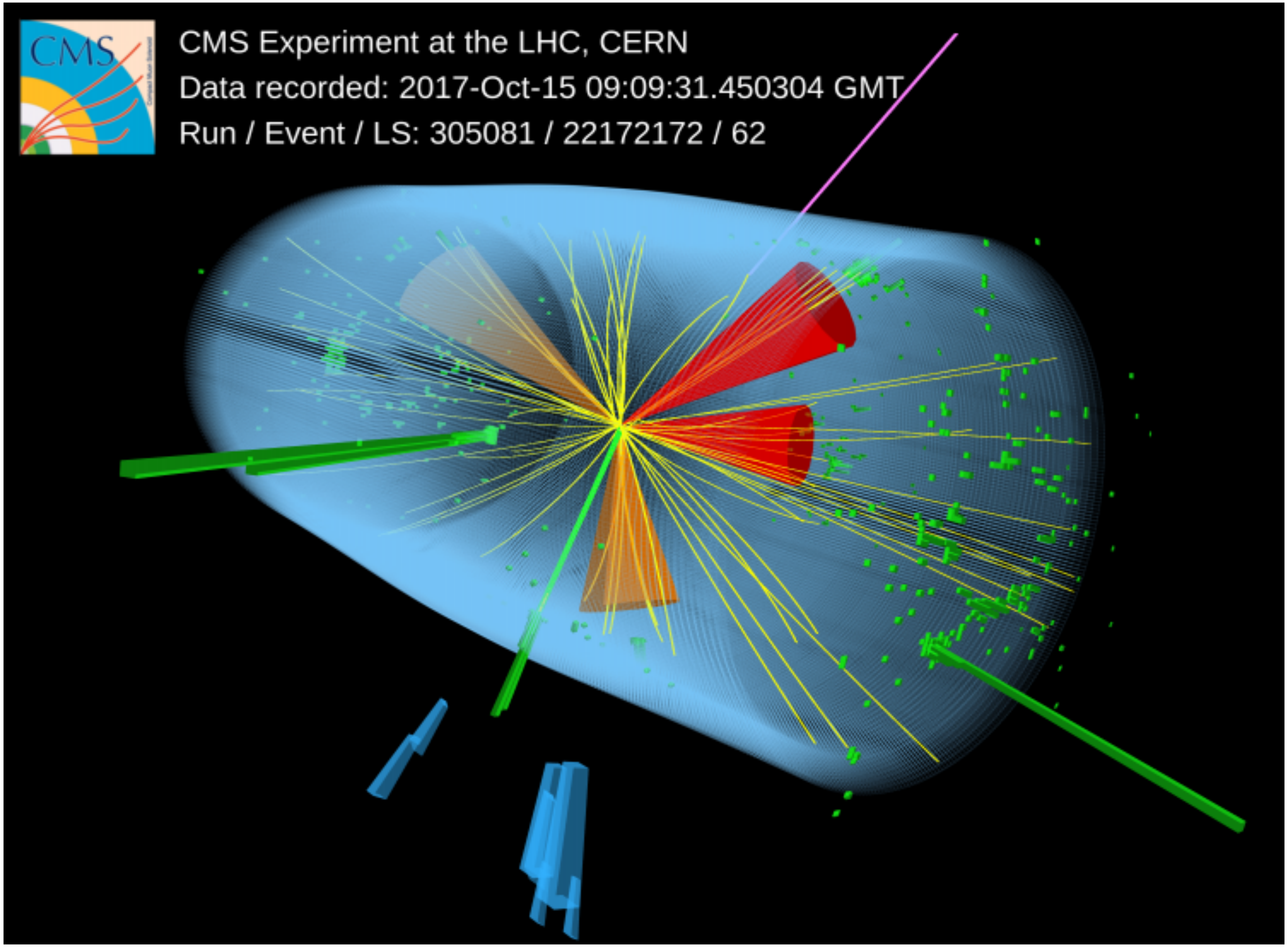
Conceptual Advances in DL for
Research on Universe and Matter
September 2022

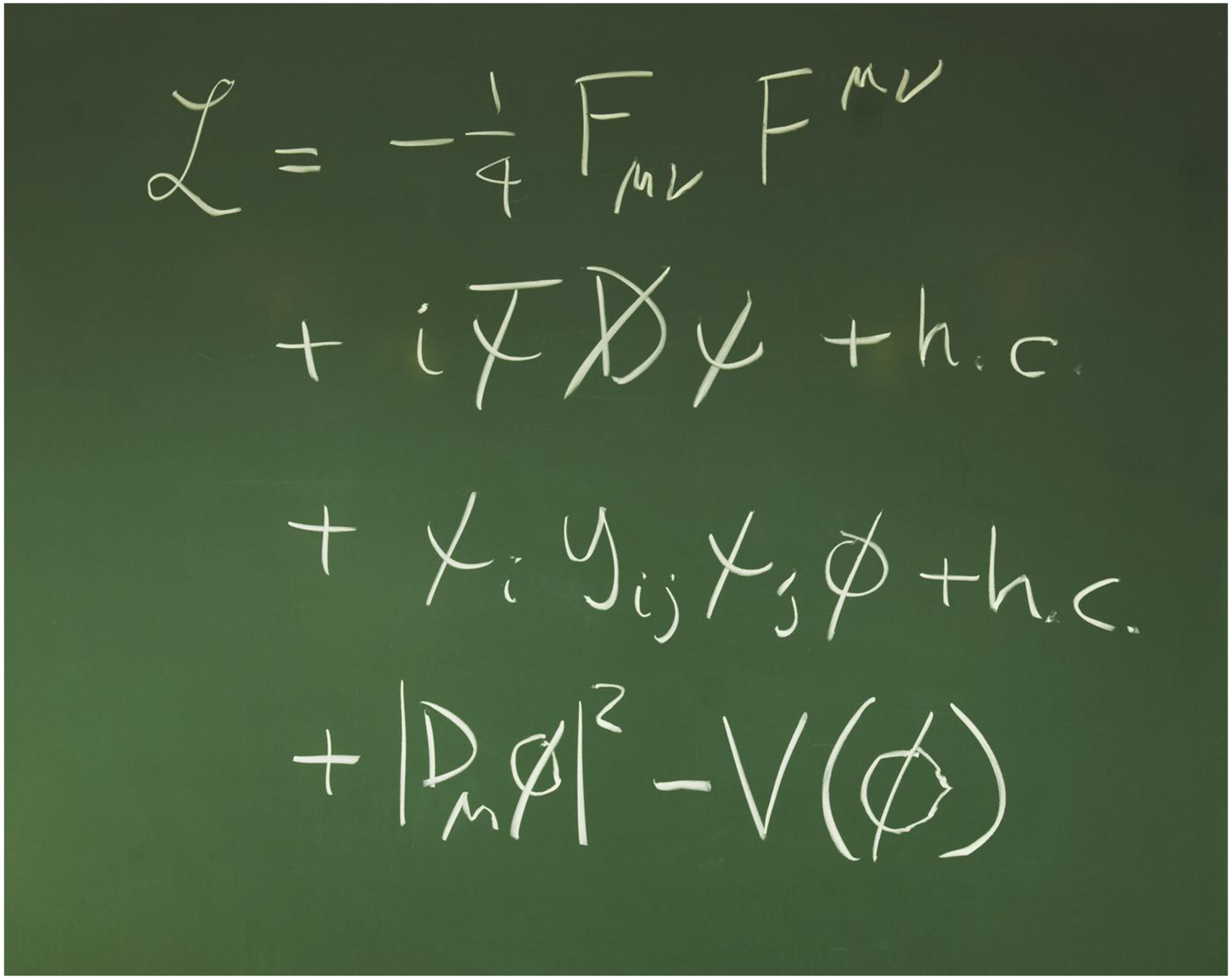
Autoencoders for anomaly detection in particle physics

Michael Krämer
(Institute for Theoretical Particle Physics and Cosmology, RWTH Aachen University)

Particle collisions at the LHC



The Standard Model of Particle Physics



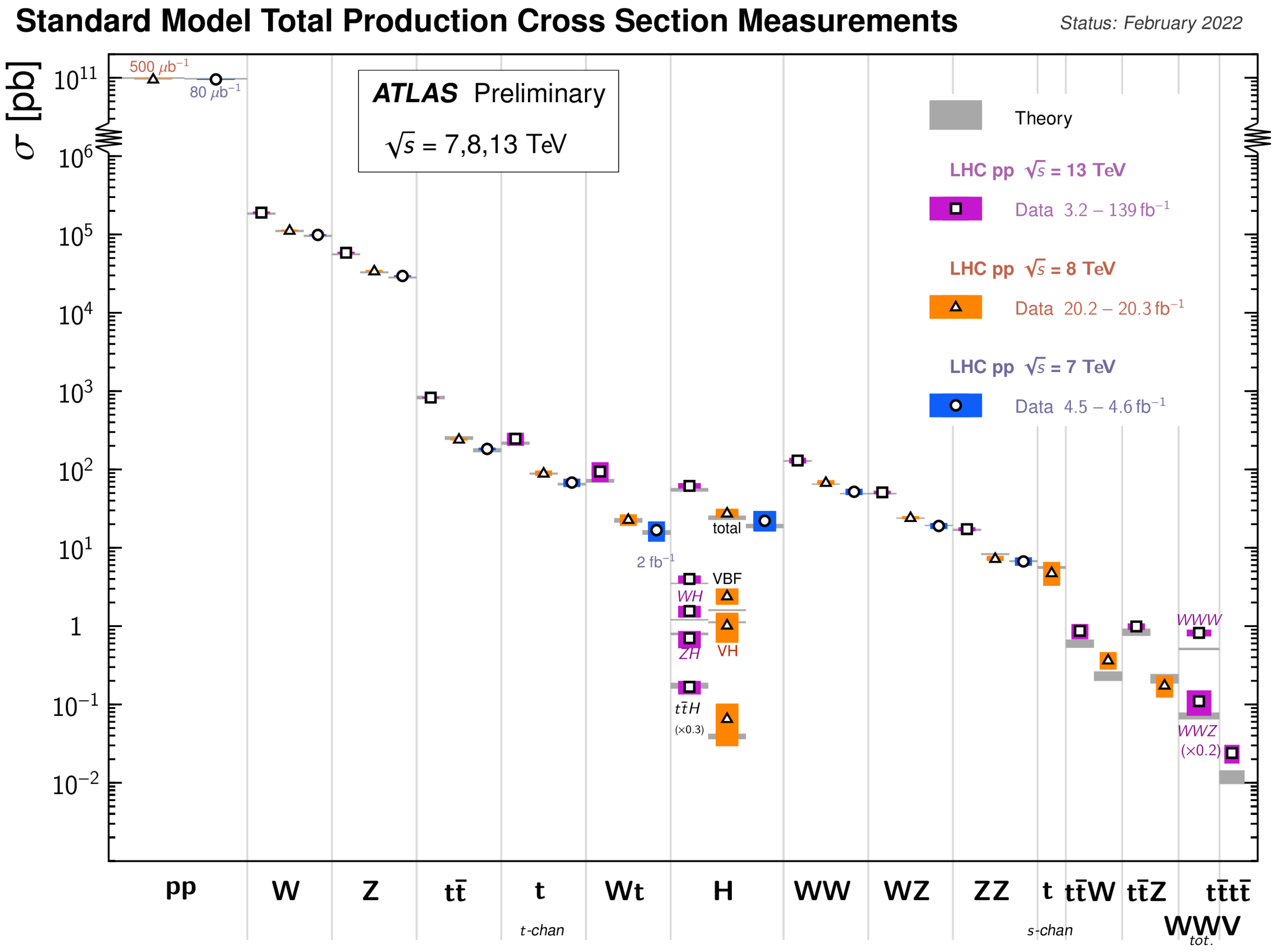
$$\mathcal{L} = -\frac{1}{4} F_{\mu\nu} F^{\mu\nu}$$

$$+ i \bar{\psi} \not{D} \psi + h.c.$$

$$+ \psi_i y_{ij} \psi_j \phi + h.c.$$

$$+ |D_\mu \phi|^2 - V(\phi)$$

The Standard Model of Particle Physics


Standard Model Total Production Cross Section Measurements

Status: February 2022

Physics beyond the Standard Model?

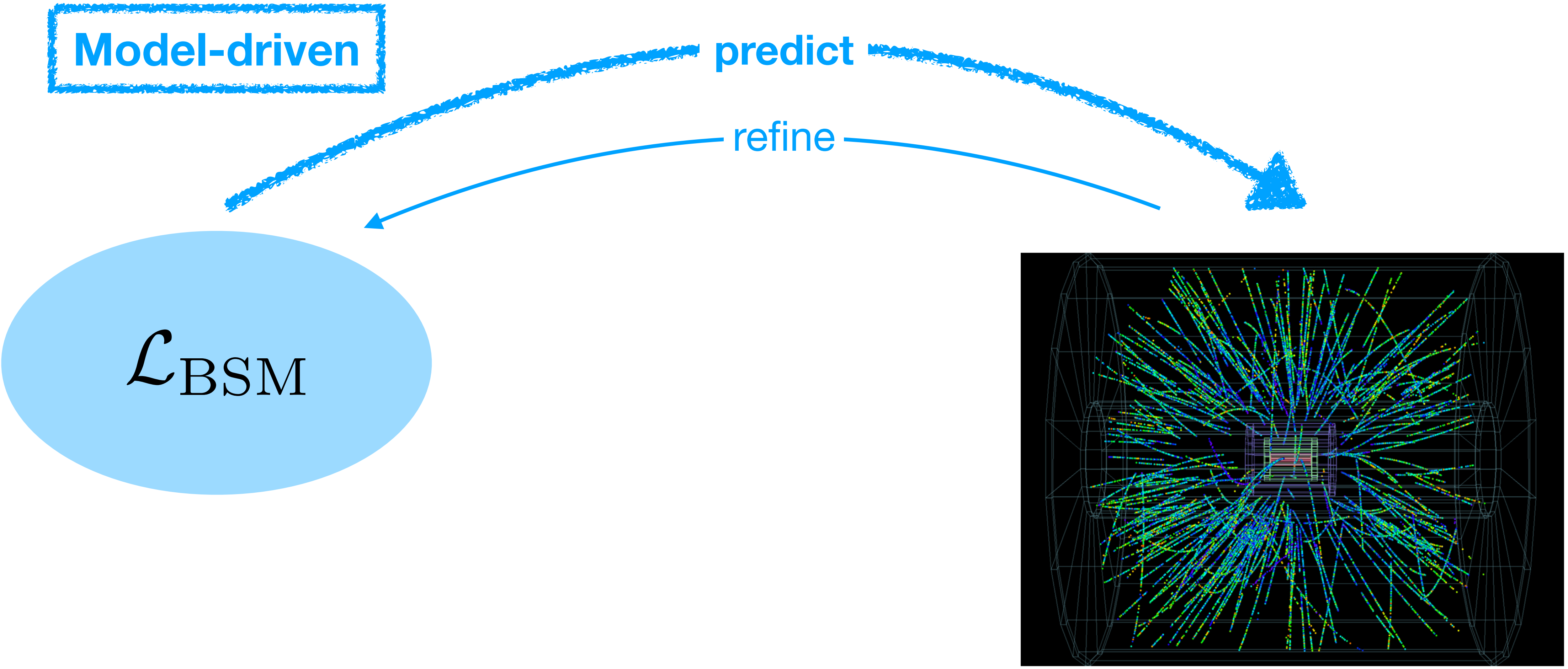
There are potential anomalies and conceptual shortcomings of the SM

- Particle physics anomalies: LFV in B-meson decays, $(g\text{-}2)_\mu$, …

- Cosmic enigmas: dark matter, matter-antimatter asymmetry, …

- Conceptual questions: origin of EWSB, mass hierarchies, unification,…

Physics beyond the Standard Model?



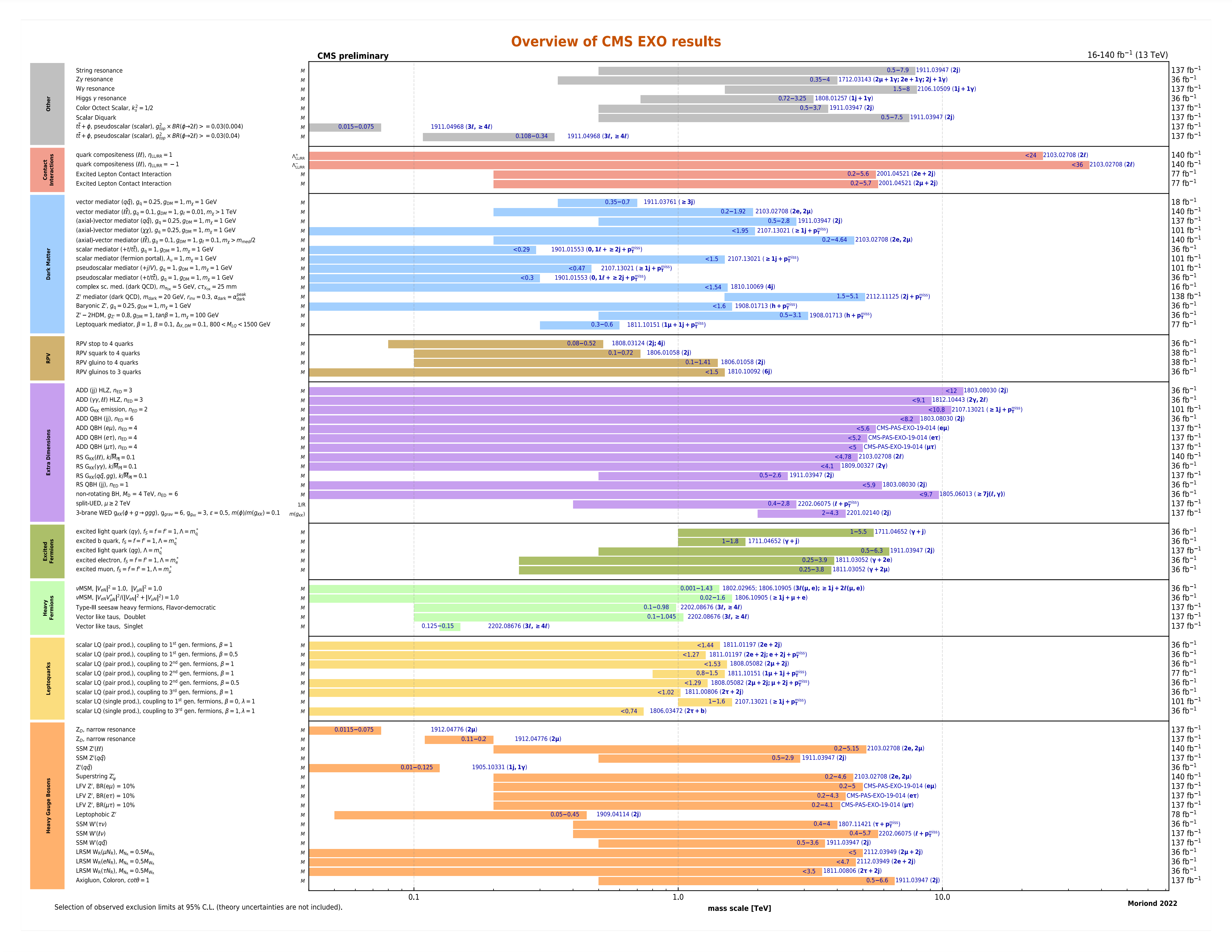
Model-driven

predict

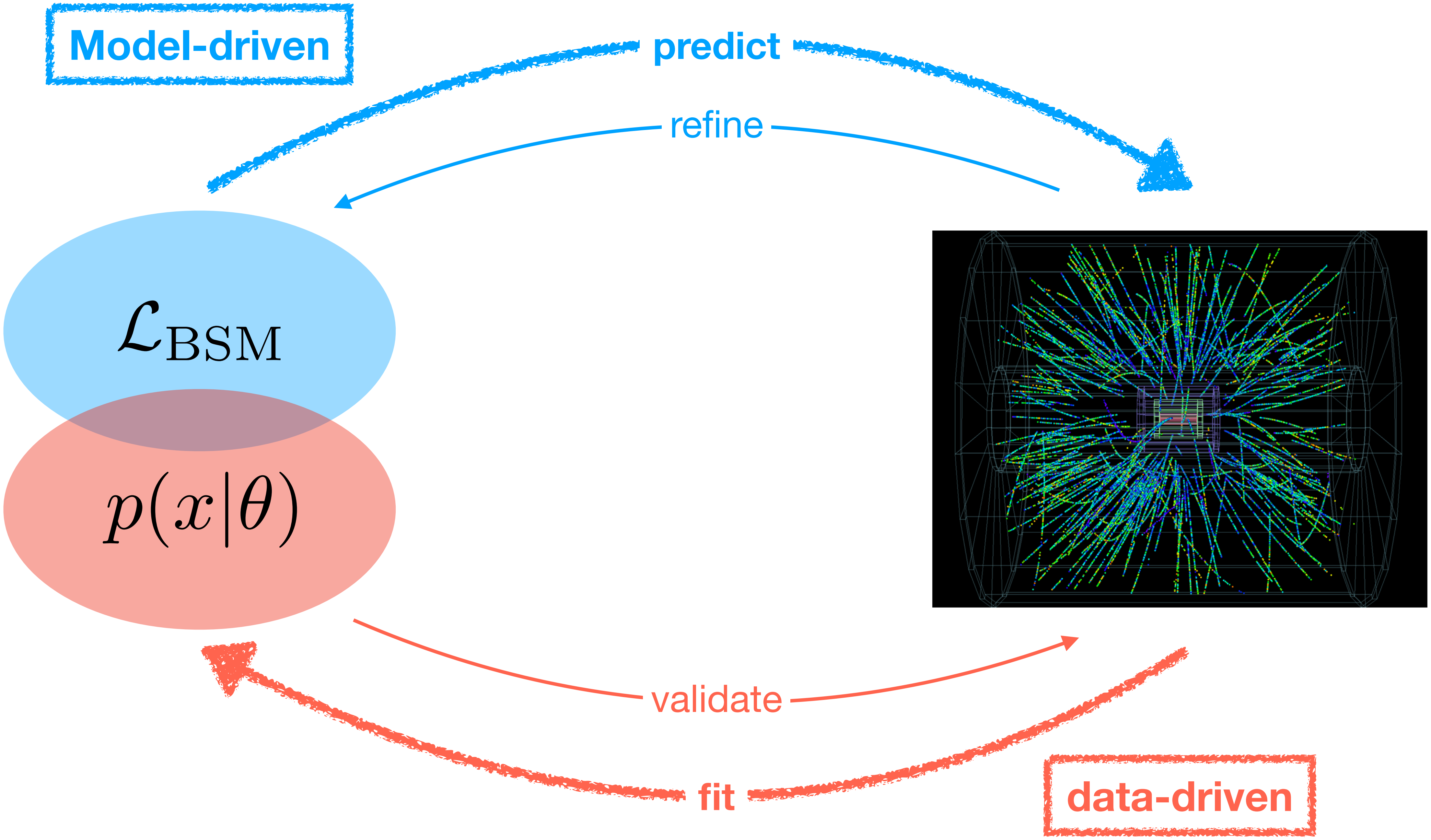
refine

$\mathcal{L}_{\text{BSM}}$

Physics beyond the Standard Model?



Physics beyond the Standard Model?



Model-driven

predict

refine

$\mathcal{L}_{\mathrm{BSM}}$

$p(x|\theta)$

validate

fit

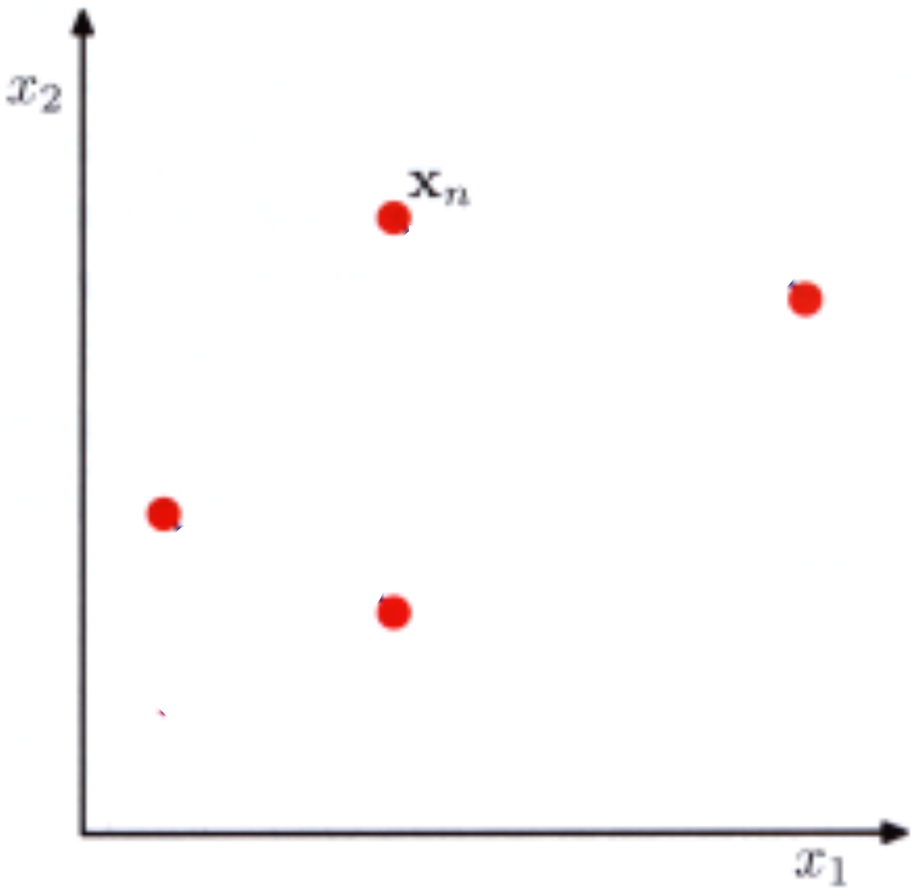
data-driven

Outline

- Unsupervised learning
- Autoencoders
- Autoencoders for anomaly detection
- Anomaly searches in particle physics

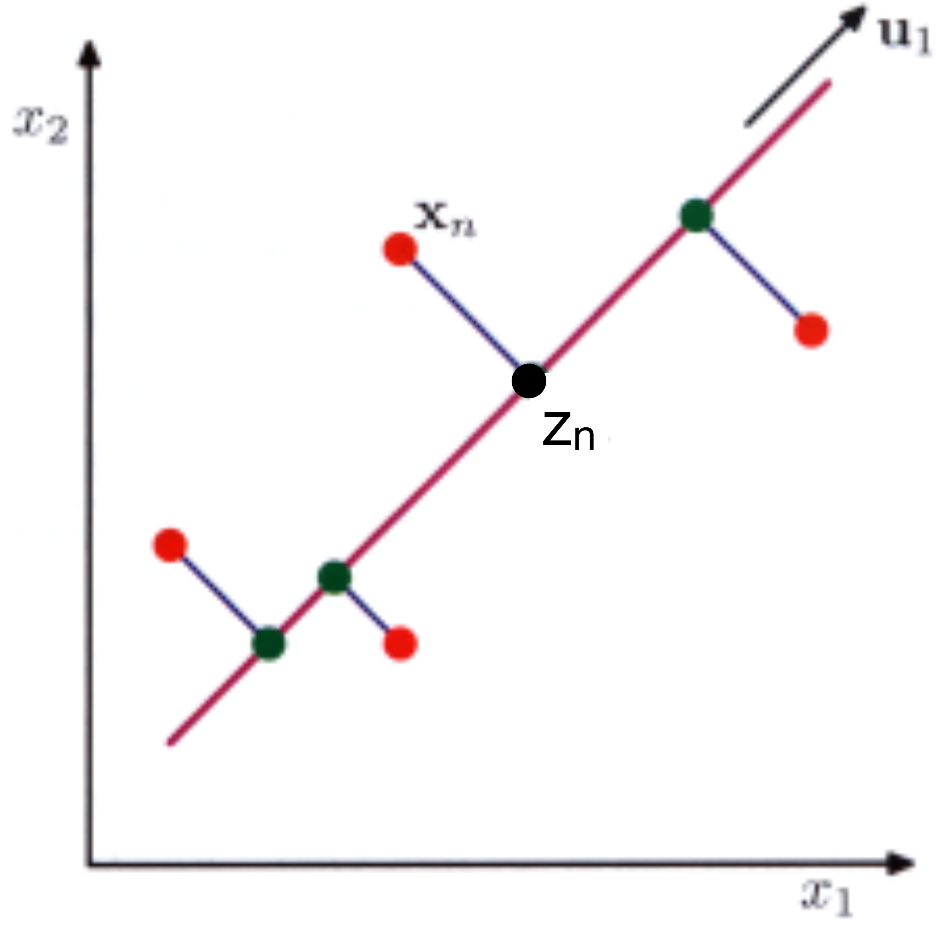
The principle component analysis

- Data reduction
- Decorrelation of features



The principle component analysis

- Data reduction
- Decorrelation of features



Search a mapping

$$\{\mathbf{x}_n\} \in \mathbb{R}^D \rightarrow \{\mathbf{z}_n\} \in \mathbb{R}^M \text{ with } M < D$$

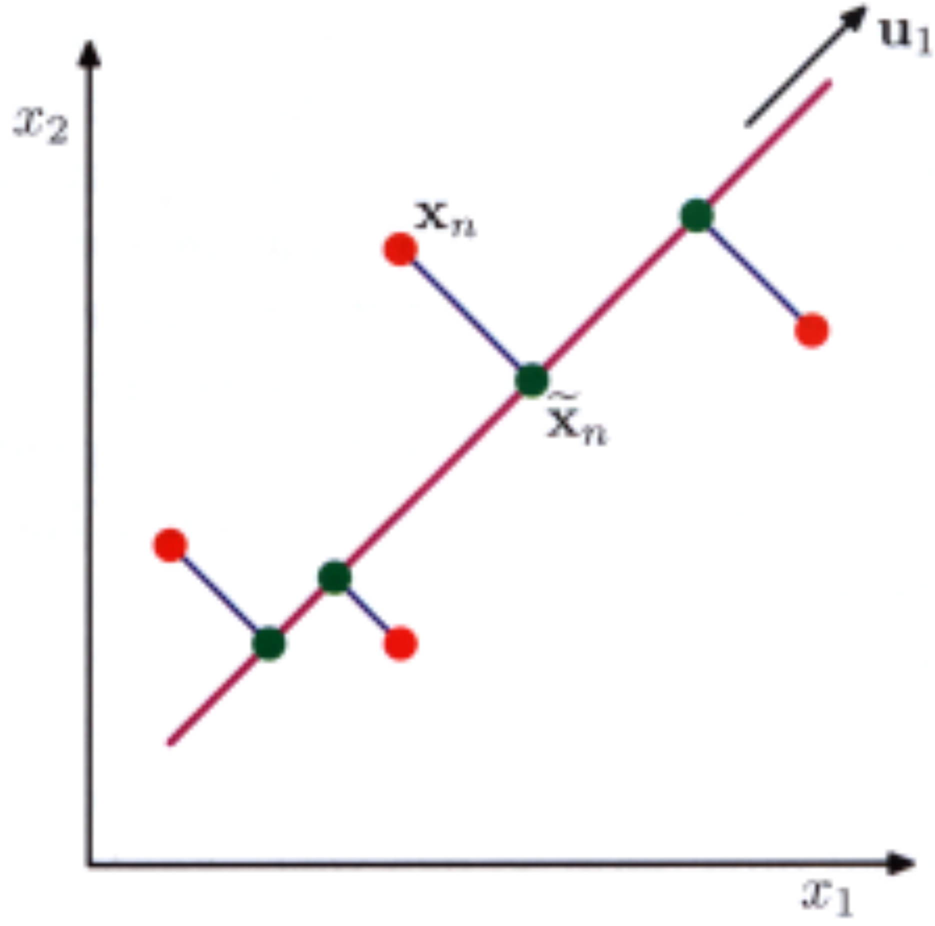
with minimal loss of information.

Want to find an encoding function f(**x**) = **z** and a decoding function g(**z**) = $\tilde{\mathbf{x}}$ such that the reconstruction error $\| \mathbf{x} - \tilde{\mathbf{x}} \|^2 = \| \mathbf{x} - g(\mathbf{z}) \|^2$ is minimal.

PCA: choose g(**z**) = D**z** → f(**x**) = $D^T\mathbf{x}$

The principle component analysis

- Data reduction
- Decorrelation of features



Bishop: Pattern recognition and machine learning

Compute data covariance matrix

$$\mathbf{S} = \frac{1}{N}\sum_{n=1}^{N}(\mathbf{x}_n - \overline{\mathbf{x}})(\mathbf{x}_n - \overline{\mathbf{x}})^{\mathrm{T}}$$

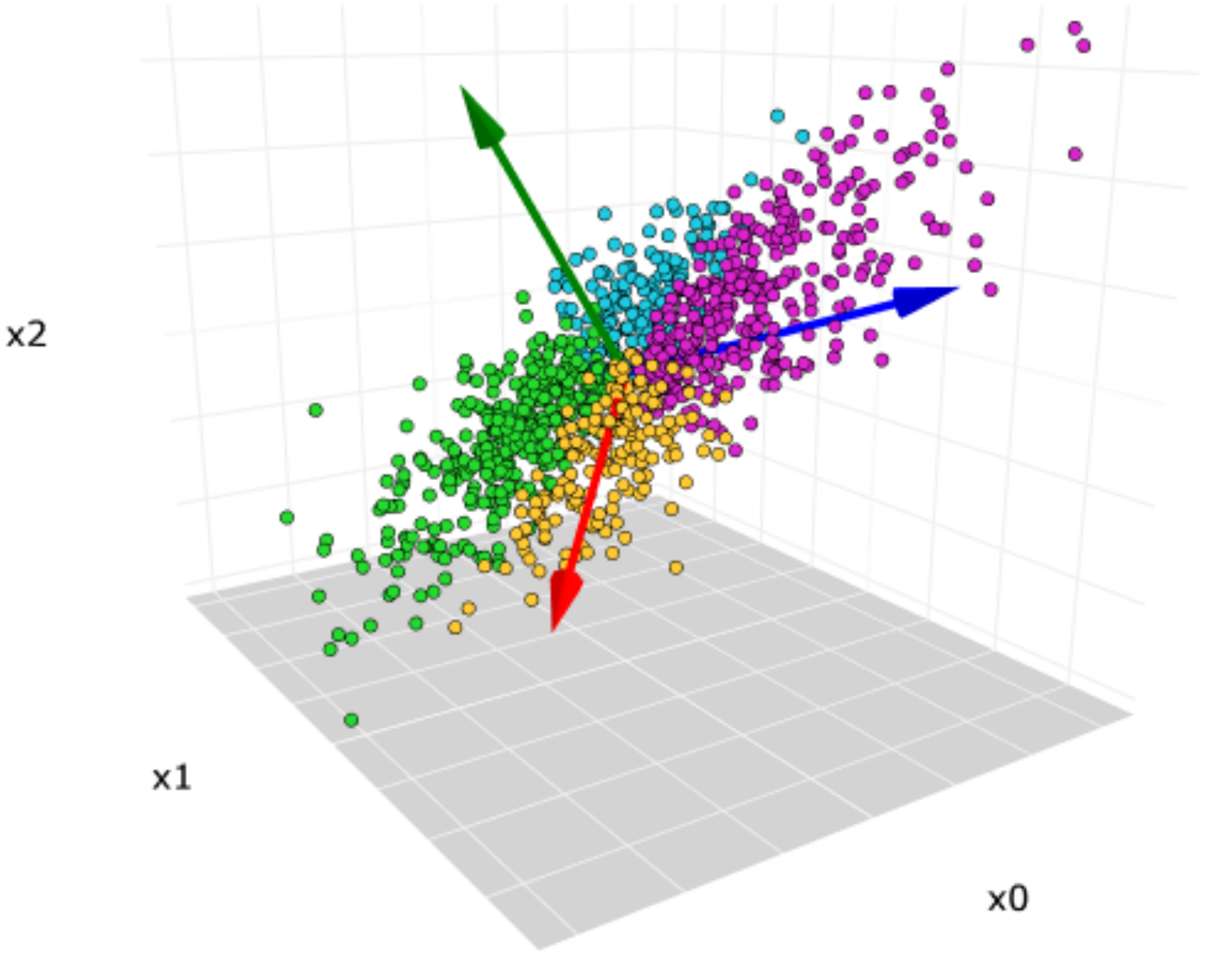
and its eigenvector decomposition.

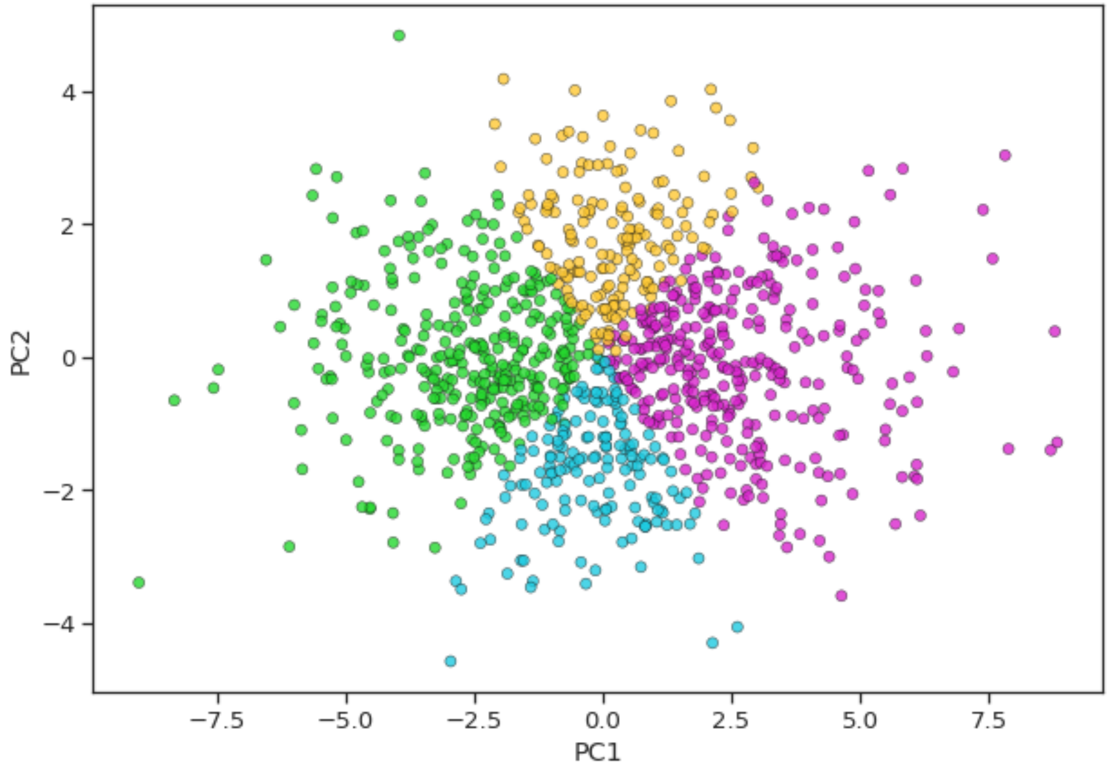
Vector **$\mathbf{u}_1$** is eigenvector with largest eigenvalue,

$$\mathbf{u}_1^{\mathrm{T}}\mathbf{S}\mathbf{u}_1 = \lambda_1$$

The principle component analysis

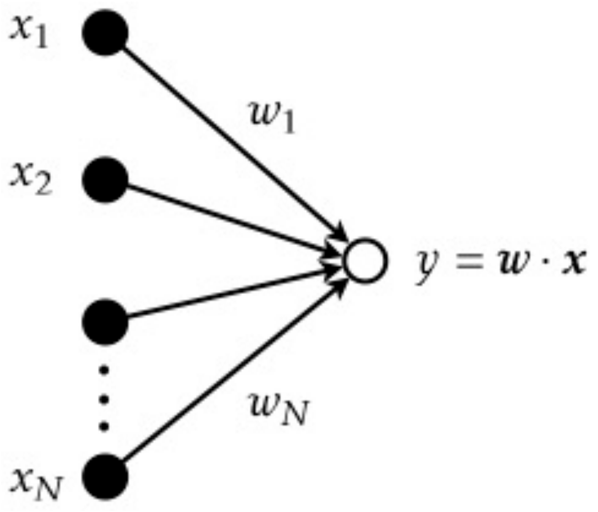
- Data reduction
- Decorrelation of features





Casey Chang: https://towardsdatascience.com/principal-component-analysis-pca-explained-visually-with-zero-math-1cbf392b9e7d

Hebbian learning



B. Mehlig: https://arxiv.org/abs/1901.05639v4

Weight update: $\boldsymbol{w}' = \boldsymbol{w} + \delta\boldsymbol{w} \quad \text{with} \quad \delta\boldsymbol{w} = \eta\, y\, \boldsymbol{x}\,.$

The output |y| becomes the larger, the more often an input feature occurs in the data.

Can write this as DGL: $\tau \frac{d\mathbf{w}}{dt} = \langle y\mathbf{x} \rangle = \langle \mathbf{x} \cdot \mathbf{x} \rangle \mathbf{w} \quad \text{with} \quad \tau \propto 1/\eta$

Writing **w** in terms of the eigenvectors of the covariance matrix, $\mathbf{w} = \sum_i c_i(t)\mathbf{u}_i,$ we get:

$$\mathbf{w} = \sum_i c_i(0) e^{\lambda_i t/\tau} \mathbf{u}_i, \text{ and thus } \mathbf{w} \propto \mathbf{u}_1 \text{ for large } t \gg \tau.$$

Hebbian learning implements the principal component analysis.

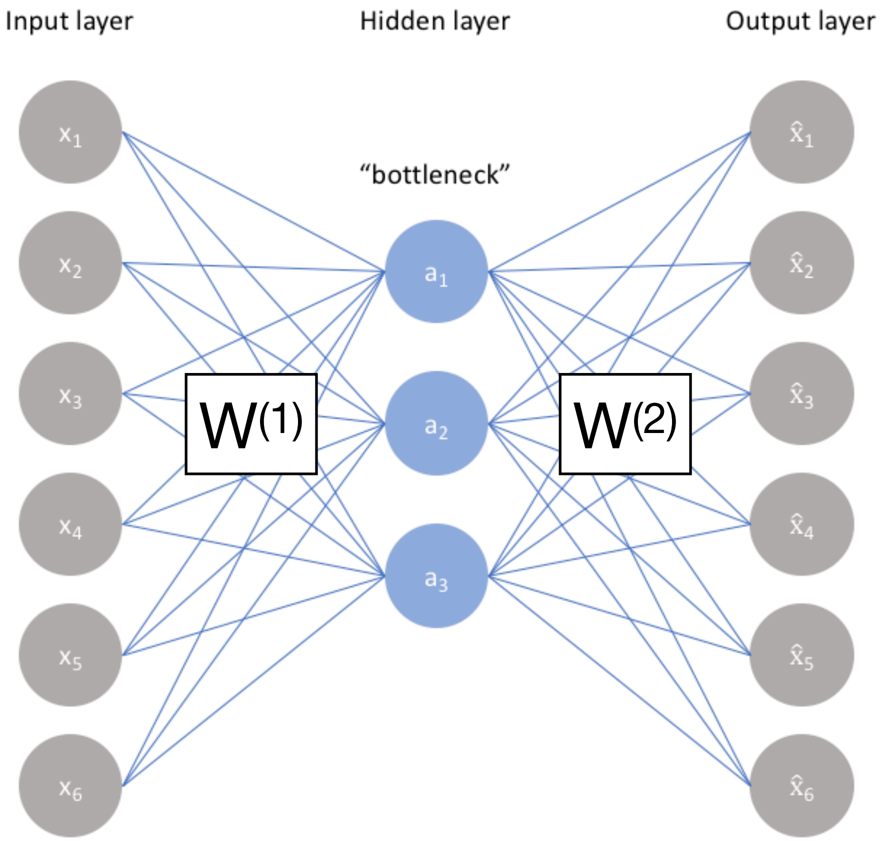
Outline

- Unsupervised learning
- Autoencoders
- Autoencoders for anomaly detection
- Anomaly searches in particle physics

Autoencoders

Let us try to implement a principal component analysis with a neural network.

Recall: we need an encoding function $f(\mathbf{x}) = D^T\mathbf{x}$ and a decoding function $g(\mathbf{z}) = D\mathbf{z}$ such that $|| \mathbf{x} - g(\mathbf{z}) ||^2$ is minimal.



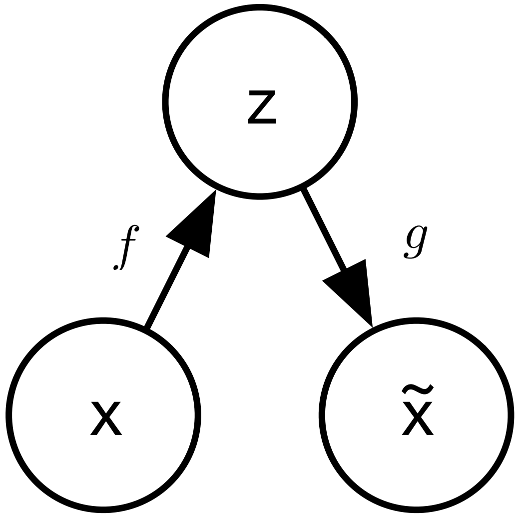
$$\mathbf{z} = f(\mathbf{x}) = W^{(1)}\,\mathbf{x}$$

and

$$\tilde{\mathbf{x}} = g(\mathbf{z}) = W^{(2)}\,\mathbf{z} = W^{(2)}\,W^{(1)}\,\mathbf{x}$$

Training the weights $W^{(1)}$ and $W^{(2)}$ to minimise the mean square error between input and output, the linear neural network (nearly) implements a principal component analysis.

Autoencoders

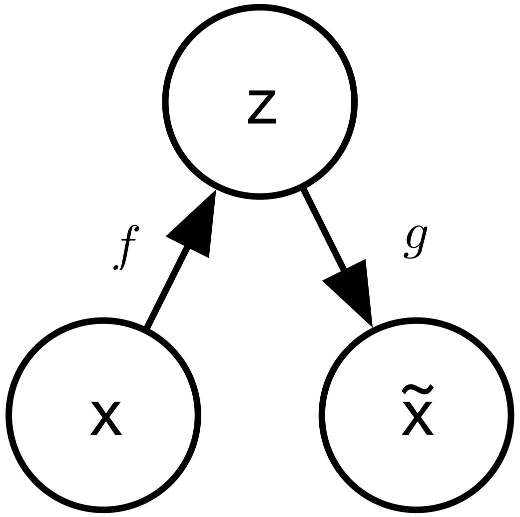


An autoencoder is a neural network that tries to learn an approximation to the identity function $\tilde{\mathbf{x}} = g(f(\mathbf{x})) \approx \mathbf{x}$

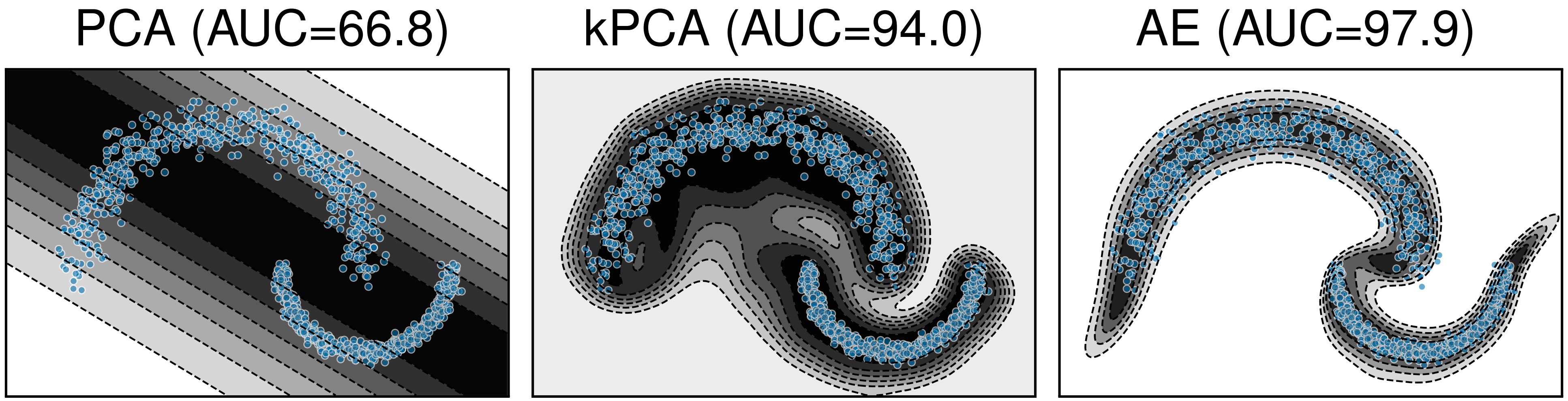
Learning the identity function itself is not very useful, but by placing constraints on the network, such as by limiting the number of hidden units, one can discover interesting structures about the data:

- latent space z has lower dimension than x;
- f or g have low capacity (e.g. linear g);
- introduce regularisation, e.g. sparse autoencoders.

Autoencoders



An autoencoder is a neural network that tries to learn an approximation to the identity function $\tilde{\mathbf{x}} = g(f(\mathbf{x})) \approx \mathbf{x}$


PCA (AUC=66.8)
kPCA (AUC=94.0)
AE (AUC=97.9)

Ruff et al., https://arxiv.org/abs/2009.11732

Variational autoencoders

One can combine the idea of an autoencoder with the concept of generative modeling.

Bayesian view: $$\underset{\substack{\nearrow \\ \text{evidence}}}{p(\mathbf{x})} = \int \underset{\substack{\uparrow \\ \text{latent prior}}}{p(\mathbf{z})} \underset{\substack{\nwarrow \\ \text{likelihood}}}{p(\mathbf{x}|\mathbf{z})} d\mathbf{z}$$

Goal: maximise $p_\theta(\mathbf{x})$ by learning $p_\theta(\mathbf{z})$ and $p_\theta(\mathbf{x}|\mathbf{z})$: $$\theta^* = \underset{\theta}{\operatorname{argmin}}\, \mathbb{E}_{\mathbf{x}\sim p_{\mathrm{data}}}(-\ln p_\theta(\mathbf{x}))$$

Difficult to evaluate in practice → introduce recognition model $q_\theta(\mathbf{z}|\mathbf{x})$ as an approximation to true posterior $p(\mathbf{z}|\mathbf{x})$

AE terminology:

$q_\theta(\mathbf{z}|\mathbf{x})$ → encoder

$p_\theta(\mathbf{x}|\mathbf{z})$ → decoder

In a variational autoencoder, $q_\theta(\mathbf{z}|\mathbf{x})$ is a multivariate Gaussian, parametrised by a neural network.

Variational autoencoders

How can we use the recognition model $q(\vec{z}|\vec{x})$ to maximize likelihood?

$$\ln p(\vec{x}) = \mathbb{E}_{z \sim q(\vec{z}|\vec{x})} \ln \left(p(\vec{x}) \frac{p(\vec{z}|\vec{x})}{p(\vec{z}|\vec{x})} \right)$$

$$= \mathbb{E}_{z \sim q(\vec{z}|\vec{x})} \left(\ln \left(\frac{p(\vec{z},\vec{x})}{q(\vec{z}|\vec{x})} \right) + \ln \left(\frac{q(\vec{z}|\vec{x})}{p(\vec{z}|\vec{x})} \right) \right)$$

$$= \mathbb{E}_{z \sim q(\vec{z}|\vec{x})} \ln \left(\frac{p(\vec{z},\vec{x})}{q(\vec{z},\vec{x})} \right) + \underbrace{KL\left(q(\vec{z}|\vec{x}) \| p(\vec{z},\vec{x})\right)}_{\geq 0}$$

$$\geq \mathbb{E}_{z \sim q(\vec{z}|\vec{x})} \ln \left(\frac{p(\vec{z},\vec{x})}{q(\vec{z},\vec{x})} \right)$$

⇝ evidence lower bound (ELBO)

Variational autoencoders

Rather than maximizing log-likelihood, minimize negative log-likelihood:

$$-\ln p(\vec{x}) \leq \mathbb{E}_{\vec{z}\sim q(\vec{z}|\vec{x})} \ln\left(\frac{q(\vec{z}|\vec{x})}{p(\vec{z},\vec{x})}\right)$$

$$= \mathbb{E}_{\vec{z}\sim q(\vec{z}|\vec{x})} \ln\left(\frac{q(\vec{z}|\vec{x})}{p(z)\, p(\vec{x}|\vec{z})}\right)$$

$$= \mathbb{E}_{\vec{z}\sim q(\vec{z}|\vec{x})} \left(\ln\left(\frac{q(\vec{z}|\vec{x})}{p(\vec{z})}\right) - \ln p(\vec{x}|\vec{z})\right)$$

$$= KL\left(q(\vec{z}|\vec{x}) \| p(\vec{z})\right) - \underbrace{\mathbb{E}_{\vec{z}\sim q(\vec{z}|\vec{x})}\left(-\ln p(\vec{x}|\vec{z})\right)}_{\sim \text{ reconstruction error}}$$

⇝ autoencoder structure: $\vec{x} \xrightarrow{q} \vec{z} \xrightarrow{p} \vec{x}$

Variational autoencoders

Variational autoencoder:

Minimize bound to negative log likelihood:

$$\theta^* = \underset{\theta}{\operatorname{argmin}}\, \mathbb{E}_{\vec{x}\sim p_{data}}\left(-\ln p_\theta(\vec{x})\right)$$

$$= \underset{\theta}{\operatorname{argmin}} \sum_{i=1}^{N} \left(-\ln p_\theta(\vec{x}^{(i)})\right)$$

$$\approx \underset{\theta}{\operatorname{argmin}} \sum_{i=1}^{N} \left[KL\left(q_\theta(\vec{z}|\vec{x}^{(i)}) \| p(\vec{z})\right) + \mathbb{E}_{\vec{z}\sim q_\theta(\vec{z}|\vec{x}^{(i)})}\left(-\ln p_\theta(\vec{x}^{(i)}|\vec{z})\right)\right]$$

In VAE, $q_\theta(\vec{z}|\vec{x})$ is a multivariate Gaussian, parametrized by neural network:

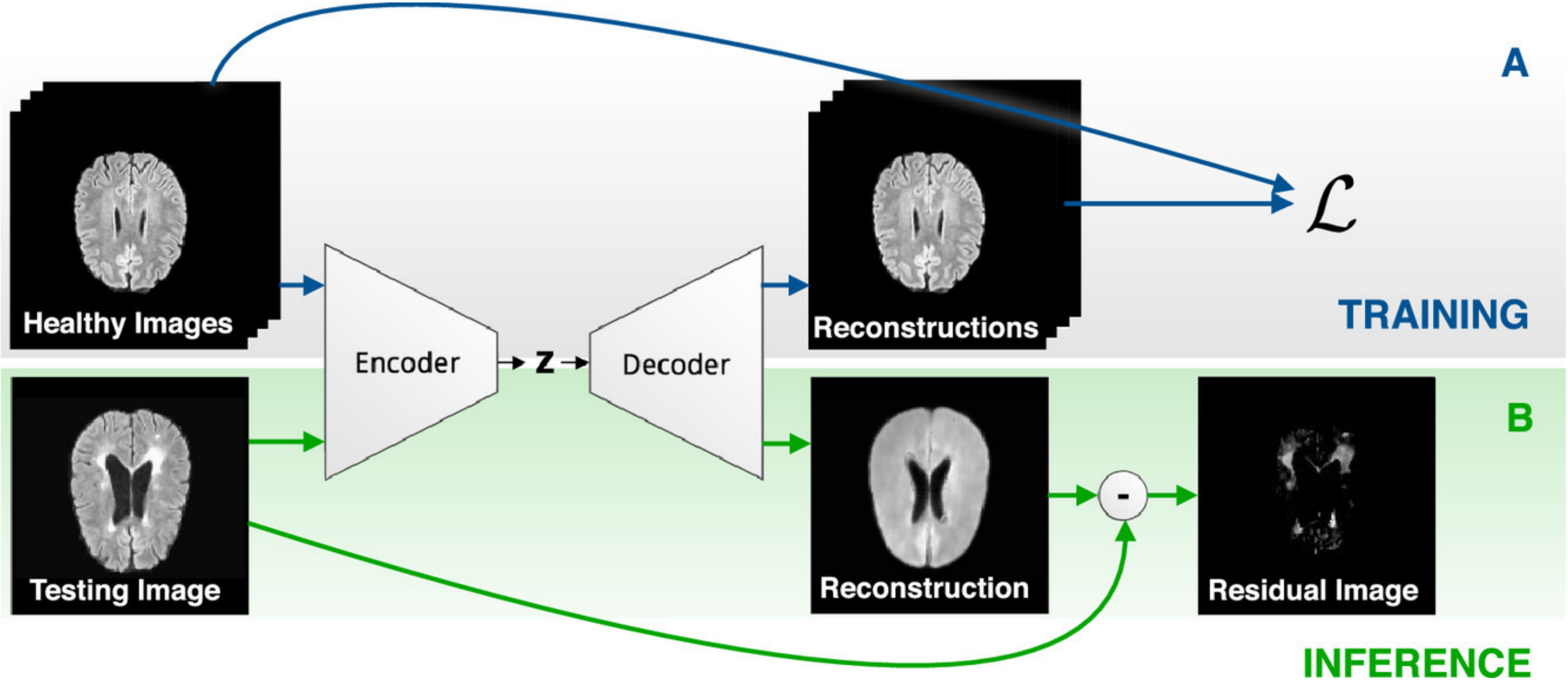
$$q_\theta(\vec{z}|\vec{x}) = \frac{1}{(2\pi)^{Q/2}} \frac{1}{|\Sigma_\theta(\vec{x})|^{1/2}} \exp\left(-\frac{1}{2}\left(\vec{z}-\vec{\mu}_\theta(\vec{x})\right)^T \Sigma_\theta^{-1}(\vec{x}) \left(\vec{z}-\vec{\mu}_\theta(\vec{x})\right)\right)$$

→ mean $\vec{\mu}$ and covariance Σ are functions of the data and determined by a neural network.

Outline

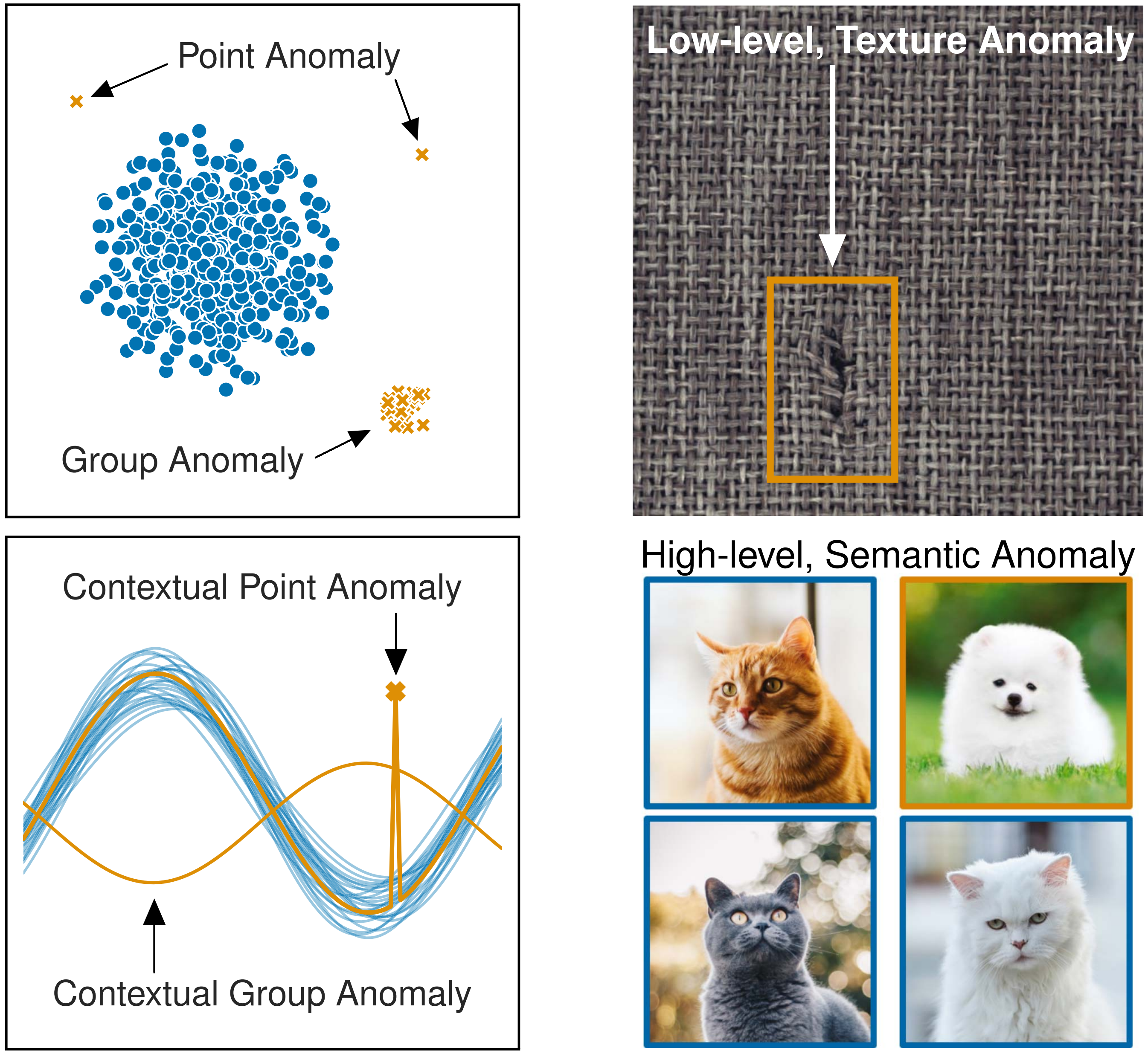
- Unsupervised learning
- Autoencoders
- Autoencoders for anomaly detection
- Anomaly searches in particle physics

Anomaly detection with autoencoders



Baur et al., https://arxiv.org/abs/2004.03271

Anomaly detection

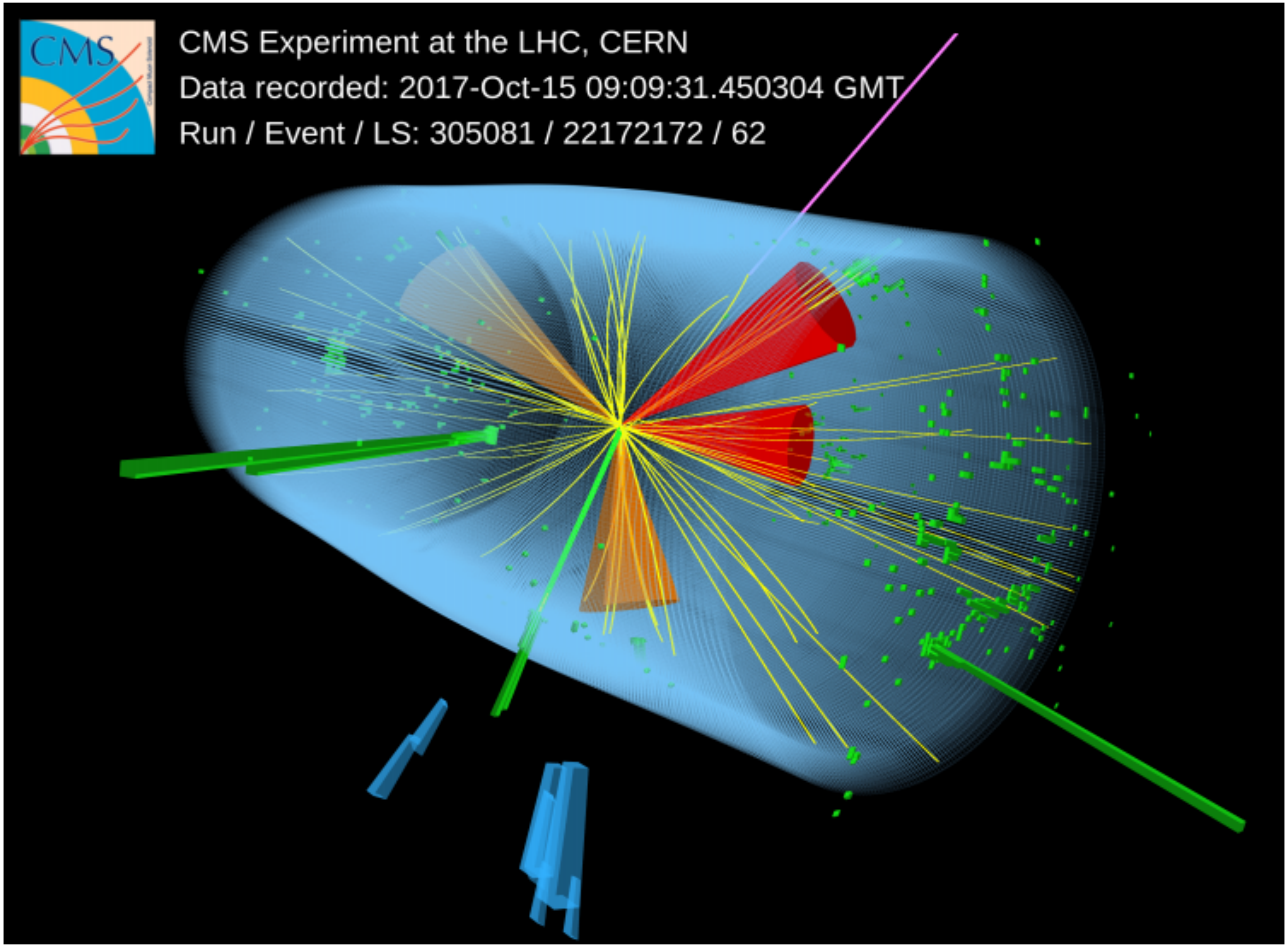


Ruff et al., https://arxiv.org/abs/2009.11732

Outline

- Unsupervised learning
- Autoencoders
- Autoencoders for anomaly detection
- Anomaly searches in particle physics

Particle collisions at the LHC



Particle collisions at the LHC

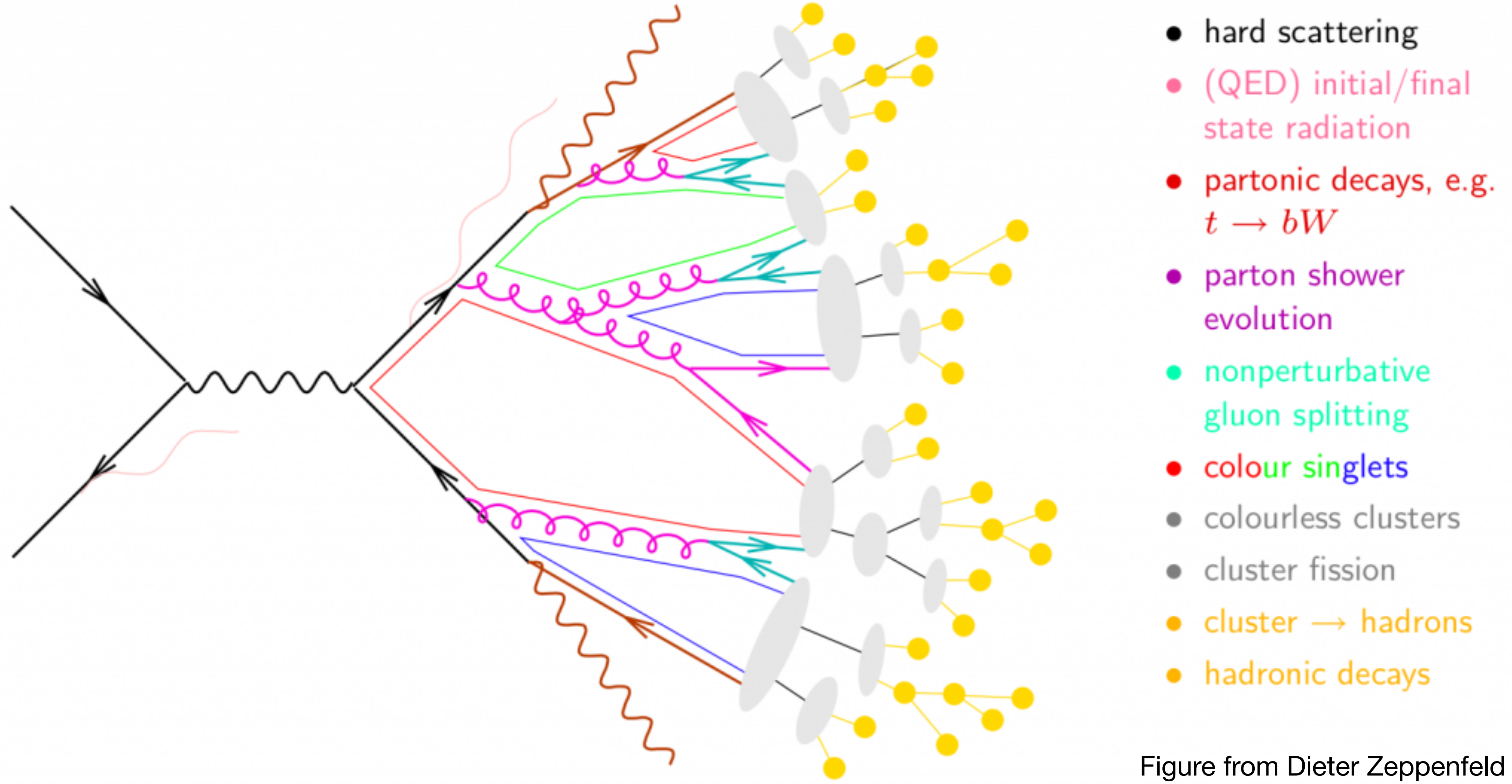
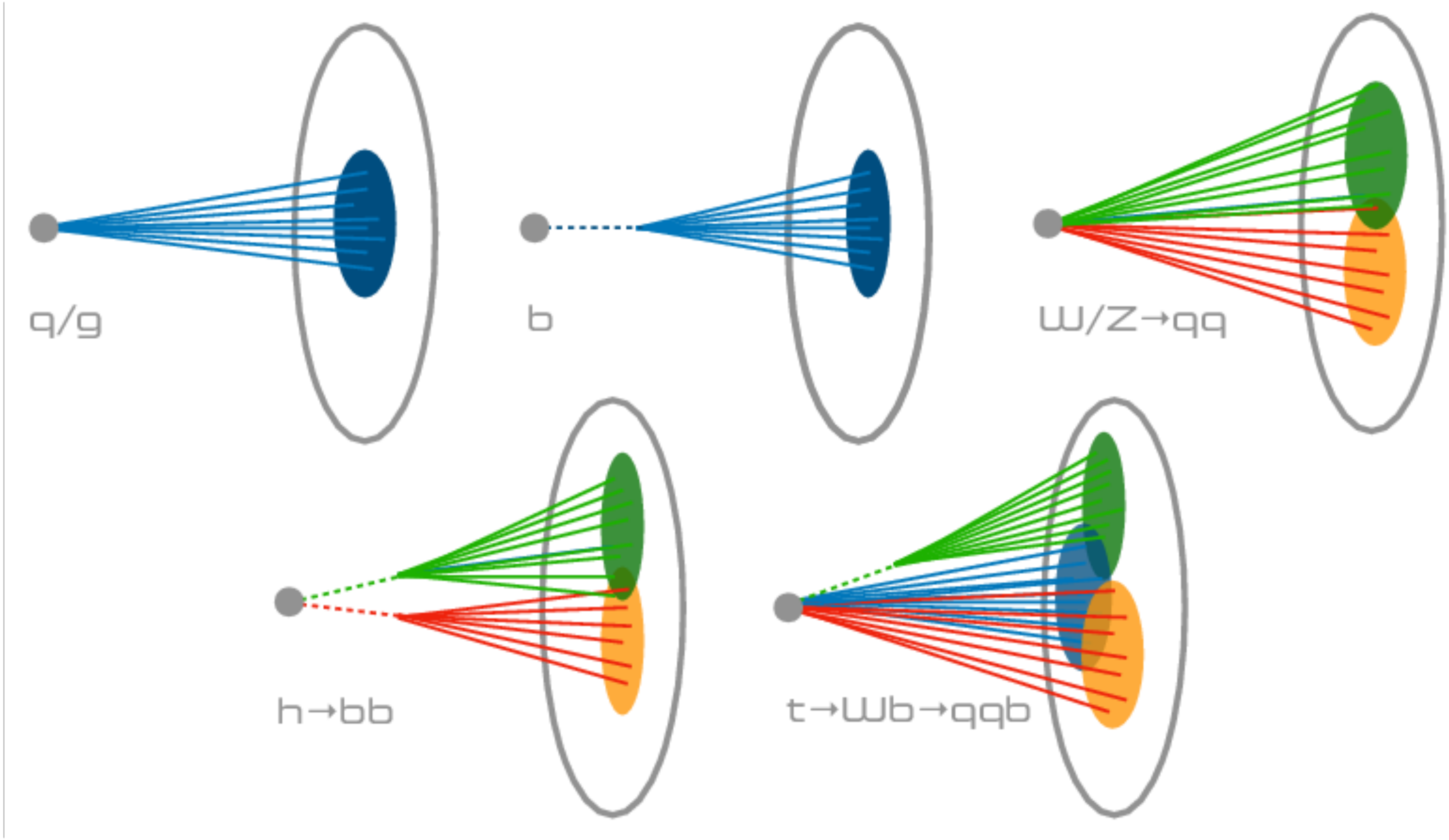


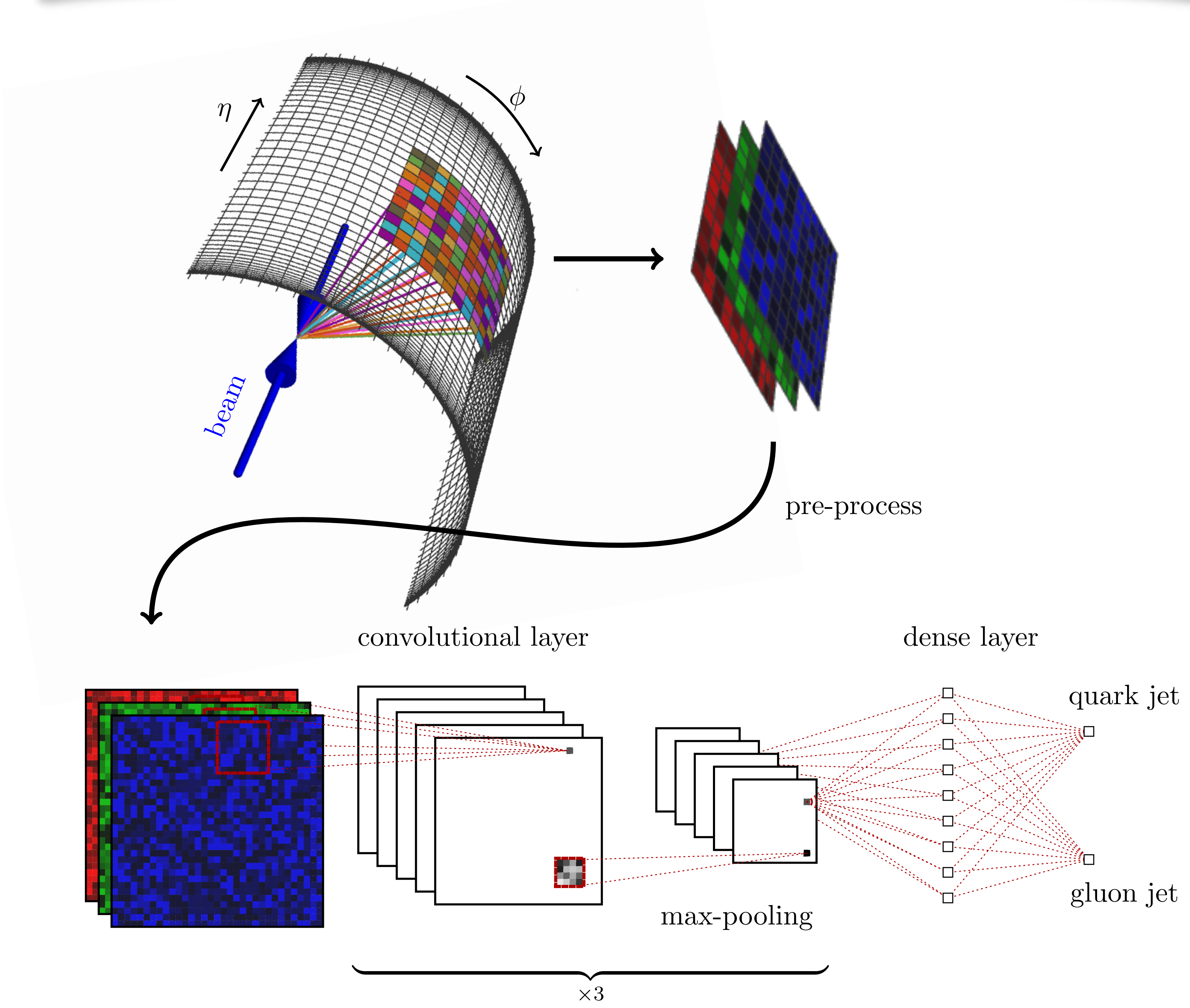
Figure from Dieter Zeppenfeld

Jets at the LHC



Moreno et al., https://arxiv.org/abs/1909.12285

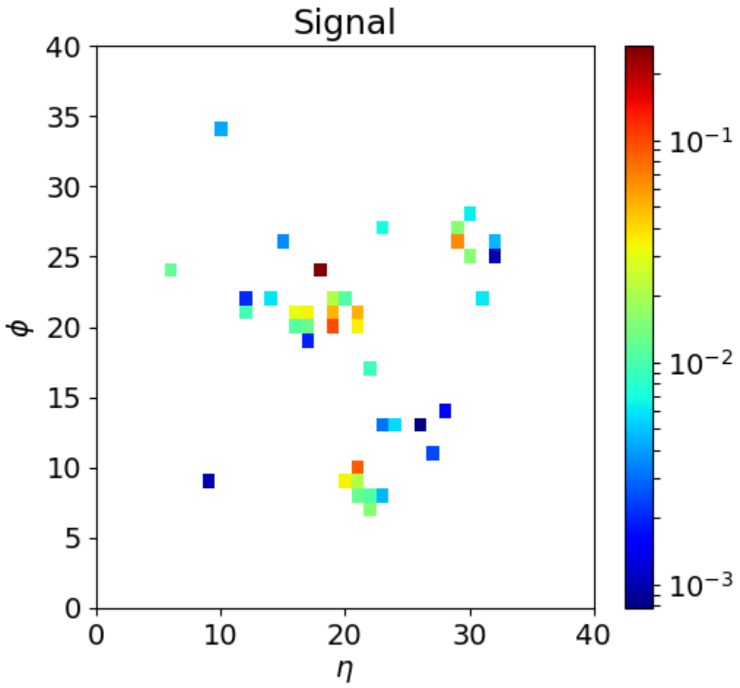
Convolutional neural networks for jet images



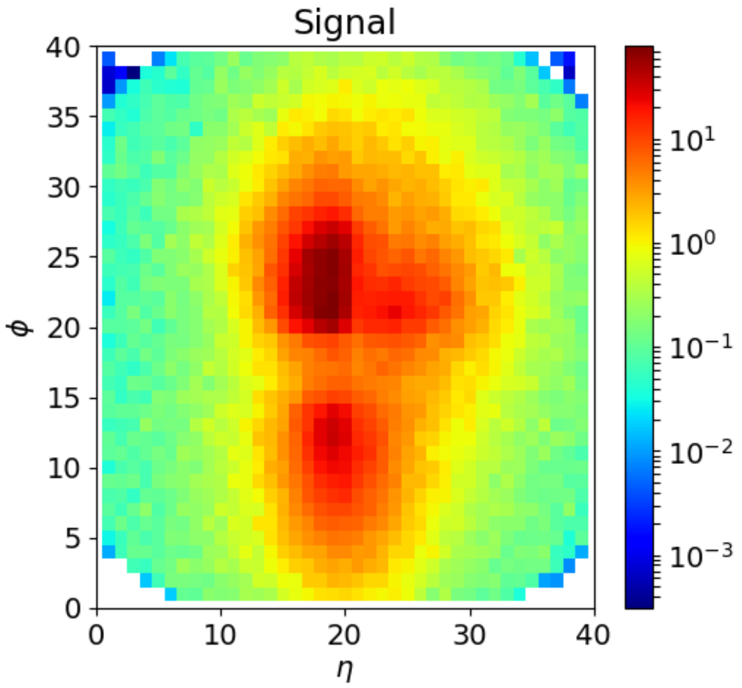
Komiske PT, Metodiev EM, Schwartz MD. J. High Energy Phys. 01:110 (2017)

Jet images

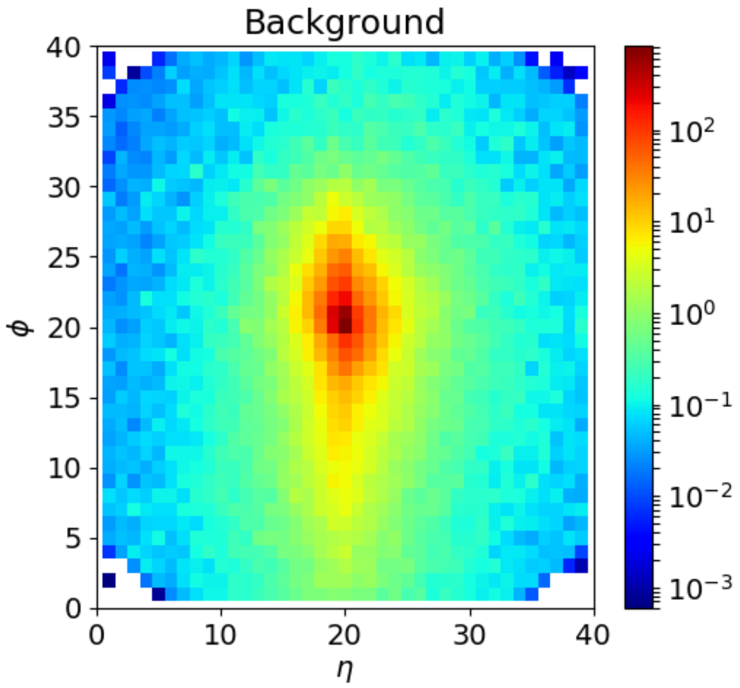
Typical single top-jet



Average top-jet

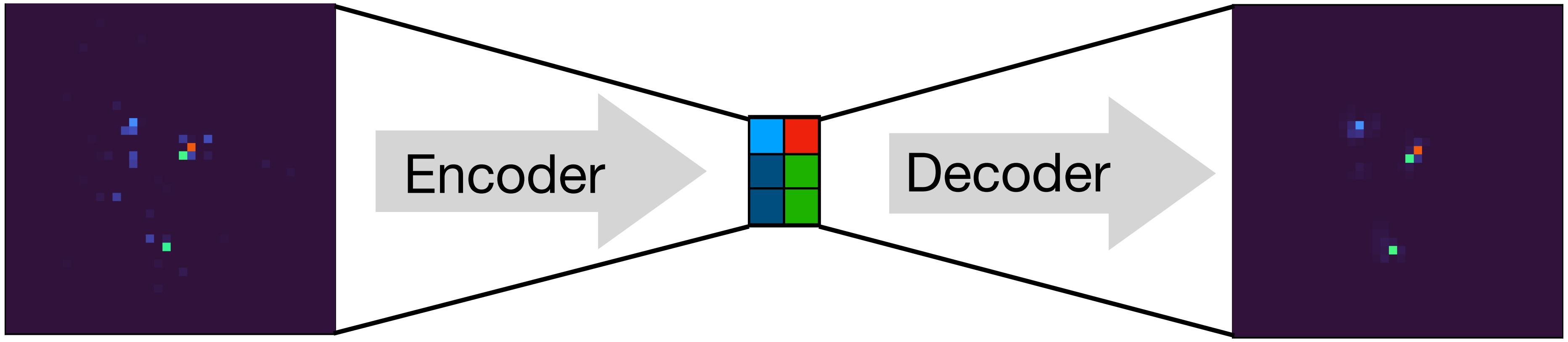


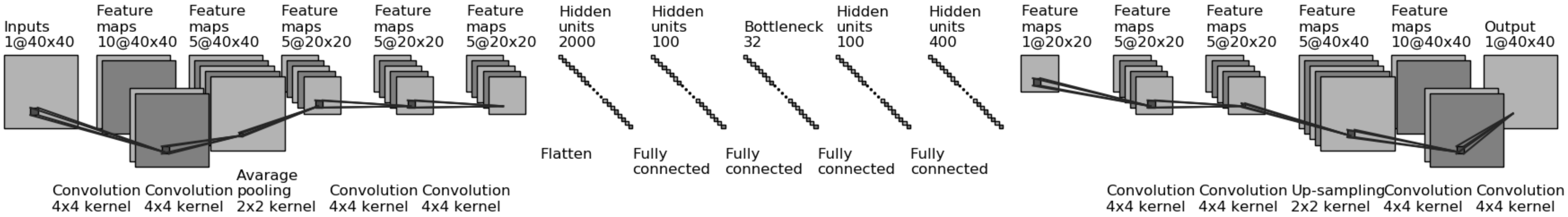
Average QCD-jet



Kasieczka et al., SciPost Phys. 7, 014 (2019)

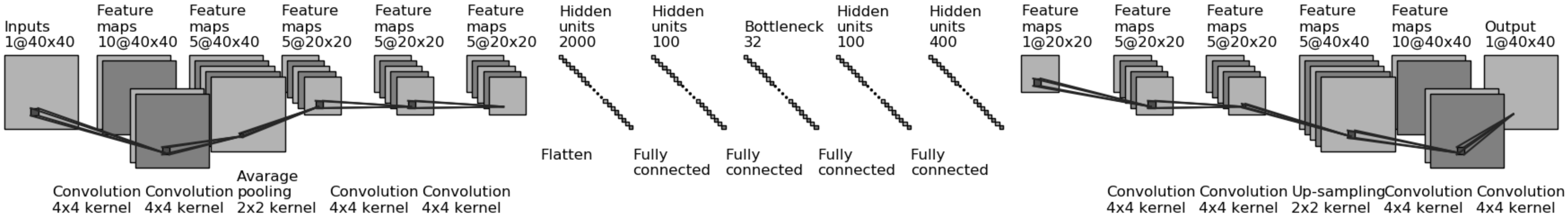
Anomaly detection with autoencoders



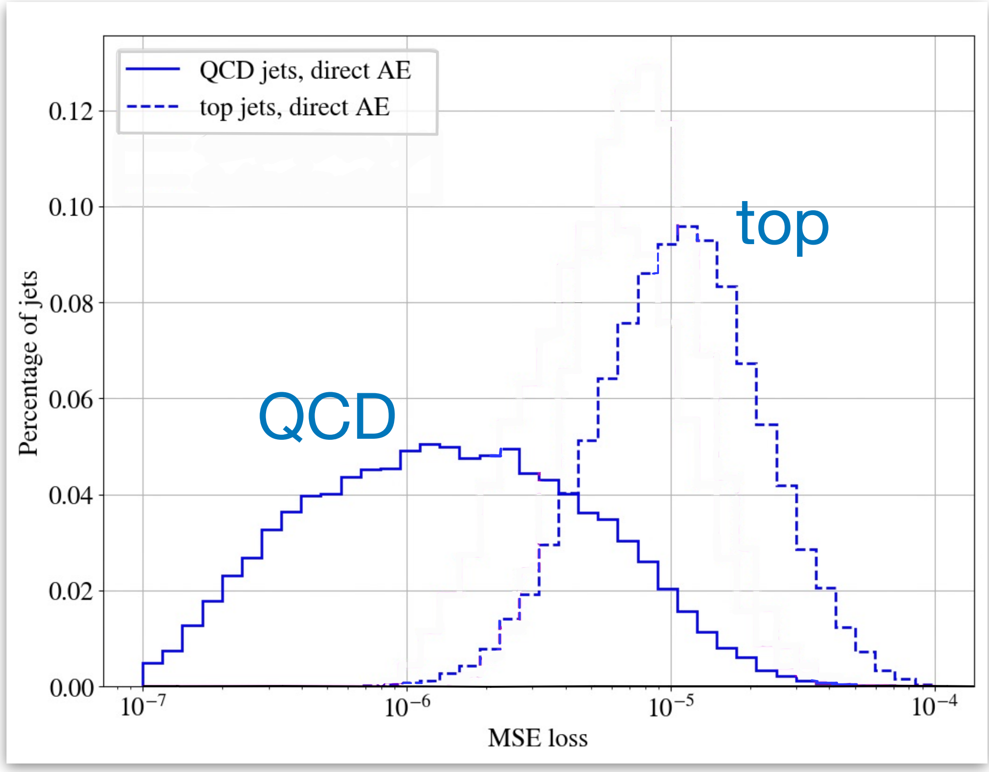


cf Heimel, Kasieczka, Plehn, Thompson, SciPost Phys. 6, 030 (2019); Farina, Nakai, Shih, PRD 101 (2020)

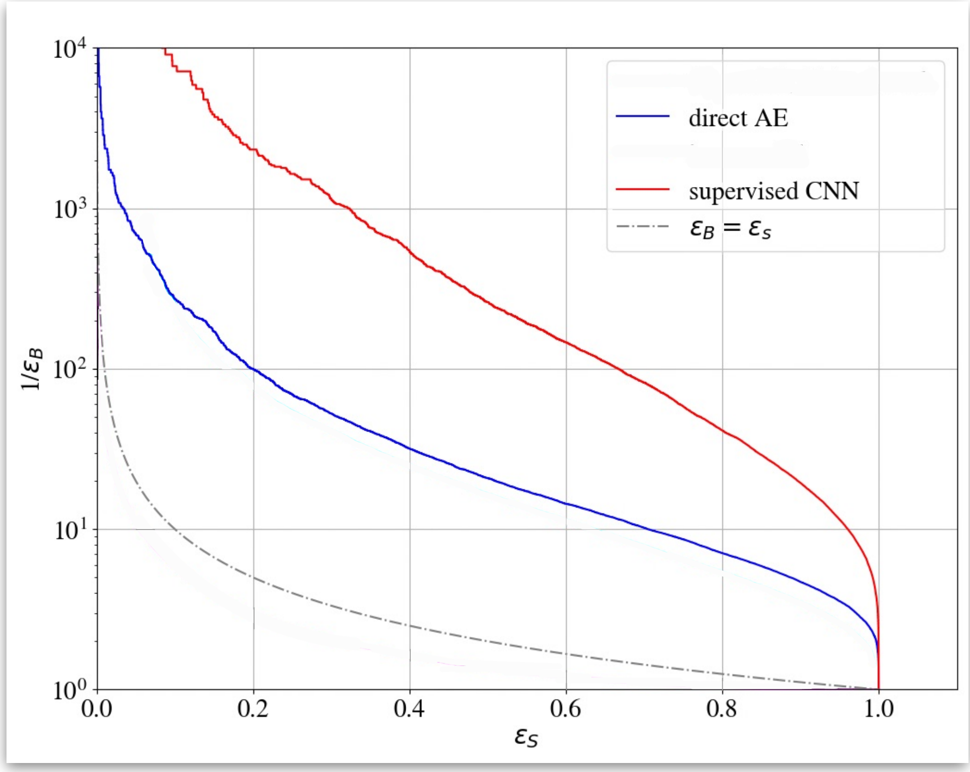
Anomaly detection with autoencoders



% of jets



Background rejection

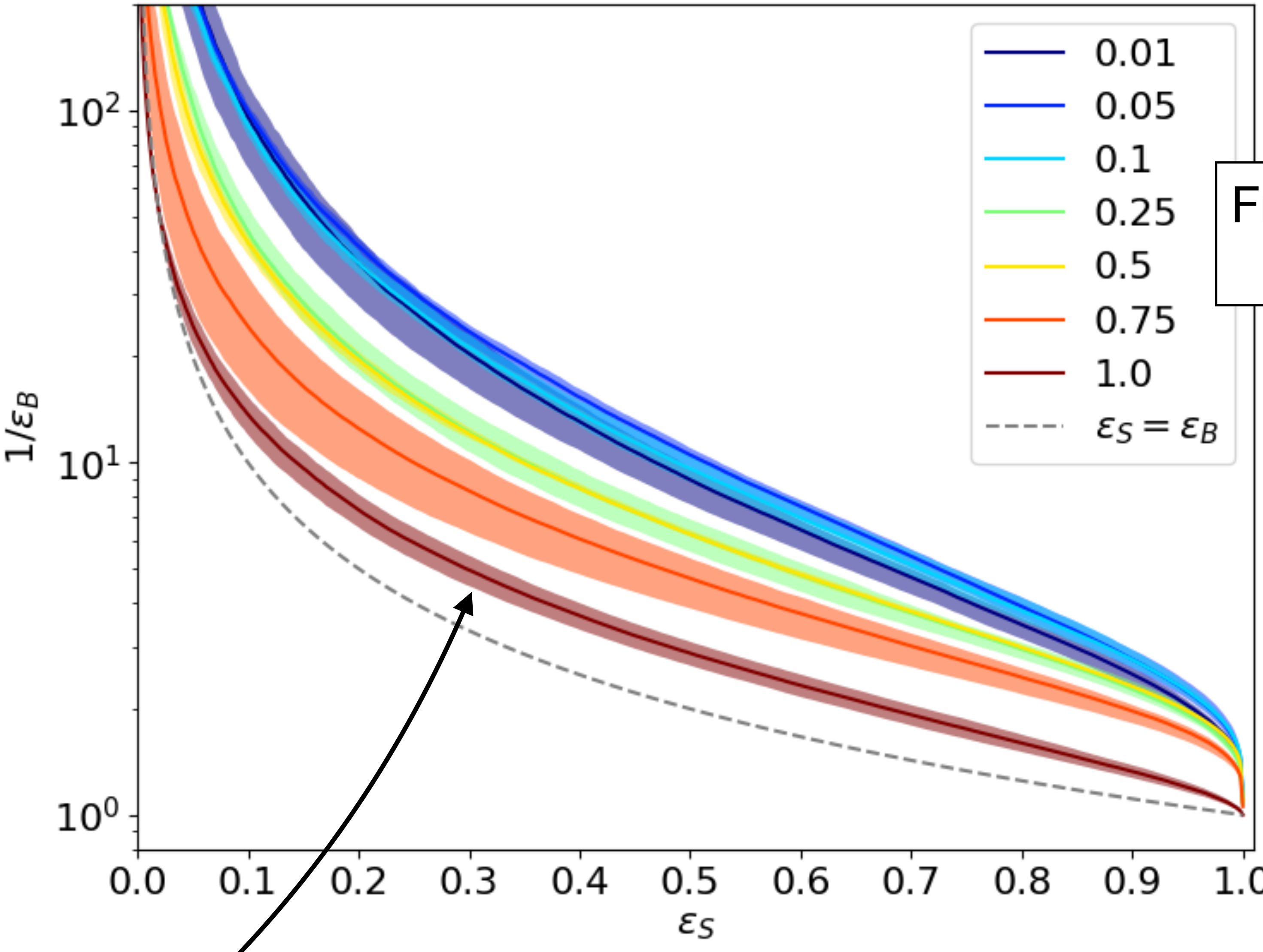


MSE loss

Top tagging efficiency

What does the autoencoder learn?

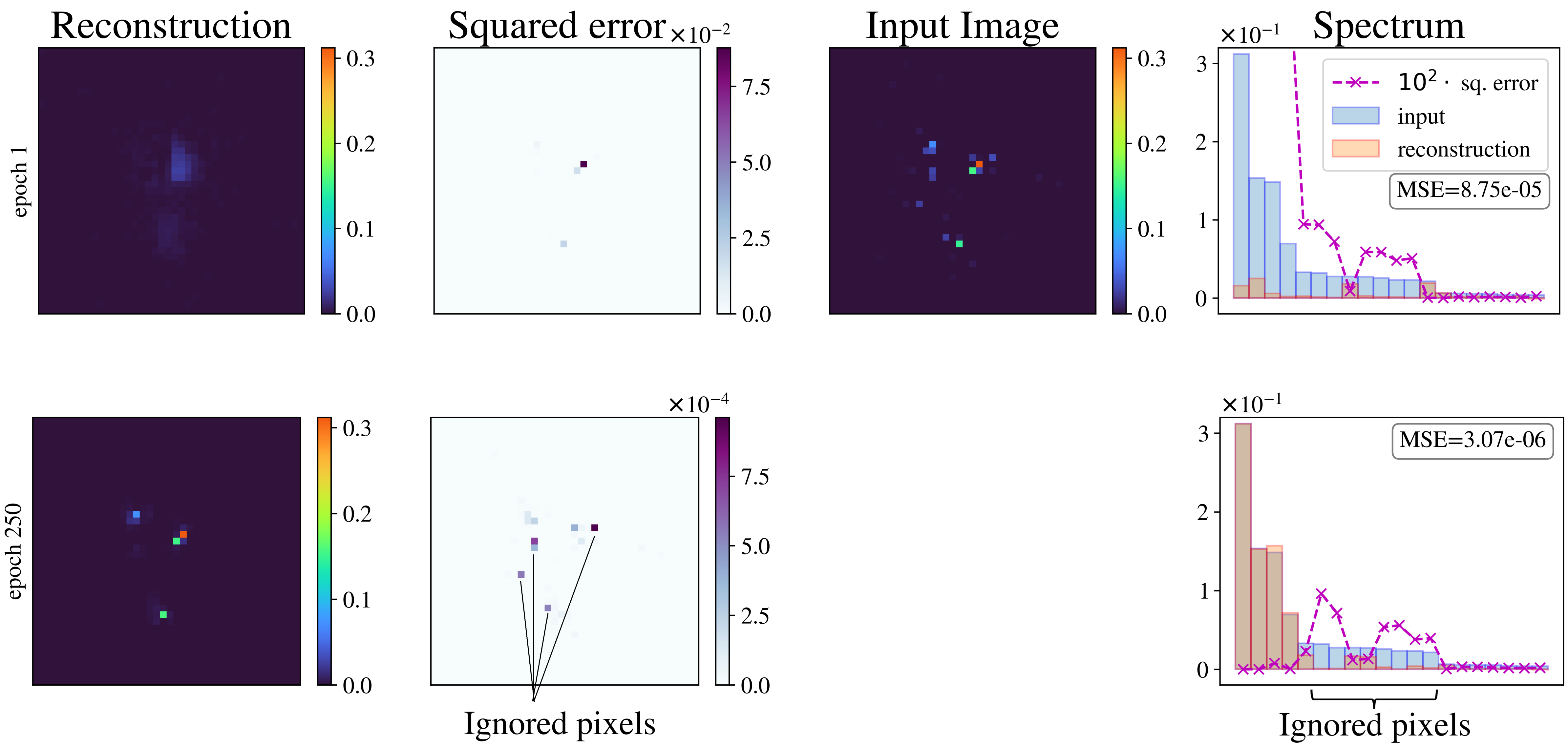
Top tagging: increase the fraction of top events (anomalies) in the training sample:



Fraction of top events in training sample

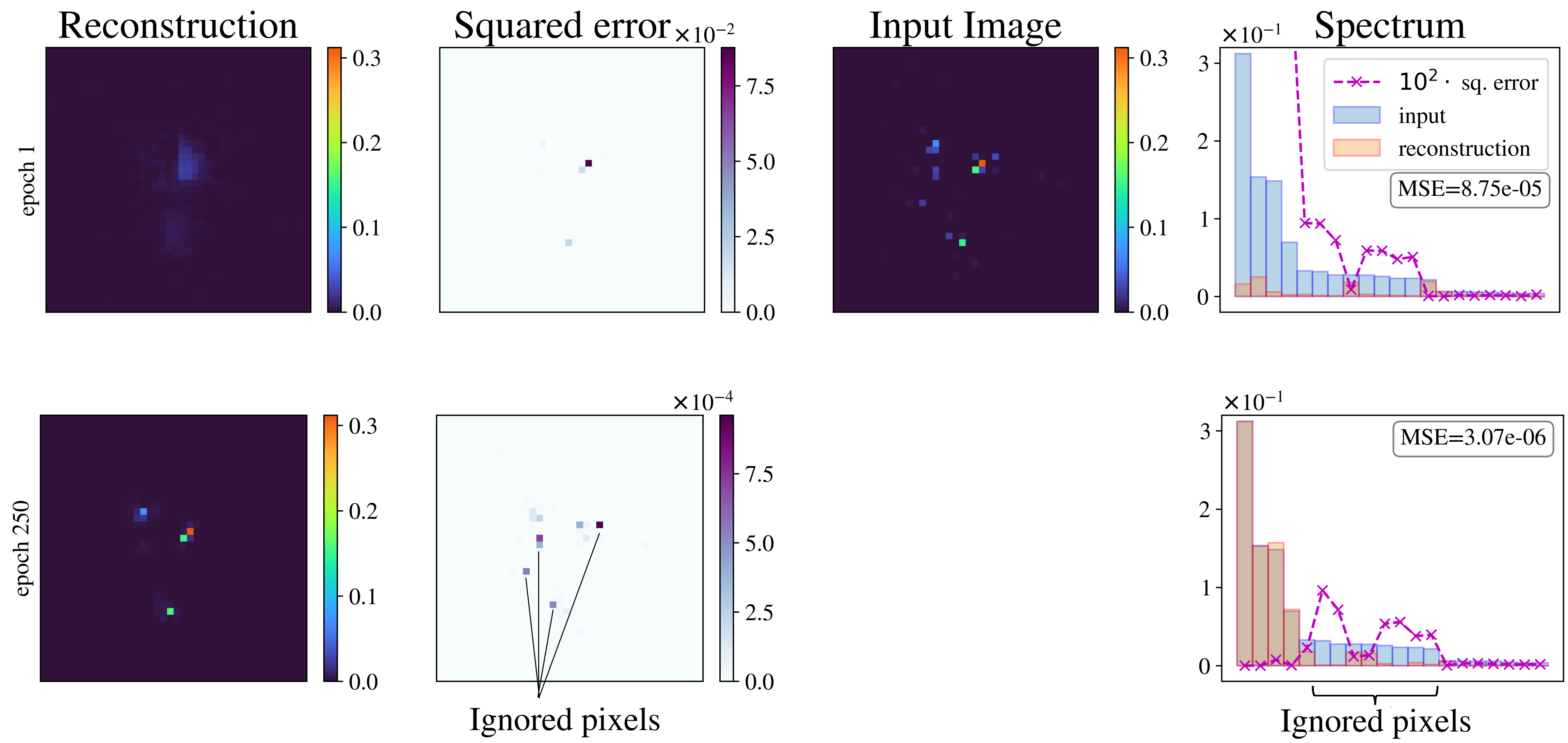
Training on top-jets only, the AE still identifies top-jets as anomalous

What does the autoencoder learn?



Vanilla autoencoder shows a very limited reconstruction capability.

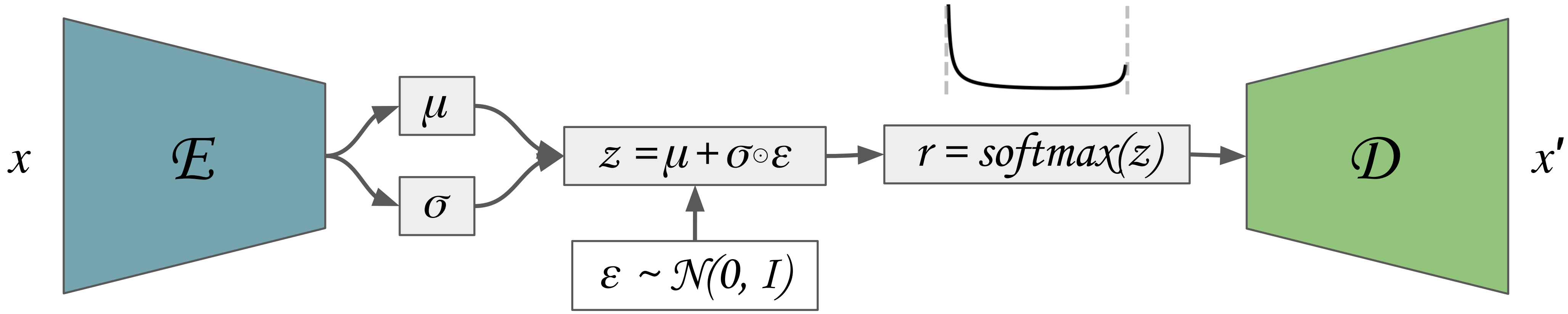
What does the autoencoder learn?



Vanilla autoencoder shows a complexity bias, it tends to better reconstruct ``simpler” images.

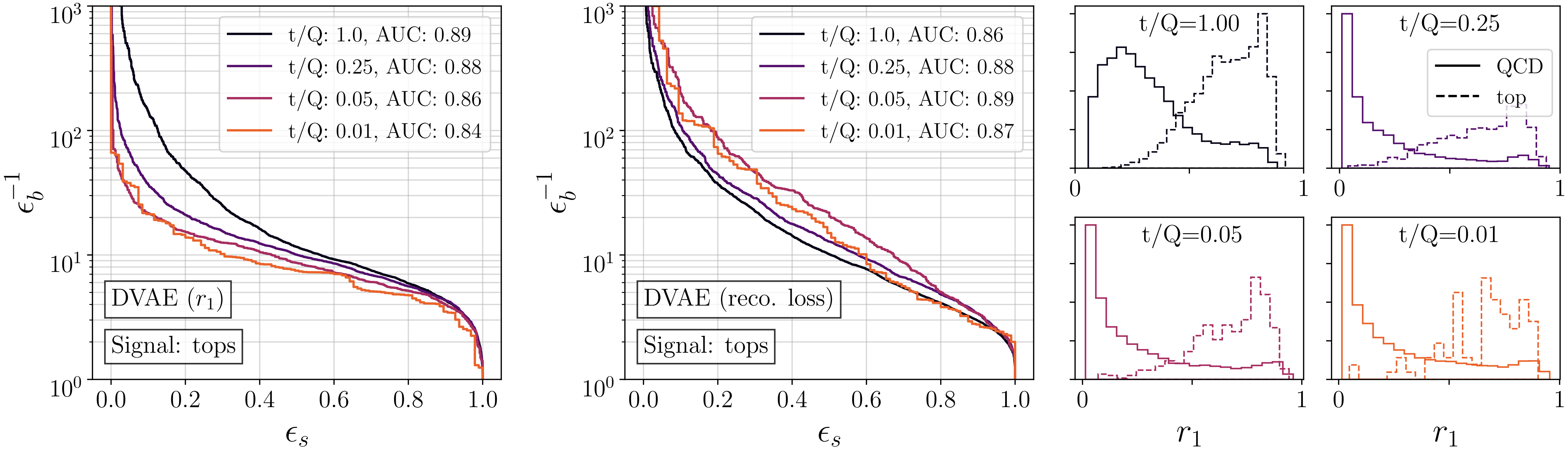
Anomaly detection with autoencoders: outlook

- Regularise the latent space? [Cerri et al., JHEP 05 (2019) 036, Cheng et al., e-Print: 2007.01850 [hep-ph], Dillon et al., SciPost Phys. 11 (2021) 061, …]



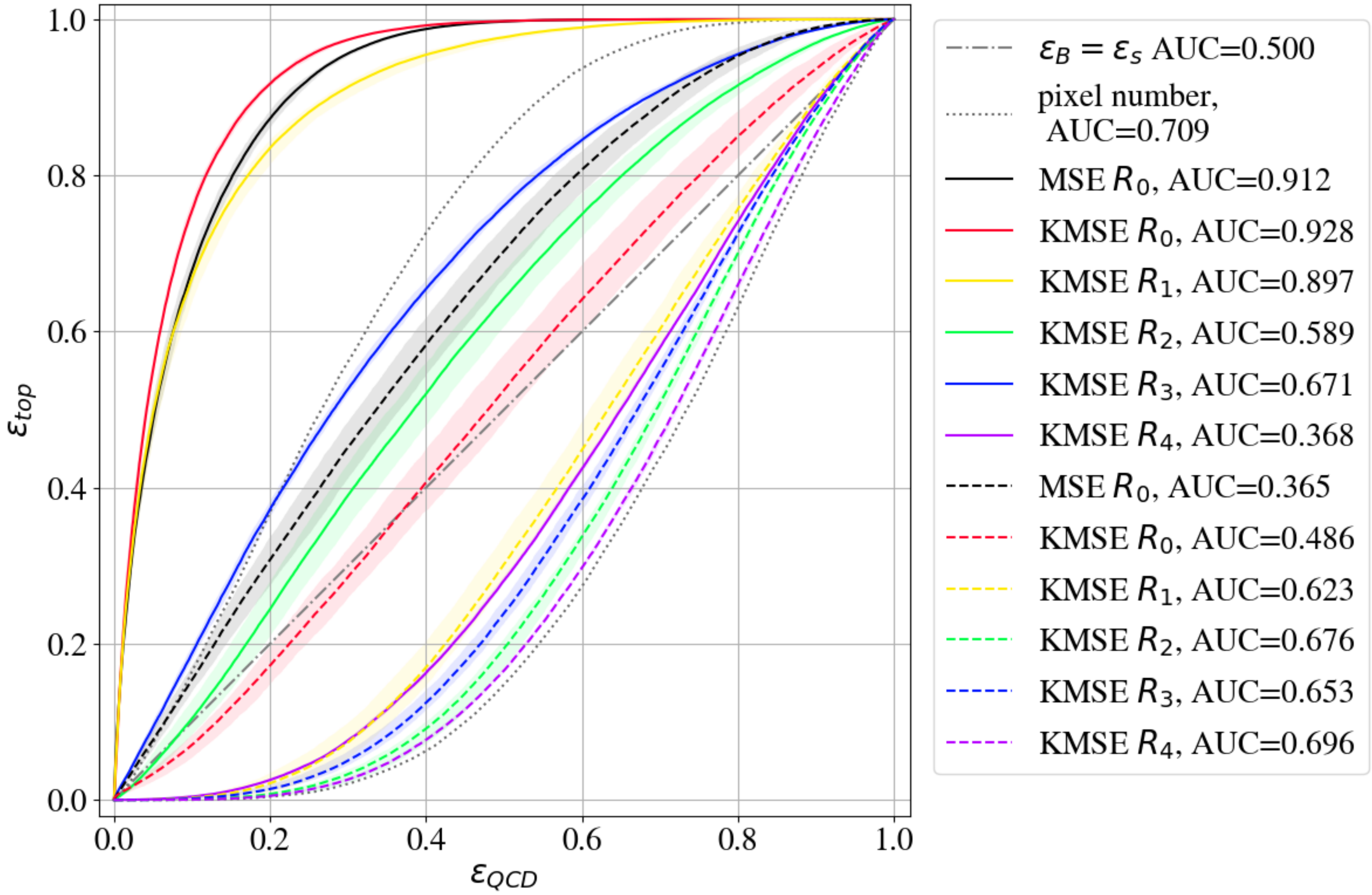
Anomaly detection with autoencoders: outlook

- Regularise the latent space? [Cerri et al., JHEP 05 (2019) 036, Cheng et al., e-Print: 2007.01850 [hep-ph], Dillon et al., SciPost Phys. 11 (2021) 061, …]



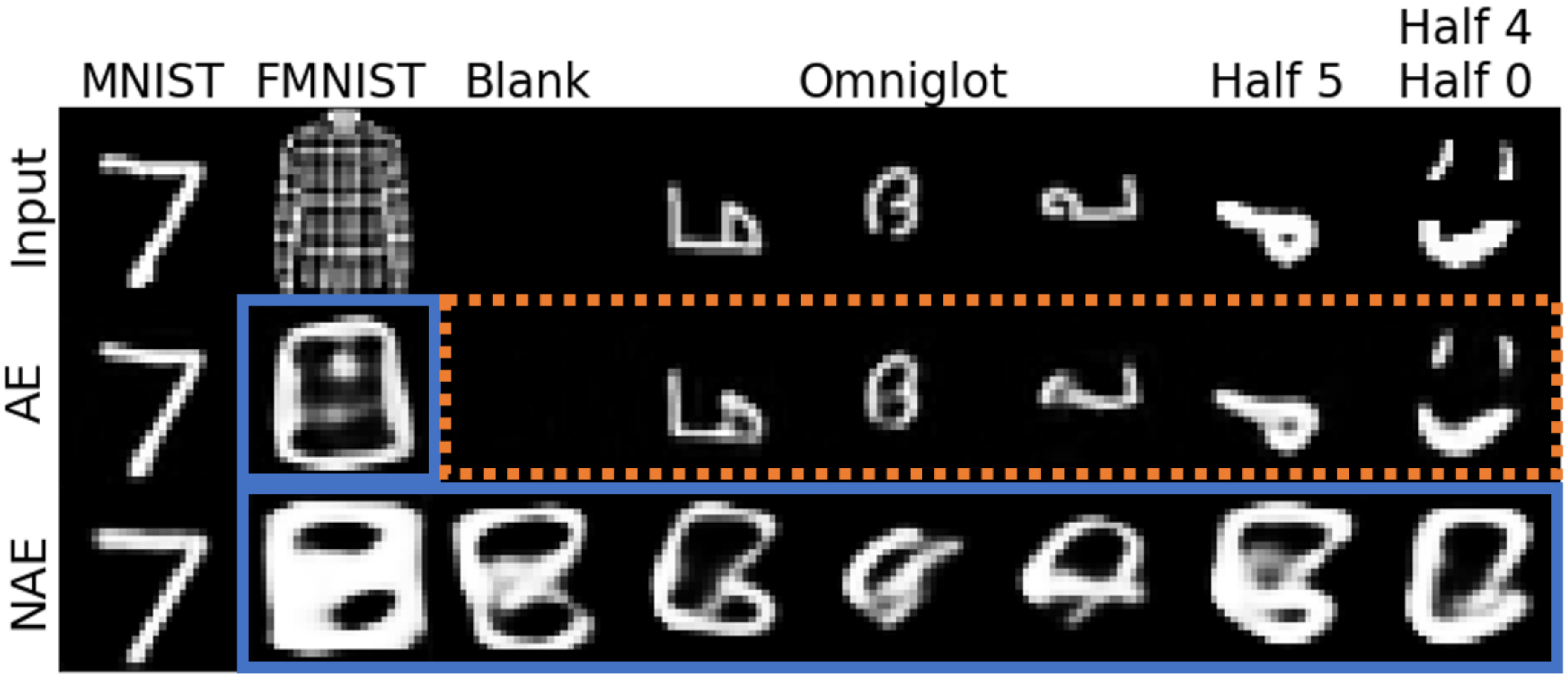
Anomaly detection with autoencoders: outlook

- Improve the performance of the AE through preprocessing: smearing and re-weighting of pixels? [Finke et al., JHEP 06 (2021) 161, Buss et al., e-Print: 2202.00686 [hep-ph], …]



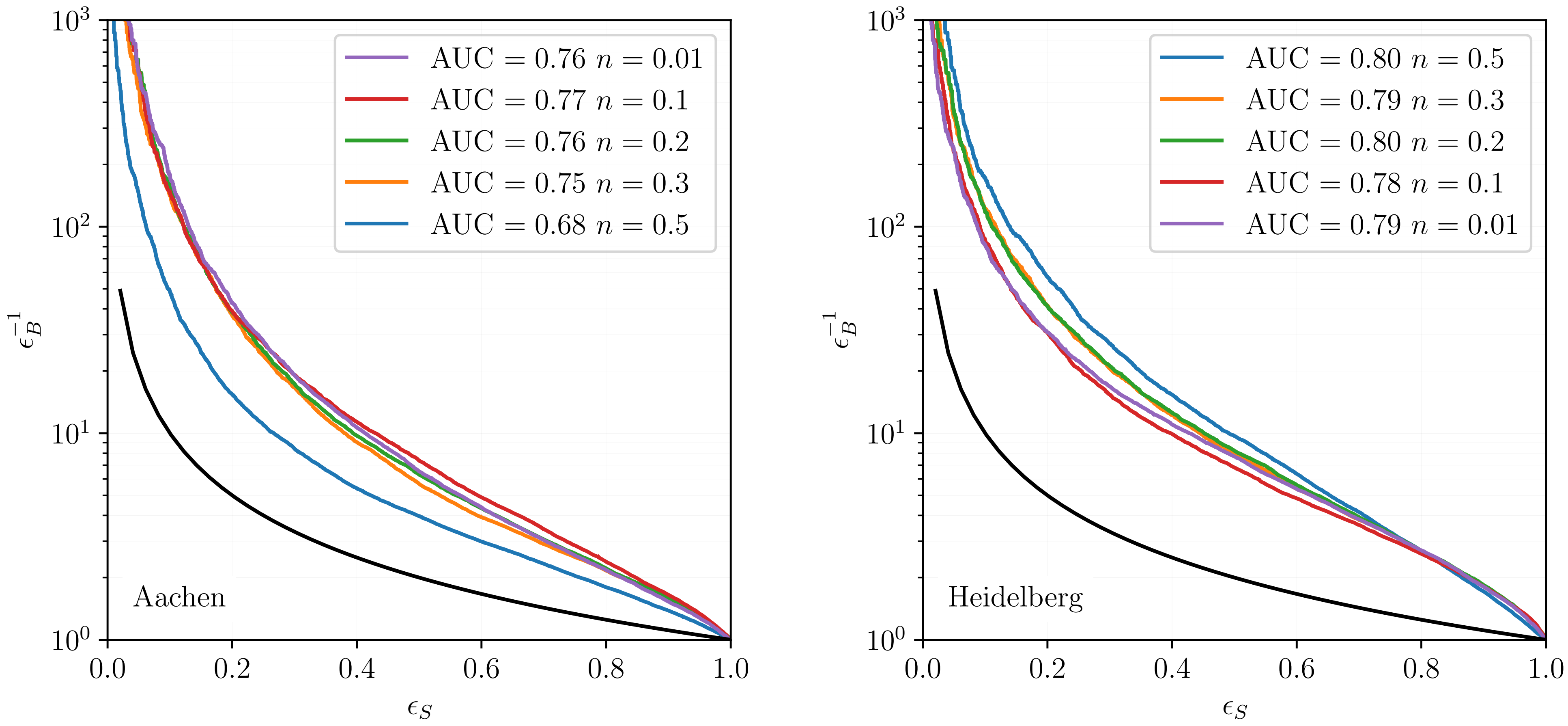
Anomaly detection with autoencoders: outlook

- Introduce normalising condition to prevent outlier reconstruction? [Yoon et al., arXiv:2105.05735 [cs.LG], Dillon et al., arXiv:2206.14225 [hep-ph], …]



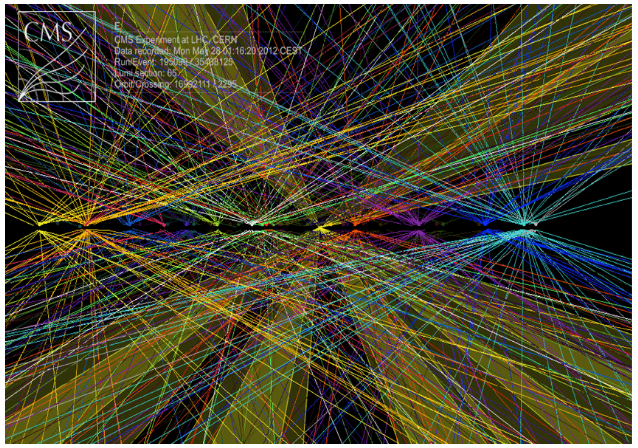
Anomaly detection with autoencoders: outlook

- Introduce normalising condition to prevent outlier reconstruction? [Yoon et al., arXiv:2105.05735 [cs.LG], Dillon et al., arXiv:2206.14225 [hep-ph], …]



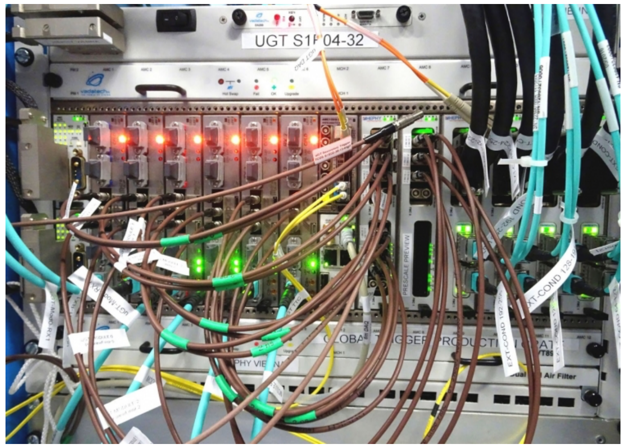
Model independence: searching for anomalies at the LHC

Collisions



40M events/sec

Level 1 trigger

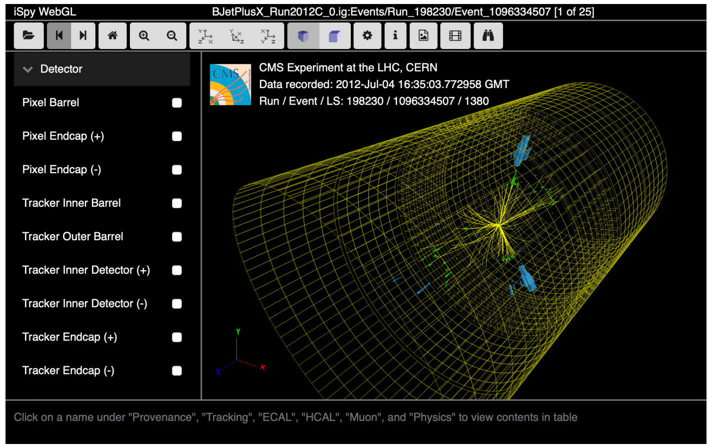


100k events/sec

High-level trigger

1k events/sec

Data analysis



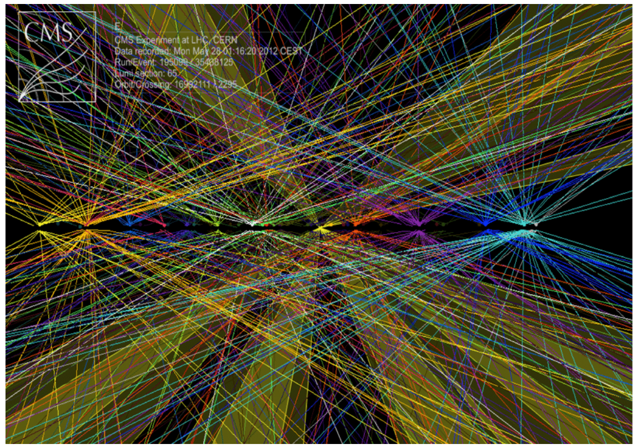
BJetPlusX_Run2012C_0.ig:Events/Run_198230/Event_1096334507 [1 of 25]

CMS Experiment at the LHC, CERN
Data recorded: 2012-Jul-04 16:35:03.772958 GMT
Run / Event / LS: 198230 / 1096334507 / 1380

Click on a name under "Provenance", "Tracking", "ECAL", "HCAL", "Muon", and "Physics" to view contents in table

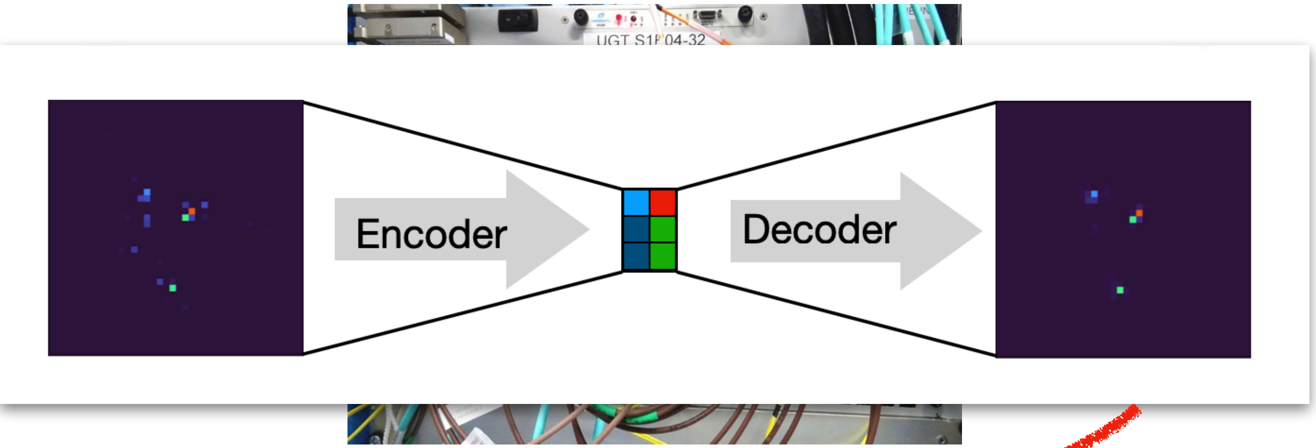
Model independence: searching for anomalies at the LHC

Collisions



40M ever

Level 1 trigger



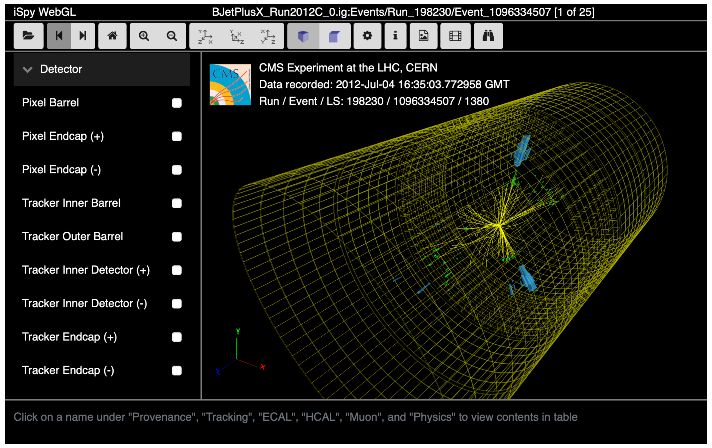
100k events/sec

New physics?

High-level trigger

1k events/sec

Data analysis



iSpy WebGL

BJetPlusX_Run2012C_0.ig:Events/Run_198230/Event_1096334507 [1 of 25]

Detector

- Pixel Barrel
- Pixel Endcap (+)
- Pixel Endcap (-)
- Tracker Inner Barrel
- Tracker Outer Barrel
- Tracker Inner Detector (+)
- Tracker Inner Detector (-)
- Tracker Endcap (+)
- Tracker Endcap (-)

CMS Experiment at the LHC, CERN
Data recorded: 2012-Jul-04 16:35:03.772958 GMT
Run / Event / LS: 198230 / 1096334507 / 1380

Click on a name under "Provenance", "Tracking", "ECAL", "HCAL", "Muon", and "Physics" to view contents in table

Anomaly detection data challenge: https://mpp-hep.github.io/ADC2021/

Anomaly detection with autoencoders: outlook

- Regularise the latent space? [Cerri et al., JHEP 05 (2019) 036, Cheng et al., e-Print: 2007.01850 [hep-ph], Dillon et al., SciPost Phys. 11 (2021) 061, …]
- Improve the performance of the AE through preprocessing: smearing and re-weighting of pixels? [Finke et al., JHEP 06 (2021) 161, Buss et al., e-Print: 2202.00686 [hep-ph], …]
- Introduce normalising condition to prevent outlier reconstruction? [Yoon et al., arXiv:2105.05735 [cs.LG], Dillon et al., arXiv:2206.14225 [hep-ph], …]
- Can we construct a fast autoencoder to detect anomalies in real time?

Thank you