Deep Generative Models in Particle Physics

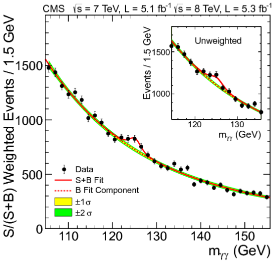
— Physics in the AI era, University of Pisa —

Claudius Krause

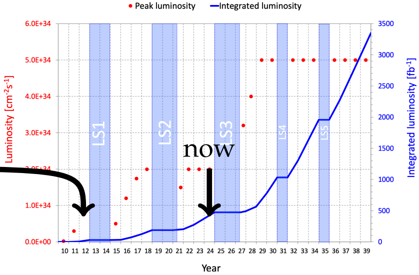
Institute of High Energy Physics (HEPHY), Austrian Academy of Sciences (OeAW)

September 27, 2024

We will have a lot more data in the near future.



2012

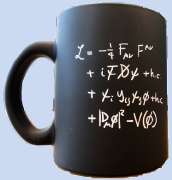


CMS Collaboration [arXiv:1207.7235, Phys.Lett.B]

https://lhc-commissioning.web.cern.ch/schedule/HL-LHC-plots.htm

- We will have 20× more data.

⇒ We want to understand every aspect of it based on 1st principles! (and find New Physics)



A (simplified) View on Particle Physics Analyses

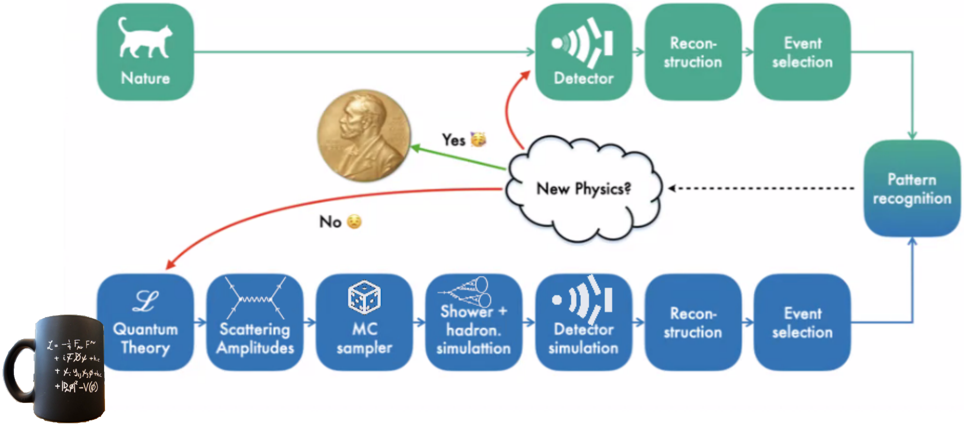


Figure by R. Winterhalder

Deep Generative (DGMs) Models are Random Number Generators

DGMs are ML models that "generate" new samples of a (complicated) $p(x)$.

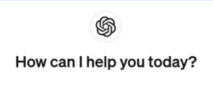
They can be understood as fancy random number generators, with the numbers being:

- pixels of an image

"Albert Einstein smiling while having fun coding" via `midjourney.com`

$\Rightarrow$ image generators like `MidJourney`, `DALL·E`

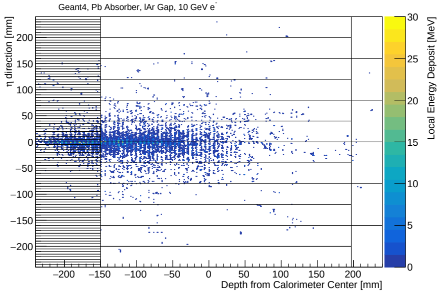
- translated to words

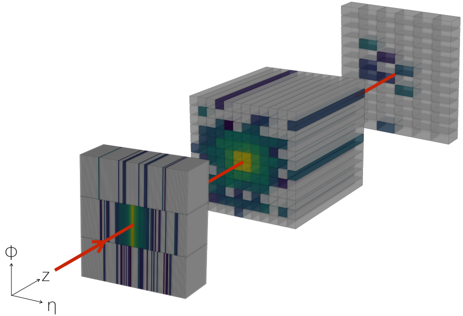


$\Rightarrow$ chatbots like `ChatGPT`, `GitHub CoPilot`

DGMs can help to speed-up bottlenecks in simulation

- particle – matter interactions are stochastic: described by $p(\text{shower}|\text{init. cond.})$
- Example: particle showers in the calorimeters
  DGMs generate samples $\sim 10,000\times$ faster than a physics simulation with GEANT4.



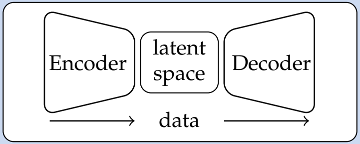


First study on toy dataset: CaloGAN by Paganini, de Oliveira, Nachman [1705.02355, PRL; 1712.10321, PRD]

The Landscape of Generative Models.

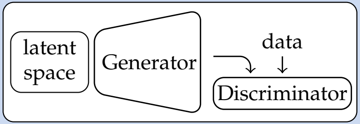
Variational Autoencoder (VAE)

⇒ Compressing data through a bottleneck.



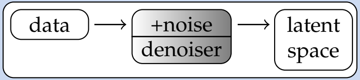
Generative Adversarial Network (GAN)

⇒ Generator and Discriminator play a game against each other.



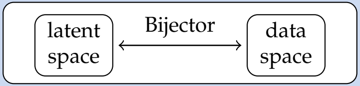
Diffusion Models

⇒ Gradually add noise and revert.



Normalizing Flows

⇒ Bijective map to a known distribution.



All types of DGMs are used for detector simulation

Table 2: Current top 15 SOTA models based on the granularity of the signatures they can generate. This list does not provide a fair comparison for the surrogate models simulating jet signatures as they inherently carry rather low granularities.

Model	Algorithm	Representation	Conditioning	Experiment	**Granularity ↑**
IEA-GAN [69, 141]	GAN	grid/set	sensor position (radius and angle)	Belle II PXD (2023,2021)	$40\times250\times768 = \mathbf{7,680,000}$ ch
WGAN [142]	GAN	grid	random	Belle II PXD (2019)	$40\times250\times768 = \mathbf{7,680,000}$ ch
YonedaVAE [28]	VAE/ARM	multi-set	sensor position and Luminosity	Belle II PXD (2023)	$\mathbf{110,000}$ points
3DGAN [143, 144]	GAN	grid	incident energy and angle	CLIC ECAL (2021, 2020)	$25 \times 51 \times 51 = \mathbf{65,025}$ ch
BIB-AE [133]	VAE/GAN/NF	grid	incident energy and angle	ILD ECAL (2023)	$30 \times 60 \times 30 = \mathbf{54,000}$ ch
CaloScore v2 [145]	Diffusion	grid	incident energy and time information	CaloChallenge D3 (2023)	$45 \times 50 \times 18 = \mathbf{40,500}$ ch

Hashemi/Krause [arXiv:2312.09597, Rev.Phys.]

All types of DGMs are used for detector simulation

Table 2: Current top 15 SOTA models based on the granularity of the signatures they can generate. This list does not provide a fair comparison for the surrogate models simulating jet signatures as they inherently carry rather low granul

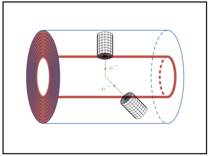
How can we compare them to each other?

Model					rity ↑
IEA-GAN [69, 141]	GAN		tion (radius and angle)	II PXD (2023,2021)	$40\times250\times768 =$ $\mathbf{7,680,000}$ ch
WGAN [142]	GAN	grid	random	Belle II PXD (2019)	$40\times250\times768 =$ $\mathbf{7,680,000}$ ch
YonedaVAE [28]	VAE/ARM	multi-set	sensor position and Luminosity	Belle II PXD (2023)	$\mathbf{110,000}$ points
3DGAN [143, 144]	GAN	grid	incident energy and angle	CLIC ECAL (2021, 2020)	$25\times51\times51 =$ $\mathbf{65,025}$ ch
BIB-AE [133]	VAE/GAN/NF	grid	incident energy and angle	ILD ECAL (2023)	$30\times60\times30 =$ $\mathbf{54,000}$ ch
CaloScore v2 [145]	Diffusion	grid	incident energy and time information	CaloChallenge D3 (2023)	$45\times50\times18 =$ $\mathbf{40,500}$ ch

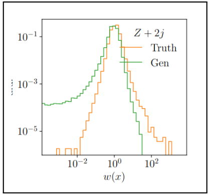
Hashemi/Krause [arXiv:2312.09597, Rev.Phys.]

Deep Generative Models in Particle Physics

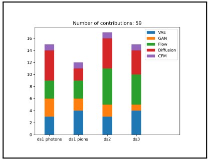
I: Common Datasets



II: Evaluation Metrics



III: Results



Let's go back to 2022 ...

- The immense progress of ML in the past decade led to awesome results for calorimeter simulation surrogates!

⇒ We have seen the use of GANs, VAEs, Normalizing Flows, Diffusion models, and their derivates on a variety of datasets.

ATLAS toy dataset
CaloGAN, CaloFlow

ILD dataset
BIB-AE, L2LFlows

ATLAS official dataset
FastCaloGAN, AtlFast3

...
...

⇒ No systematic comparison of methods available!

Introducing: Fast Calorimeter Simulation Challenge 2022

Why a challenge?

- Evaluate existing models on common datasets.

$\Rightarrow$ A challenge creates a survey of DGMs with pros and cons.

$\Rightarrow$ Winners are strong candidates for the new generation of FastSim.

- Trigger development of new generative models.

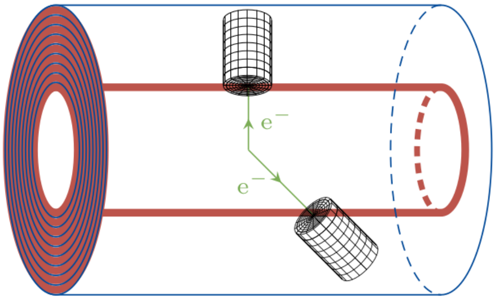
$\Rightarrow$ The datasets will also be benchmarks for new models in the future.

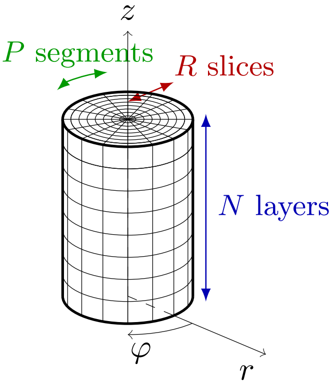
- Improve our understanding of common struggles, advantages, disadvantages, and scaling behavior.
- Learn about the evaluation of DGMs.

- Previous challenges on top tagging and anomaly detection were very successful.

CaloChallenge Showers are voxelized in cylindrical coordinates.

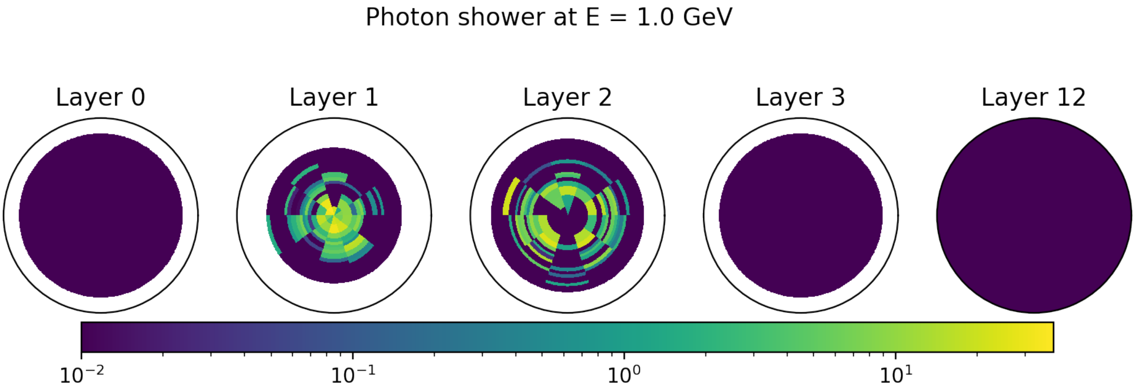
- There 4 datsets in increasing complexity / dimensionality.
- Particles enter perpendicular to front surface:





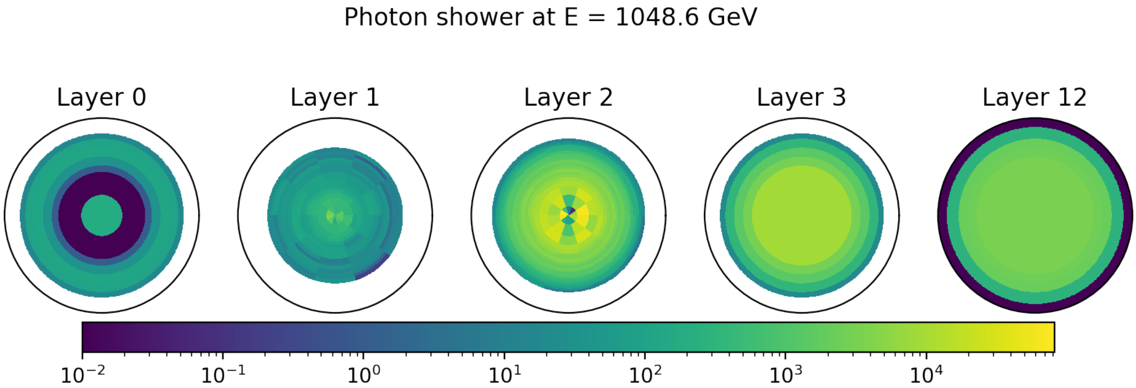
CaloChallenge Showers are voxelized in cylindrical coordinates.

- Showers are usually sparse.
- Energy depositions span several orders of magnitude.



CaloChallenge Showers are voxelized in cylindrical coordinates.

- Showers are usually sparse.
- Energy depositions span several orders of magnitude.



The Fast Calorimeter Simulation Challenge 2022

The main task: $\boxed{\text{Develop a model that samples from } p(\text{shower}|E_{\text{incident}})}$

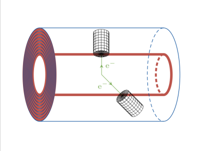
https://calochallenge.github.io/homepage/

Michele Faucci Giannelli, Gregor Kasieczka, **CK**, Ben Nachman, Dalila Salamani, David Shih, and Anna Zaborowska

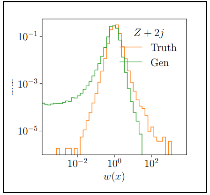
- Dataset 1: AtlFast3 trainig data (γ: 368, π: 533 voxels) $E_{\text{inc}} \in [256\text{ MeV}, 4.2\text{ TeV}]$
  [2109.02551, Comput.Softw.Big Sci.]
- Dataset 2: Par04 simulated detector (e^-: 6480 voxels) $E_{\text{inc}} \in [1\text{ GeV}, 1\text{ TeV}]$
- Dataset 3: Par04 simulated detector (e^-: 40500 voxels) $E_{\text{inc}} \in [1\text{ GeV}, 1\text{ TeV}]$

Deep Generative Models in Particle Physics

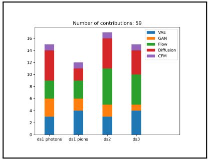
I: Common Datasets



II: Evaluation Metrics



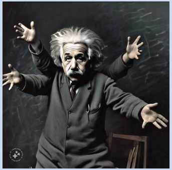
III: Results



How to evaluate generative models?

In text / image / video generation: "by eye".
$\Rightarrow$ Our brains are incredible good at this task, but it doesn't scale.

imagined with Meta AI.



In high-energy physics: need to find something better!
$\Rightarrow$ We want to correctly cover $p(x)$ of the entire phase space.

1. Can look at histograms of derived features / observables.

$\Rightarrow$ To quantify, we use the *separation power* of high-level feature histograms:

$$S(h_1, h_2) = \frac{1}{2} \sum_{i=1}^{n_{\text{bins}}} \frac{(h_{1,i} - h_{2,i})^2}{h_{1,i} + h_{2,i}}$$

But: this is just a 1-dim projection!

A Classifier provides the “ultimate metric”.

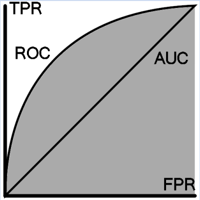
According to the Neyman-Pearson Lemma we have:

- The likelihood ratio is the most powerful test statistic to distinguish two samples.
- A powerful classifier trained to distinguish the samples should therefore learn (something monotonically related to) $w = \frac{p_{\text{data}}}{p_{\text{model}}}$.
- If this classifier is confused, we conclude $\Rightarrow$ $p_{\text{data}}(x) = p_{\text{model}}(x)$

$\Rightarrow$ This captures the full phase space incl. correlations. CK/D. Shih [2106.05285, PRD]

2. Now, the AUC provides a single number to compare different models.

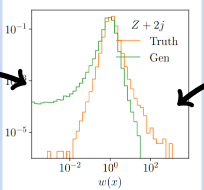
   But: are AUCs of different models really comparable?



A Classifier tells us much more about the model.

Failure modes of the model can now be seen in the $w = \frac{p_{\text{data}}}{p_{\text{model}}}$ histogram:

Data manifold over-populated by model: $\Rightarrow$ missmodeled feature



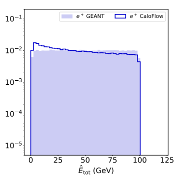
Data manifold not populated by model: $\Rightarrow$ missed feature

R. Das, CK, et al. [2305.16774, SciPost]

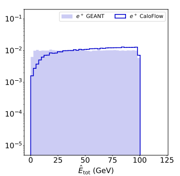
Cluster plots show where events lie in phase space:

figures by B. Schmidthaler / M. Rosendorf

small weights:



large weights:



How to decide which model is closest to the reference: the Multiclass Classifier

A multi-class classifier:

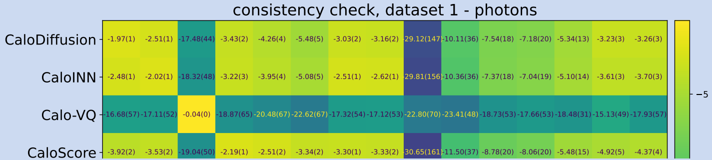
Train on submission 1 vs. submission 2 vs. … vs. submission n
and evaluate the *log posterior*:

$$L = \langle \log \left(p(x_{\in \text{class } i} | x_{\text{taken from } j}) \right) \rangle \qquad j \in \{\text{submission } k, \textsc{Geant4}\}$$

3. As metric: evaluate with GEANT4

Lim et al. [2211.11765, MNRAS]

As cross-check: validate with all submissions j



Other important metrics to look at.

⇒ The *generation time*.

- on CPU/GPU architectures
- for batch sizes 1 / 100 / 10000

⇒ The *number of trainable parameters*.

- as proxy for model size
- in training / generation

Other important metrics to look at.

⇒ The *generation time*.

- on CPU/GPU architectures
- for batch sizes 1 / 100 / 10000

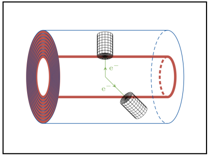
- start singularity container
- load model weights + biases
- generate samples
- save them to `.hdf5`

⇒ The *number of trainable parameters*.

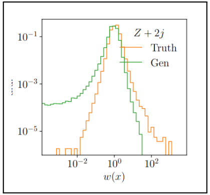
- as proxy for model size
- in training / generation

Deep Generative Models in Particle Physics

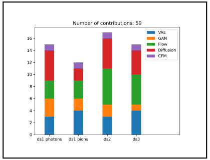
I: Common Datasets



II: Evaluation Metrics



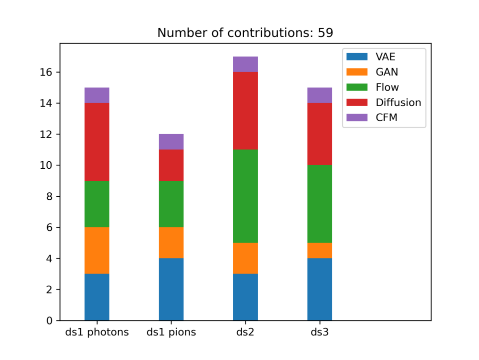
III: Results



A little disclaimer: the ~~final~~ preliminary results of the CaloChallenge

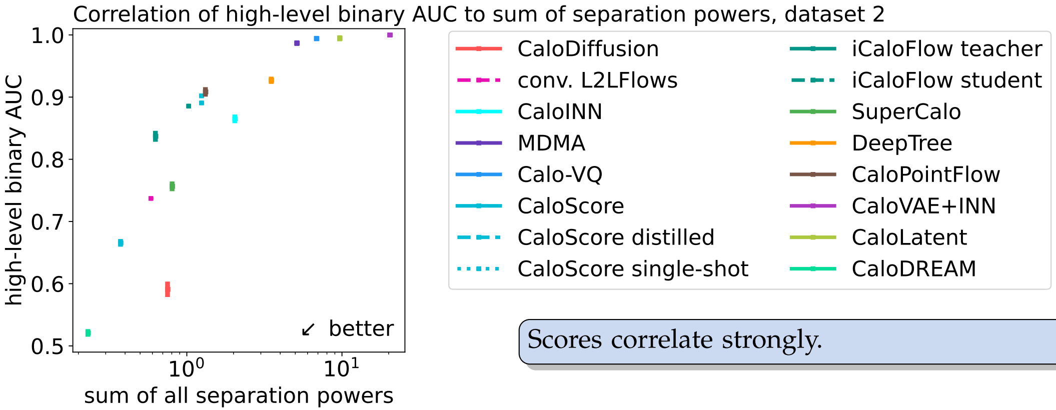
In the following, I will share preliminary results of the CaloChallenge.

The final write-up will be ready in a few weeks, with a lot more content.



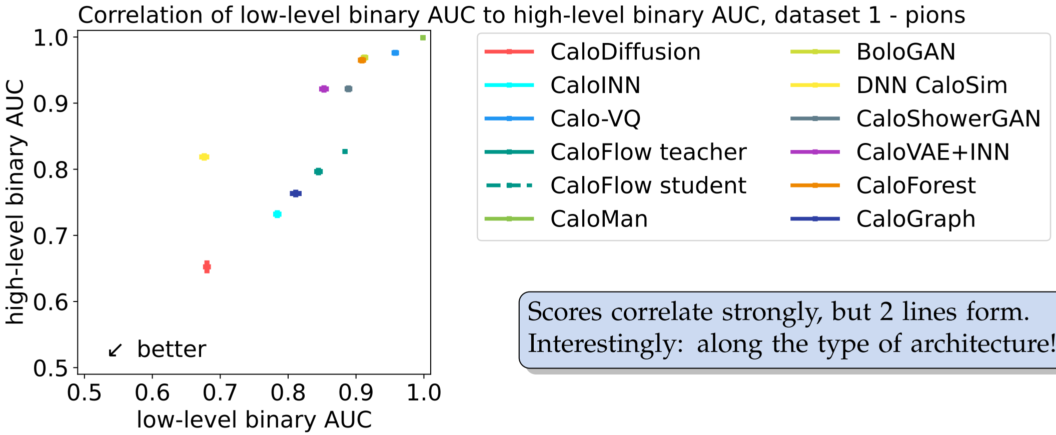
- We received 59 submissions for all datasets.
- They were generated by 23 different models.
- All types of DGM architectures were used.

Comparing different quality metrics: high-level features



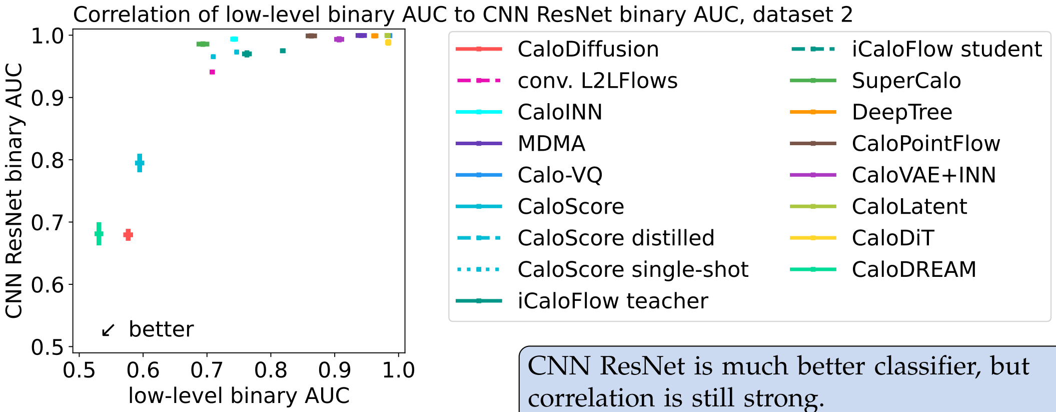
Scores correlate strongly.

Comparing different quality metrics: classifier input



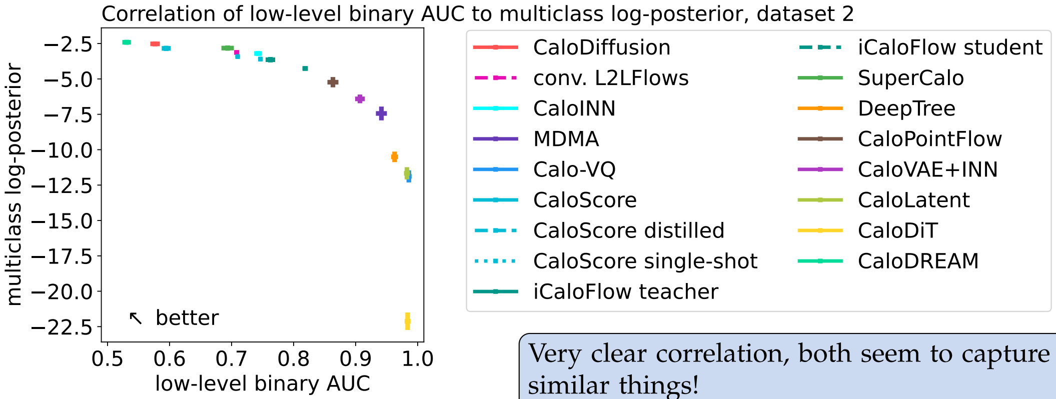
Scores correlate strongly, but 2 lines form. Interestingly: along the type of architecture!

Comparing different quality metrics: classifier architecture



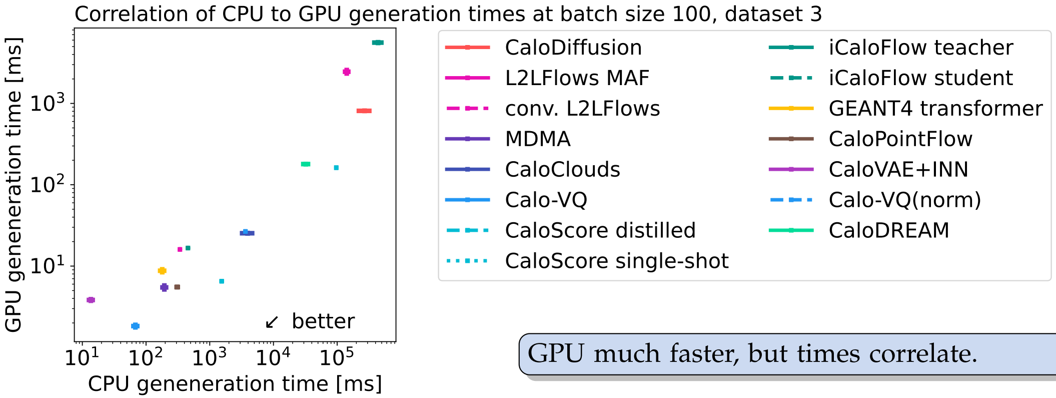
CNN ResNet is much better classifier, but correlation is still strong.

Comparing different quality metrics: binary vs. multiclass



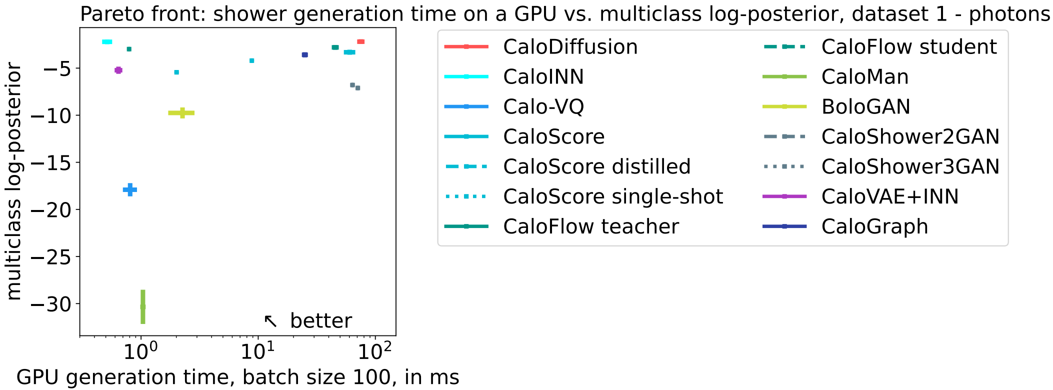
Very clear correlation, both seem to capture similar things!

Comparing different timing metrics: CPU vs. GPU

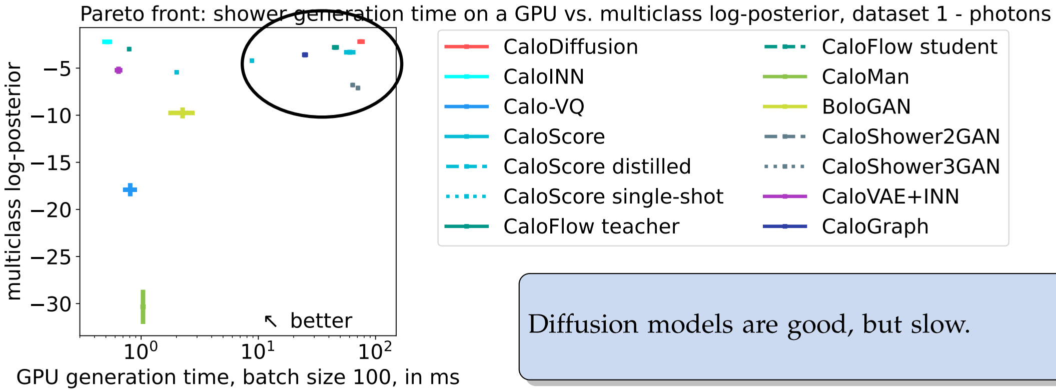


GPU much faster, but times correlate.

Pareto Fronts: Quality vs. Generation Time

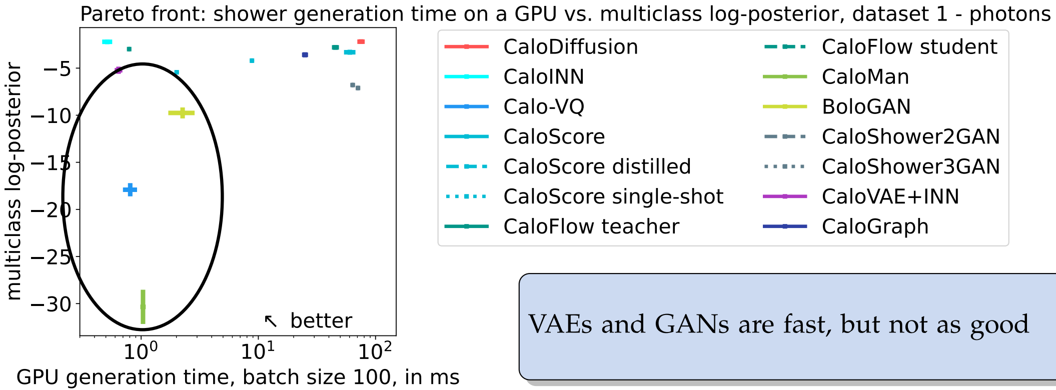


Pareto Fronts: Quality vs. Generation Time



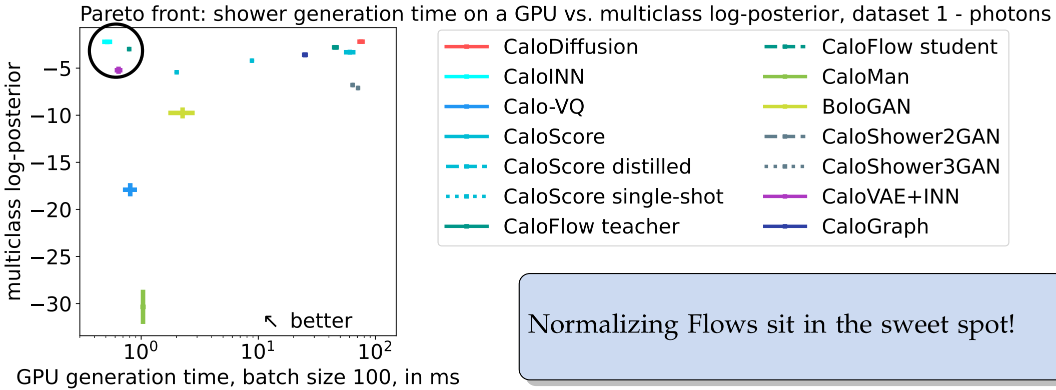
Diffusion models are good, but slow.

Pareto Fronts: Quality vs. Generation Time



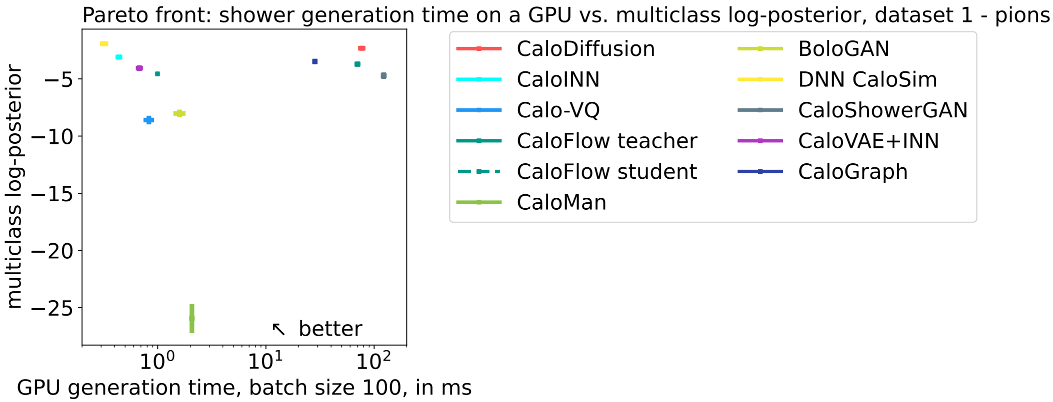
VAEs and GANs are fast, but not as good

Pareto Fronts: Quality vs. Generation Time

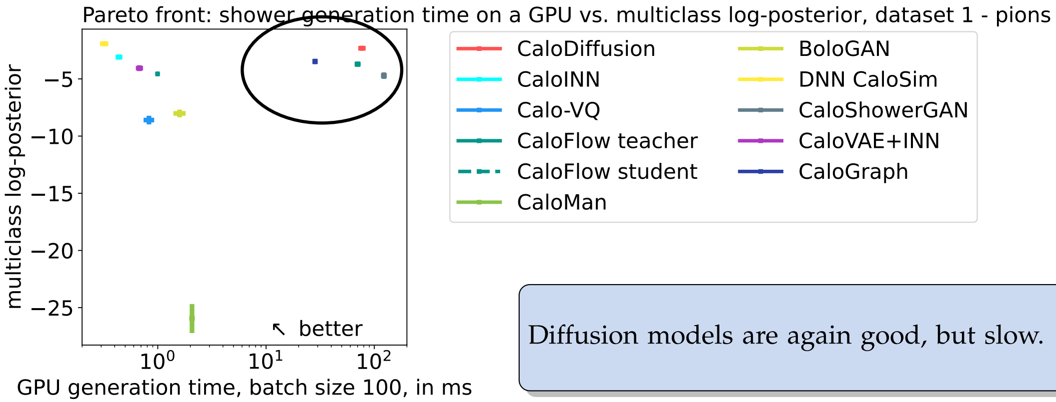


Normalizing Flows sit in the sweet spot!

Pareto Fronts: Quality vs. Generation Time

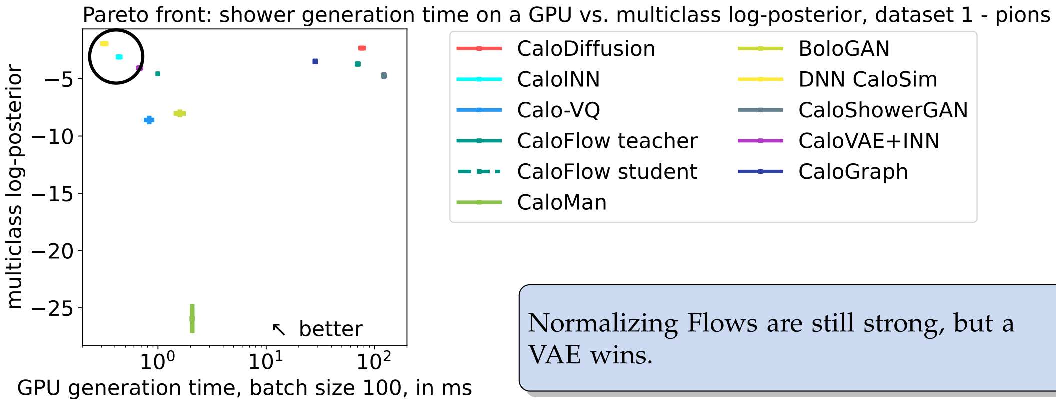


Pareto Fronts: Quality vs. Generation Time



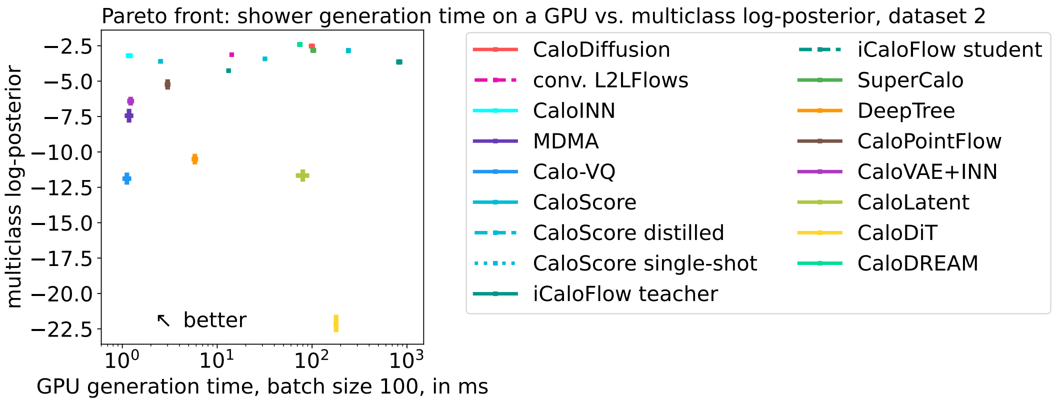
Diffusion models are again good, but slow.

Pareto Fronts: Quality vs. Generation Time

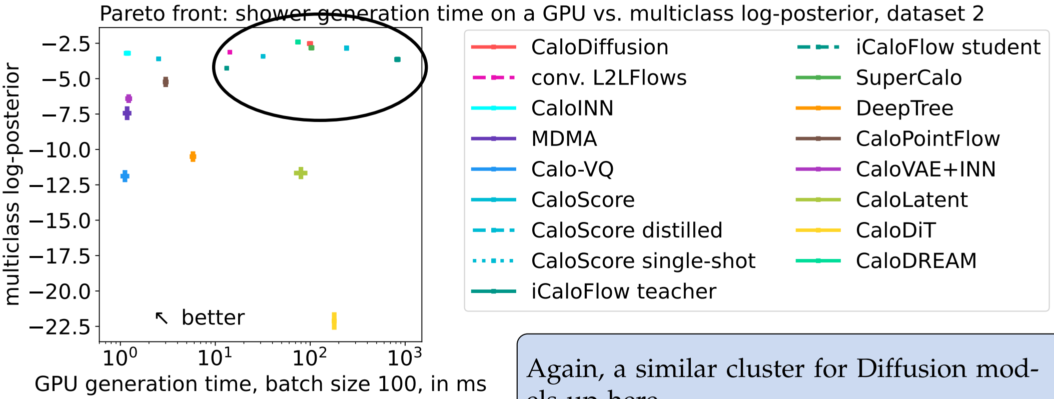


Normalizing Flows are still strong, but a VAE wins.

Pareto Fronts: Quality vs. Generation Time

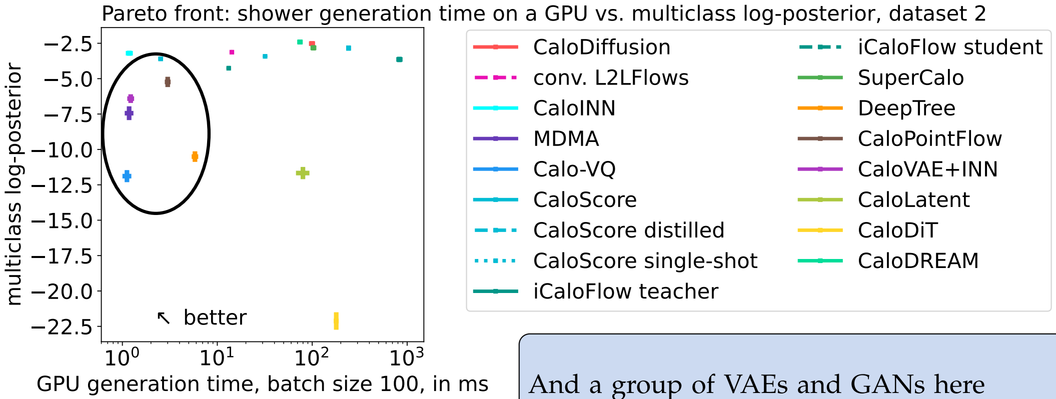


Pareto Fronts: Quality vs. Generation Time



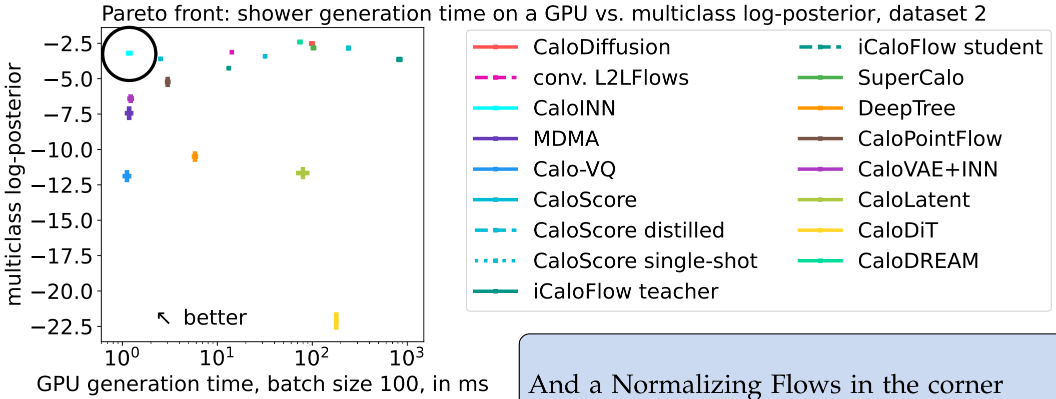
Again, a similar cluster for Diffusion models up here.

Pareto Fronts: Quality vs. Generation Time



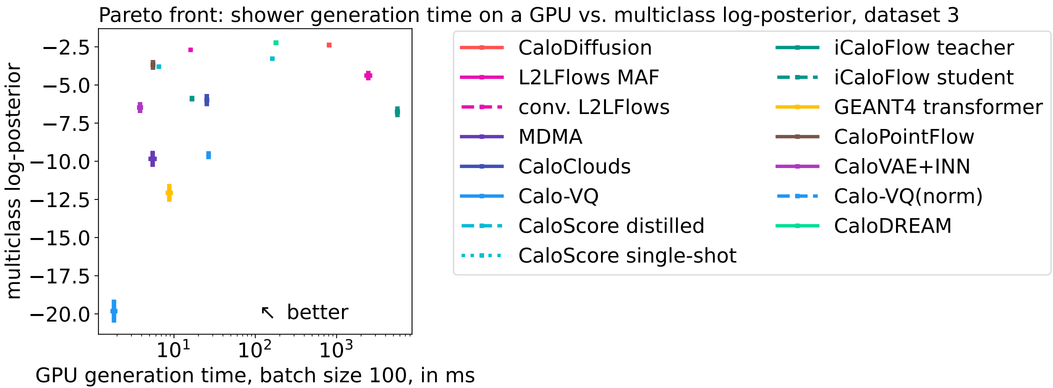
And a group of VAEs and GANs here

Pareto Fronts: Quality vs. Generation Time

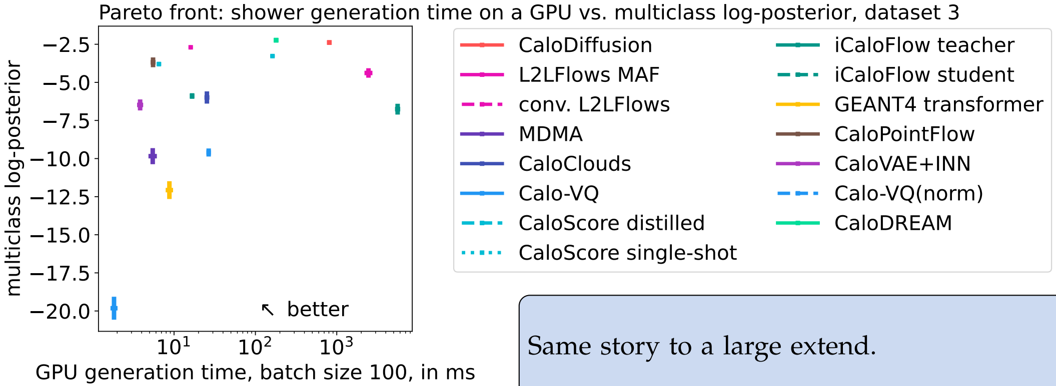


And a Normalizing Flows in the corner

Pareto Fronts: Quality vs. Generation Time

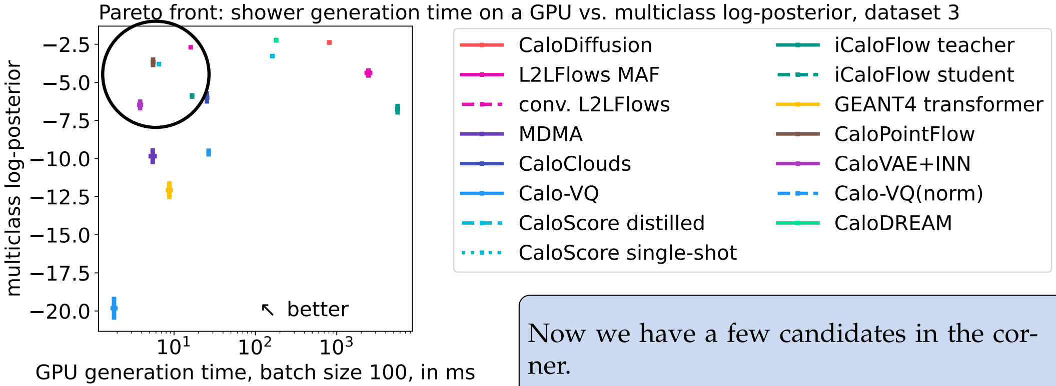


Pareto Fronts: Quality vs. Generation Time



Same story to a large extend.

Pareto Fronts: Quality vs. Generation Time



Now we have a few candidates in the corner.

Deep Generative Models in Particle Physics

- DGMs will play an important role HEP simulation in the next years.
- There are lots of different use cases and architectures.
- For deployment, we need to ensure they are faithful on the entire phase space!

$\Rightarrow$ We require evaluation tools that capture everything.

- I introduced classifiers for this job.

Deep Generative Models in Particle Physics

- DGMs will play an important role HEP simulation in the next years.
- There are lots of different use cases and architectures.
- For deployment, we need to ensure they are faithful on the entire phase space!

$\Rightarrow$ We require evaluation tools that capture everything.

- I introduced classifiers for this job.

So, where do we stand now?

- A challenge provides the perfect setting to survey the state-of-the-art.
- I showed correlations between metrics and Pareto Fronts of current DGMs based on the CaloChallenge datasets.