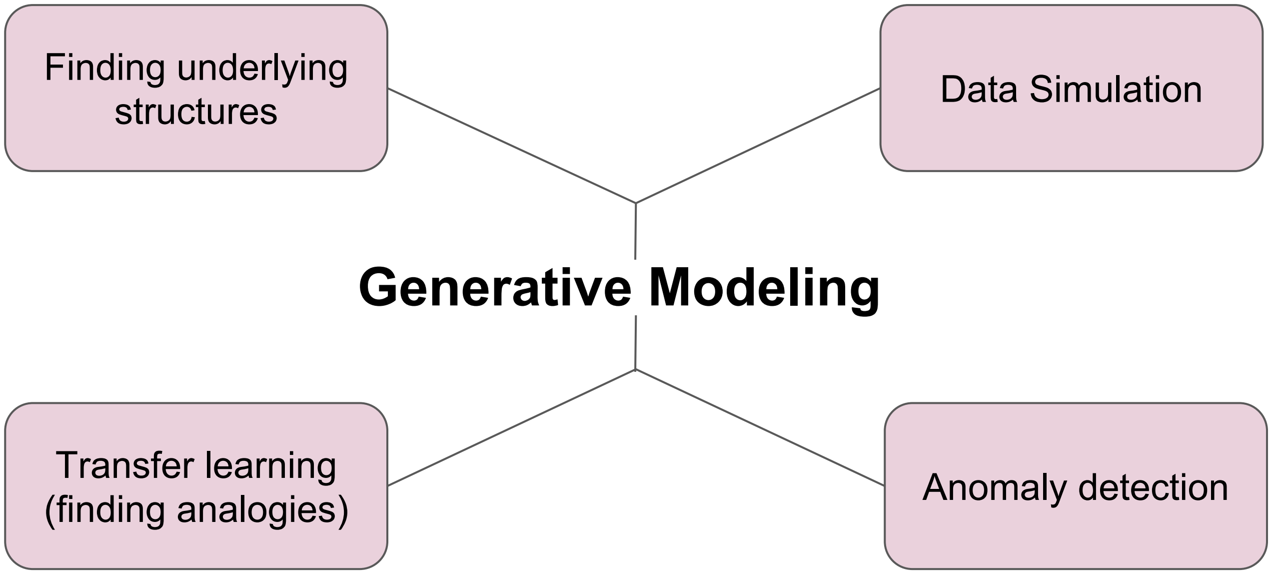
Understanding
$p(y,x) = p(y|x)\,p(x)$



High probability to get **red** label
x
Low probability of the **object**
=
Uncertain decision

What is generative modeling about ?

Understanding $p(y,x) = p(y|x)\,p(x)$



What is generative modeling about ?

Understanding $p(y,x) = p(y|x)\, p(x)$

- Learn the true **data distribution** of the training set **to reproduce it.**
- Popular generative models Variational Autoencoders (**VAEs**) & Generative Adversarial Networks(**GANs**): deep neural networks that learn the approximation function of the true (& sparse) distribution,

Data Simulation

Generative Modeling

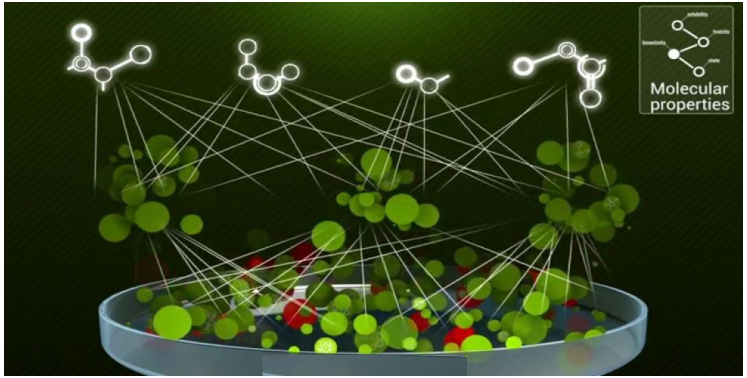
Noise $\sim N(0,1)$

Generate

Generative models : domain application

Learning to generate speech : Den Oord et al, 2016

Drug Discovery : Chen et al, 2018

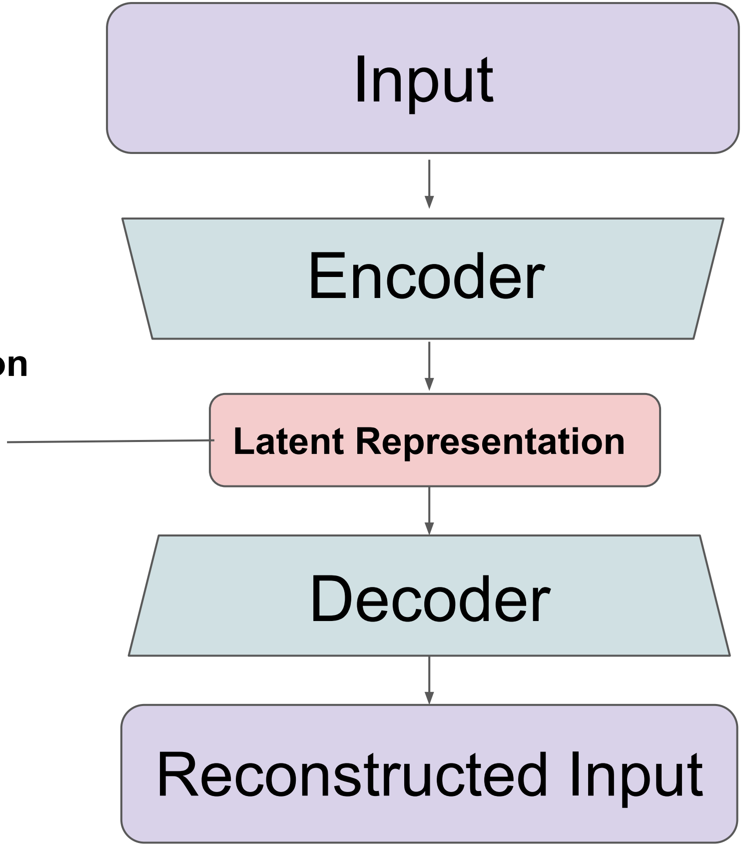


Learning to generate images : Brock et al, 2018

Variational Autoencoders

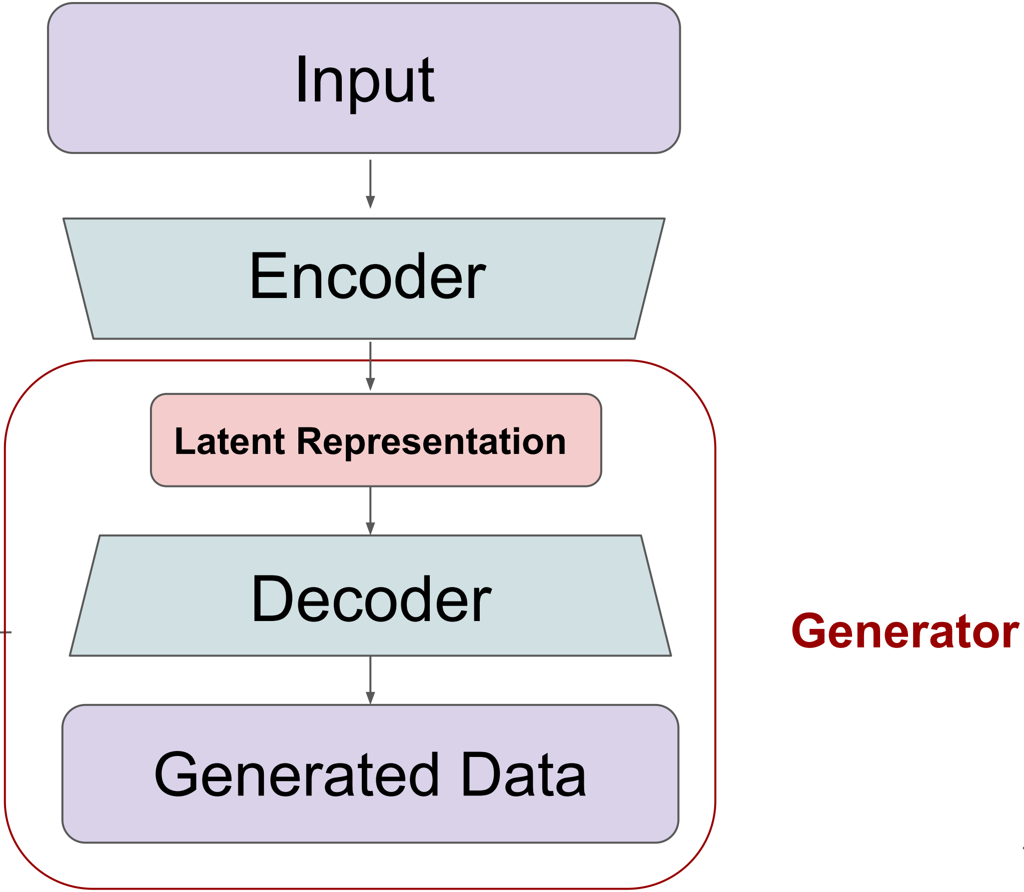
[Kingma & Willing, 2014]
[Rezende & al 2014]

- **Lower dimensional representation** of the input data
- Minimal information needed to reconstructed back the input



Variational Autoencoders

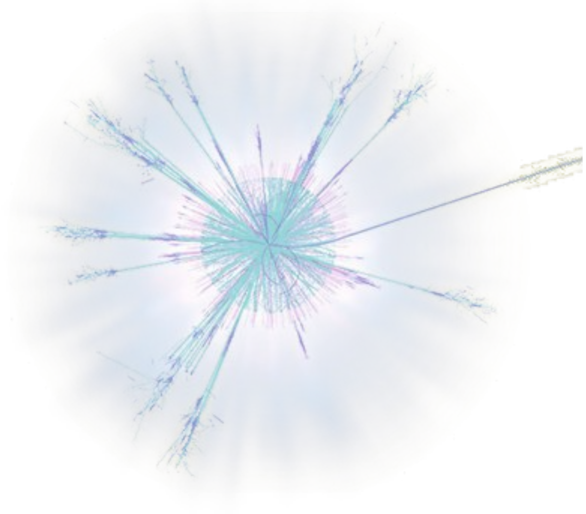
[Kingma & Willing, 2014]
[Rezende & al 2014]



Once the VAE (encoder + decoder) model trained, to generate new data : use only the **decoder** part as a generator by sampling from the distribution of the latent space

ATLAS PUB NOTE

Generative models for HEP ATL-SOFT-PUB-2018-001

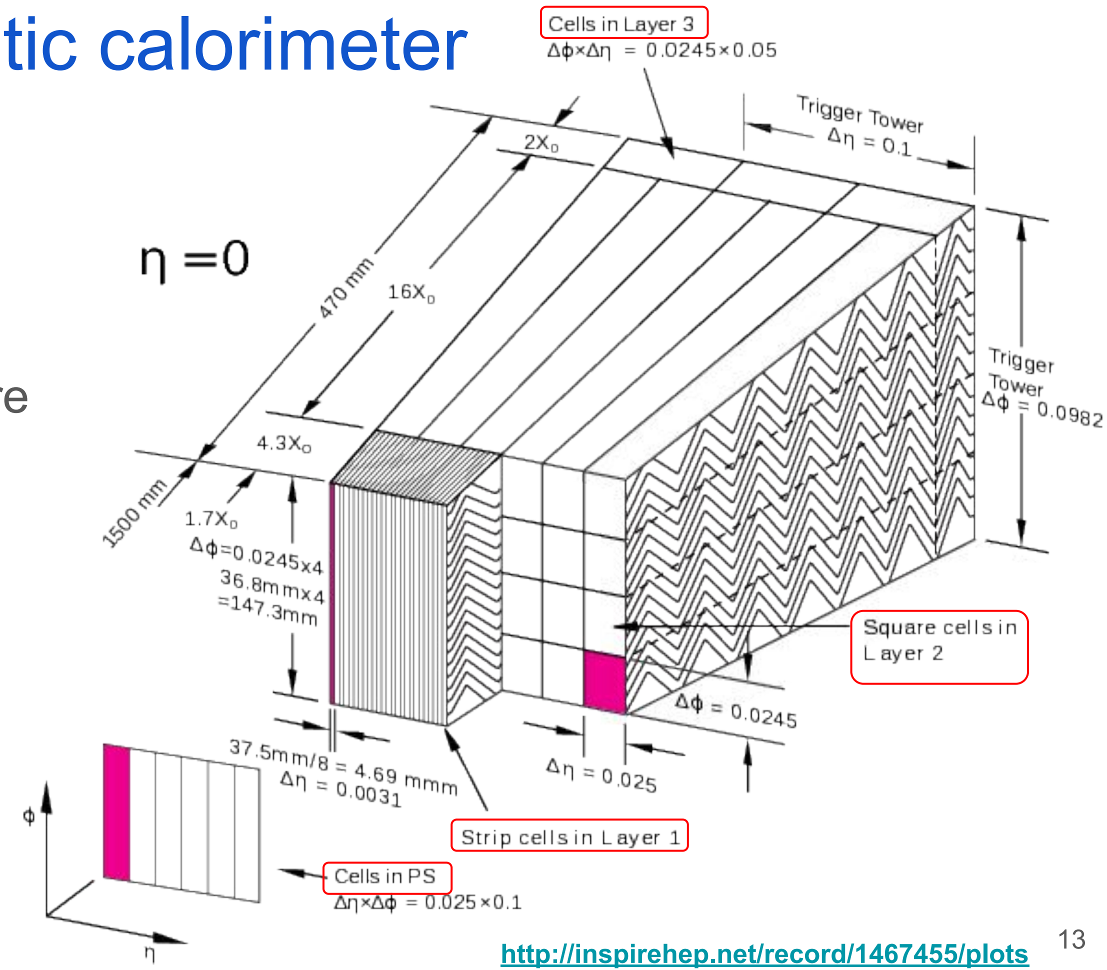


Generative models for **HEP (Showering)**

- Model the shower process.
- Take into account the ATLAS calorimeter geometry.
- Validation : shower shape variables distribution comparison.
- First application of deep generative models for fast shower simulation in ATLAS: Public Note ATL-SOFT-PUB-2018-001.

ATLAS electromagnetic calorimeter

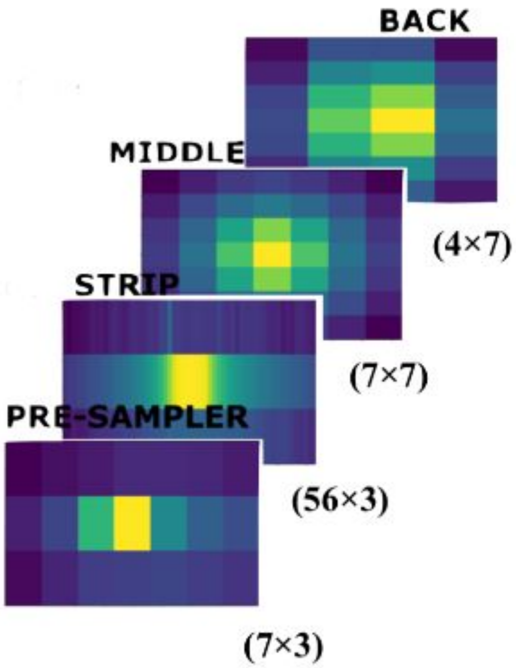
- **Pre-Sampler** : some energy deposit
- **Strip**: very granular in η, more energy deposit
- **Middle** : thickest layer, maximum energy deposit
- **Back** : little energy deposit



http://inspirehep.net/record/1467455/plots

Dataset & preprocessing

- Single photon samples in the electromagnetic calorimeter (4 layers with different granularities).
- Pseudorapidity **0.20 < |η| < 0.25**.
- Energies in **[1, 260] GeV** logarithmically spaced.
- A total of **266 cells** (7x3, 56x3, 7x7 and 4x7) are considered for energy deposits.

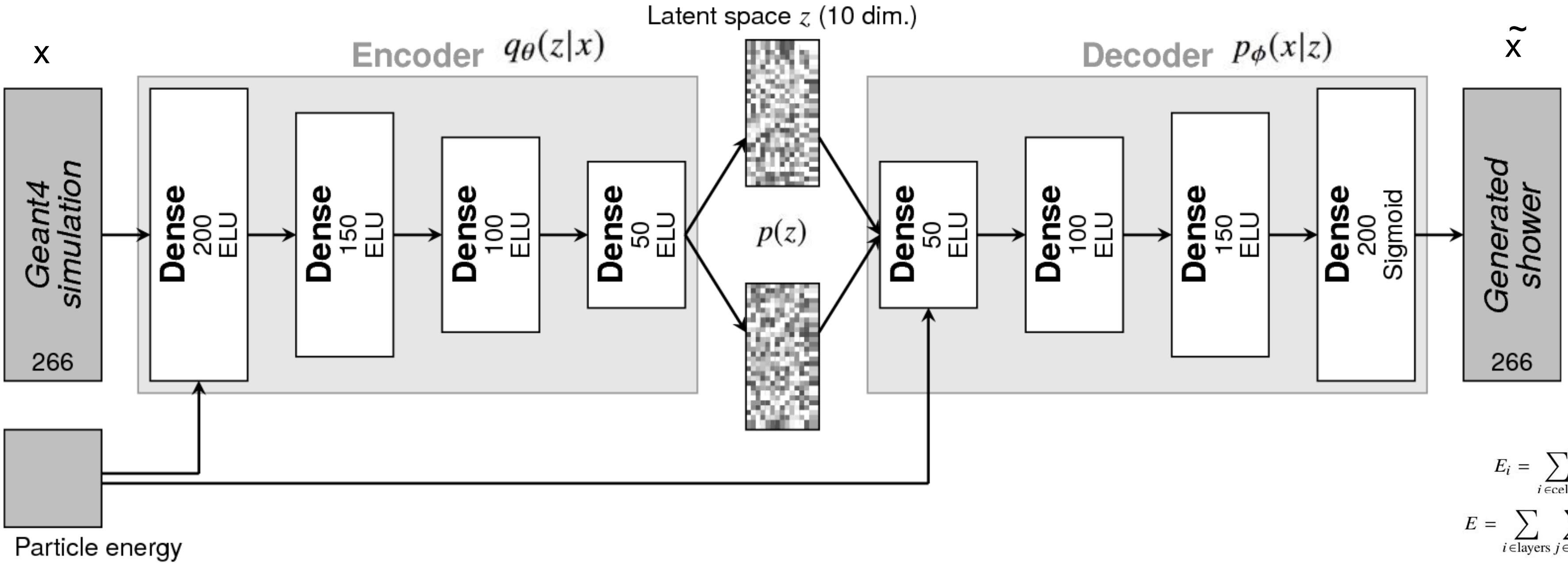


TensorFlow Keras

Using **HDF5** format

Reference

VAE model architecture



$$E_i = \sum_{i \in \text{cells}} E_{ij}$$

$$E = \sum_{i \in \text{layers}} \sum_{j \in \text{cells}} E_{ij}$$

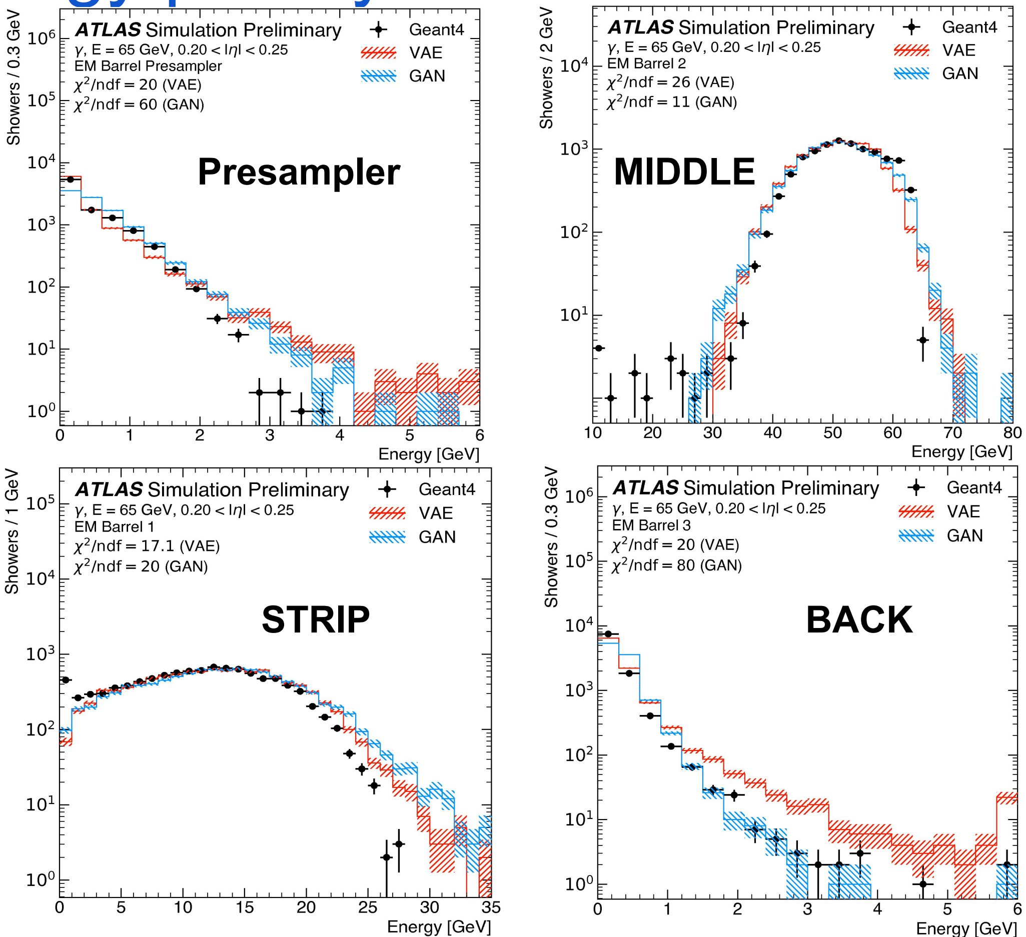
$$L_{\text{VAE}}(x, \tilde{x}) = w_{\text{reco}} E_{z \sim q_\theta(z|x)}[\log p_\phi(x|z)] - w_{\text{KL}} \text{KL}(q_\theta(z|x) || p(z)) + w_{E_{\text{tot}}} L_{E_{\text{tot}}}(x, \tilde{x}) + \sum_i^M w_i L_{E_i}(x, \tilde{x})$$

Reconstruction Loss **KL Loss** **Total Energy Loss** **Energy fraction per layer Loss**

Reference

Generation results: energy per layer

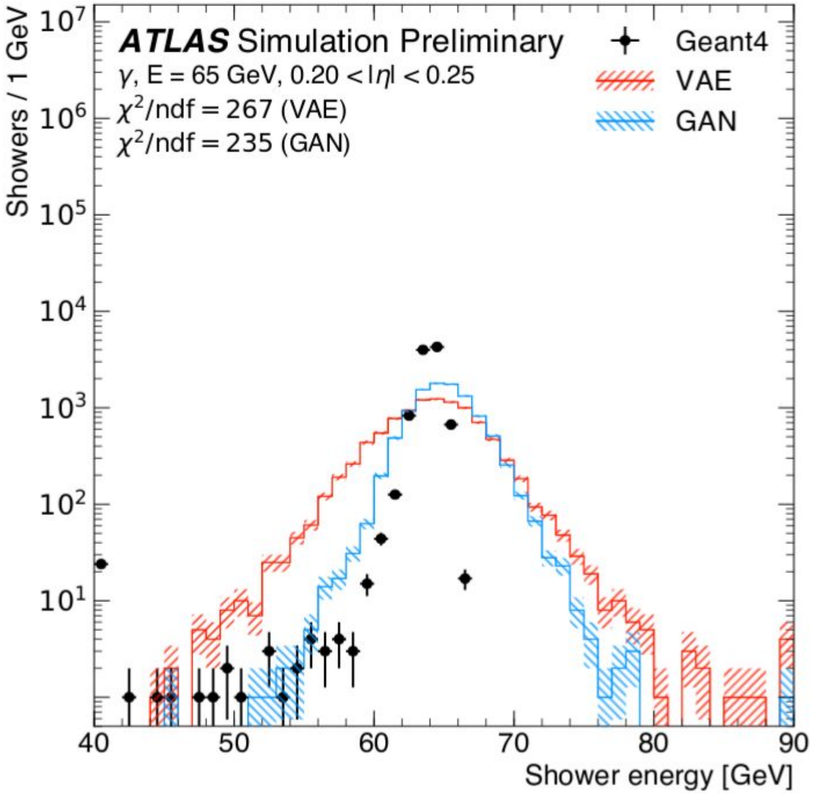
- Energy deposited in the individual electromagnetic calorimeter layers for photons 65 GeV.
- Challenges posed by layers with low (and sparse) energy deposits, i.e. late showers.



Reference

Generation results: total energy

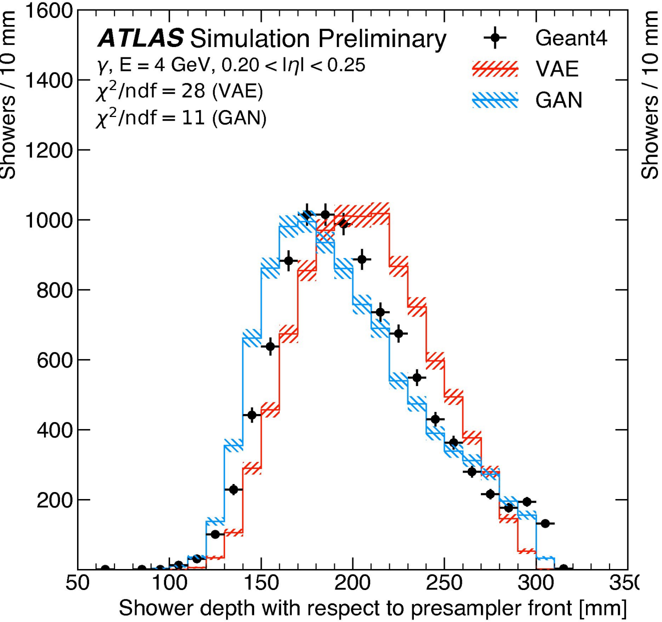
- Total energy response of the calorimeter to photons with an energy of approximately 65GeV
- Modeling the correlation between the layers ?



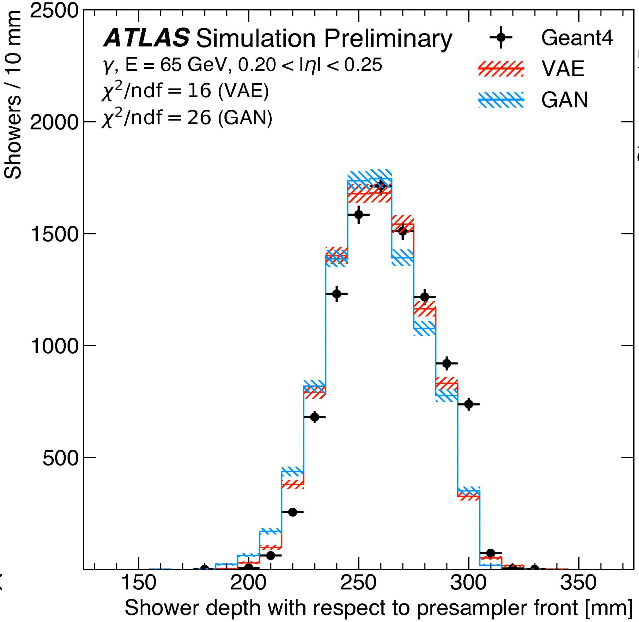
Reference

Generation results: reconstructed longitudinal shower center

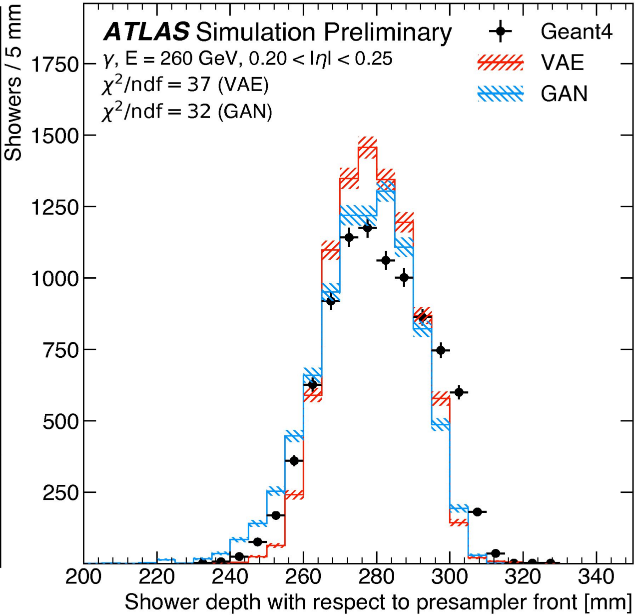
Energy = 4 GeV



Energy = 65 GeV



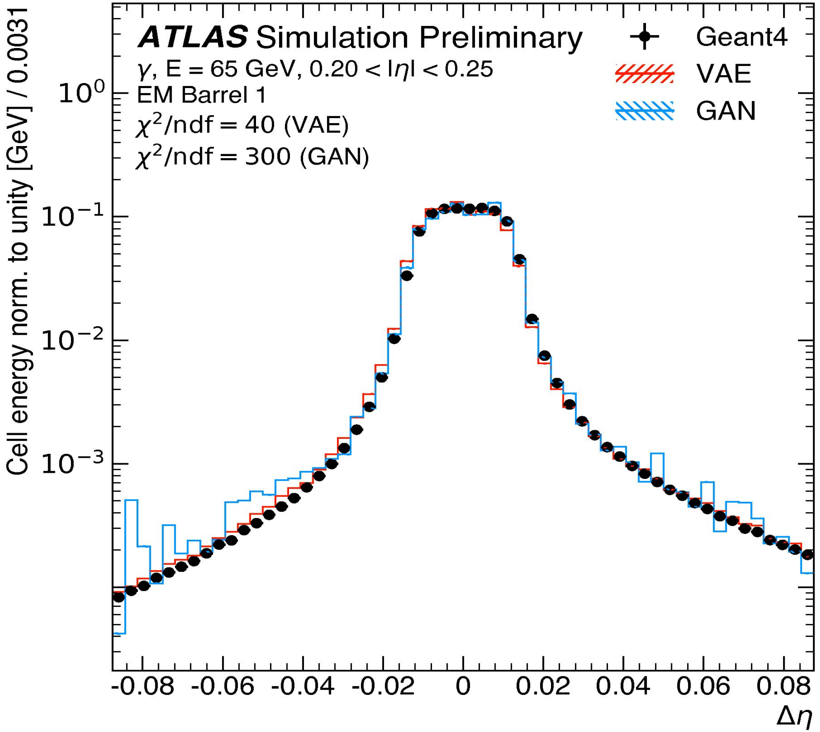
Energy = 260 GeV



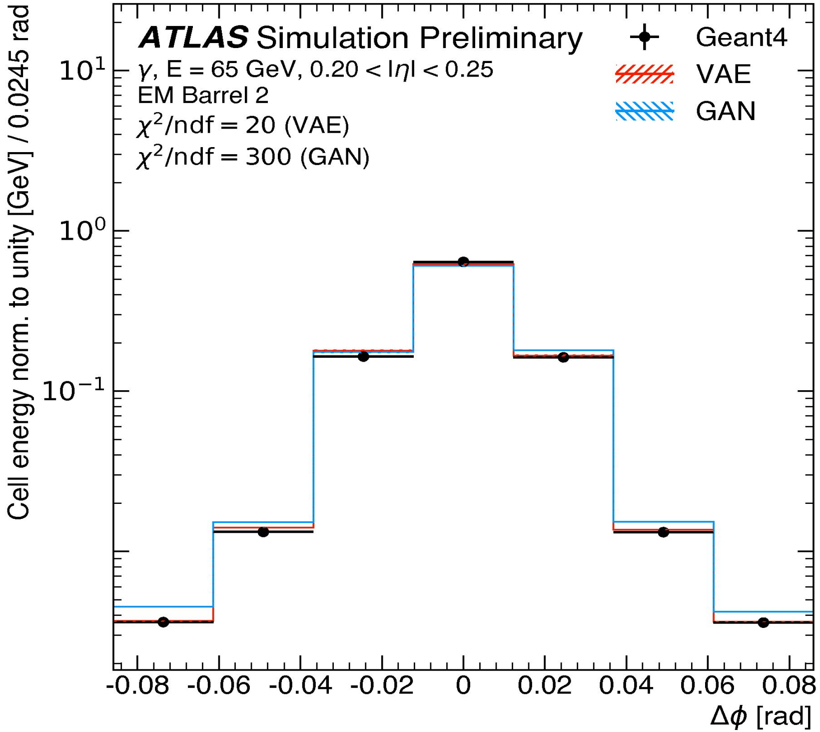
Reference

Generation results: Average energy vs Δη , Δϕ

Average energy vs Δη Layer : STRIP



Average energy vs Δϕ Layer : MIDDLE

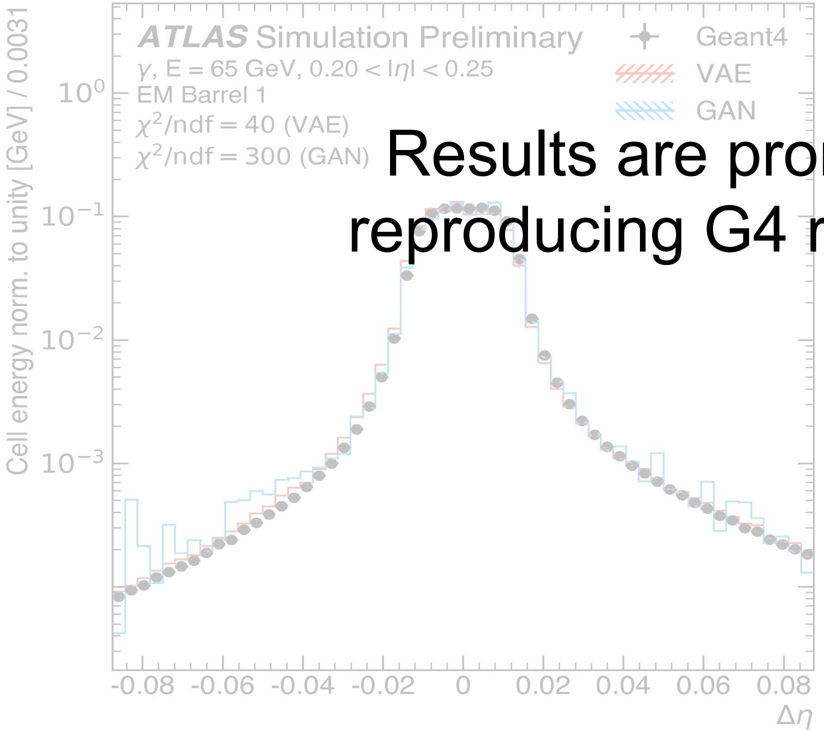


Reference

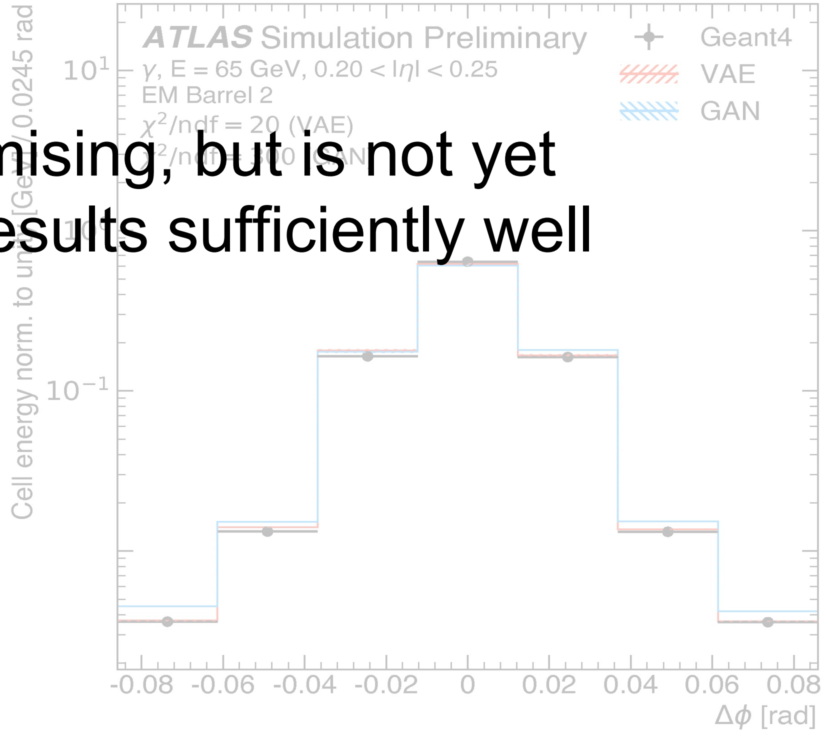
Generation results: Average energy vs Δη , Δϕ

Results are promising, but is not yet reproducing G4 results sufficiently well

Average energy vs Δη Layer : STRIP



Average energy vs Δϕ Layer : MIDDLE

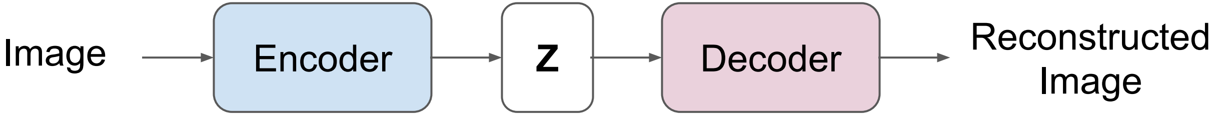


Latent Space Analysis

- Latent representation: **data compression**, **dimensionality reduction** & **feature learning**
- Motivation for the current study
  - **Better understanding of the latent space to improve results**
  - Defining what makes a given latent representation better than another is not trivial.
    Understanding this represents one of the main determinants of the performance of VAEs
  - **How to get a Gaussian latent space** ?

Motivation

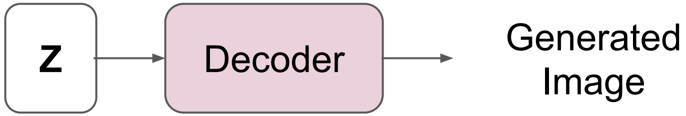
Training the VAE model (encoder + decoder)

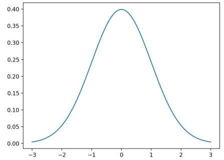


Force the Z distribution to be N-dim (N=latent space dimension) Gaussian :

- **Gain in memory** : no need to store the latent space histograms
- **Easy generation** : randomly sample from a N-dim Gaussian

Generation



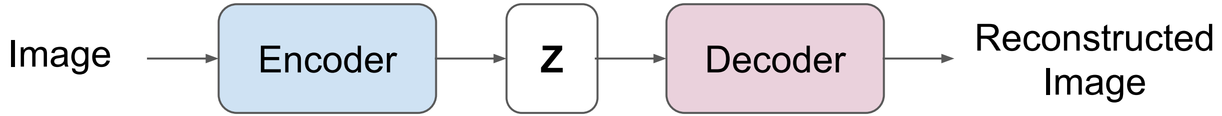


Models' architecture

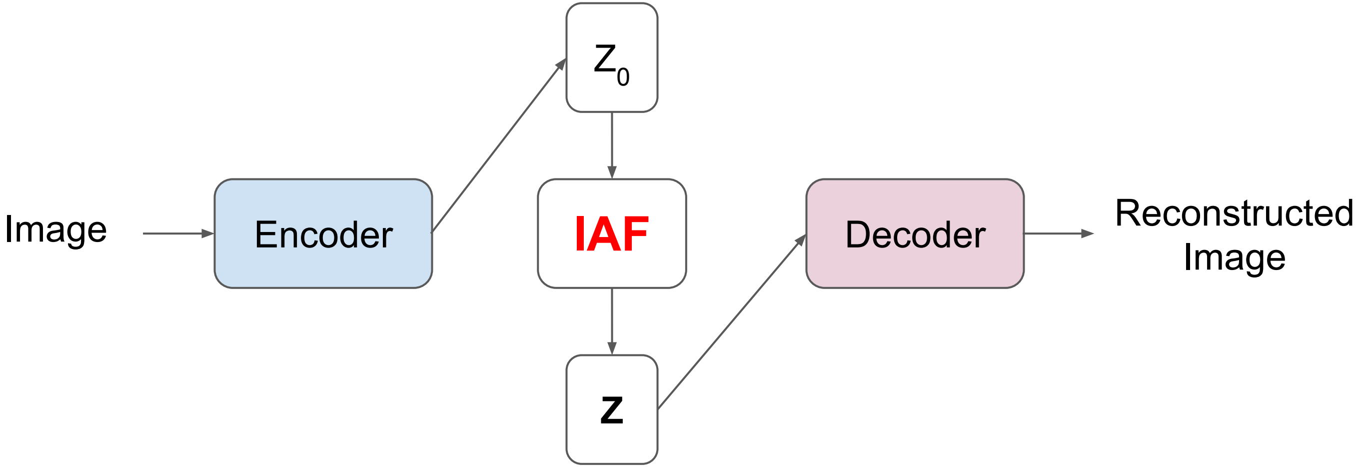
Info training dataset + parameters

- Photons 65 GeV energy in 0.2<|η|<0.25
- 9k Geant4 full simulation events in EMB2 (7x7 energy grid)
- Latent space dimension = 5
- Simplified version of the PUB note model

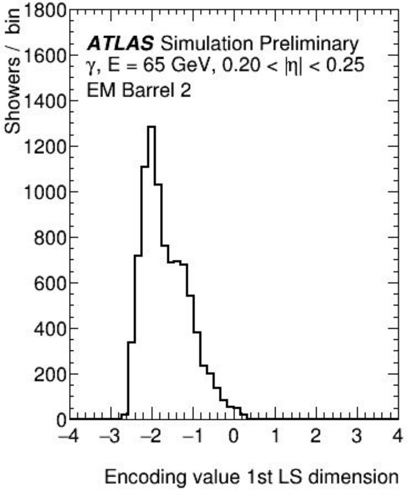
Model 1 : Training the VAE model (encoder + decoder)

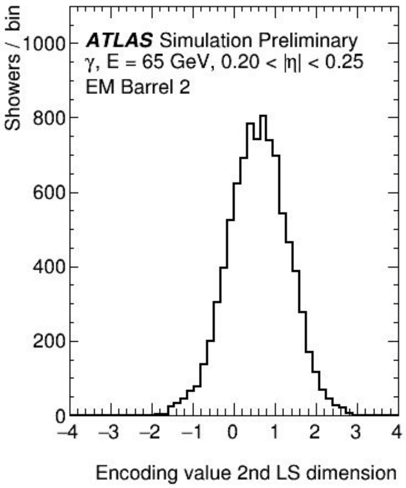


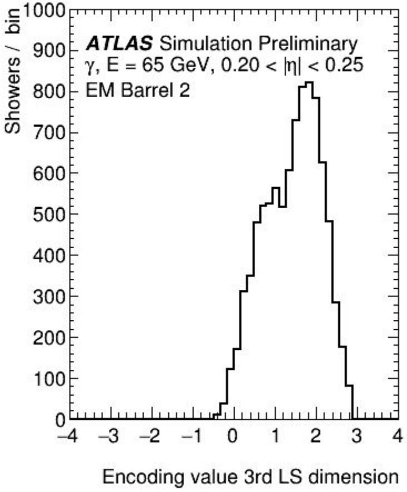
Model 2 : Training the Inverse Autoregressive Flow (**IAF**) VAE model (encoder + decoder)

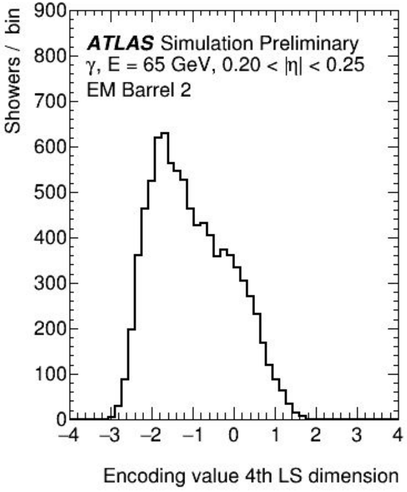


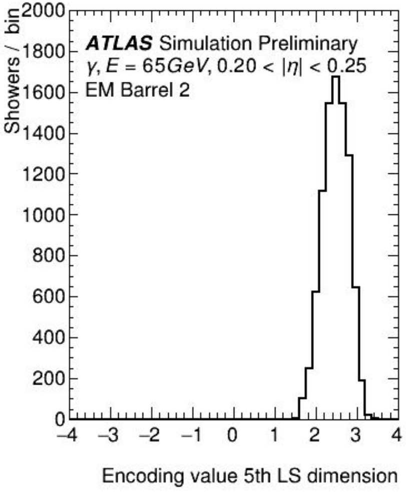
5D latent space variable distributions [From Model 1]











- Latent space distributions are not well modelled by Gaussians

https://atlas.web.cern.ch/Atlas/GROUPS/PHYSICS/PLOTS/SIM-2019-003/

[Rezende and Mohamed, 2016]

Variational Inference with Normalizing Flows

- Invertible transformations with a tractable Jacobian.
- Transforming the PDF
  - **x** n-dimensional random variable, **f(x)** joint density function, x can be transformed to y via an **invertible** (1-to-1) and **differentiable** function **H** with joint density **g(y)**.

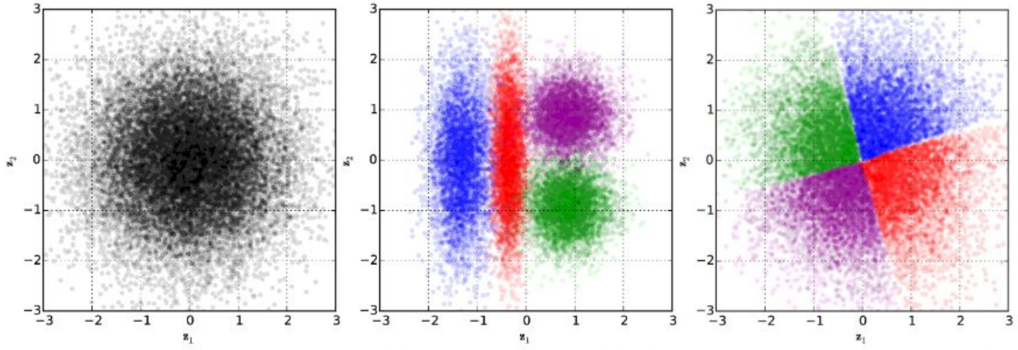
y = H(x)

$$g(y) = f(x)\left|\det\left(\frac{d\mathbf{x}}{d\mathbf{y}}\right)\right|$$
$$= f(H^{-1}(\mathbf{y}))\left|\det\left(\frac{dH^{-1}(\mathbf{y})}{d\mathbf{y}}\right)\right|$$

- Transform an initial distribution with a sequence of invertible mappings
  - Sample from a simple distribution (eg.diagonal gaussian) : $z_0 \in R^D$
  - Apply a sequence of invertible transformations : $f_k : R^D \rightarrow R^D$
  - $z_0 \rightarrow z_k$, $z_k = \mathbf{f_k} \circ \ldots \circ \mathbf{f_2} \circ \mathbf{f_1}(\mathbf{z_0})$

Inverse Autoregressive transformations

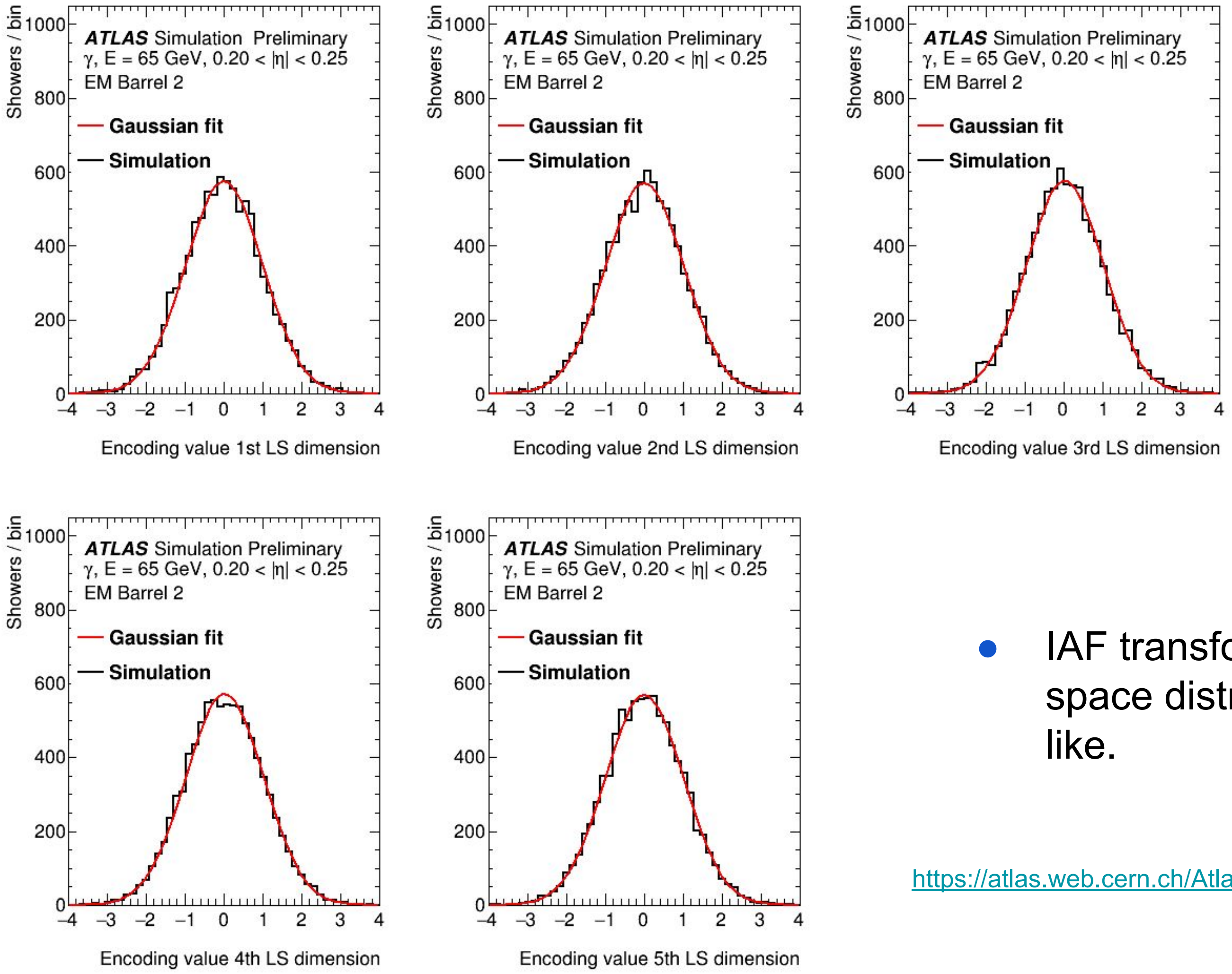
- Inverse Autoregressive Flow (IAF) is a type of Normalizing Flow.
- The flow consists of a chain of invertible transformations, each transformation is based on an **gaussian autoregressive function** :
  - Input : a variable with some specified ordering (multidimensional tensor)
  - Output : (μ,σ) for each element of the input variable conditioned on the previous elements.



(a) Prior distribution (b) Posteriors in standard VAE (c) Posteriors in VAE with IAF

[Kingma et al, 2017]

5D latent space variable distributions [From Model 2]



- IAF transformations make the latent space distributions more Gaussian like.

https://atlas.web.cern.ch/Atlas/GROUPS/PHYSICS/PLOTS/SIM-2019-003/

Conclusion & Outlook

- Proof of concept for generative deep learning models for simulating particle showers.
- Promising results and active development towards achieving required accuracy.
- Latent space investigated in more detail
  - Non-gaussian in simple VAE case, so Gaussian sampling unlikely to give good results in generation
  - IAF VAE gives much nicer latent space properties - very Gaussian
    - Impact on physics under study

Backup

Hyperparameters optimization for VAE - PUB Note

Hyperparameter	Values
Latent space dim.	[1, ..., **10**, ..., 100]
Reco. weight	(0, ..., **1**, ..., 3]
KL weight	(0, ..., $\mathbf{10^{-4}}$, ..., 1]
E_{tot} weight	[0, ..., $\mathbf{10^{-2}}$, ..., 1]
E_i weights	[0, ..., $\mathbf{8 \times 10^{-2}}$, ..., 1]
	[0, ..., $\mathbf{6 \times 10^{-1}}$, ..., 1]
	[0, ..., $\mathbf{2 \times 10^{-1}}$, ..., 1]
	[0, ..., $\mathbf{10^{-1}}$, ..., 1]
Hidden layers (encoder)	1, 2, 3, **4**, 5
Hidden layers (decoder)	1, 2, 3, **4**, 5
Units per layer	[180, ..., **200**, ..., 266]
	[120, ..., **150**, ..., 180]
	[80, ..., **100**, ..., 120]
	[10, ..., **50**, ..., 80]
Activation func.	**ELU**, ReLU, SELU, LeakyReLU, PReLU
Kernel init.	zeros, ones, random normal, random uniform, truncated normal, **variance scaling**, `glorot_normal`
Bias init.	zeros, **ones**, random normal, random uniform, truncated normal, variance scaling, `glorot_normal`
Optimizer	**RMSprop**, Adam, Adagrad, Adadelta, Nadam
Learning rate	[10^{-2}, ... ,$\mathbf{10^{-4}}$, ... , 10^{-6}]
Mini-batch size	50, **100**, 150 , 1000