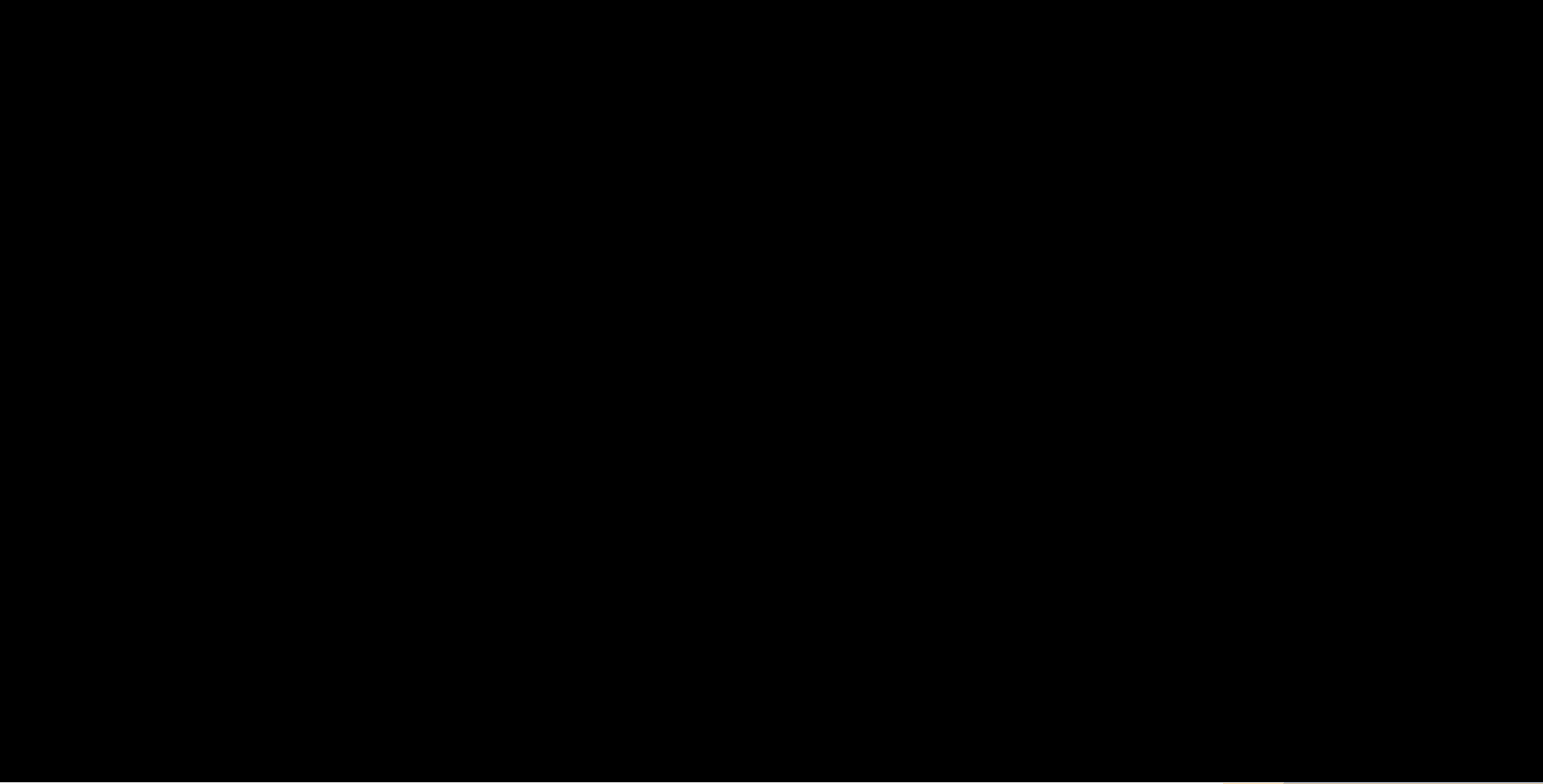
Machine learning methods for collider physics

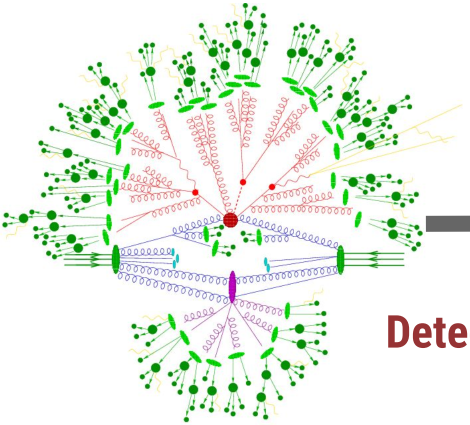
Vinicius M. Mikuni



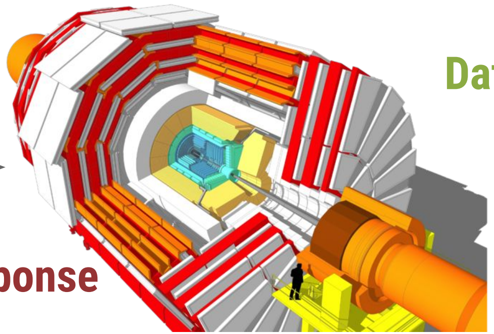
Particle colliders

Animation from [business insider]

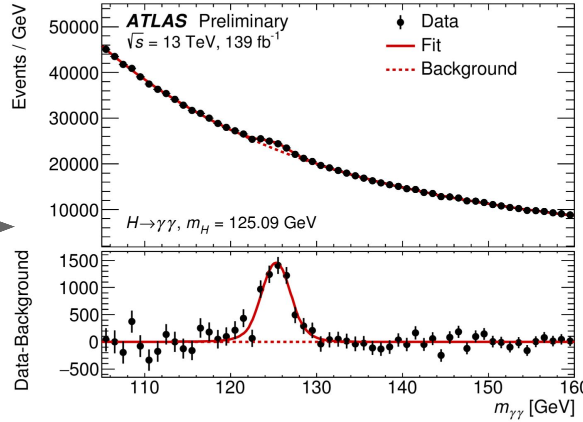
A generic collider physics workflow



Detector response



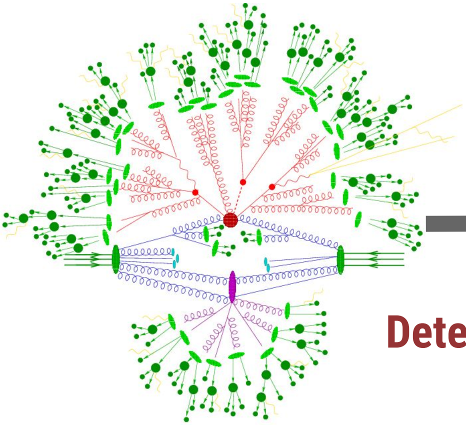
Data analysis



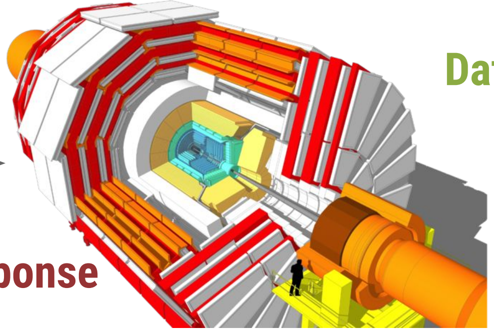
Picture from arXiv:1411.4085

A generic collider physics workflow

Surrogate



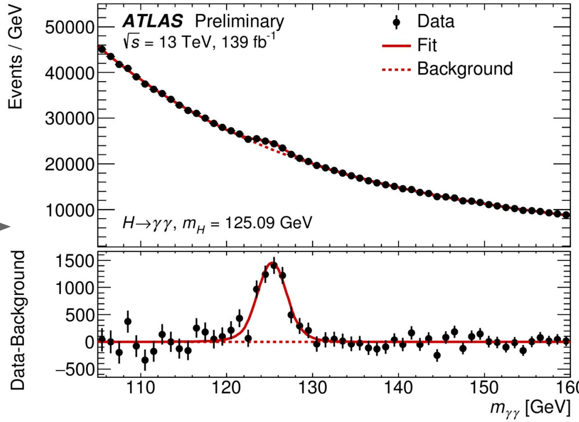
Detector response



Unfolding

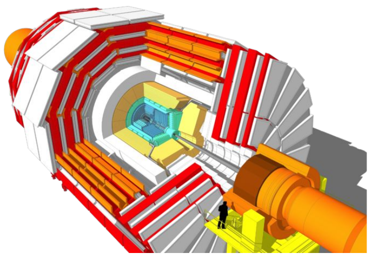
Data analysis

Anomaly detection



Picture from arXiv:1411.4085

Surrogate modeling for detector simulation



- Detector simulation is **computationally expensive**:
  - Full detector simulation of a particle can take up to **a minute** and we still need **billions of particles simulated**
- For previous LHC runs, detector simulation used around **40% of all computing resources** and may go beyond the available budget for future runs

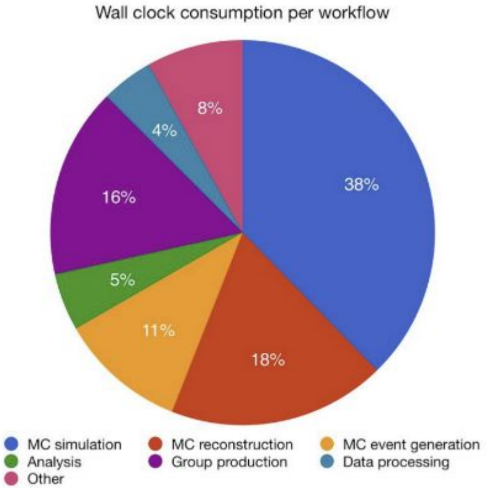
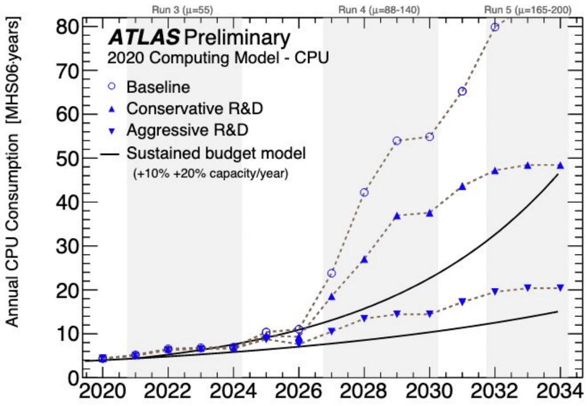
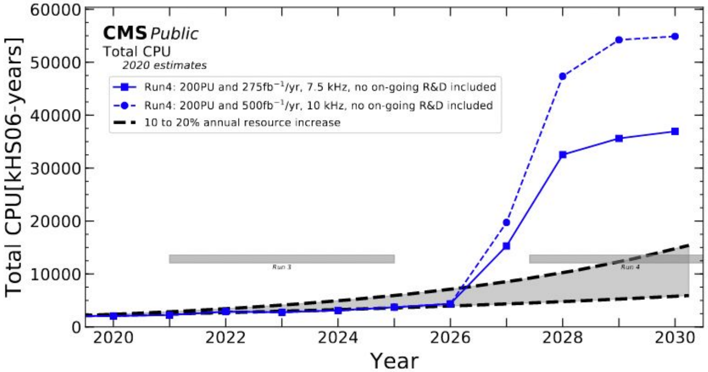
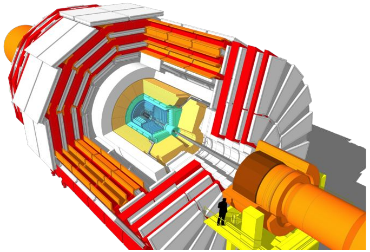


Figure 1: ATLAS CPU hours used by various activities in 2018

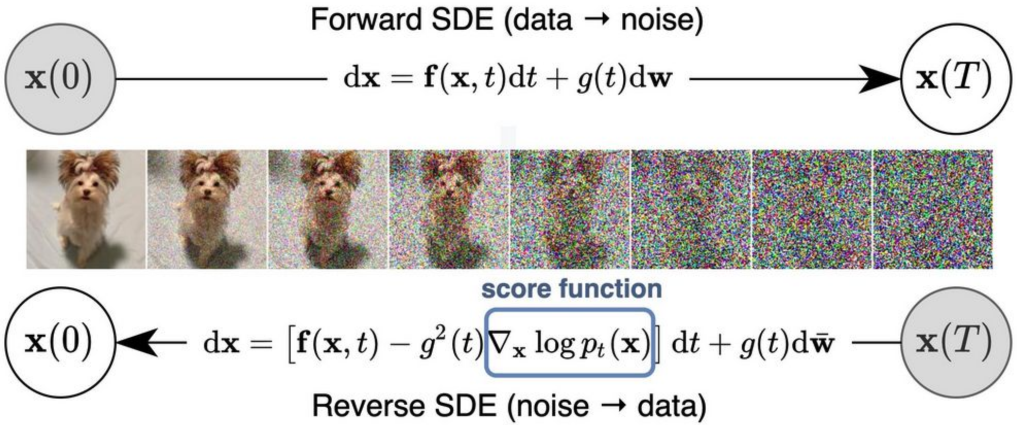


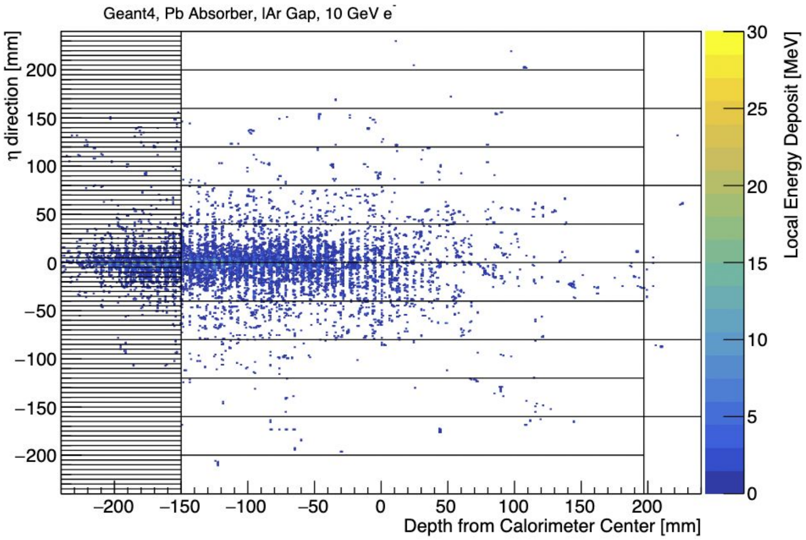


Surrogate modeling for detector simulation

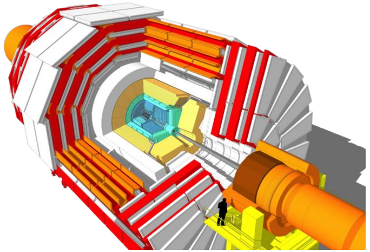


- Replace the calorimeter simulation with a **surrogate model** that learns to reproduce the detector response
- Use new state-of-the-art generative model based on diffusion models: **data is slowly perturbed** by a noise and the **network learns** how to perform **denoising**





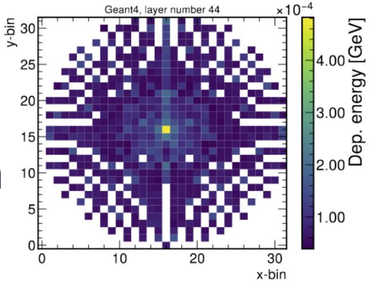
Surrogate modeling for detector simulation



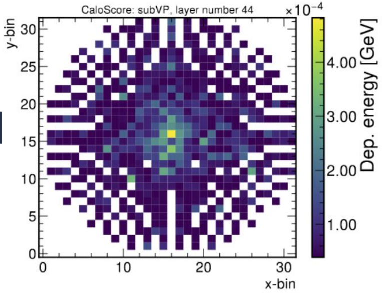
- Train on **Perlmutter** using **16 GPUs** at a time on datasets of different number of **pixels**:
  - **300**
  - **6500**
  - **46000**
- Approximately **4 hours** to train the model
- **Accurate** representation of the full simulation
- Around **10 times faster** than full simulation: can still get faster as speed is a general challenge for diffusion-based models

Dataset	N. of voxels	N. of weights	Time to 100 showers [s]		
			CaloScore	WGAN-GP	Geant
dataset 1	384	32M	4.0	1.3	$\mathcal{O}(10^2 - 10^3)$
dataset 2	6480	1.4M	5.8	1.33	$\mathcal{O}(10^4)$
dataset 3	46080	1.7M	33.4	2.06	$\mathcal{O}(10^4)$

Full simulation



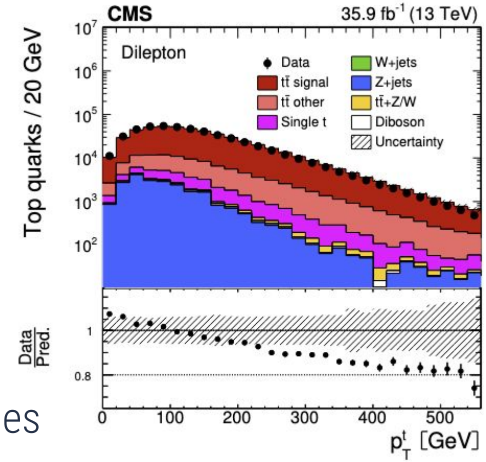
Generated



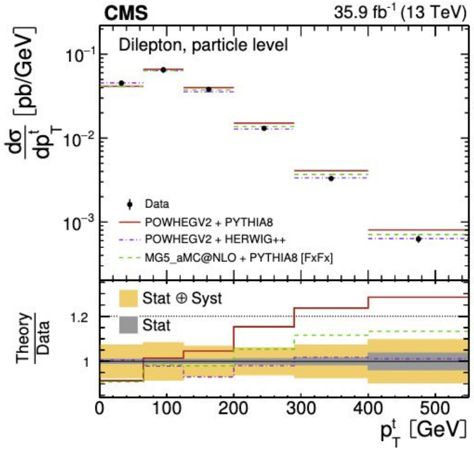
Detector unfolding

- The **opposite problem** is how to report physics measurements that are **corrected for detector effects: Unfolding**
- Easier to **compare between different theories**:
  - Don't require theorists to have expert detector knowledge to compare their predictions
  - Easier to maintain and incorporate new calibration routines for detector simulation
- Can also be seen as a **deconvolution** problem
- Standard methods require **histograms** of observables used as inputs
  - Can only correct **1 distribution at a time**
  - The **histogram cannot be modified** without redoing the full measurement

We have this



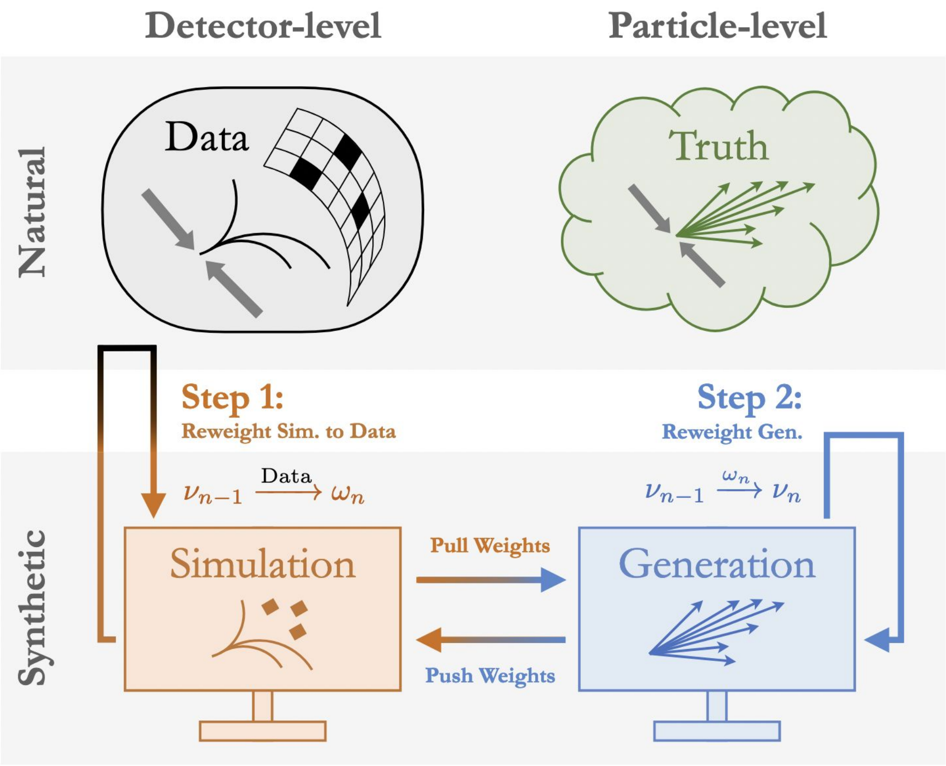
But want this



J. High Energ. Phys. **2019,** 149 (2019).

Omnifold*

* Andreassen et al. PRL 124, 182001 (2020)
For unfolding using **invertible networks** see:
- SciPost Phys. 9 (2020) 074 e-Print: [2006.06685](2006.06685)



ML is used to overcome these limitations

2 step iterative approach

- Simulated events after detector interaction are reweighted to match the data
- Create a “new simulation” by transforming weights to a proper function of the generated events

Machine learning is used to approximate **2** likelihood functions:

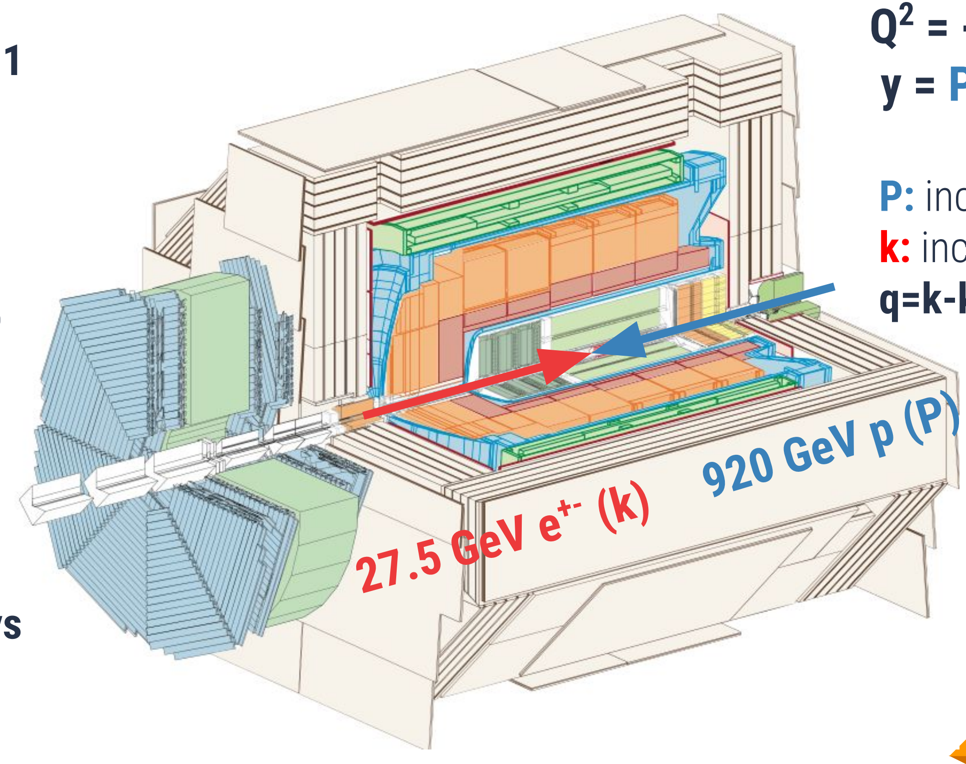
- **reconstructed simulation to Data** reweighting
- **Previous** and **new generated** reweighting

* Andreassen et al. PRL 124, 182001 (2020)

Experimental setup

Using data collected by the **H1 Experiment** during **2006** and **2007**

- Running on **Perlmutter** with **128 GPUs**
- Takes about **2 hours** to run
- Additional trainings required to estimate uncertainties: full measurement can be performed in a **few days**



$Q^2 = - q^2$

$y = Pq / pk$

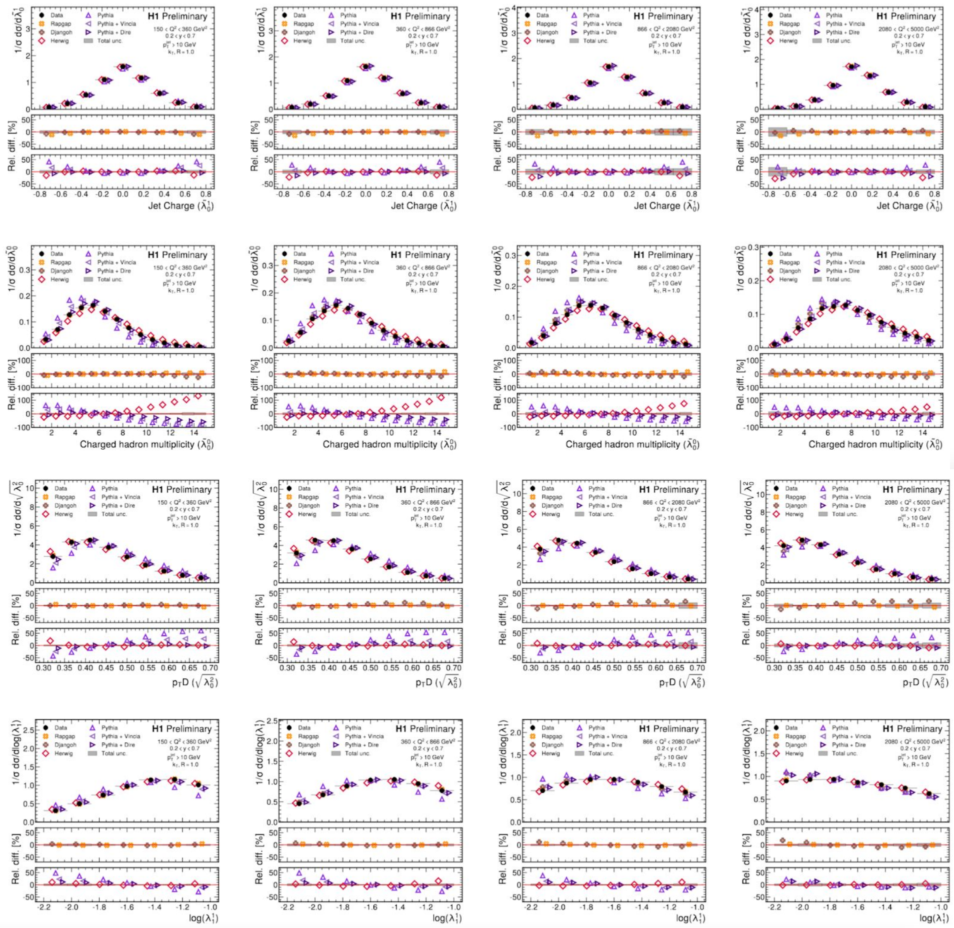
P: incoming proton 4-vector

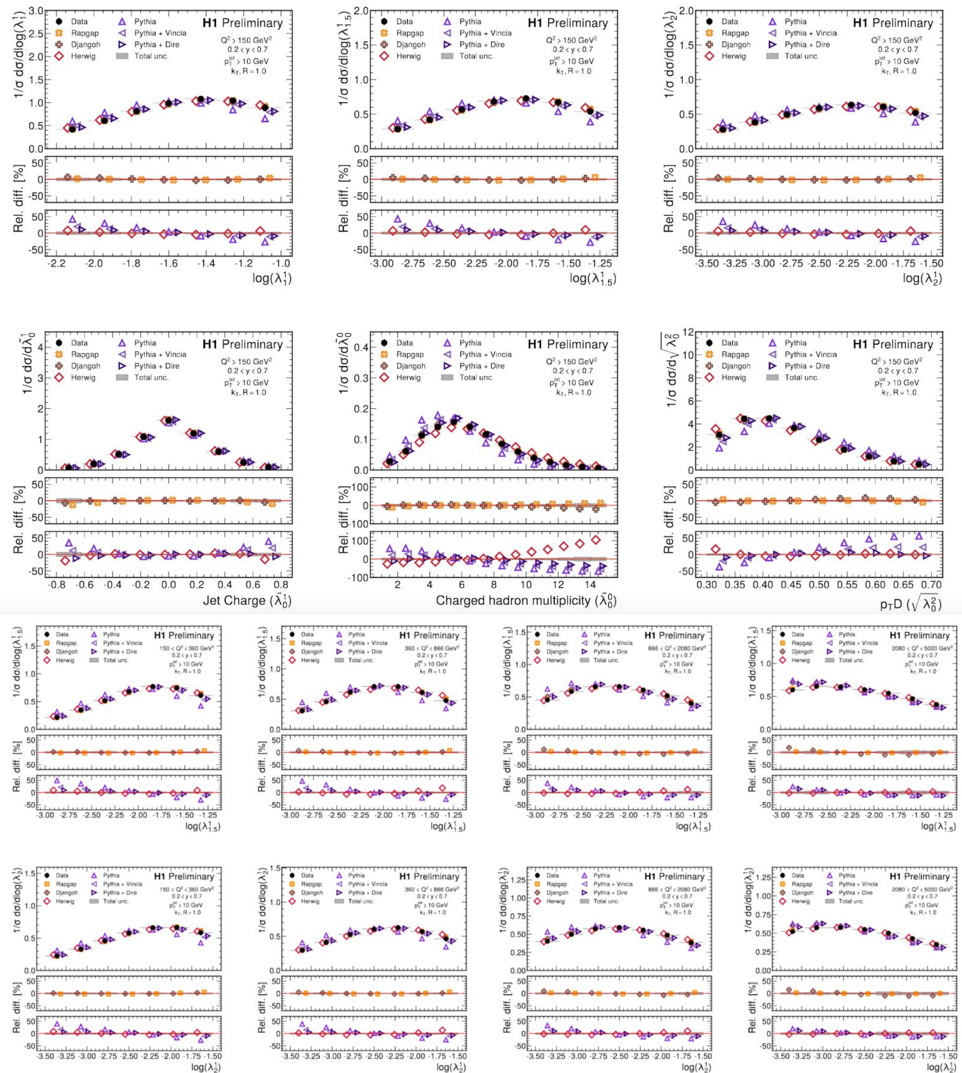
k: incoming electron 4-vector

q=k-k' : 4-momentum transfer

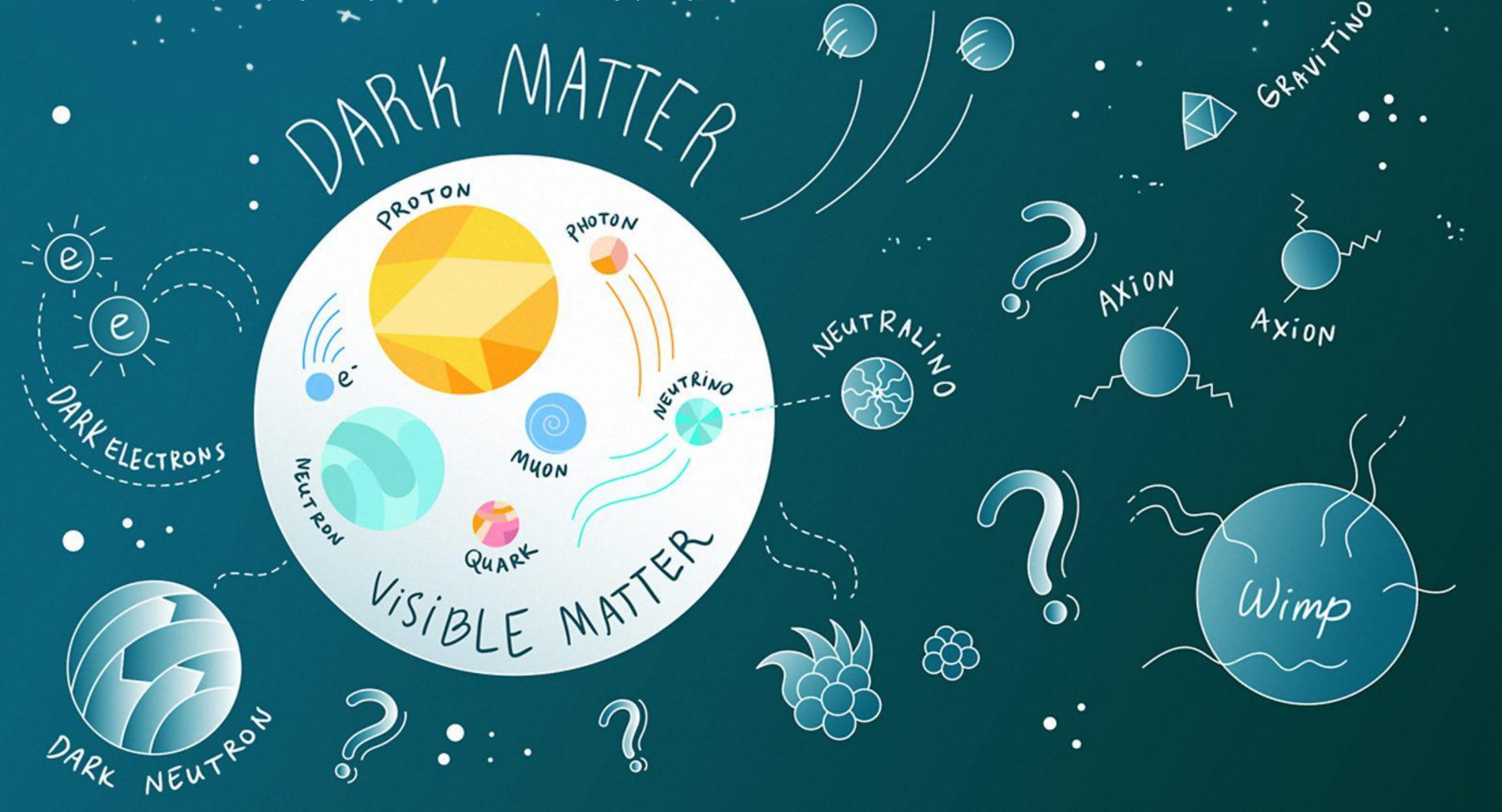
Multi-differential

Simultaneous unfolding of 30 observables!

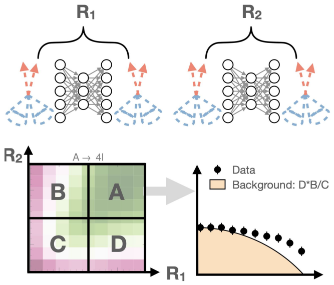




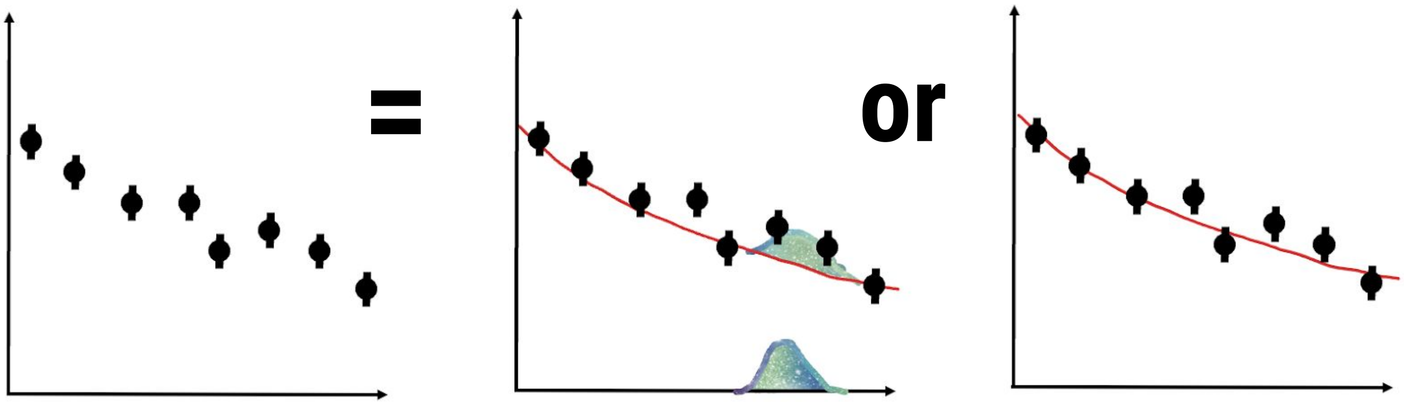
Picture from: https://www.symmetrymagazine.org/article/december-2013/four-things-you-might-not-know-about-dark-matter



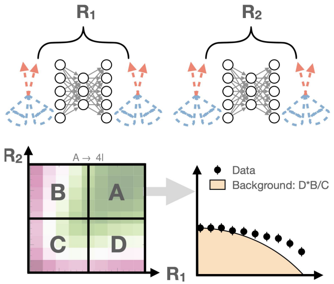
Anomaly detection



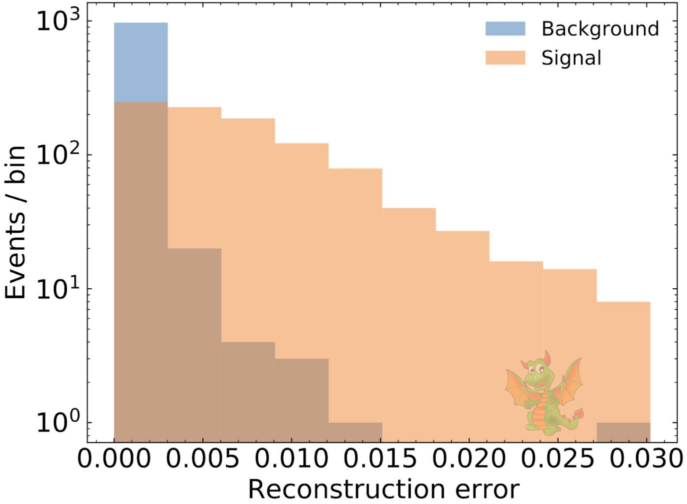
- How to look for **new physics processes** without knowing how they should look like?
- New physics should be rare: **Anomaly detection**
- Even if you are able to identify "anomalies", how to **interpret the observation**?
- A good method of anomaly detection requires:
  - A method that **identifies** particle collisions that seem to be **anomalous**
  - Able to provide context: how should **false positives** look like?



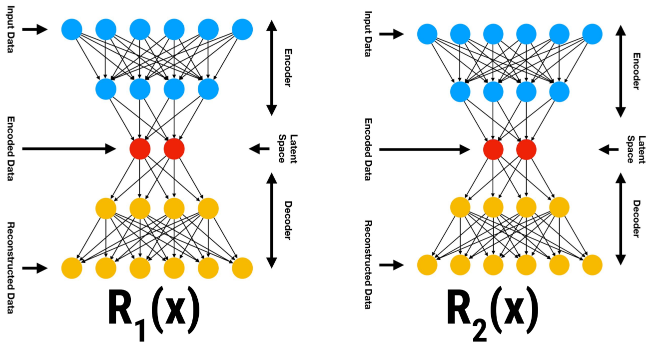
Decorrelated autoencoders



- Anomaly detection based on **autoencoders**: algorithm learns how to **compress** and **decompress** the data using background events
- Events that are **poorly decompressed** are often **rare** and point to anomalous events

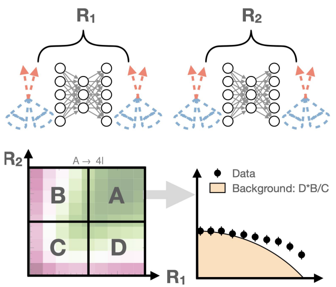


- Train **multiple autoencoders** such that their reconstruction is **independent** for the background



$$L[f_1, f_2, g_1, g_2] = \sum_i R_1(x_i)^2 + \sum_i R_2(x_i)^2 + \lambda\, \mathrm{DisCo}^2[R_1(X), R_2(X)]$$

Anomaly detection performance

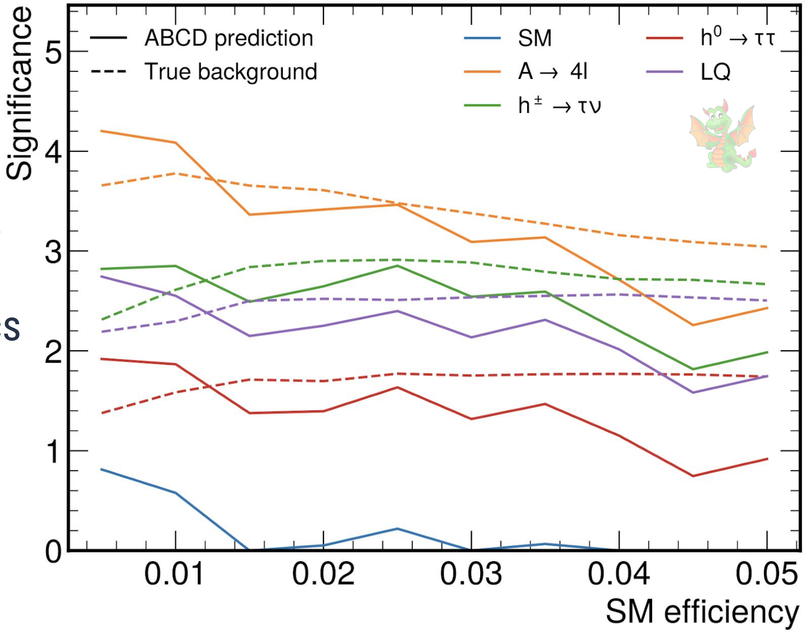


Use the independent reconstructions to estimate the **number of false positives**

- **Significance**: how often your observation is compatible with the **no new physics hypothesis**
- **1**: 1 in **3, 2**: 1 in **22, 3**: 1 in **140, 4**: 1 in **1M, 5**: 1 in **3.5 million**

No anomalies

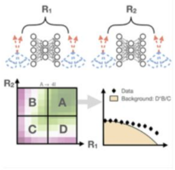
Other colors: datasets with **0.1%** anomalies and **99.9%** standard physics processes



Online-compatible unsupervised nonresonant anomaly detection

Vinicius Mikuni, Benjamin Nachman, and David Shih

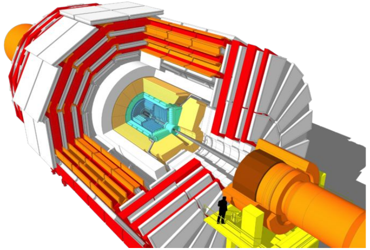
Phys. Rev. D **105**, 055006 (2022) – Published 8 March 2022



The authors of this paper employ two (or more) autoencoders to provide a complete strategy for unsupervised non-resonant anomaly detection. Both signal extraction and data-driven background estimation can be determined with decorrelated autoencoders. The method shows strong performance on test datasets and has the advantage of being online-compatible.

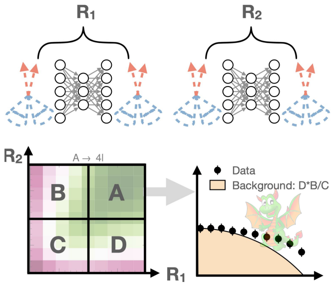
Show Abstract +

Conclusions

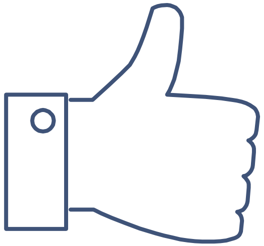


- **Full detector simulation** is expensive and not easily scalable
  - **Surrogate models using ML** can create simulations faster and with similar precision
  - Use **diffusion generative models** for the first time in particle physics
  - **More info here:** arXiv

- **Machine learning unfolding** overcomes the limitations of standard unfolding:
  - No histogram dependence
  - Able to use multiple variables at a time
  - Showcase the method using **real particle collisions**
  - **More info here:** H1prelim-22-034



- Design a method for **anomaly detection** using **decorrelated autoencoders**
- Provides a precise estimation of the **false positive rate** for observations that are considered anomalous
- **More info here:** Phys. Rev. D

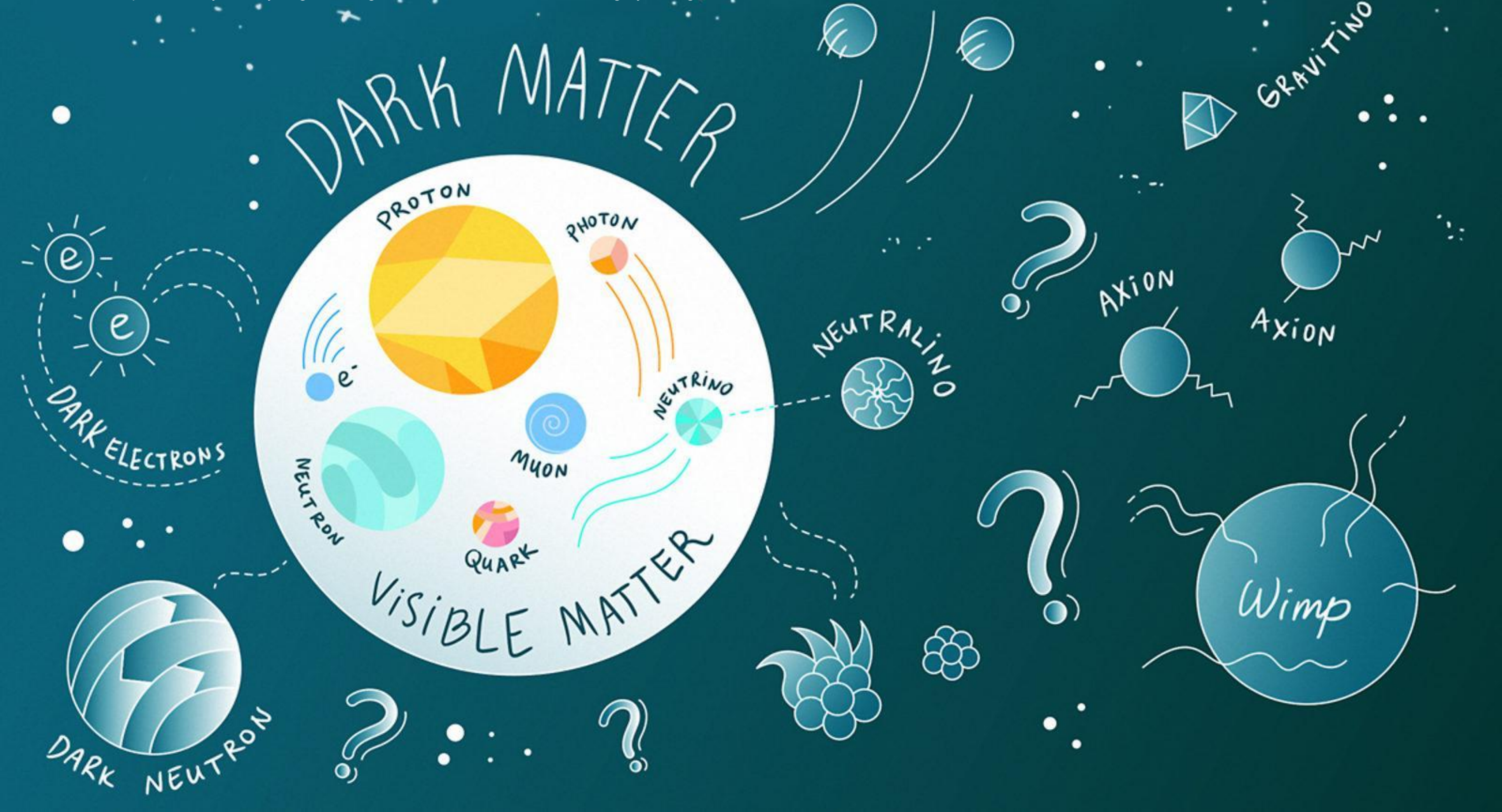


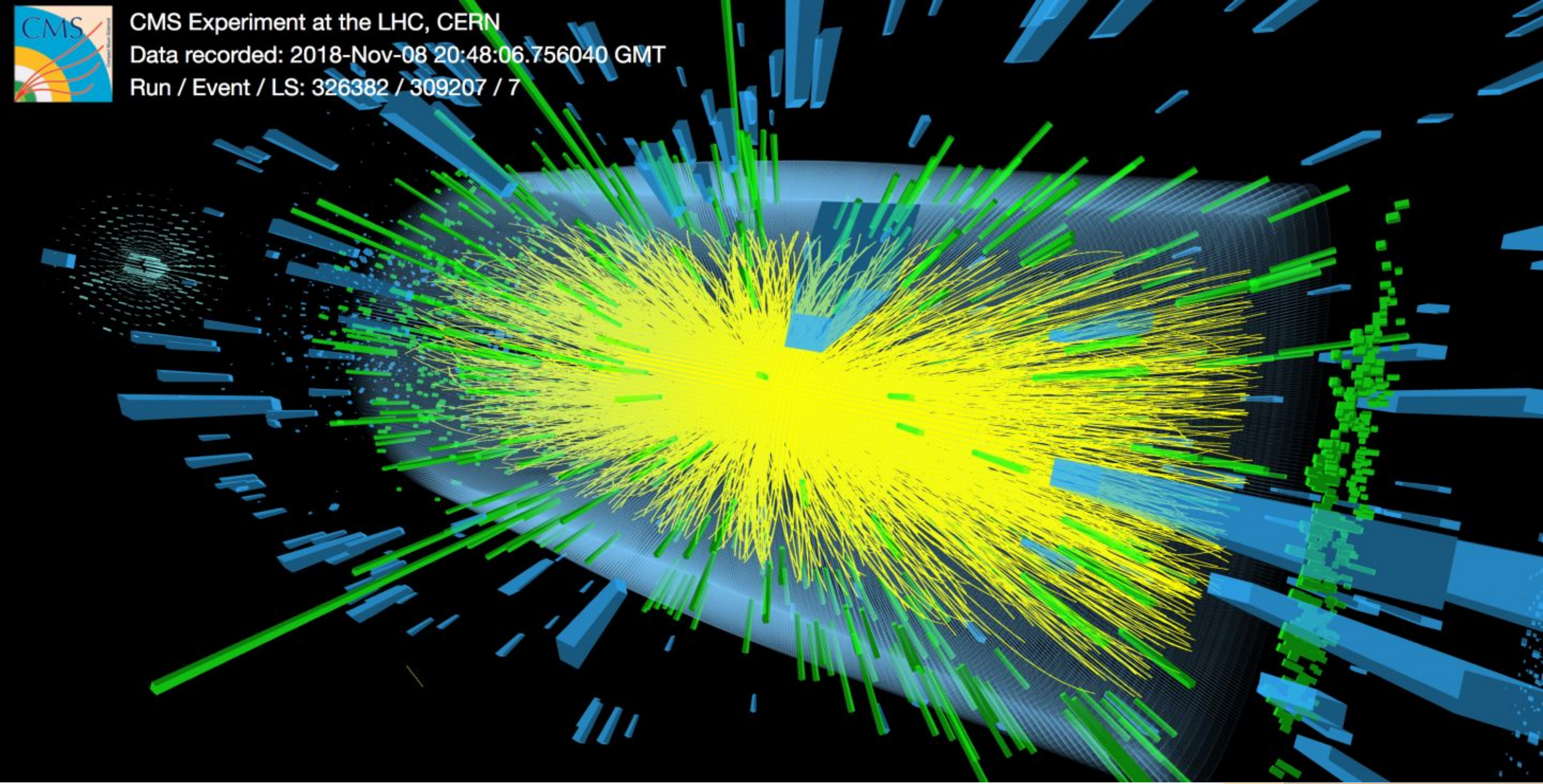
THANKS!

Any questions?

BACKUP

Picture from: https://www.symmetrymagazine.org/article/december-2013/four-things-you-might-not-know-about-dark-matter

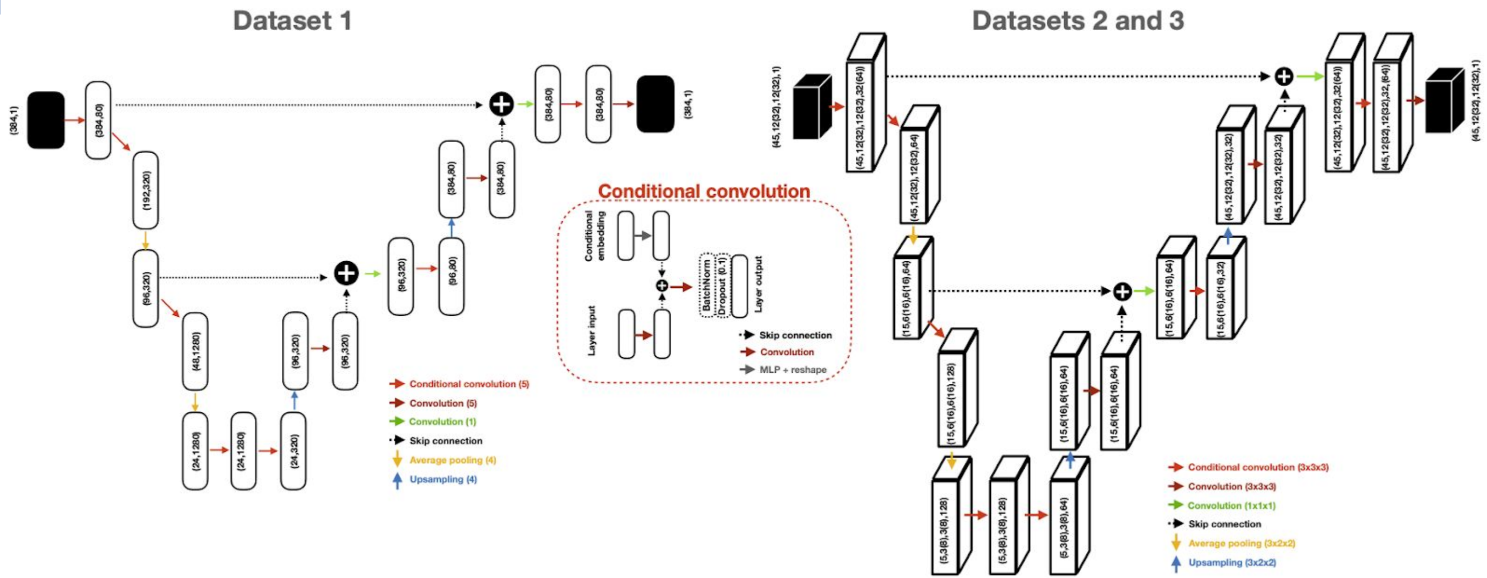




A particle detector

Surrogate model for detector simulation

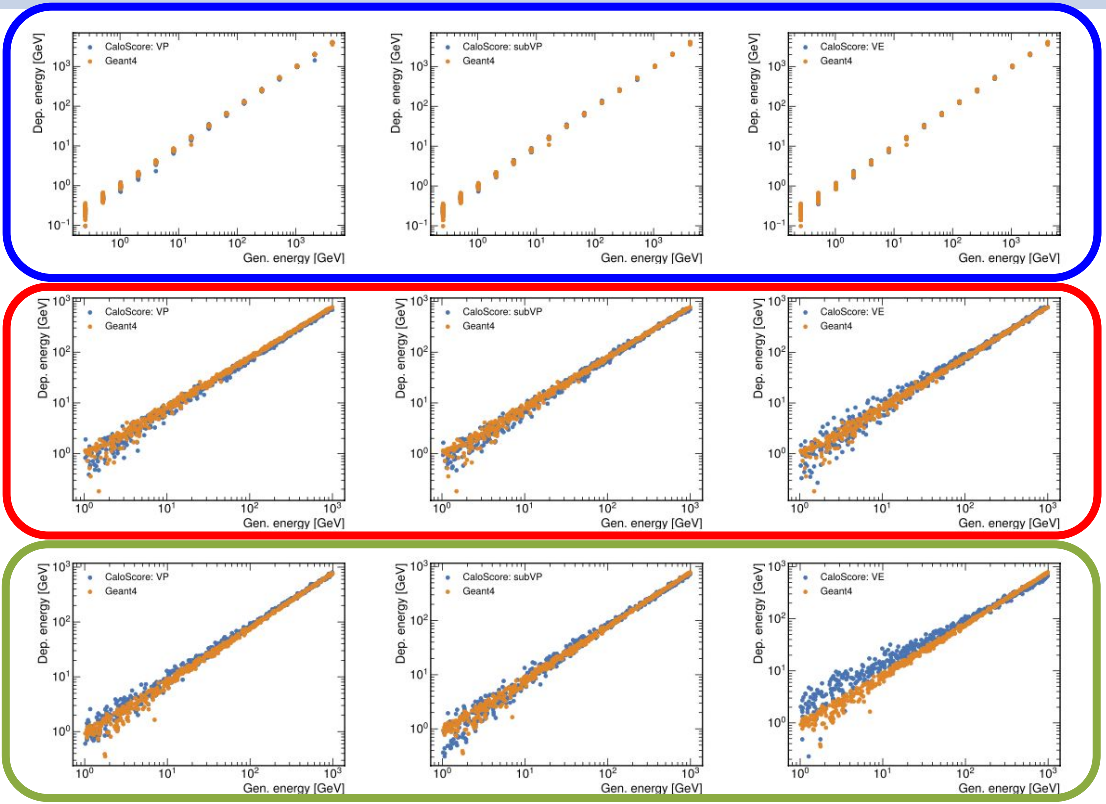
Calorimeter shower generation



Very simple **U-NET** model used to build the score function

- Lots of new developments over the years, adding attention between layers, additional skip connections, but kept it simple for this application
- **Conditional information** is added to convolutional layers as a **bias term**

Results



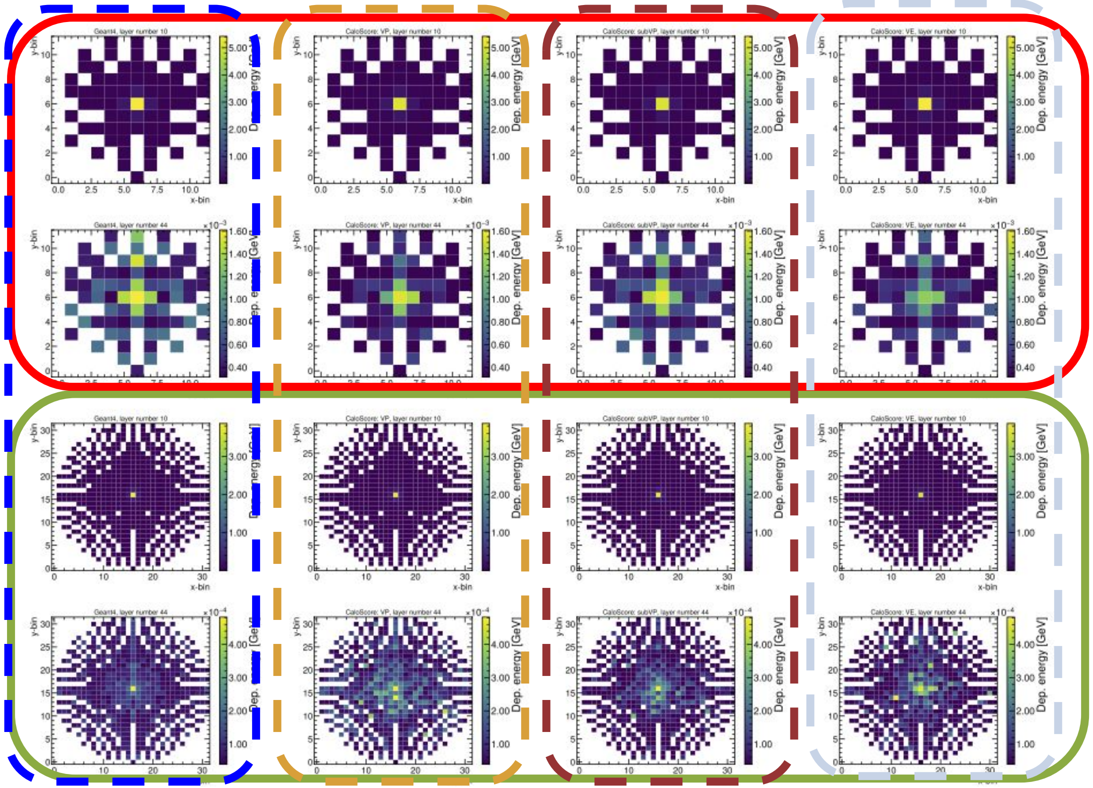
- **Deposited energy** (sum of voxels) vs. the **conditional energy**
- Good agreement between **full simulation** and **different diffusion models**
- **VE** shows the same shift observed for dataset 3

Dataset 1

Dataset 2

Dataset 3

Results



Full simulation

VP SDE

subVP SDE

VE SDE

Dataset 2

Dataset 3

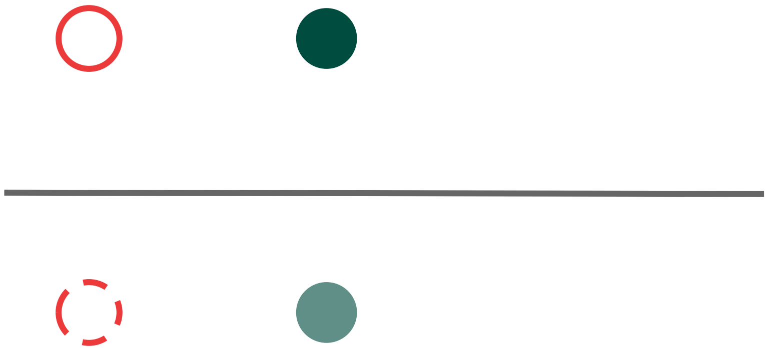
Weird shapes are a result of the coordinate transformation

Unfolding

Omnifold

Reco level

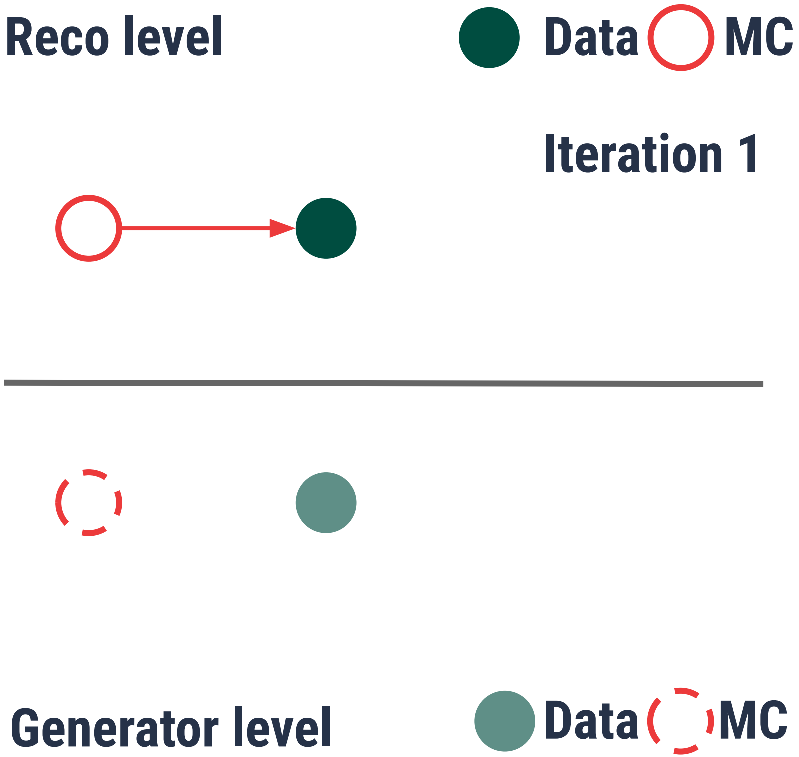
Data MC



Generator level

Data MC

Omnifold



Step 1:

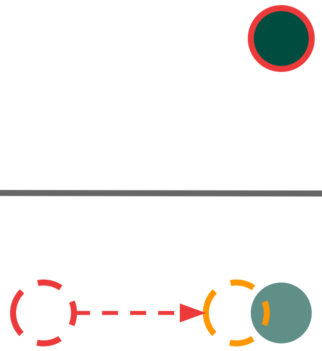
- Train a classifier to separate **data** from **MC** events
- Reweight **reco level MC** with weights:

$$W(\mathrm{reco}) = p_{\mathrm{Data}}(\mathrm{reco})/p_{\mathrm{MC}}(\mathrm{reco})$$

Omnifold

Reco level Data MC

Iteration 1



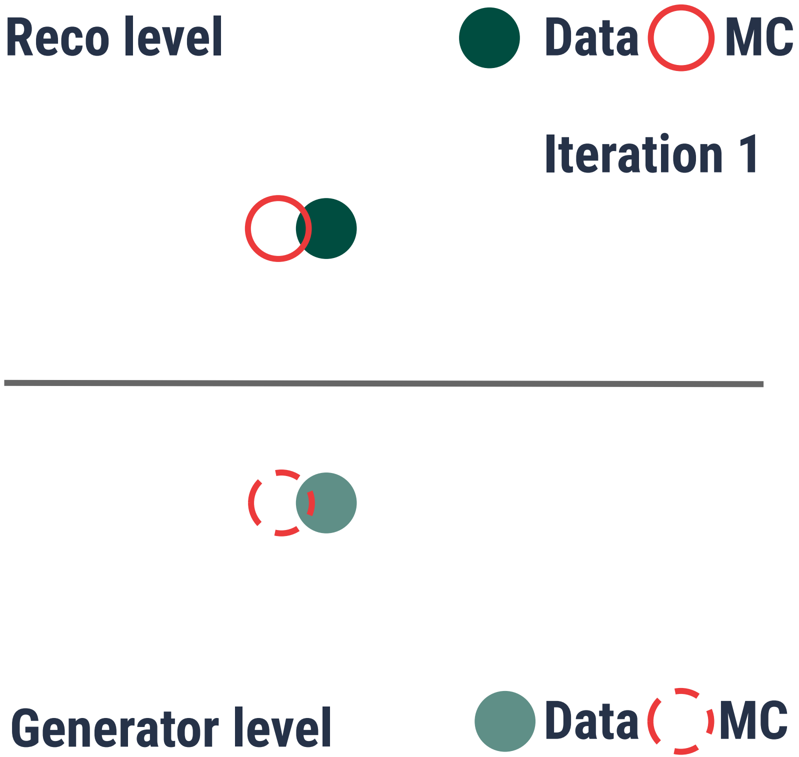
Generator level Data MC MC reweighted

Step 2:

- **Pull weights** from **step 1** to generator level events
- Train a classifier to separate **initial MC at gen level** from **reweighted MC** events
- Define a **new simulation** with weights that are a **proper function of gen level kinematics**

$$W(gen) = p_{weighted\ MC}(gen)/p_{MC}(gen)$$

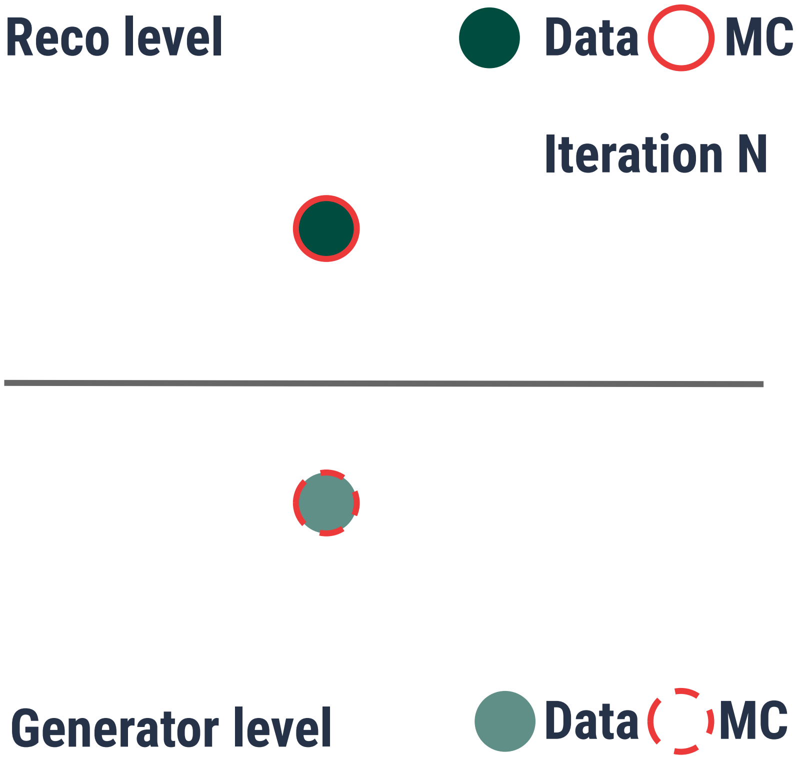
Omnifold



Start again from **step 1** using the **new simulation** after **pushing** the weights from **step 2**

- Guaranteed convergence to the maximum likelihood estimate of the generator-level distribution when number of iterations go to infinite
- In practice, less than 10 iterations are enough to achieve convergence

Omnifold



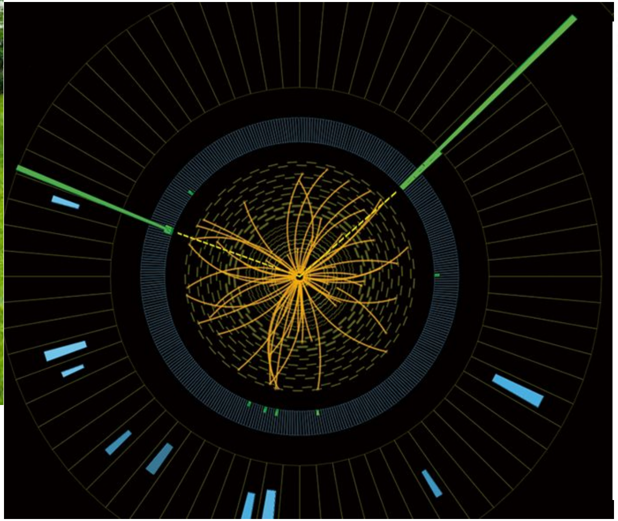
Start again from **step 1** using the **new simulation** after **pushing** the weights from **step 2**

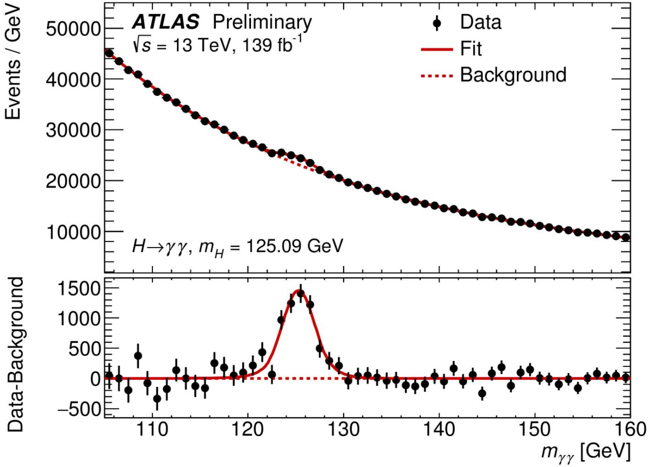
- **Guaranteed convergence** to the maximum likelihood estimate of the generator-level distribution when number of iterations goes to infinite
- In practice, **less than 10 iterations** are enough to achieve convergence

Anomaly detection

What is an anomaly anyway?

- **Anomaly detection** is often associated to **outlier detection**
- Our application is a bit different: a **single particle collision is not very informative**, only an ensemble of events are!



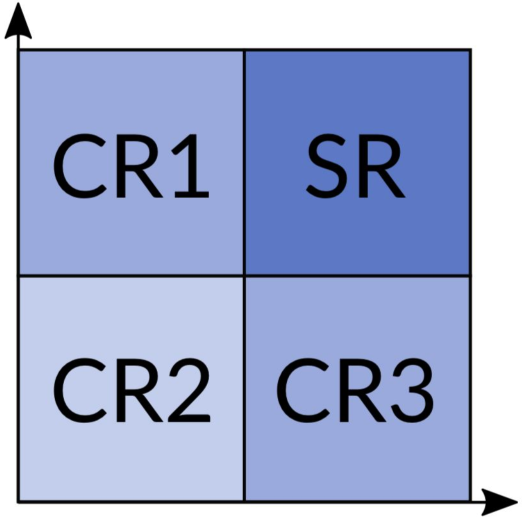


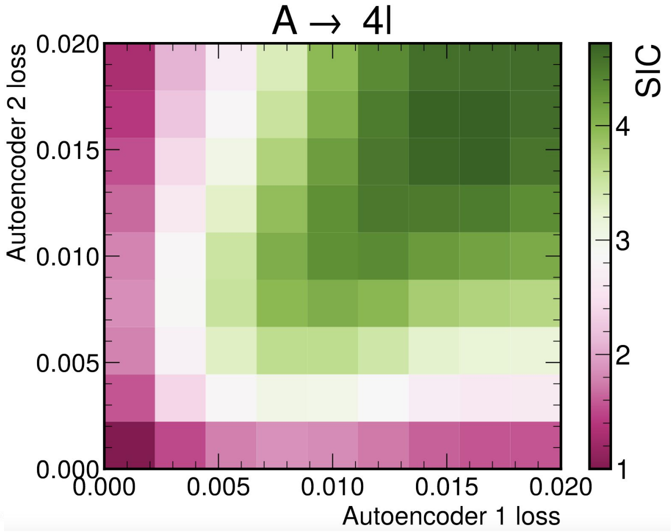
- There are also examples of outlier detection in HEP such as detector quality monitoring

ABCD

ABCD method is a popular choice of data-driven background estimation

- Requires 2 **background-independent** distributions
- **Both** distributions should provide **signal sensitivity** to avoid contamination
- Background in the signal-enriched region is described by the other background-dominated regions





SR=CR1*CR3/CR2