Machine Learning in Particle Physics

Pheno 2022

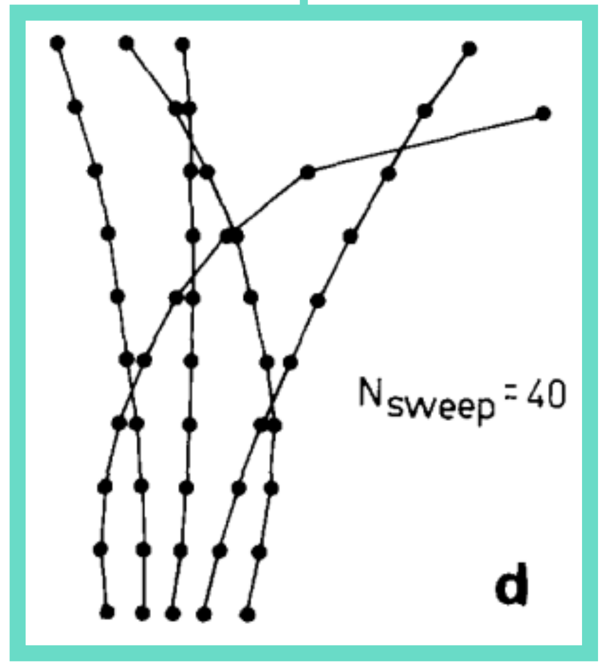
Anja Butter, ITP Heidelberg

A short history of ML in HEP

First HEP NN papers

Track finding
Denby (LAL, Orsay) ’87
Peterson (Lund) ’88
→ Jet identification

1987/88

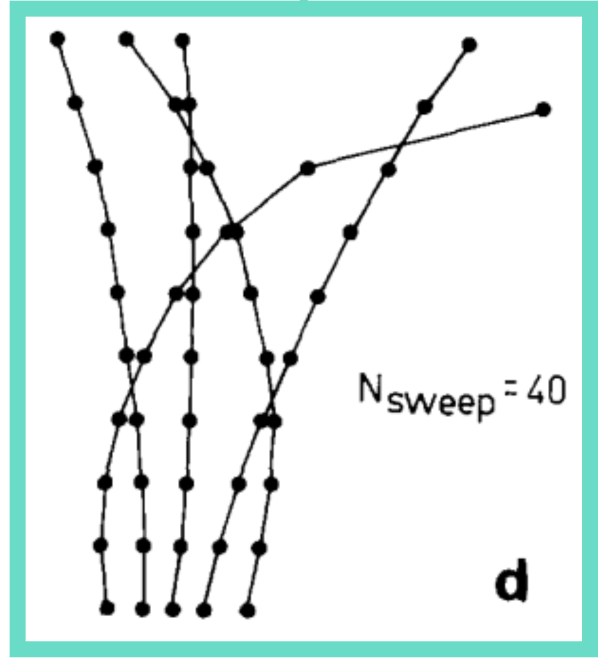


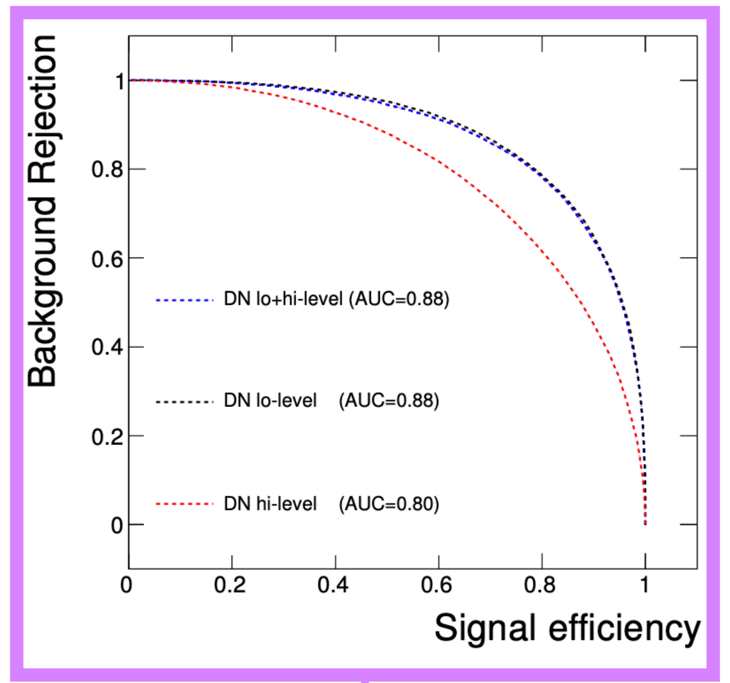
A short history of ML in HEP

First HEP NN papers

Track finding
Denby (LAL, Orsay) ’87
Peterson (Lund) ’88
→ Jet identification

1987/88





2014

Relaunch

Deep Learning in HEP
Signal vs Background

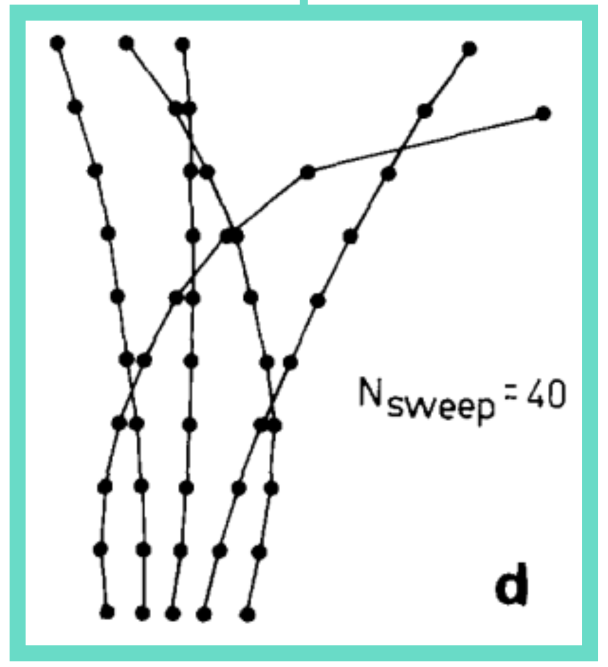
P. Baldi, P. Sadowski,
D. Whiteson

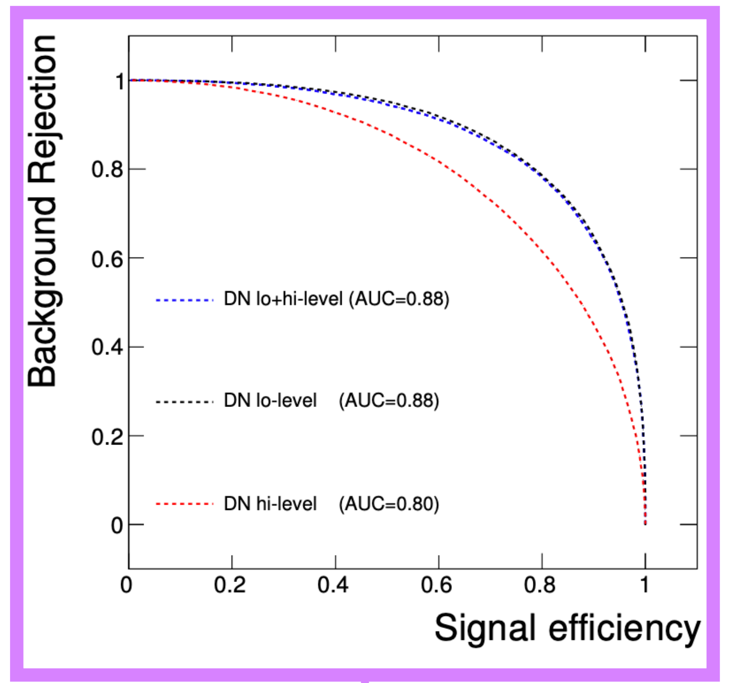
A short history of ML in HEP

First HEP NN papers

Track finding
Denby (LAL, Orsay) ’87
Peterson (Lund) ’88
→ Jet identification

1987/88





2014

Relaunch

Deep Learning in HEP
Signal vs Background

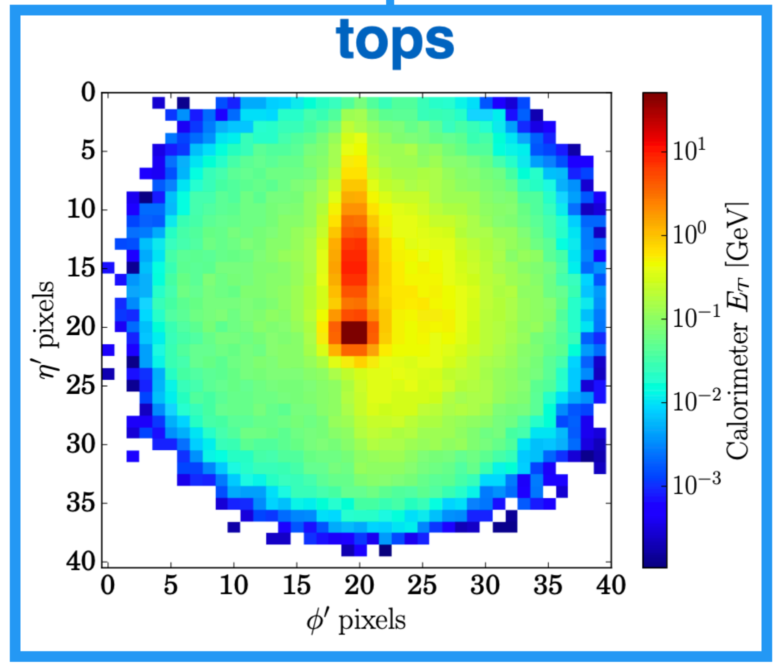
P. Baldi, P. Sadowski,
D. Whiteson

First NN@Pheno

Top tagging
M. Russel, L. Huang

CWoLa
J. Collins

2018

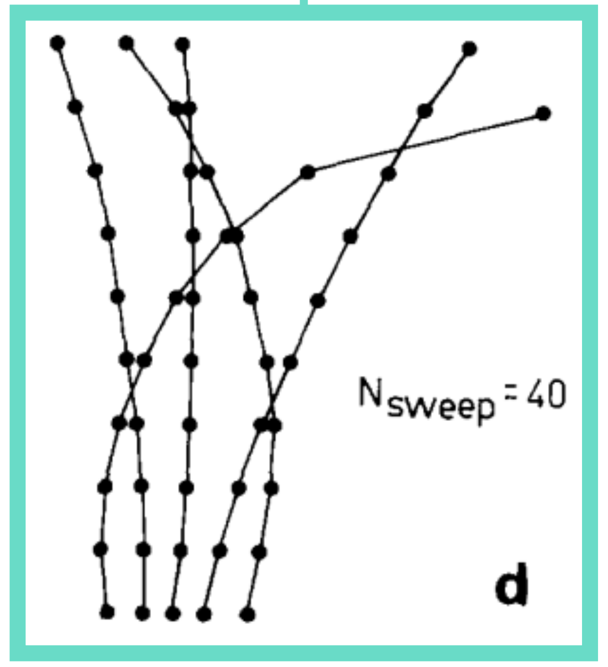


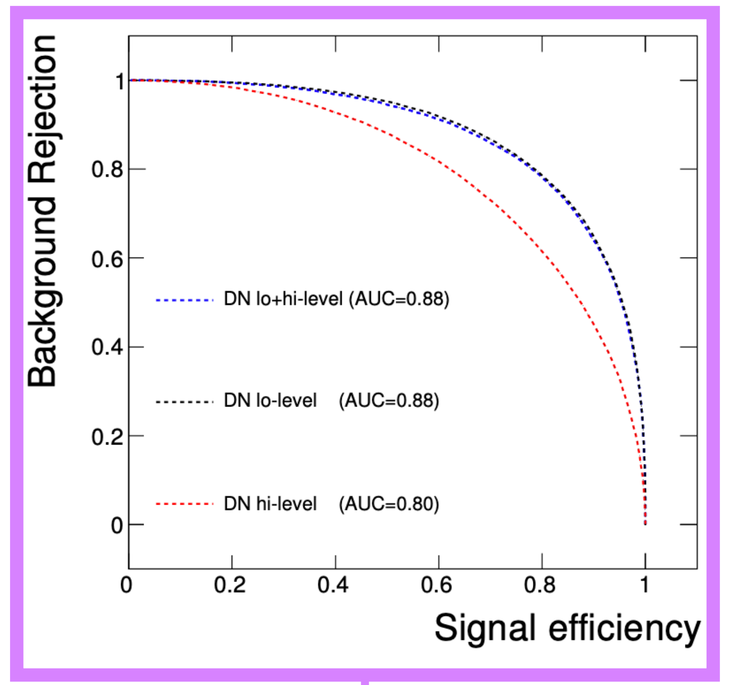
A short history of ML in HEP

First HEP NN papers

Track finding
Denby (LAL, Orsay) ’87
Peterson (Lund) ’88
→ Jet identification

1987/88





2014

Relaunch

Deep Learning in HEP
Signal vs Background

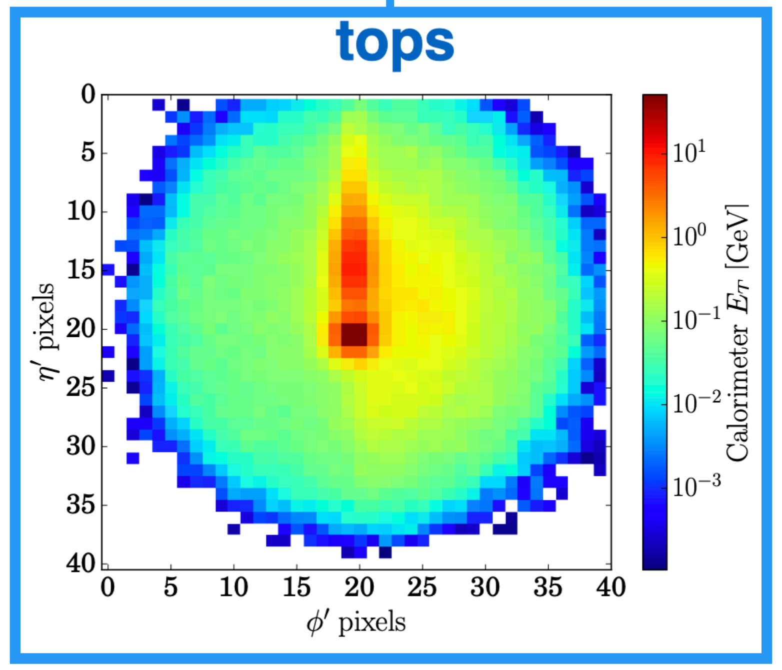
P. Baldi, P. Sadowski,
D. Whiteson

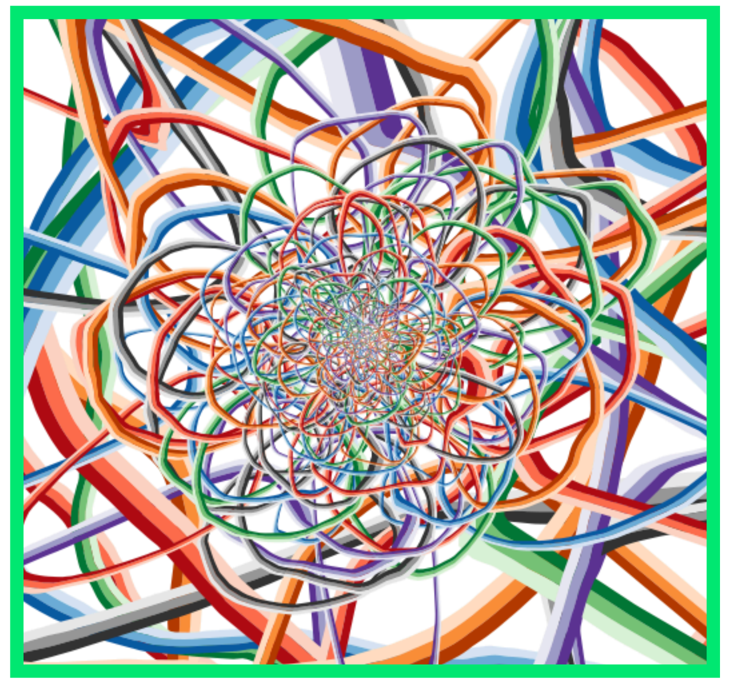
First NN@Pheno

Top tagging
M. Russel, L. Huang

CWoLa
J. Collins

2018





2019

First Pheno plenary talk

Deep Thinking

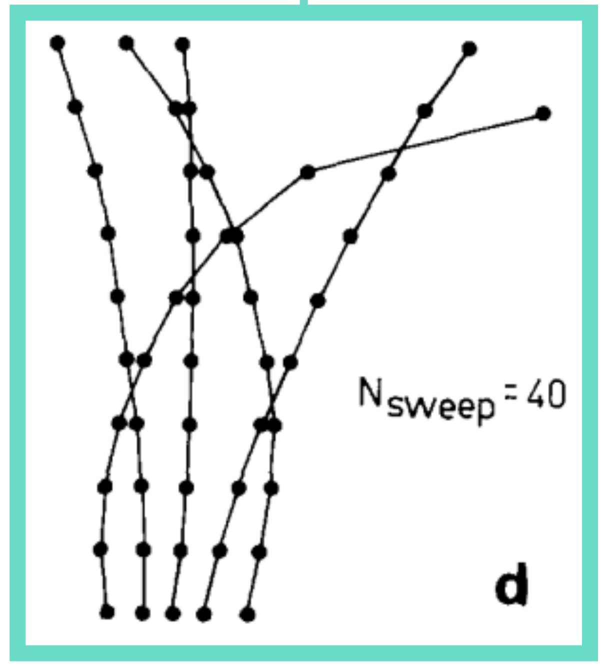
J. Thaler

A short history of ML in HEP

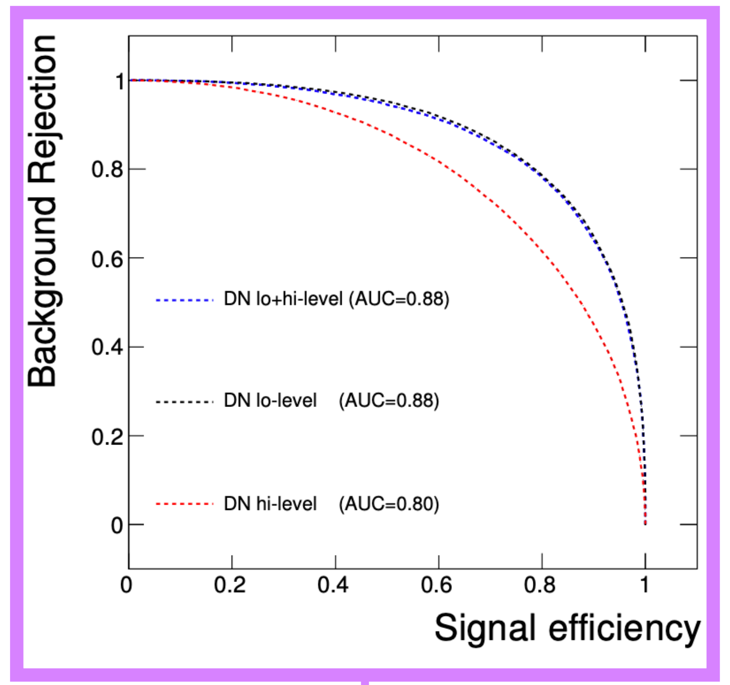
1987/88

First HEP NN papers

Track finding
Denby (LAL, Orsay) ’87
Peterson (Lund) ’88
→ Jet identification



2014



Relaunch

Deep Learning in HEP
Signal vs Background

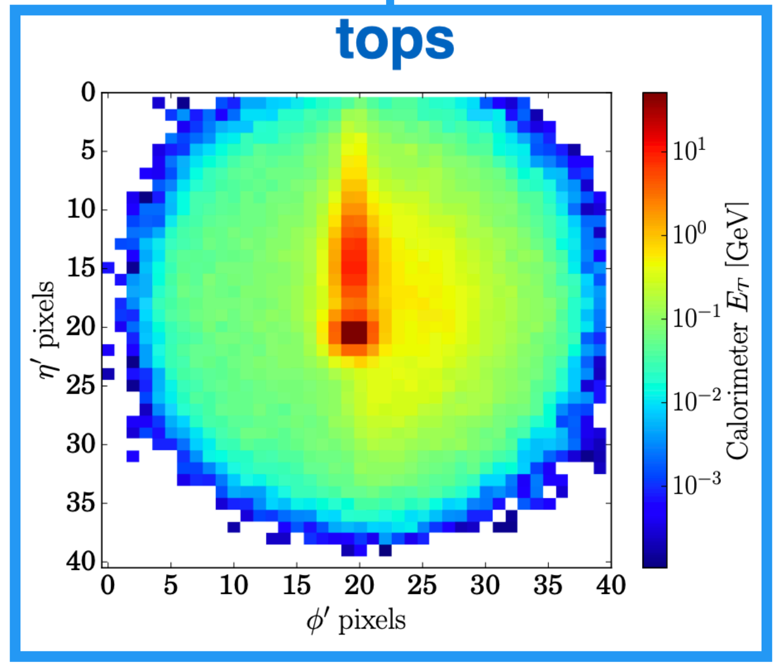
P. Baldi, P. Sadowski, D. Whiteson

2018

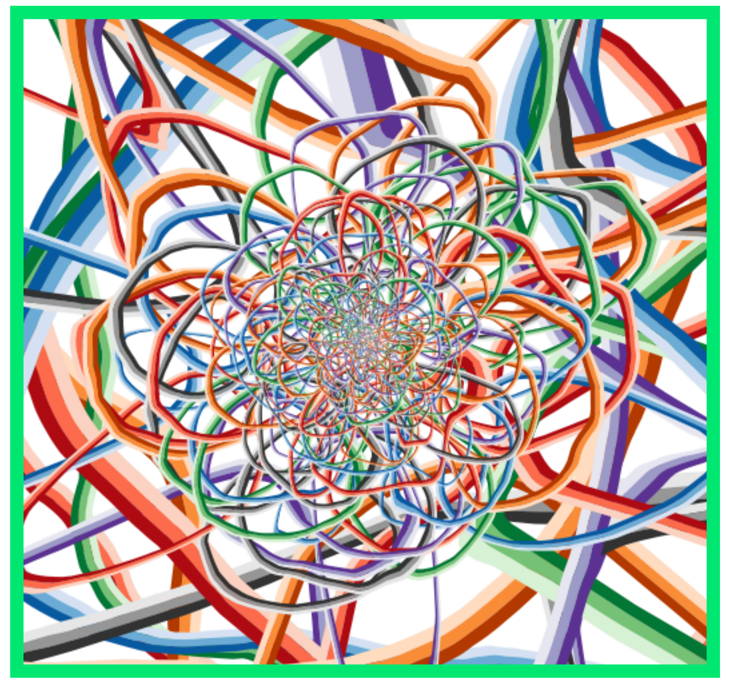
First NN@Pheno

Top tagging
M. Russel, L. Huang

CWoLa
J. Collins



2019



First Pheno plenary talk

Deep Thinking

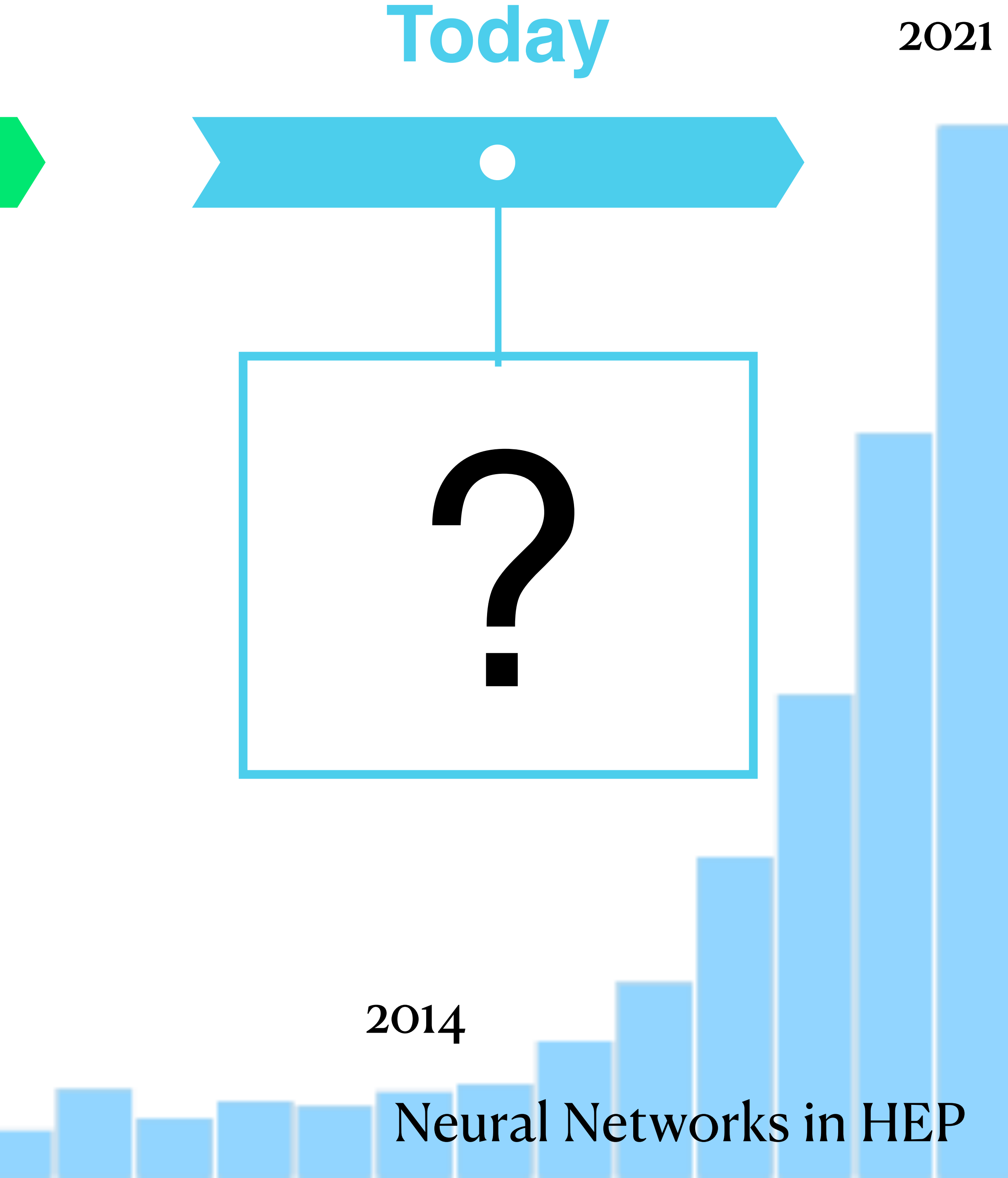
J. Thaler

Today

Pheno 2022

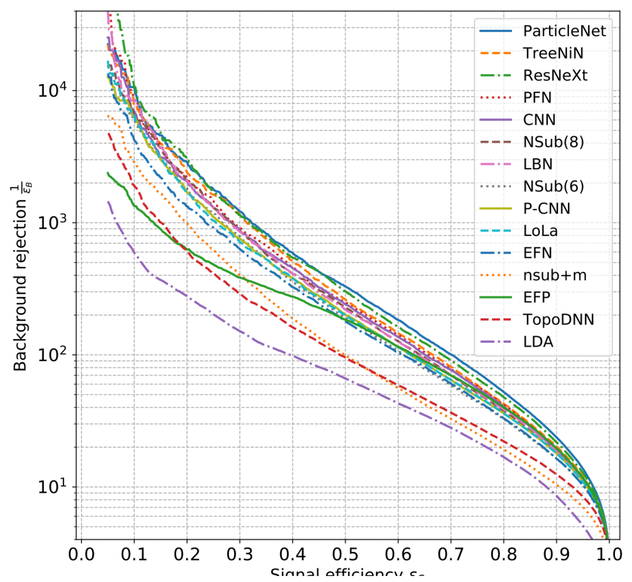
> 12 *official* ML talks

?



ML in particle physics 2022

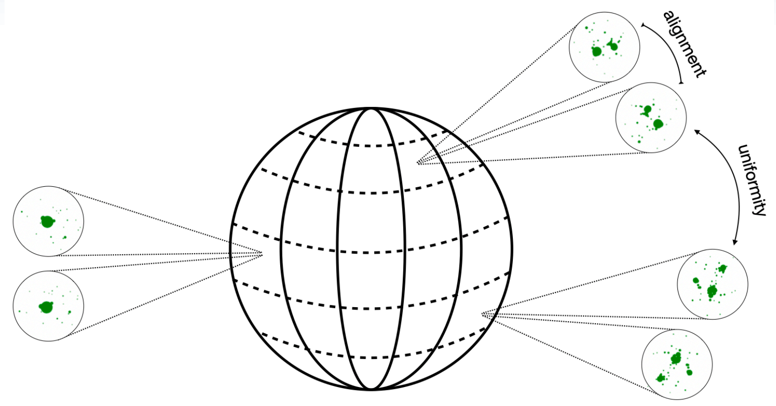
Top tagging



G. Kasieczka [1902.09914]

→ Talks by
D. Athanasakos & T. Cain

Anomaly detection



→Talks by
B. Dillon & A. Hallin

Classification

Generative models

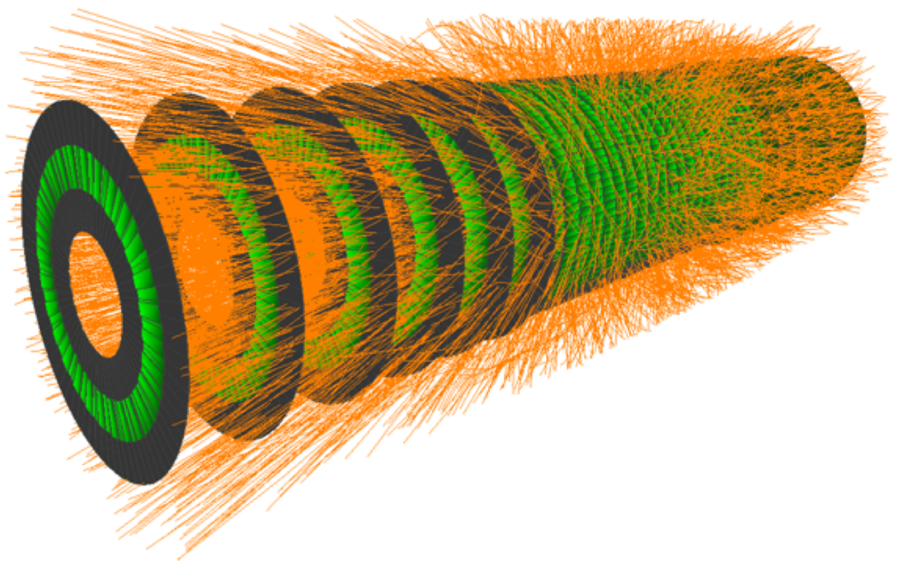
Graph networks

Bayesian networks

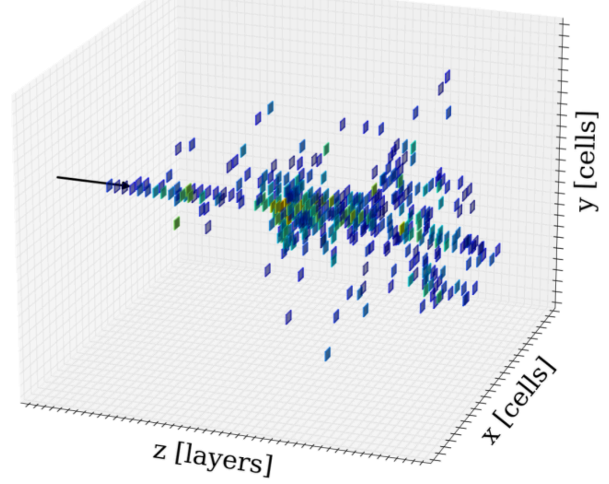
Regression

Track reconstruction

Kaggle challenge

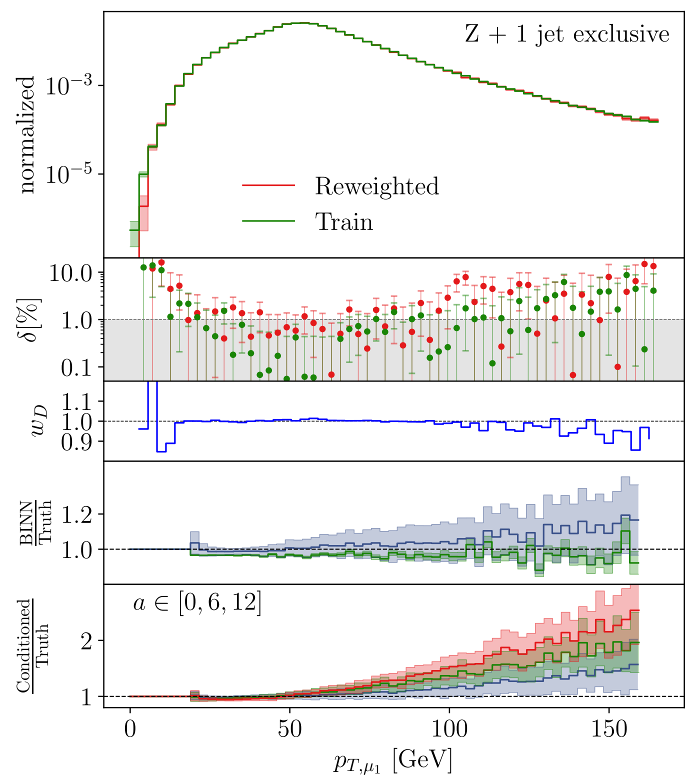


Detector simulation



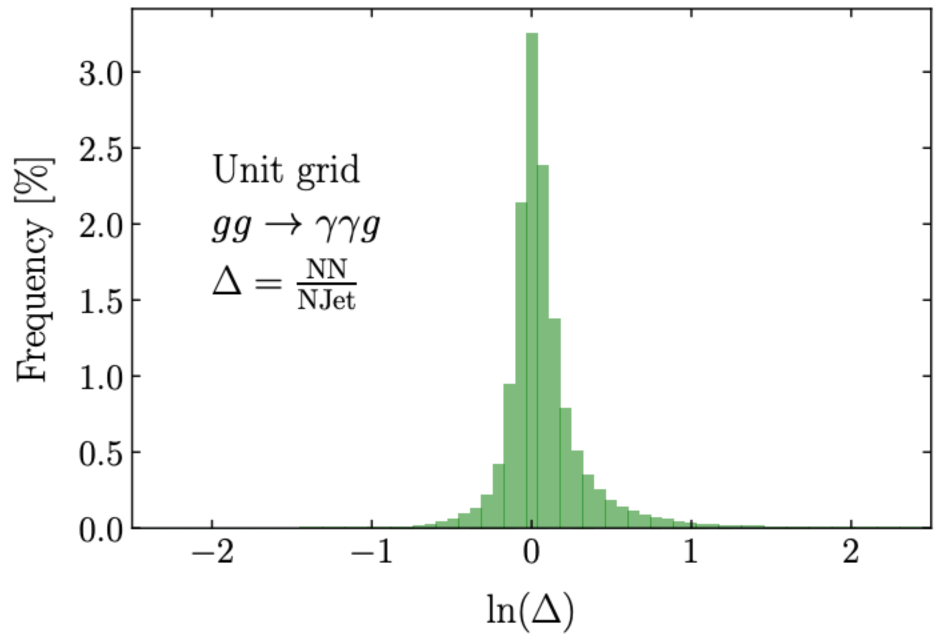
E. Buhmann et al. [2112.09709]

Event generation



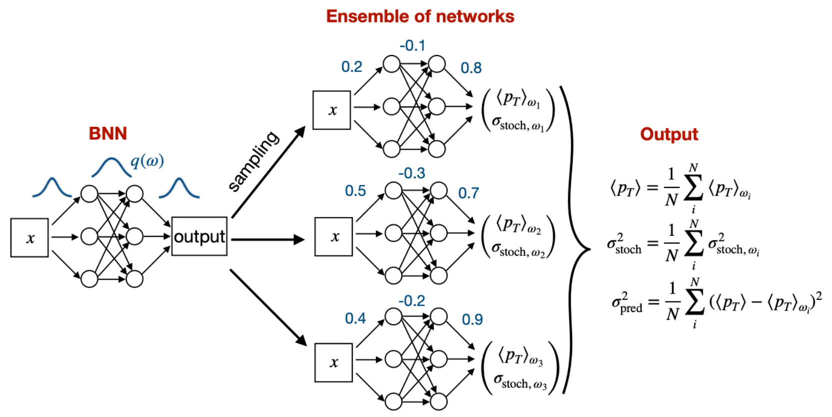
→Talk by T. Heimel

Amplitude estimation



J. Aylett-Bullock, et al. [2106.09474]

Jet calibration & uncertainties

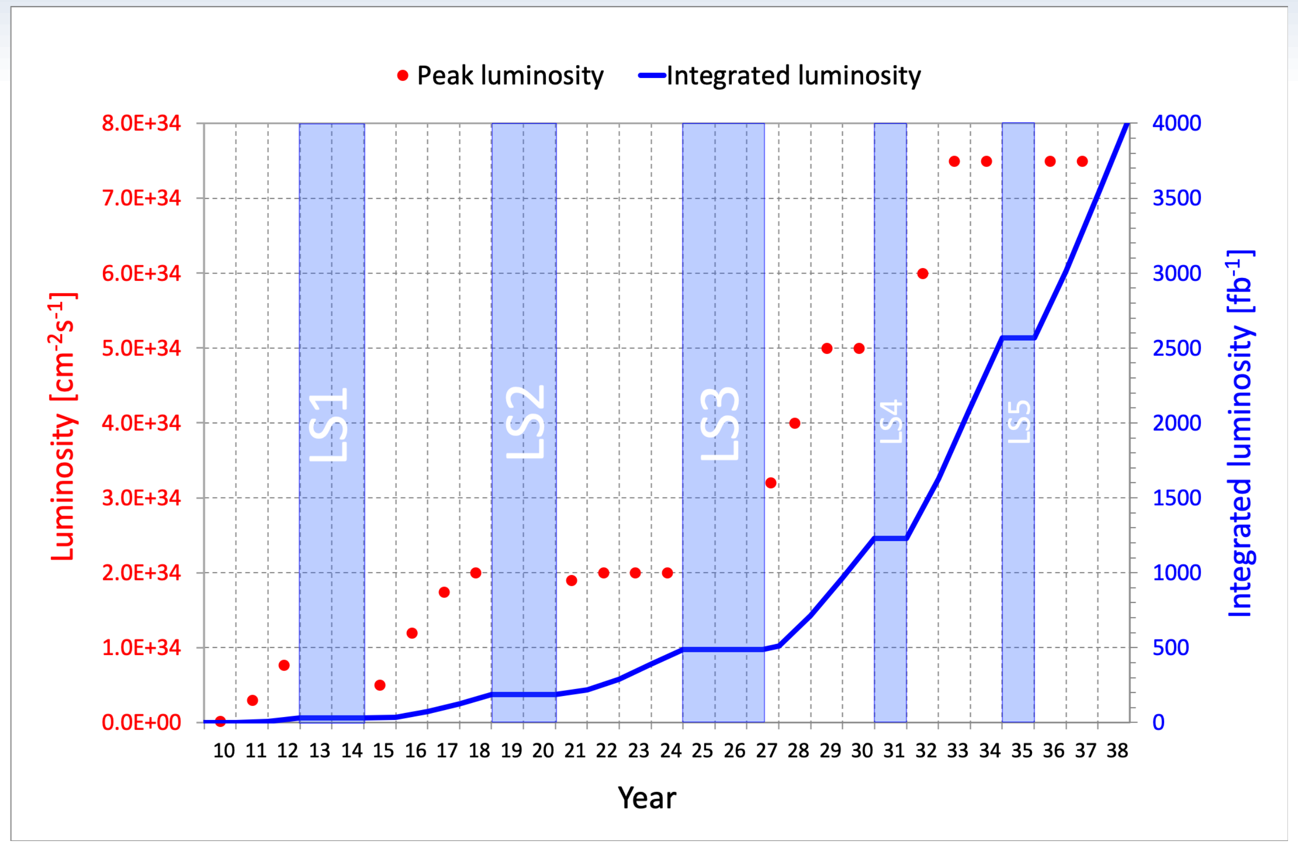


G. Kasieczka et al. [2003.11099]

Complete citations $\mathcal{O}(800)$
https://iml-wg.github.io/HEPML-LivingReview/

Open questions towards HL-LHC

A biased selection



- Facing **25 times** the amount of data
- What do we need to understand the data? (*read:* find new physics)

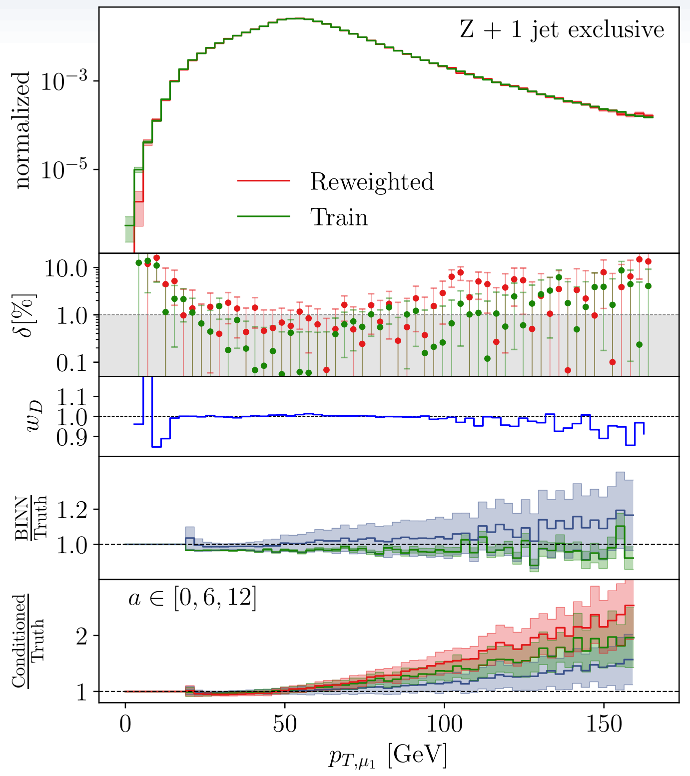
- **Precision predictions**
  - Higher order amplitudes
  - Event generation
  - Shower
  - Detector simulation

- **Optimized analysis for high-dimensional data**
  - Likelihood free inference [→ Talk by R. Barman]
    - Optimal Observables, Unfolding
  - Anomaly detection [→ Talks by B. Dillon & A. Hallin]
  - Uncertainty treatment ?

Problems beyond supervised classification/regression → How can machine learning help?

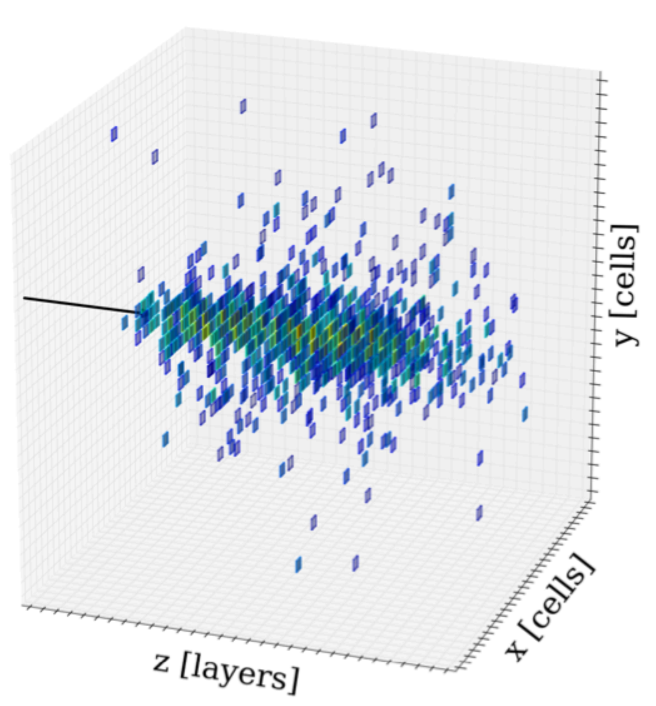
Forward simulations with generative networks

Event generation

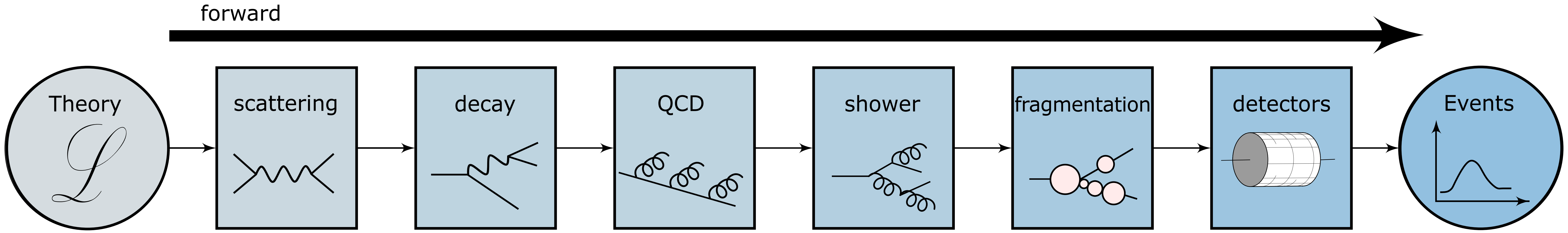


→ Otten et al.
→ Gao et al.
→ Bothmann et al.
→ Stienen et al.
→ AB, et al.
→ and many more

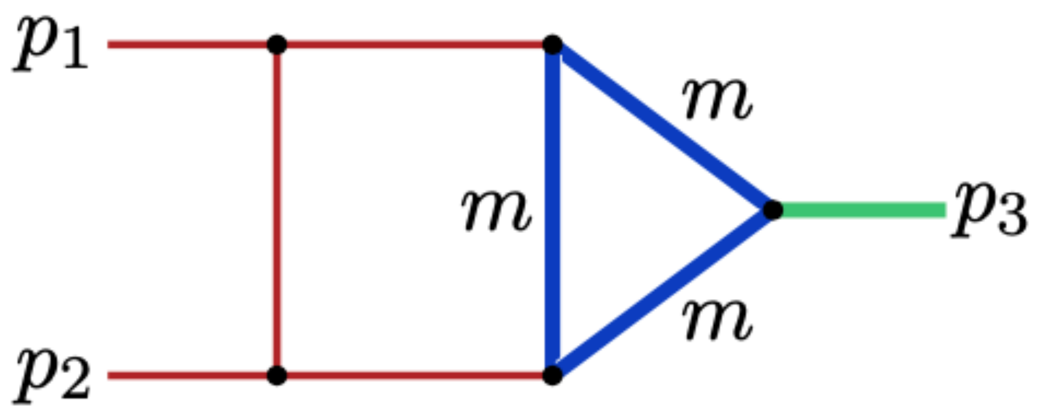
Detector simulation



→ CaloGAN by M. Paganini et al.
→ BIBAE by E. Buhman, S. Diefenbacher et al.
→ CaloFlow by C. Krause , D. Shih
→ and many more

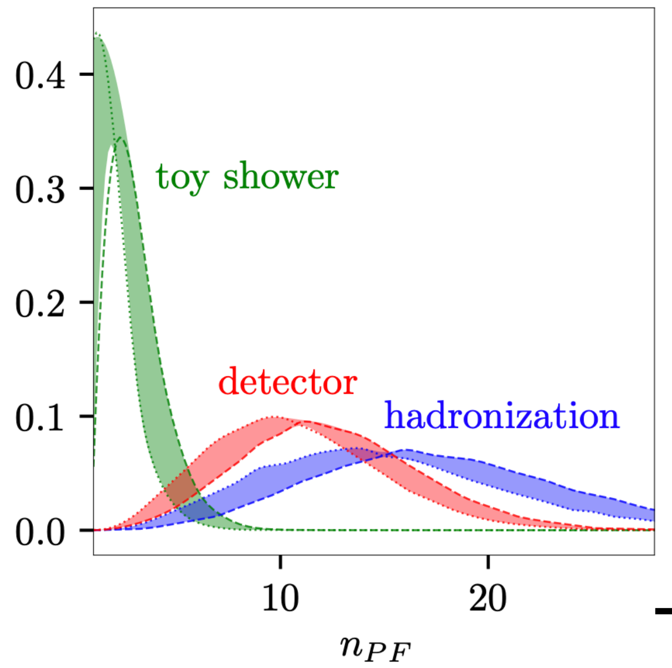


Loop amplitudes



→ R. Winterhalder, et al.

Shower simulation



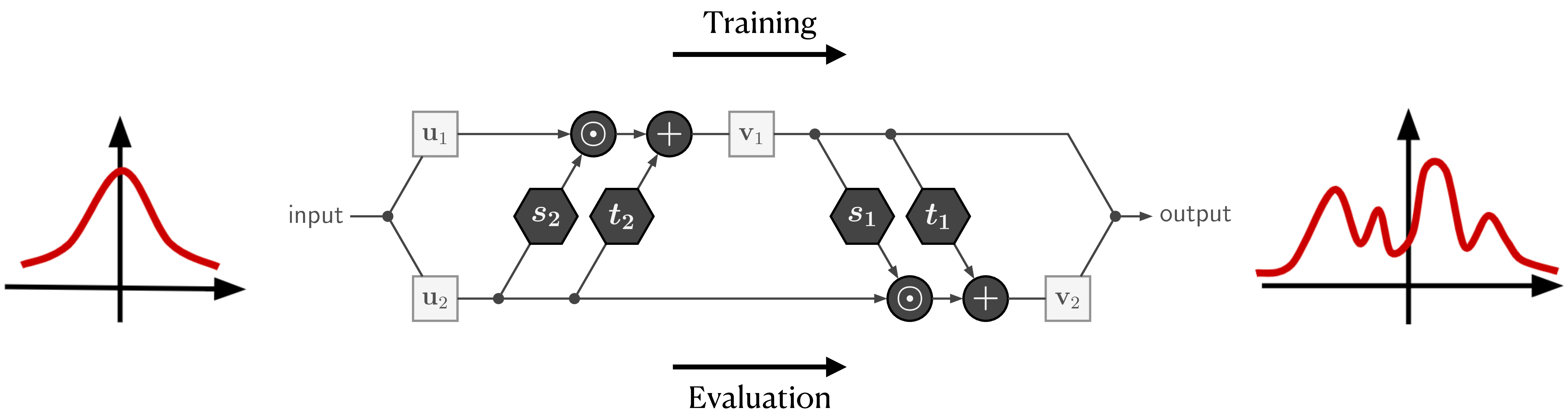
→ S. Bieringer, et al.

Particularly promising architecture
→ Normalizing flows

Normalizing flows

Invertible networks for complex transformations

+ Bijective mapping
+ Tractable Jacobian → $p_x(x) = p_z(z) \cdot J_{NN}$
+ Fast evaluation in both direction



Training on density $t(x)$
→ Minimize difference

$$\begin{aligned}\mathscr{L} &= \log p_x(x)/t(x)\\ &= \log p_z(z(x))\, J_{NN} \,/\, t(x)\end{aligned}$$

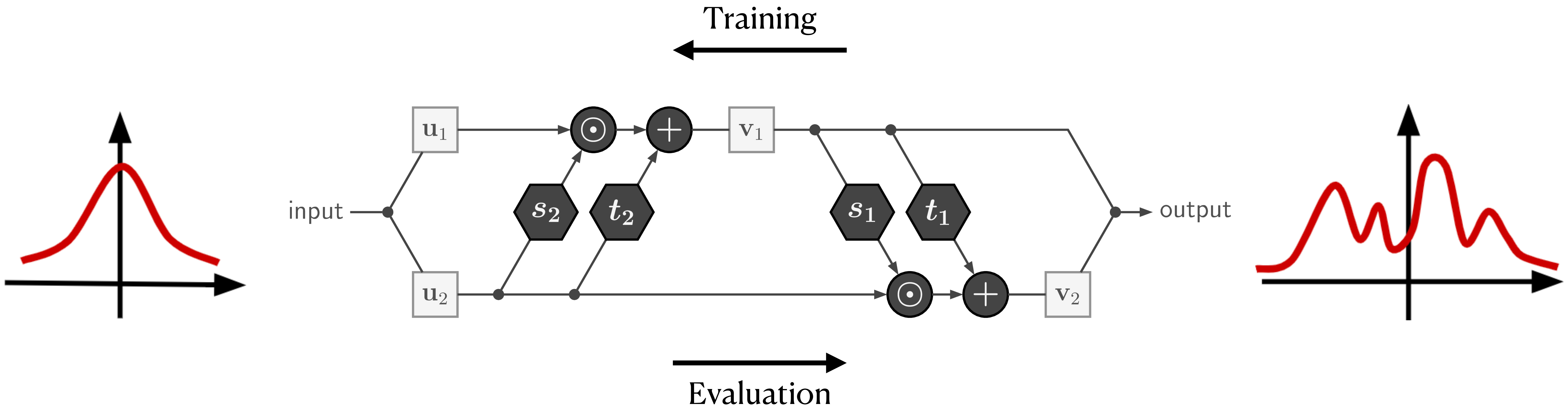
Training on samples x
→ Maximize the log-likelihood

$$\begin{aligned}\mathscr{L} &= \log p(\theta\,|\,x)\\ &= \log p(z\,|\,\theta) + \log J_{NN} + p(\theta)\end{aligned}$$

Normalizing flows

Invertible networks for complex transformations

+ Bijective mapping
+ Tractable Jacobian → $p_x(x) = p_z(z) \cdot J_{NN}$
+ Fast evaluation in both direction



Training on density $t(x)$

→ Minimize difference

$$\mathscr{L} = \log p_x(x)/t(x)$$
$$= \log p_z(z(x))\, J_{NN} \,/\, t(x)$$

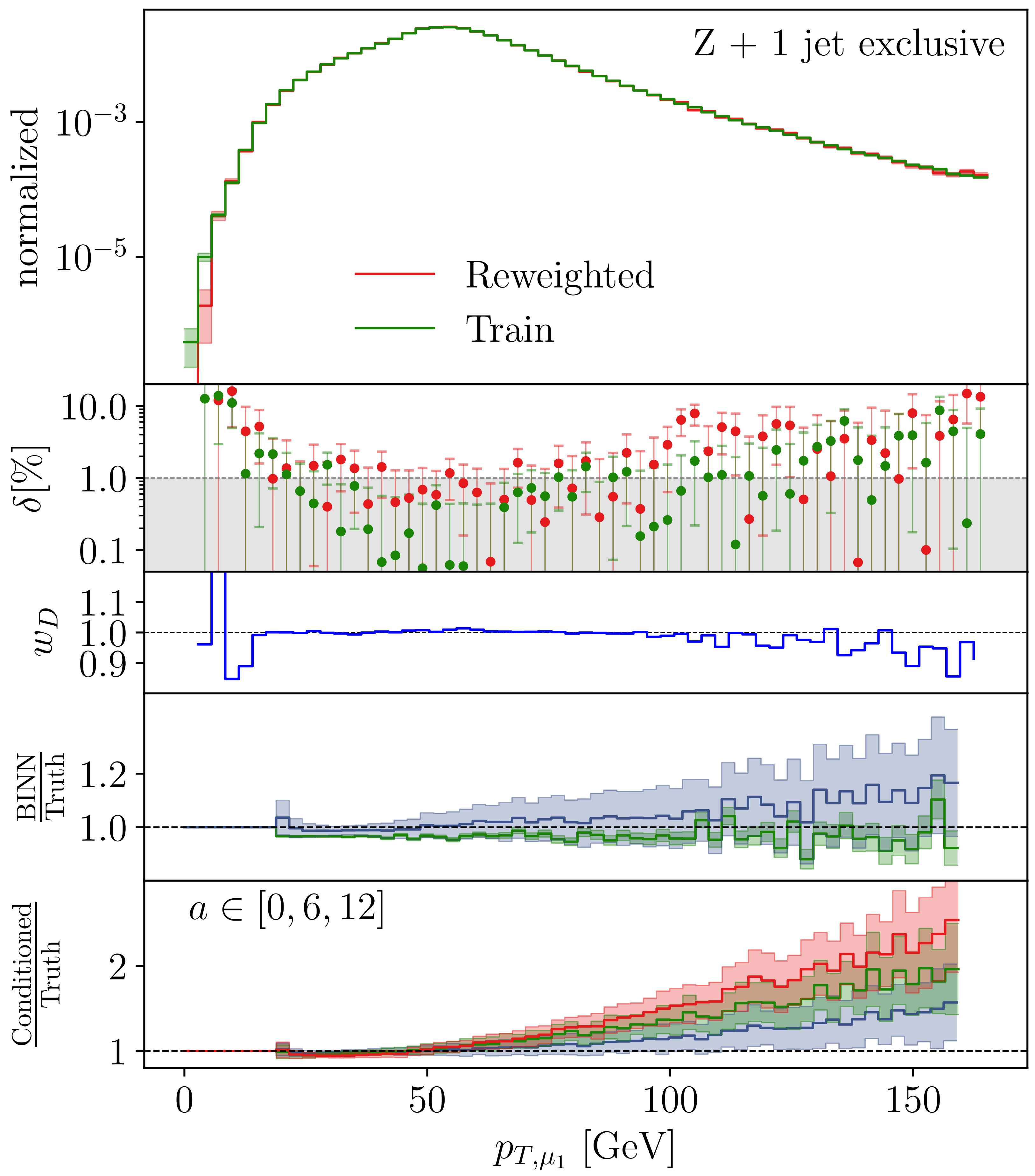
Training on samples x

→ Maximize the log-likelihood

$$\mathscr{L} = \log p(\theta\,|\,x)$$
$$= \log p(z\,|\,\theta) + \log J_{NN} + p(\theta)$$

Putting flows to work

Event generation



- Basis: INN
  - Phase space symmetries in architecture
- Control via classifier D
  - $$\frac{p_{\text{truth}}(x)}{p_{\text{INN}}(x)} = \frac{D(x)}{1 - D(x)}$$
- Precision via reweighting
  - Correct deviations of p_{INN}

➡ Uncertainty estimation via Bayesian NN

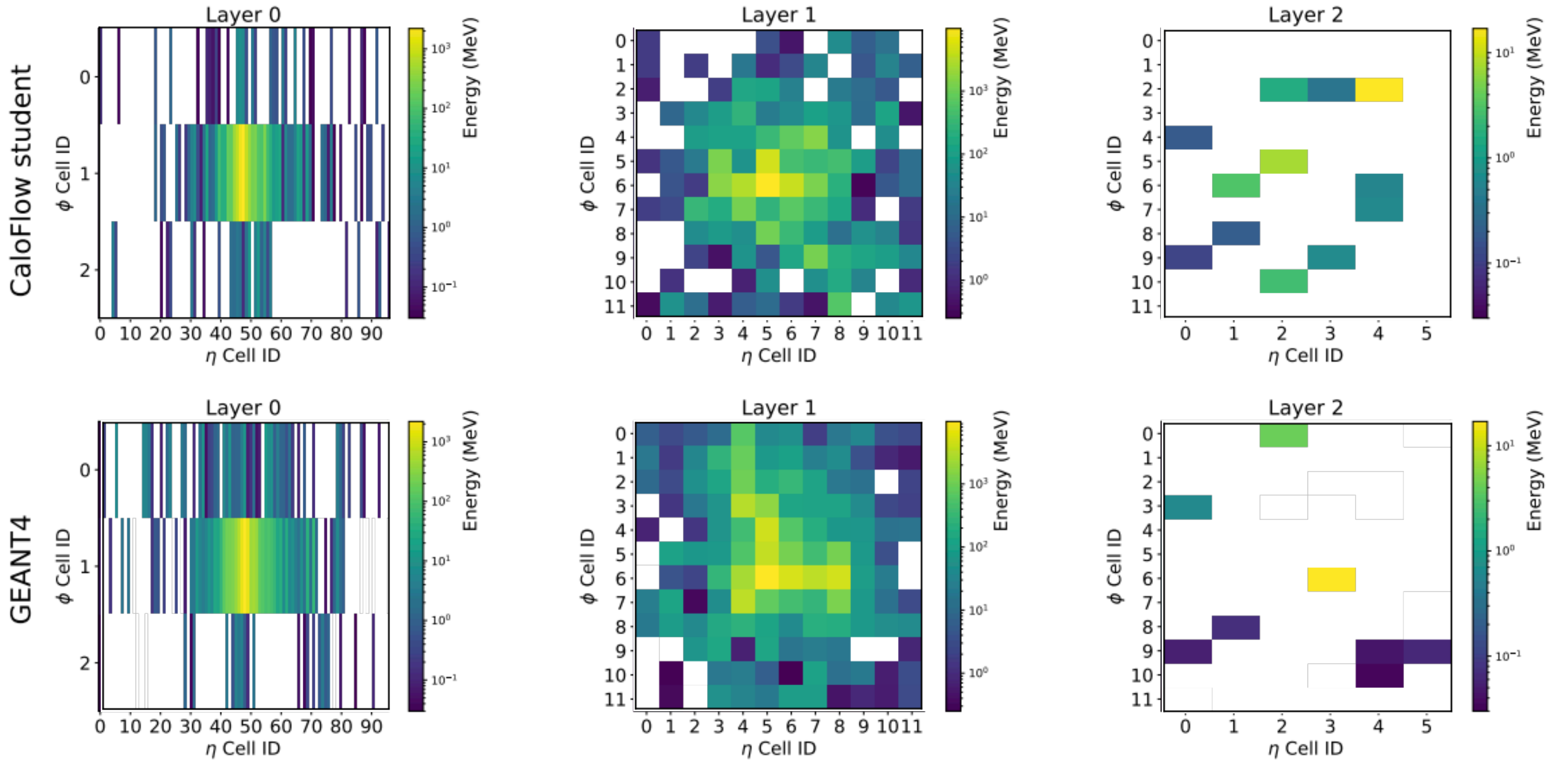
➡ Uncertainty propagation via conditioning

→ Details in talk by T. Heimel

Putting flows to work

Detector simulation

Challenge: large dimensionality $(3 \times 96, 12 \times 12, 12 \times 6)$



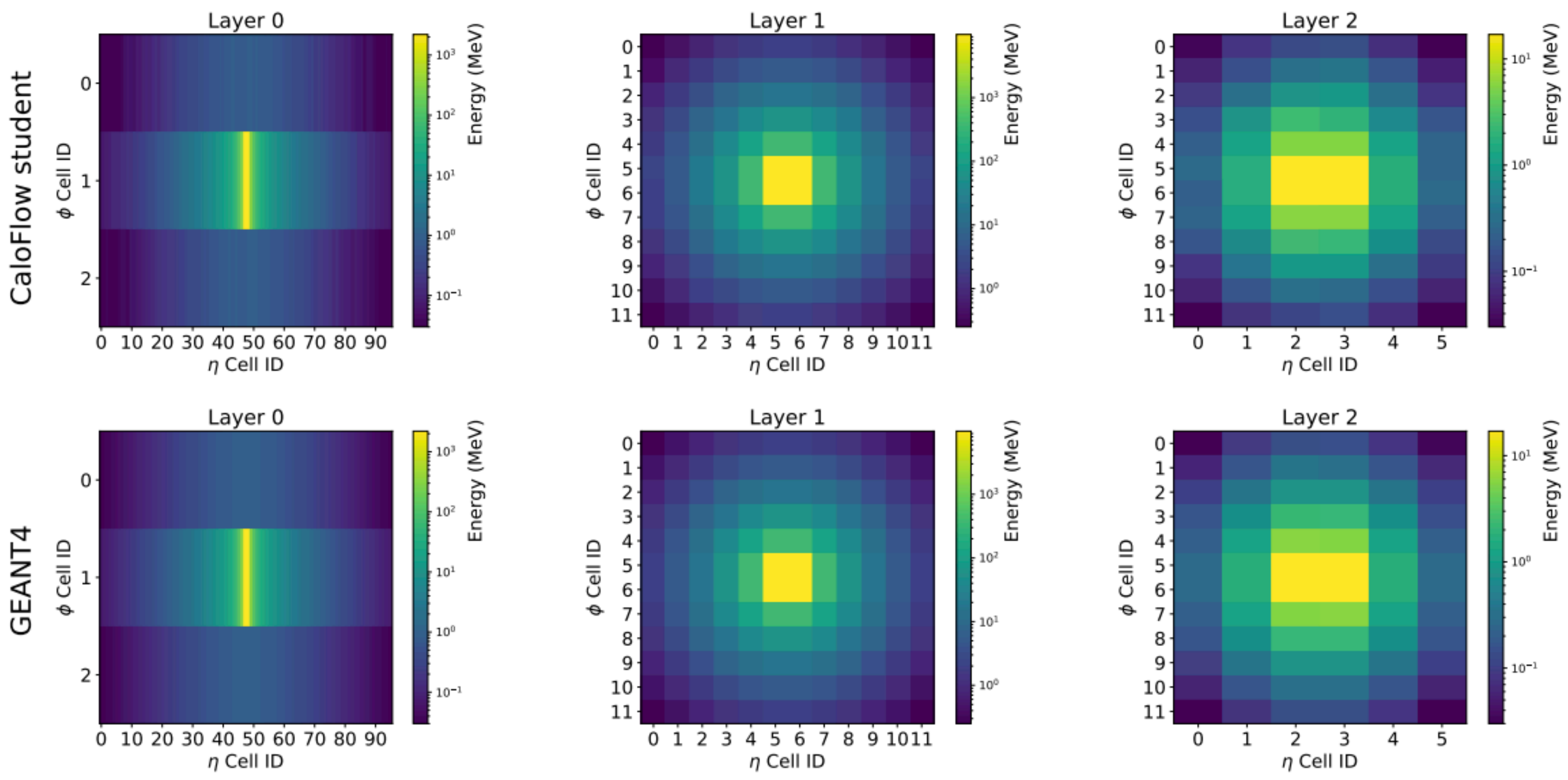
C. Krause & D. Shih [2110.11377]

π^+ shower individual & average

Putting flows to work

Detector simulation

Challenge: large dimensionality $(3 \times 96, 12 \times 12, 12 \times 6)$



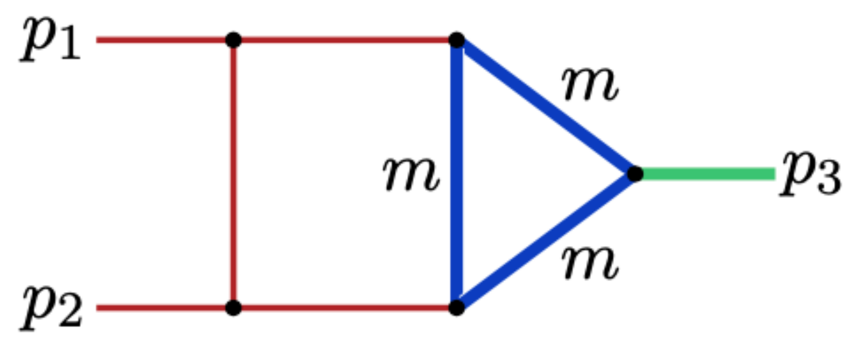
C. Krause & D. Shih [2110.11377]

π^+ shower individual & average

Multi-loop calculations with INNs

Profiting from the Jacobian

Precision predictions based on loop diagrams



Analytic expression for loop amplitude

$$G = \int_{-\infty}^{\infty} \left(\prod_{l=1}^{L} \frac{\mathrm{d}^D k_l}{i\pi^{\frac{D}{2}}} \right) \prod_{j=1}^{N} \frac{1}{(q_j^2 - m_j^2 + i\delta)^{\nu_j}}$$

$$= \int_0^1 \prod_{j=1}^{N-1} \mathrm{d}x_j \, x_j^{\nu_j - 1} \frac{U^{\nu-(L+1)D/2}}{F^{\nu-LD/2}} = \int_0^1 \prod_{j=1}^{N-1} \mathrm{d}x_j \; I(\vec{x})$$

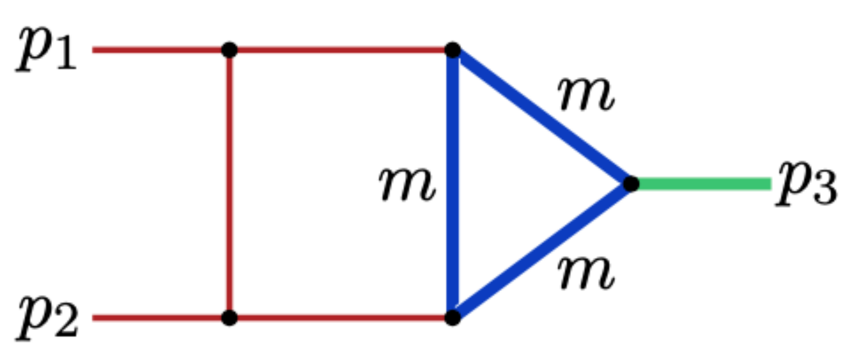
Rewrite with
Feynman parameters

Still contains singularities

Multi-loop calculations with INNs

Profiting from the Jacobian

Precision predictions based on loop diagrams



Analytic expression for loop amplitude

$$G = \int_{-\infty}^{\infty} \left(\prod_{l=1}^{L} \frac{\mathrm{d}^D k_l}{i\pi^{\frac{D}{2}}} \right) \prod_{j=1}^{N} \frac{1}{(q_j^2 - m_j^2 + i\delta)^{\nu_j}}$$

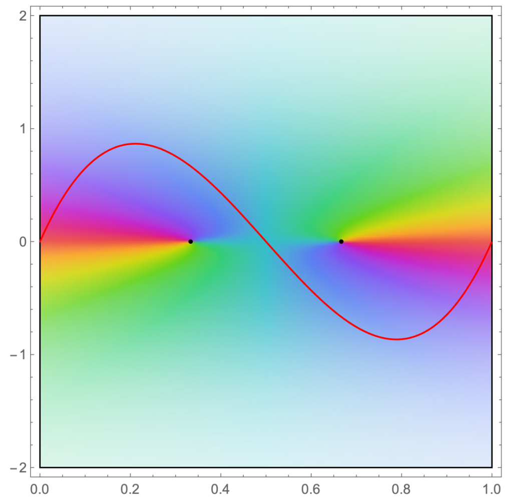
$$= \int_0^1 \prod_{j=1}^{N-1} \mathrm{d}x_j\, x_j^{\nu_j - 1} \frac{U^{\nu-(L+1)D/2}}{F^{\nu - LD/2}} = \int_0^1 \prod_{j=1}^{N-1} \mathrm{d}x_j\, I(\vec{x})$$

Rewrite with Feynman parameters

Still contains singularities

Solved by contour deformation due to Cauchy's theorem

$$\int_0^1 \prod_{j=1}^{N} \mathrm{d}x_j\, I(\vec{x}) = \int_0^1 \prod_{j=1}^{N} \mathrm{d}x_j\, \det\left(\frac{\partial \vec{z}(\vec{x})}{\partial \vec{x}}\right) I(\vec{z}(\vec{x}))$$

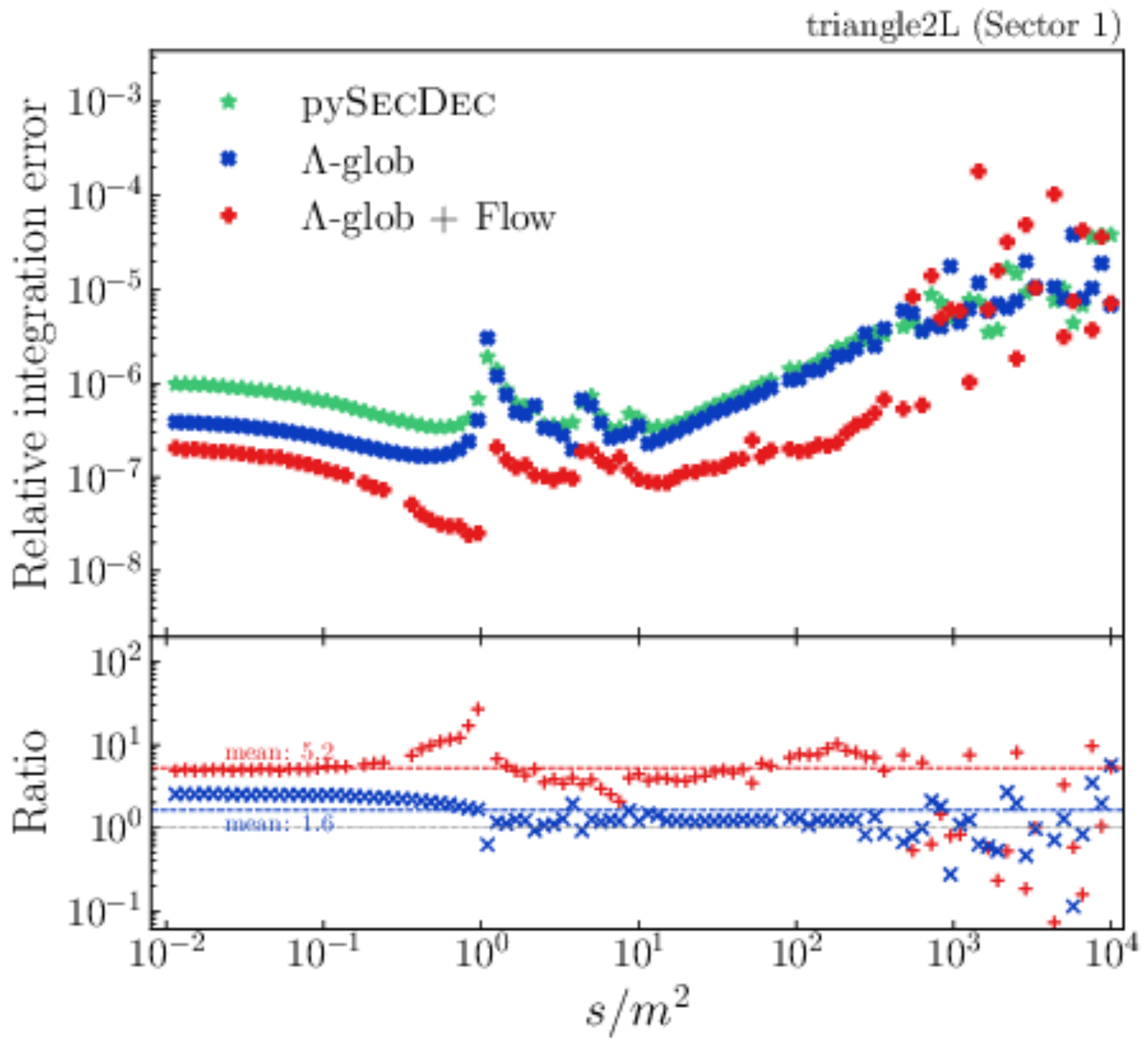


Optimal parametrization = minimal variance

Turn it into an ML Problem

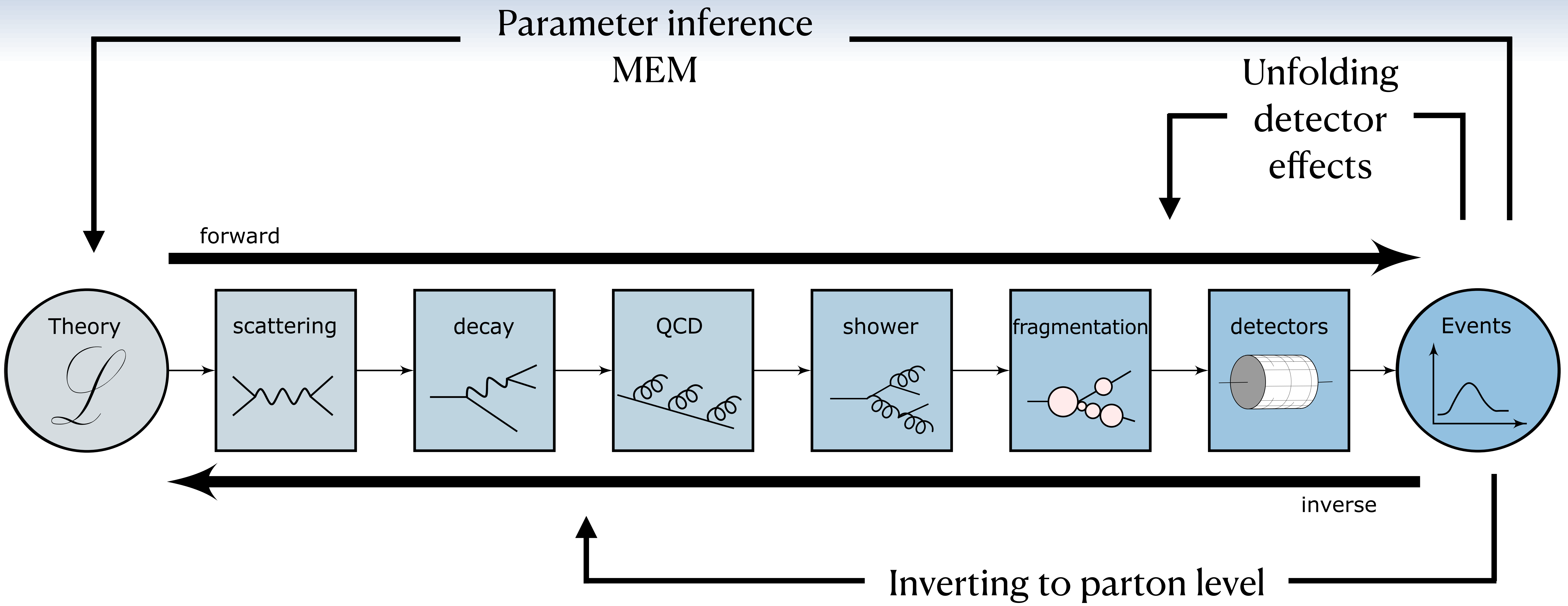
Parametrization $\rightarrow z = \mathrm{INN}(x)$

Variance $\rightarrow \mathscr{L}$



R. Winterhalder, et al. [2112.09145]

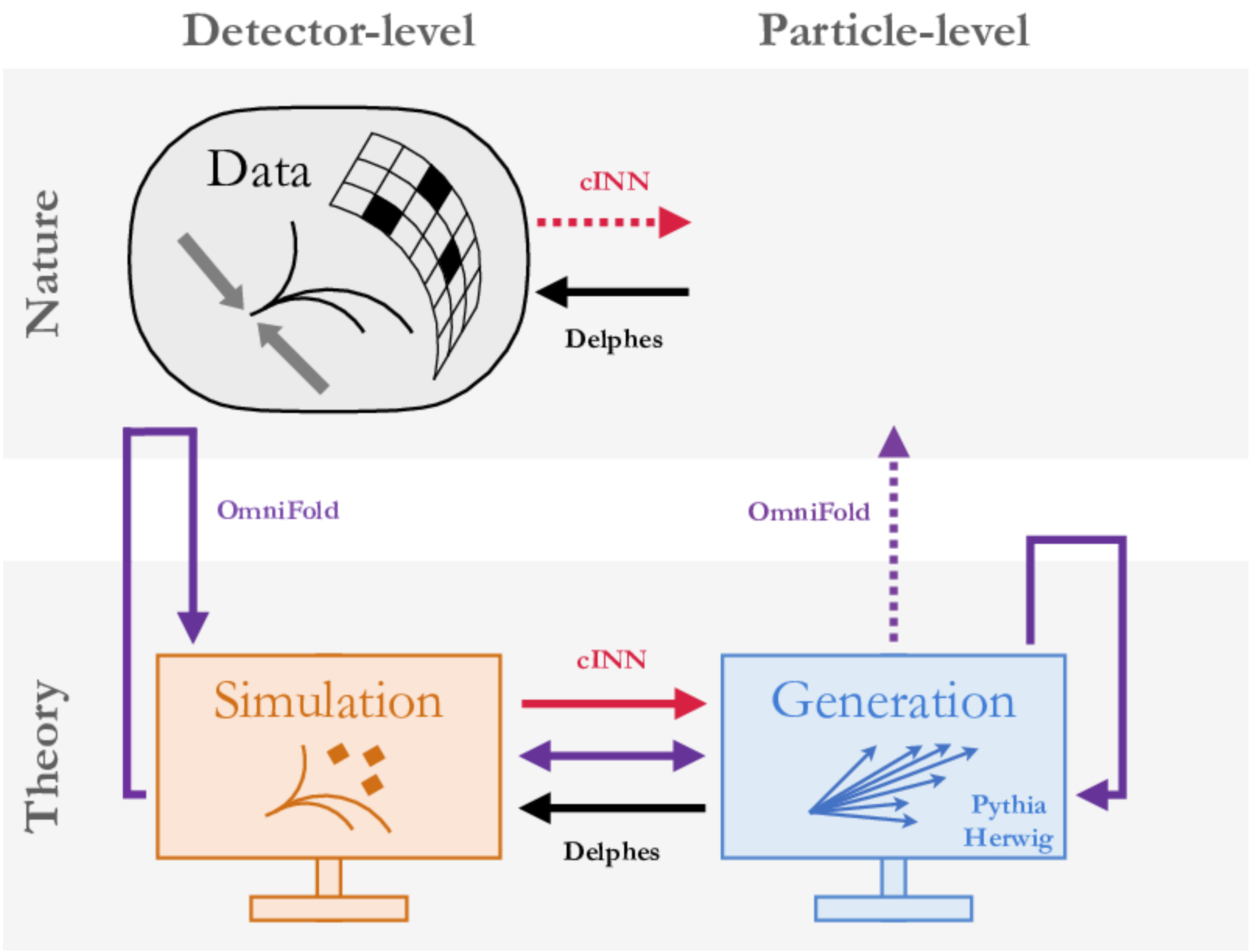
Inverting the simulation chain



Requirements

- ☐ Highdimensional
- ☐ Bin independent
- ☐ Statistically well defined

ML unfolding methods



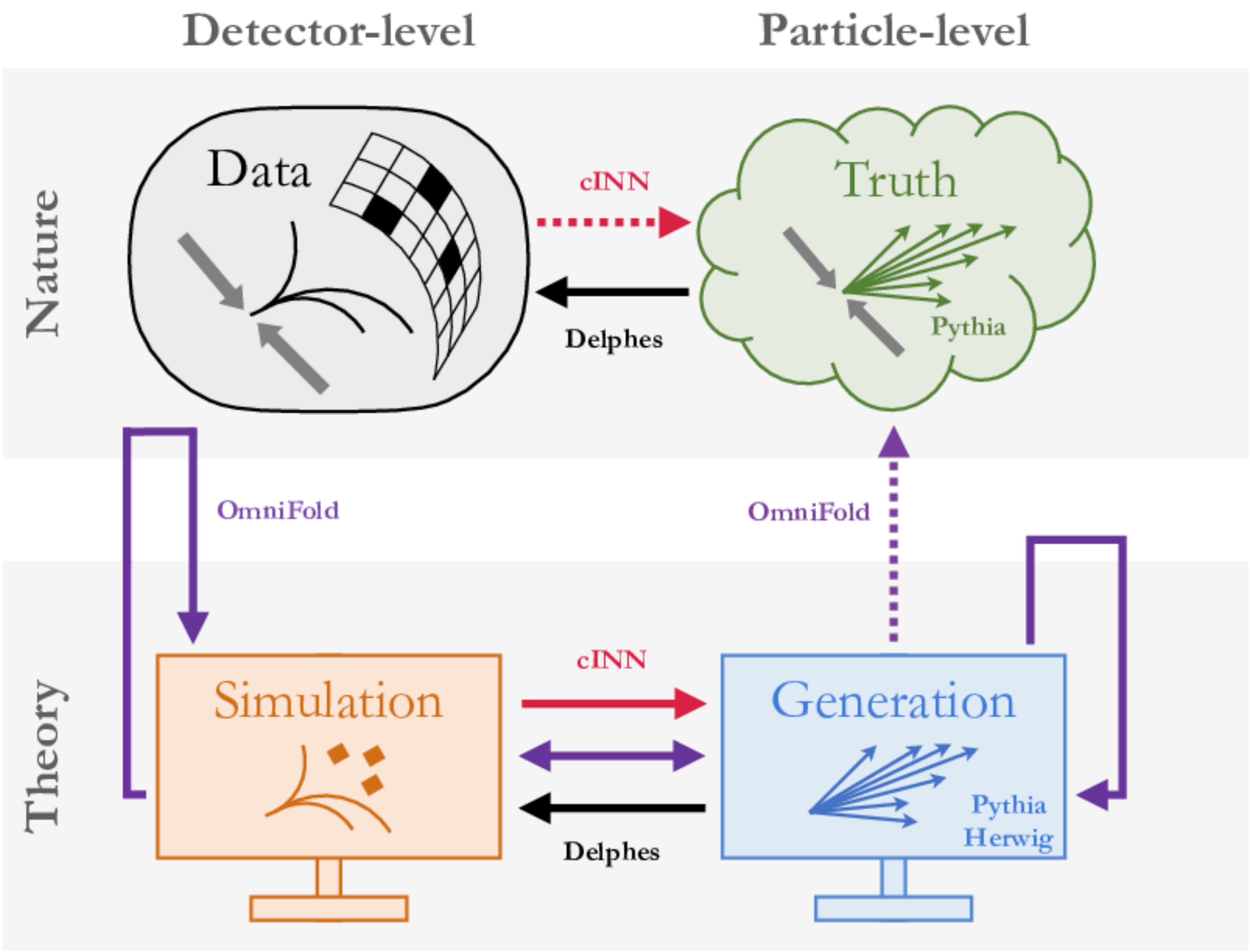
M. Arratia et al. [2109.13243]

Classifier based aproach
Output: reweighted distribution of MC events

Density based approach
Output: probability density per unfolded event

VAE alternative: OTUS by J. N. Howard et al.

ML unfolding methods



M. Arratia et al. [2109.13243]

Classifier based aproach

Output: reweighted distribution of MC events

Density based approach

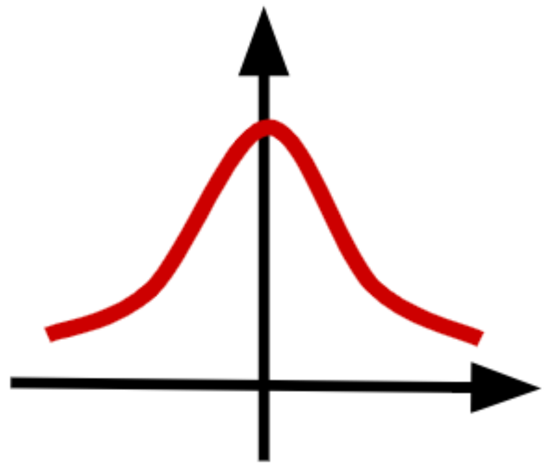
Output: probability density per unfolded event

VAE alternative: OTUS by J. N. Howard et al.

cINN unfolding

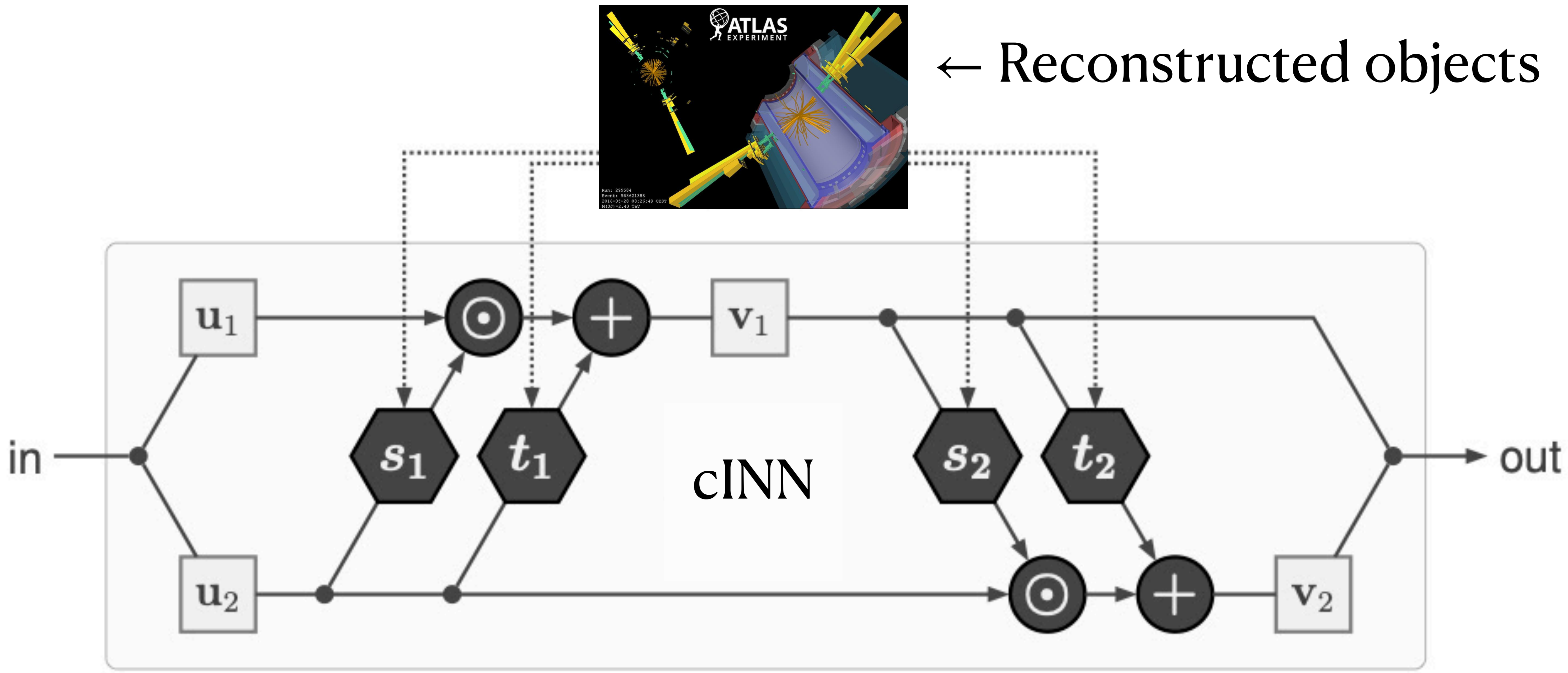
Given a reconstructed event:
What is the probability distribution at particle level?

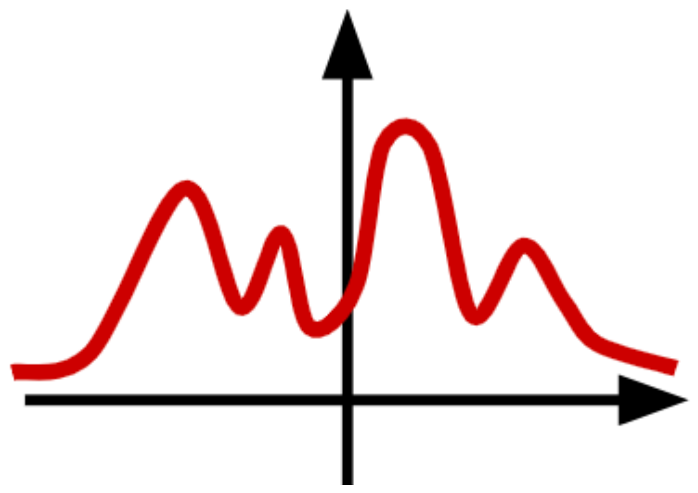
← Reconstructed objects



Normal distribution

$z \sim \mathcal{N}$





Probability distribution

$x \sim P_{\text{parton}}$

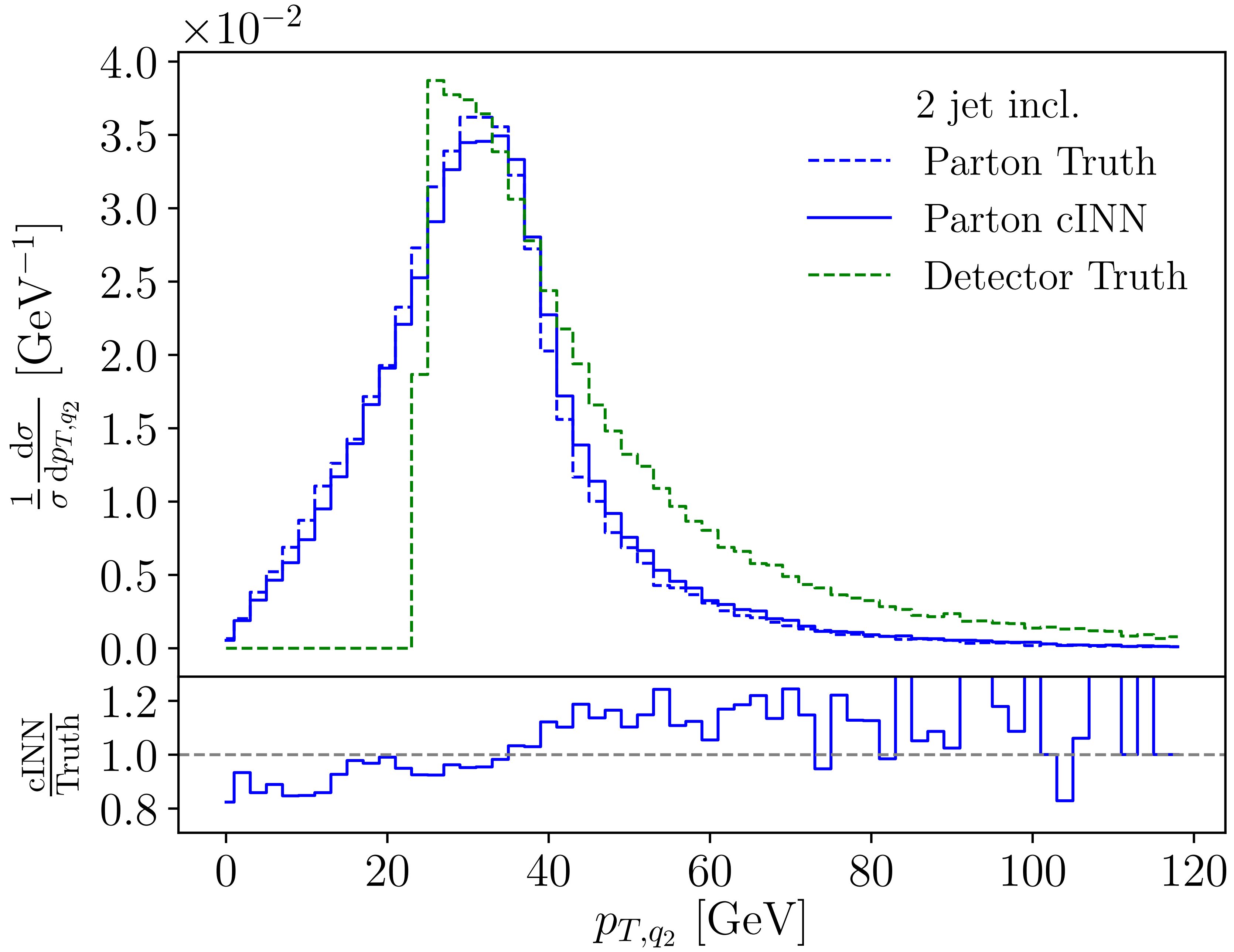
Training

Unfolding

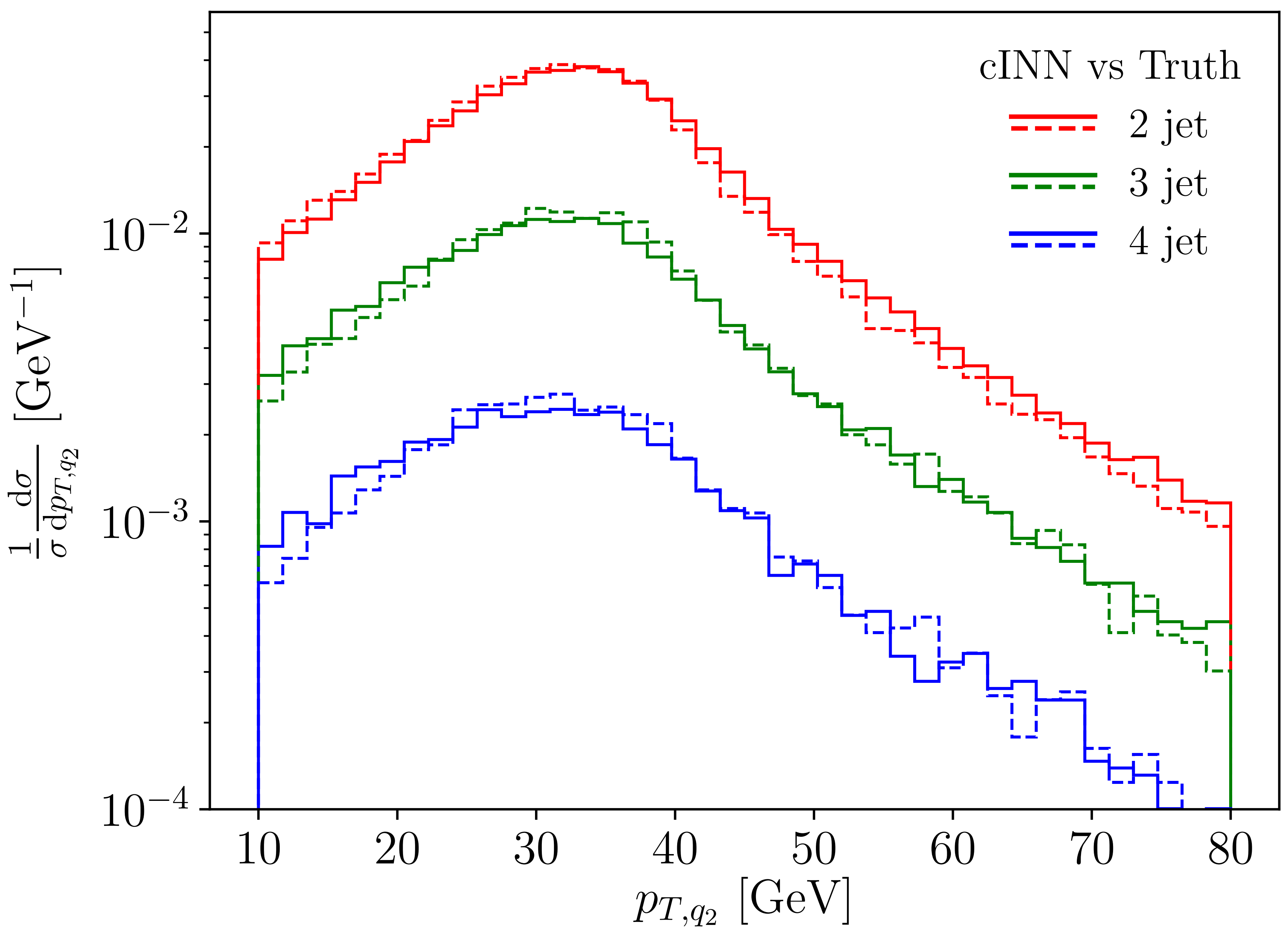
Inverting inclusive distributions

$pp > WZ > q\bar{q}l^+l^- +$ ISR → 2/3/4 jet events

Training on inclusive dataset



Evaluate exclusive 2/3/4 jet events

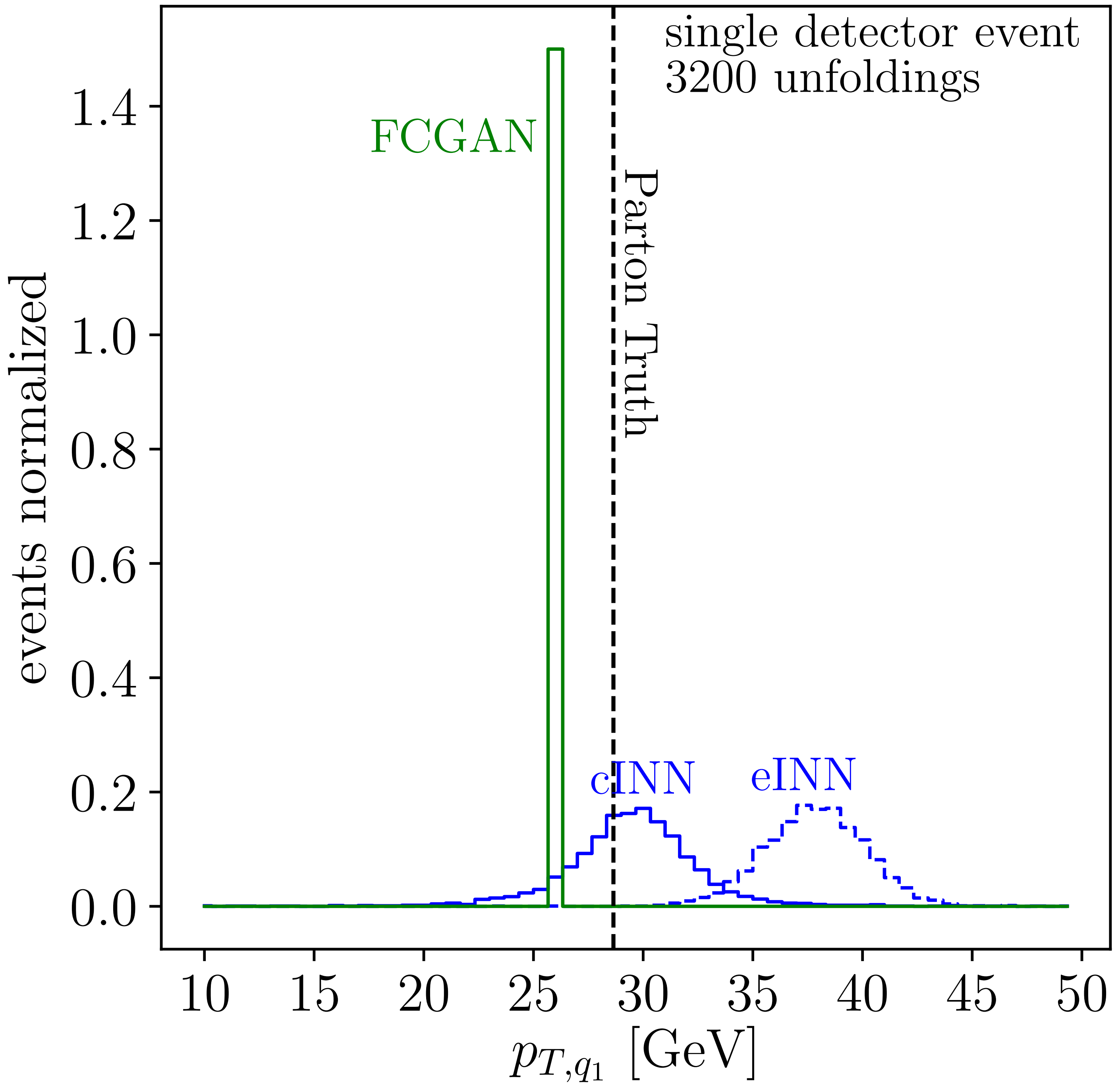


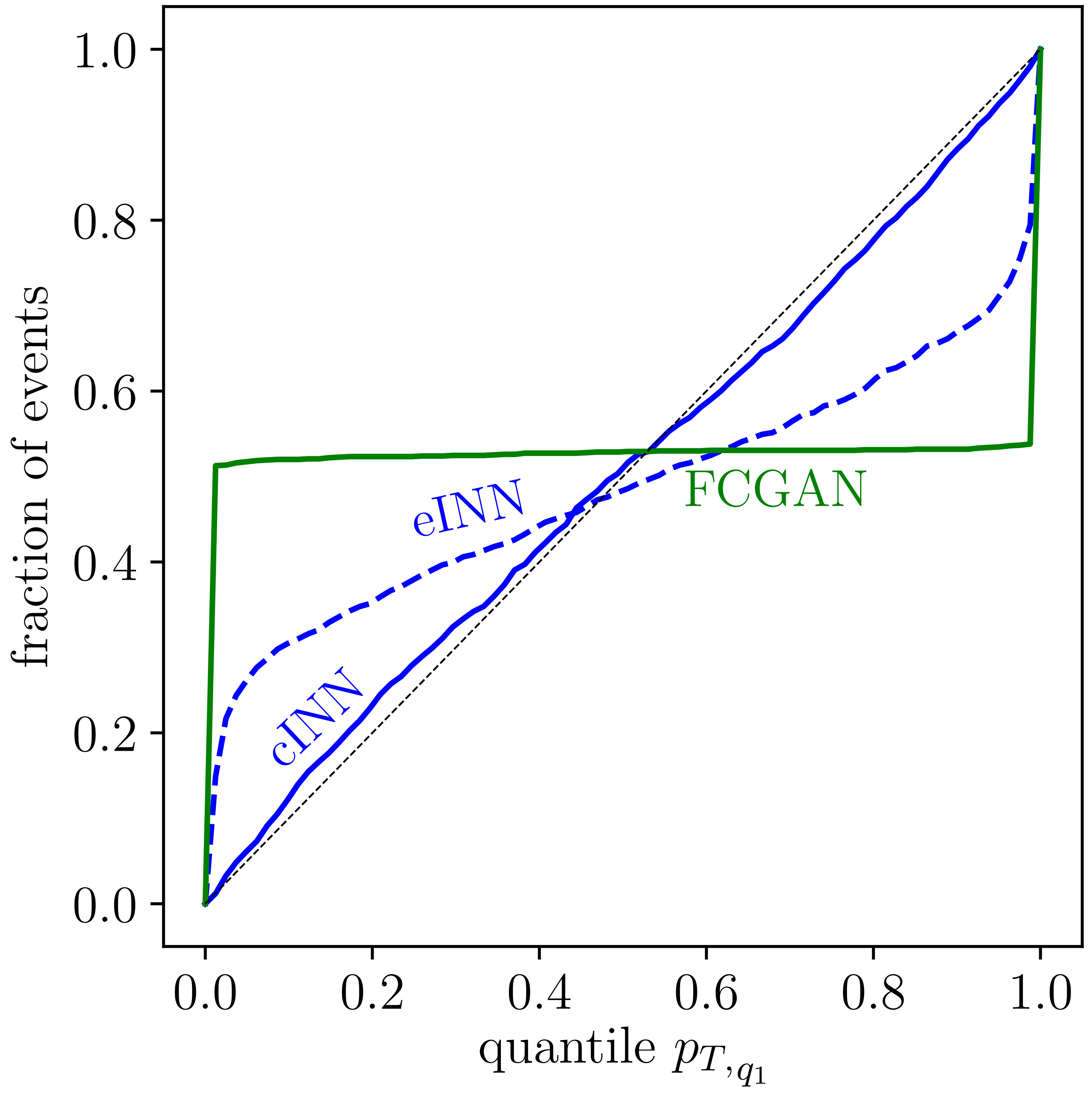
M. Bellagente et al. [2006.06685]

- [x] High-dimensional
- [x] Bin-independent
- [ ] Statistically well defined ?

Event-wise unfolding

No deterministic mapping!
Check calibration of probability density for individual event unfolding





M. Bellagente et al. [2006.06685]

- ☑ High-dimensional
- ☑ Bin-independent
- ☑ Statistically well defined

ML Uncertainties

When do we (not) need them?

- Analyses with poorly trained NNs are sub-***optimal*** but not ***wrong***

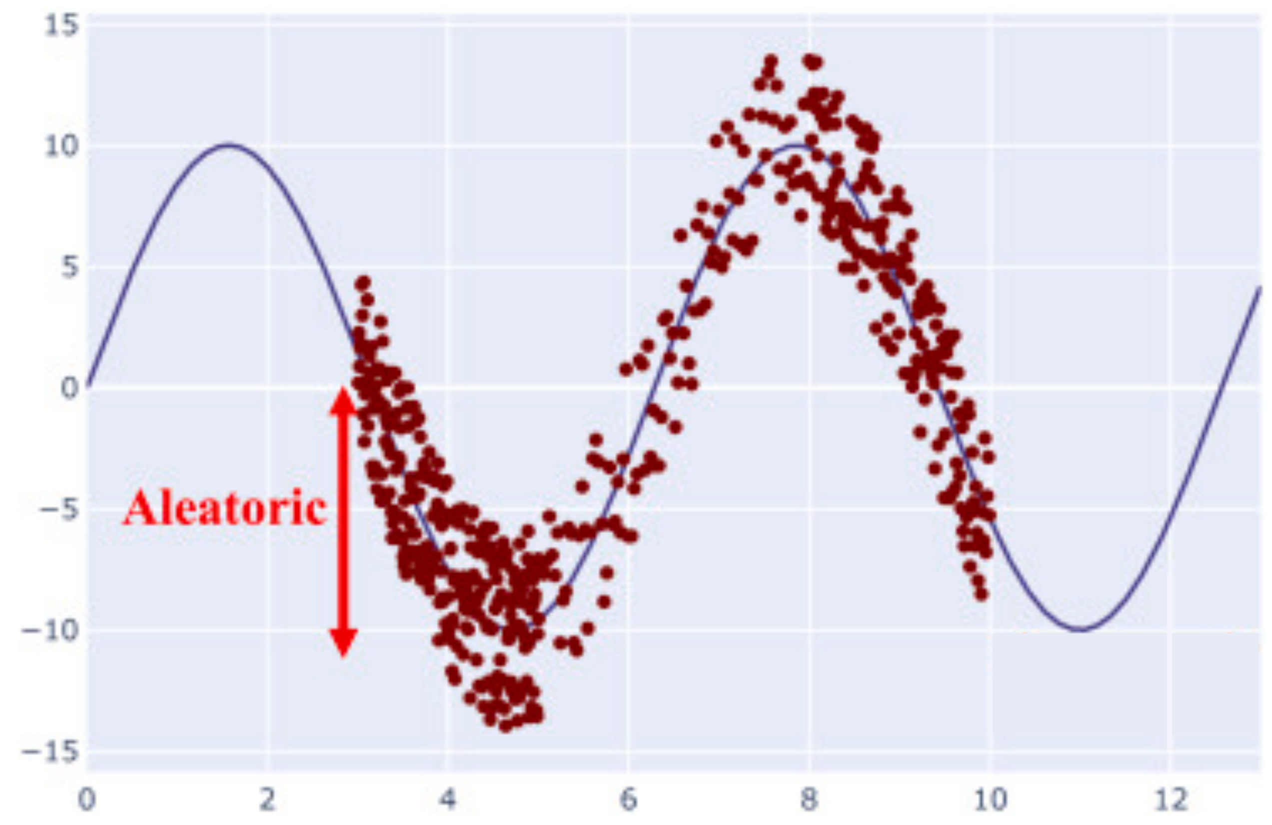
- Example 1: Enhance **Signal vs Background** with NN
  - Use NN output as observable
  - Poor NN yields **low S vs B**
  - Does not prevent correct statistical analysis

- Example 2: INN for **integration**
  - Sub-optimal contour deformation
  - **High variance** of integral
  - Not efficient but not wrong

➡ **Control** comes from **simulation** !

More details eg. B. Nachman [1909.03081]

- Example 3: Understanding a calibration output
  - Regression problem with uncertainties

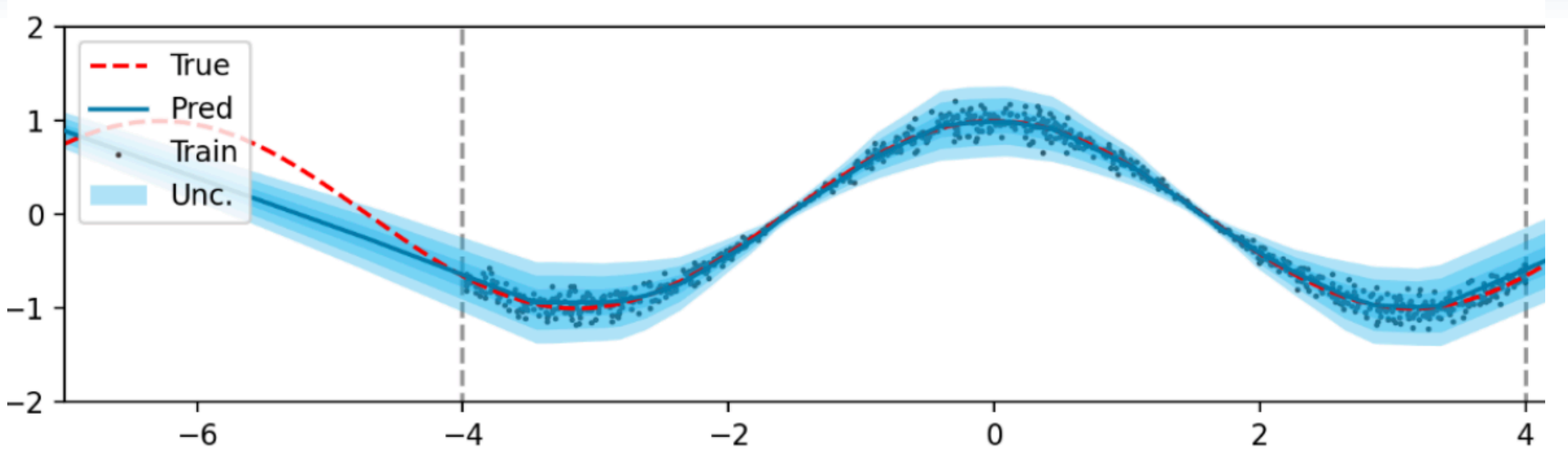


modified from M. Abdar [doi.org/10.1016/j.inffus.2021.05.008]

➡ How can we estimate this uncertainty?

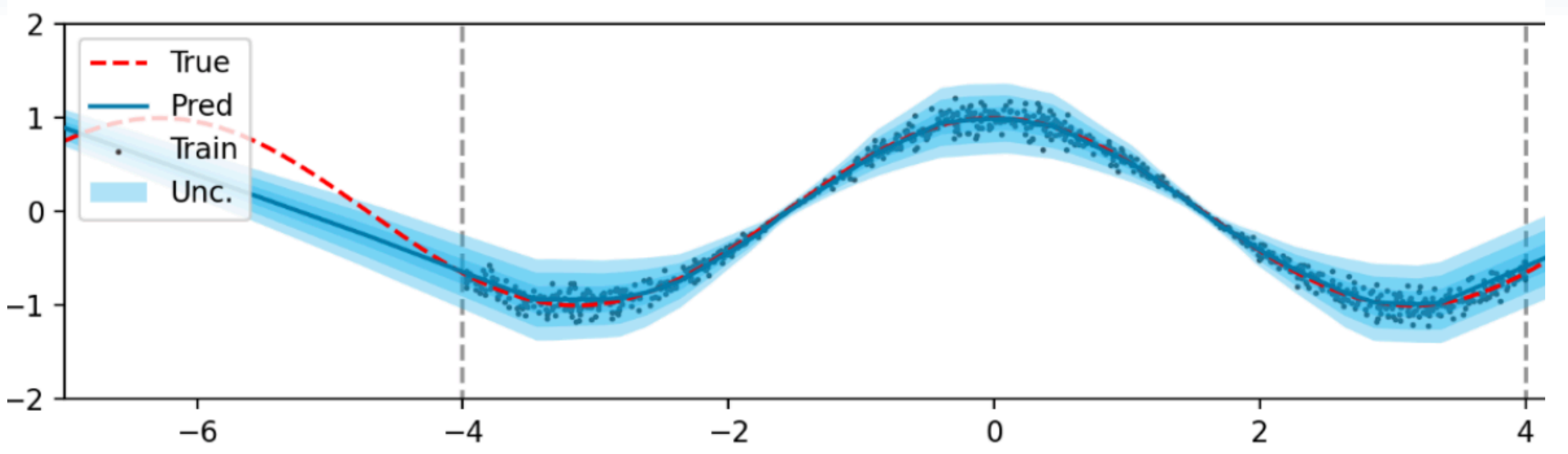
Estimating uncertainties in ML

1. **Extend** standard network **output** to include uncertainty $\rightarrow (\mu(x), \sigma(x))$
    - Gaussian approximation
    - $\mathcal{L}_{\text{Gauss}} = -\log(\sqrt{2\pi}\sigma(x)) - \frac{1}{2}\frac{(\mu(x)-y)^2}{\sigma(x)^2}$
    - Captures only $\mathbf{p}(\mathbf{y}\,|\,\mathbf{x}, \mathbf{w})$ for fixed network weights
    - w varies for different trainings!



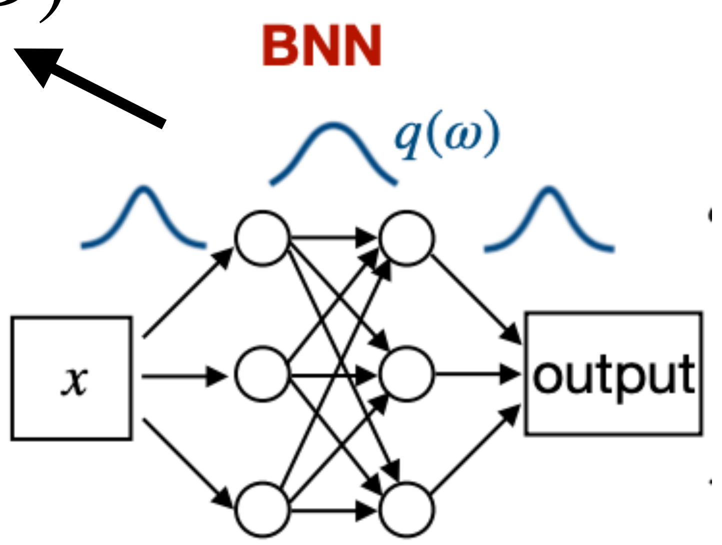
Estimating uncertainties in ML

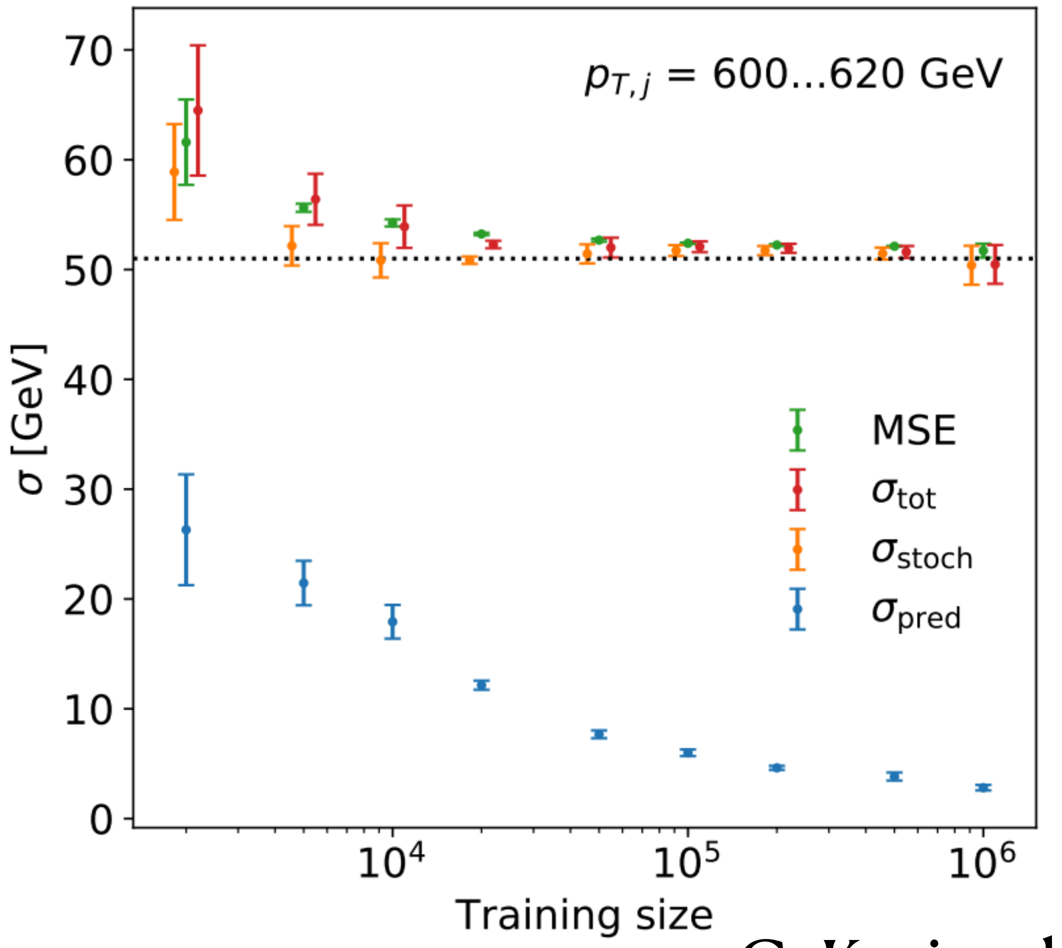
1. **Extend** standard network **output** to include uncertainty $\rightarrow (\mu(x), \sigma(x))$
    - Gaussian approximation
    - $\mathscr{L}_{\text{Gauss}} = -\log(\sqrt{2\pi}\sigma(x)) - \frac{1}{2}\frac{(\mu(x)-y)^2}{\sigma(x)^2}$
    - Captures only $\mathbf{p}(\mathbf{y}\,|\,\mathbf{x}, \mathbf{w})$ for fixed network weights
    - w varies for different trainings!



2. Estimating $\mathbf{p}(\mathbf{y}\,|\,\mathbf{x}, \mathbf{D})$ with training dataset D
    - $p(y\,|\,x, D) = \int \mathrm{d}w\; p(y\,|\,x, w)\; p(w\,|\,D)$

$\mathscr{L}_{\text{Gauss}}$





Jet calibration

$\rightarrow \sigma(x)$ captures intrinsic uncertainty

For large dataset:

$\rightarrow p(w\,|\,D)$ approaches δ-function

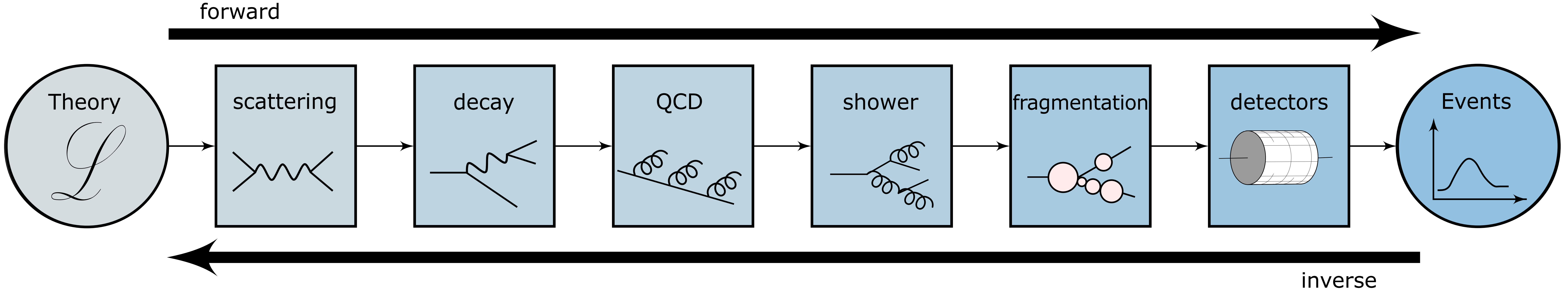
G. Kasieczka et al. [2003.11099]

3. **Alternative approaches**: Ensembling, Normalizing flows (calibration curves), ...

Summary

What ML can do for you

Better predictions with ML based precision simulations



Optimal inference with controllable networks

New data are coming sooner than you think...