UNIVERSITÄT HEIDELBERG
ZUKUNFT SEIT 1386

IRN Terascale, 26.10.2023

Diffusion Models

for LHC event generation

SPONSORED BY THE
Federal Ministry of Education and Research

Anja Butter, Nathan Huetsch, Sofia Palacios Schweitzer, Tilman Plehn, Peter Sorrenson, Jonas Spinner
arXiv: 2305.10475

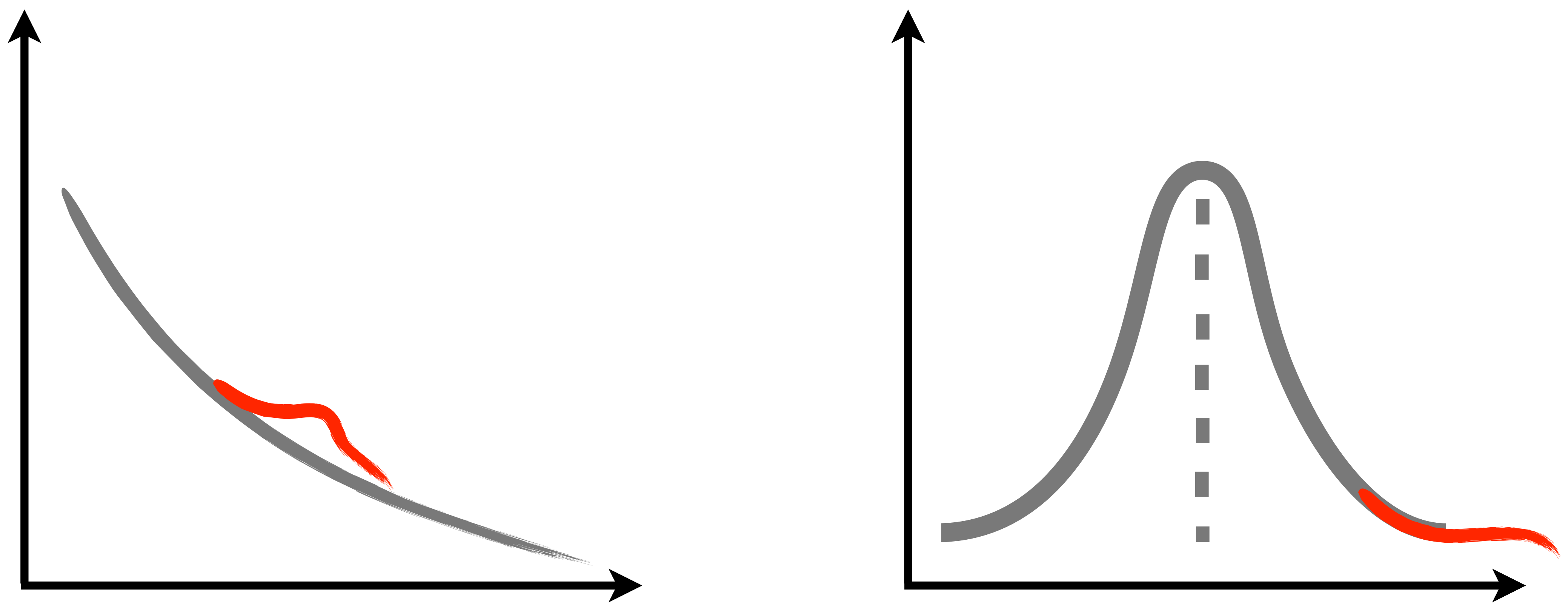
Why Event Generation?

Vast amount of data collected by collider experiments

Standard Model is probed

Theoretical predictions (simulation) needs to match experimental statistics

$5\,\sigma$?



Why ML Event Generation?

After high luminosity runs → ~ 5 times as much data

Theoretical predictions needs to be even more precise (include higher correction terms)

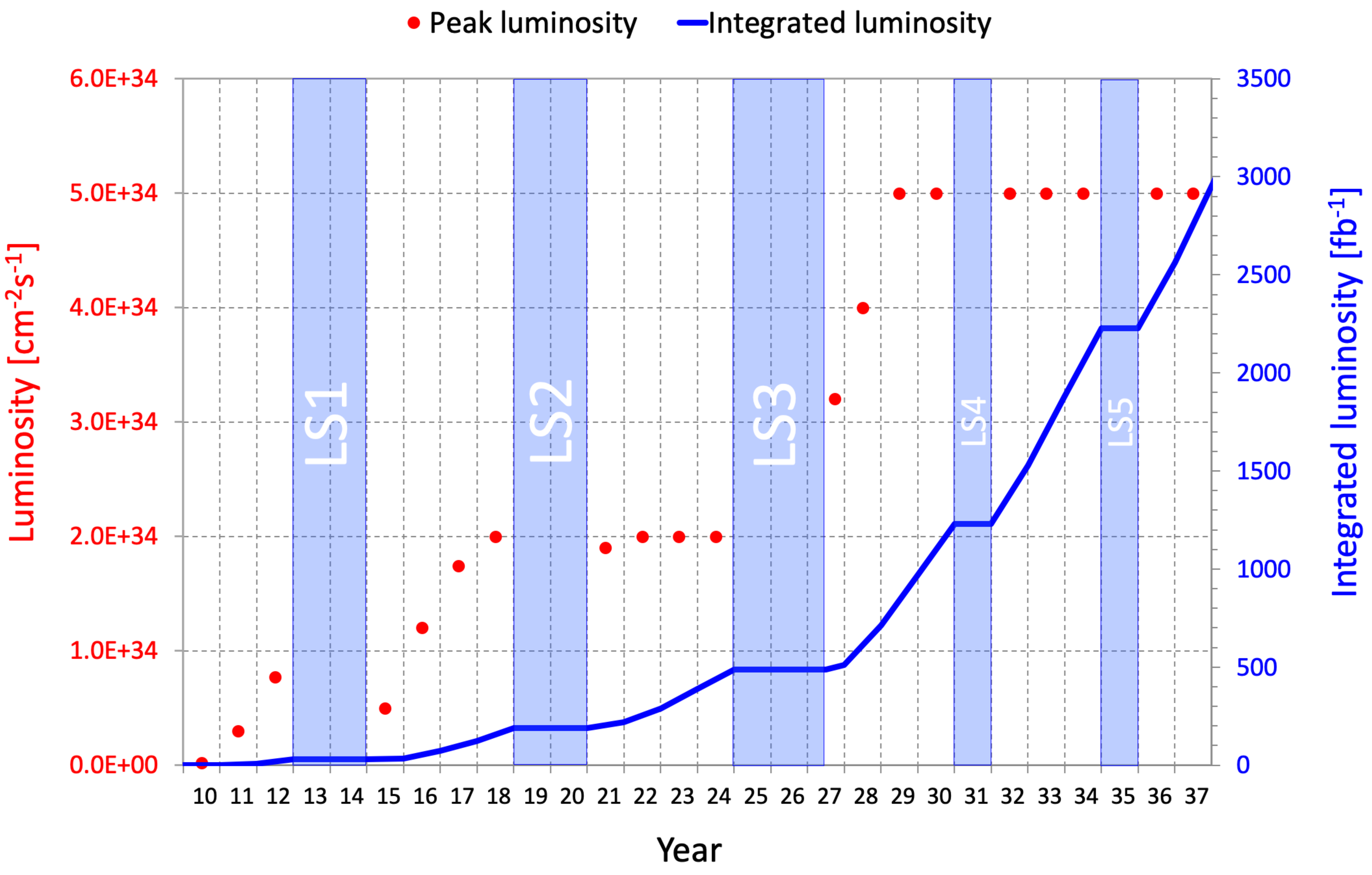


Figure from https://web.archive.org/web/20220706170326/https://lhc-commissioning.web.cern.ch/schedule/images/LHC-nominal-lumi-projection.png

Why ML Event Generation?

After high luminosity runs → ~ 5 times as much data

Theoretical predictions needs to be even more precise (include higher correction terms)

But: Currently computational expensive

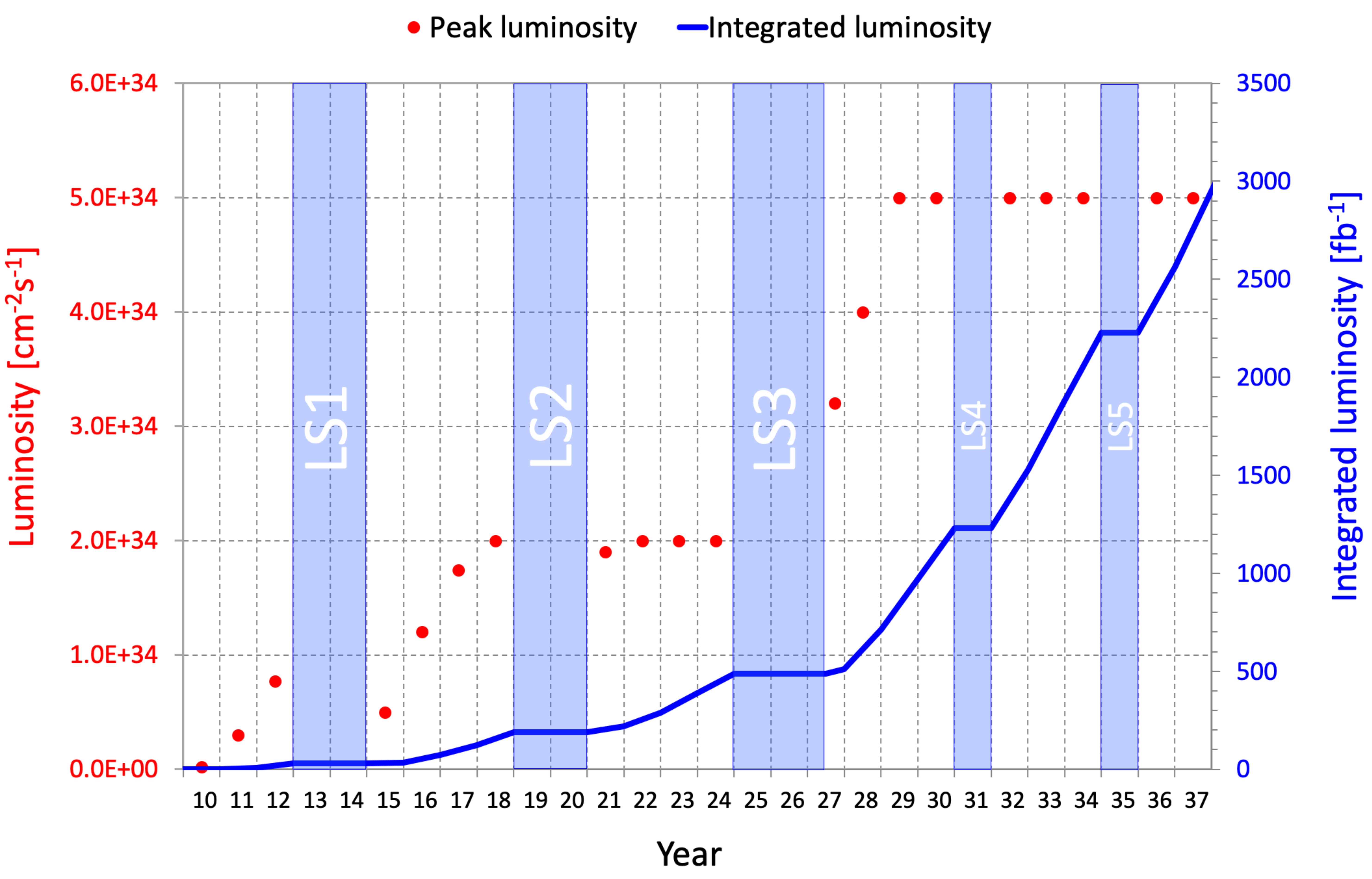


Figure from https://web.archive.org/web/20220706170326/https://lhc-commissioning.web.cern.ch/schedule/images/LHC-nominal-lumi-projection.png

Where ML Event Generation?

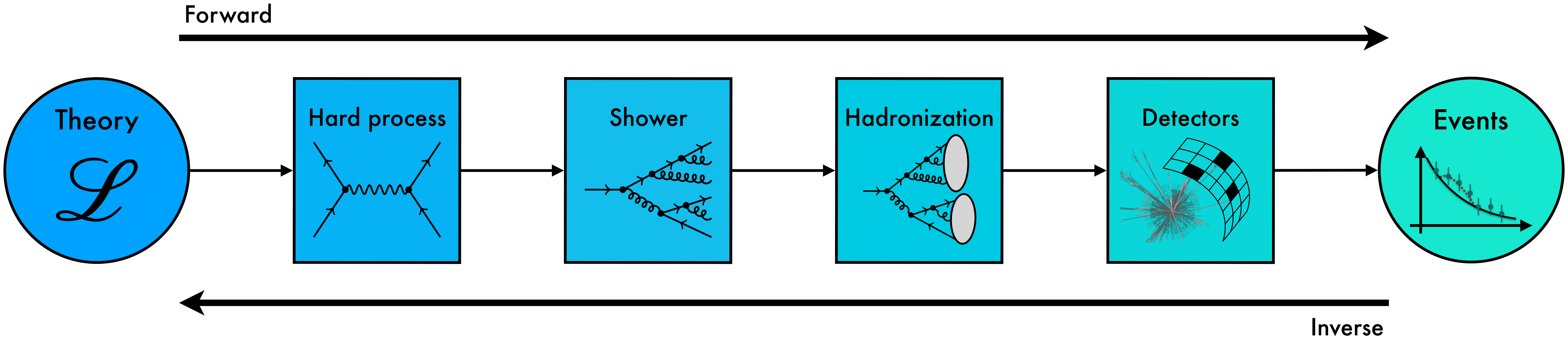


Figure from A. Butter et al.: arXiv:2203.07460, R. Winterhalder

Where ML Event Generation?

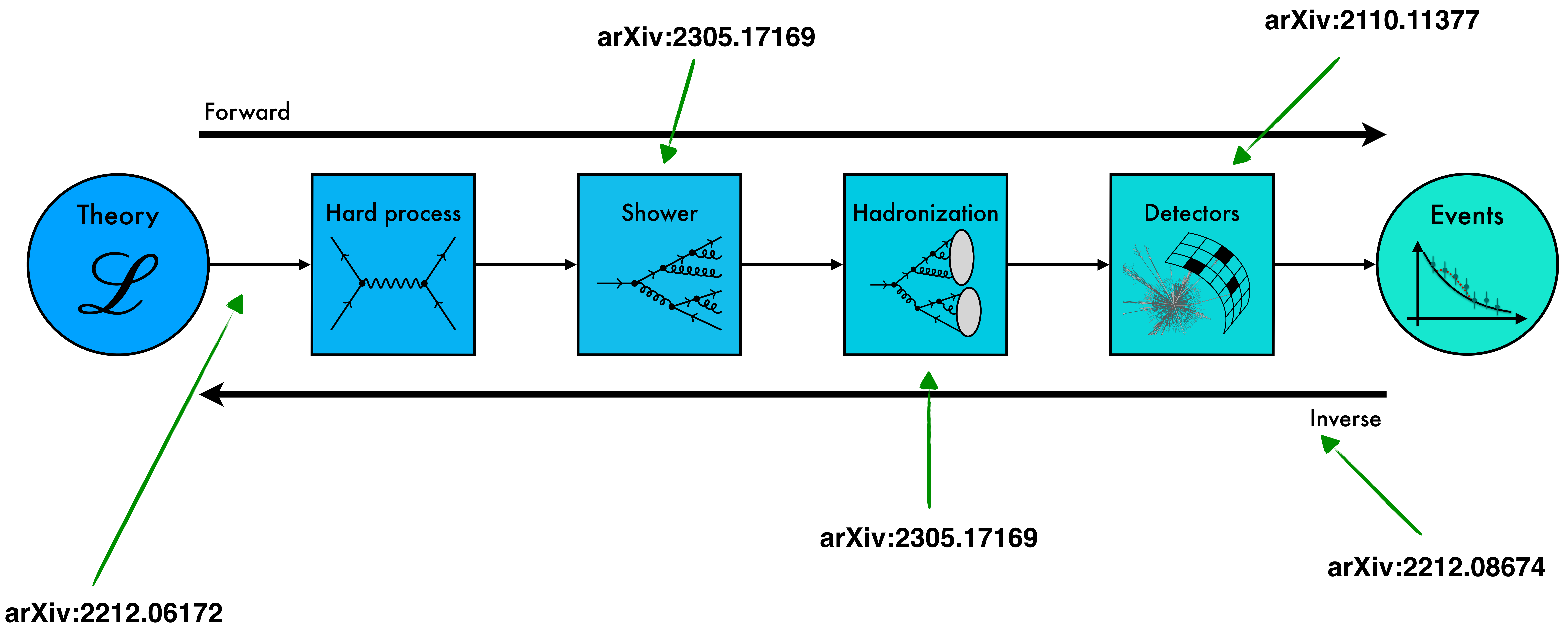


Figure from A. Butter et al.: arXiv:2203.07460, R. Winterhalder

Where ML Event Generation?

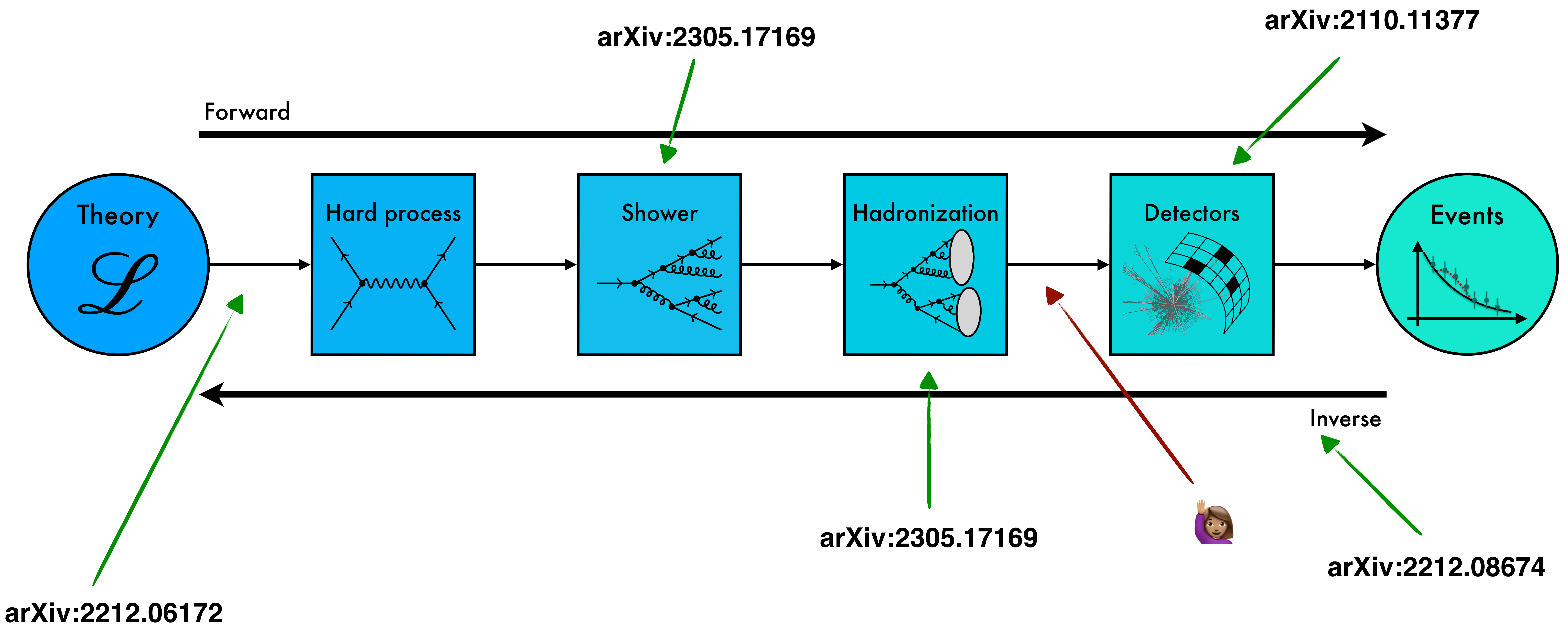


Figure from A. Butter et al.: arXiv:2203.07460, R. Winterhalder

How to be generative

Phase Space

$x \sim p(x)$

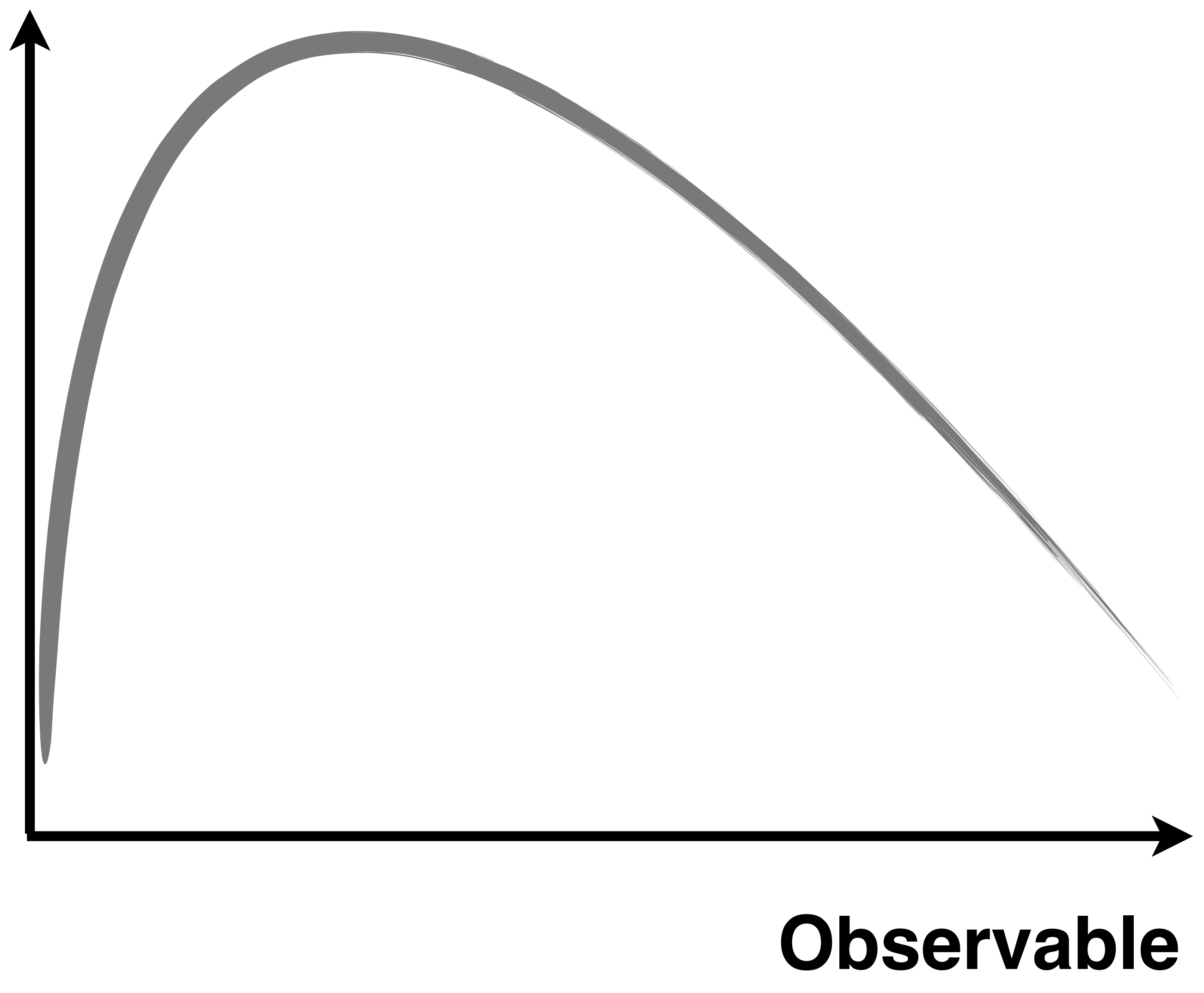
👎🏽 Difficult to sample from

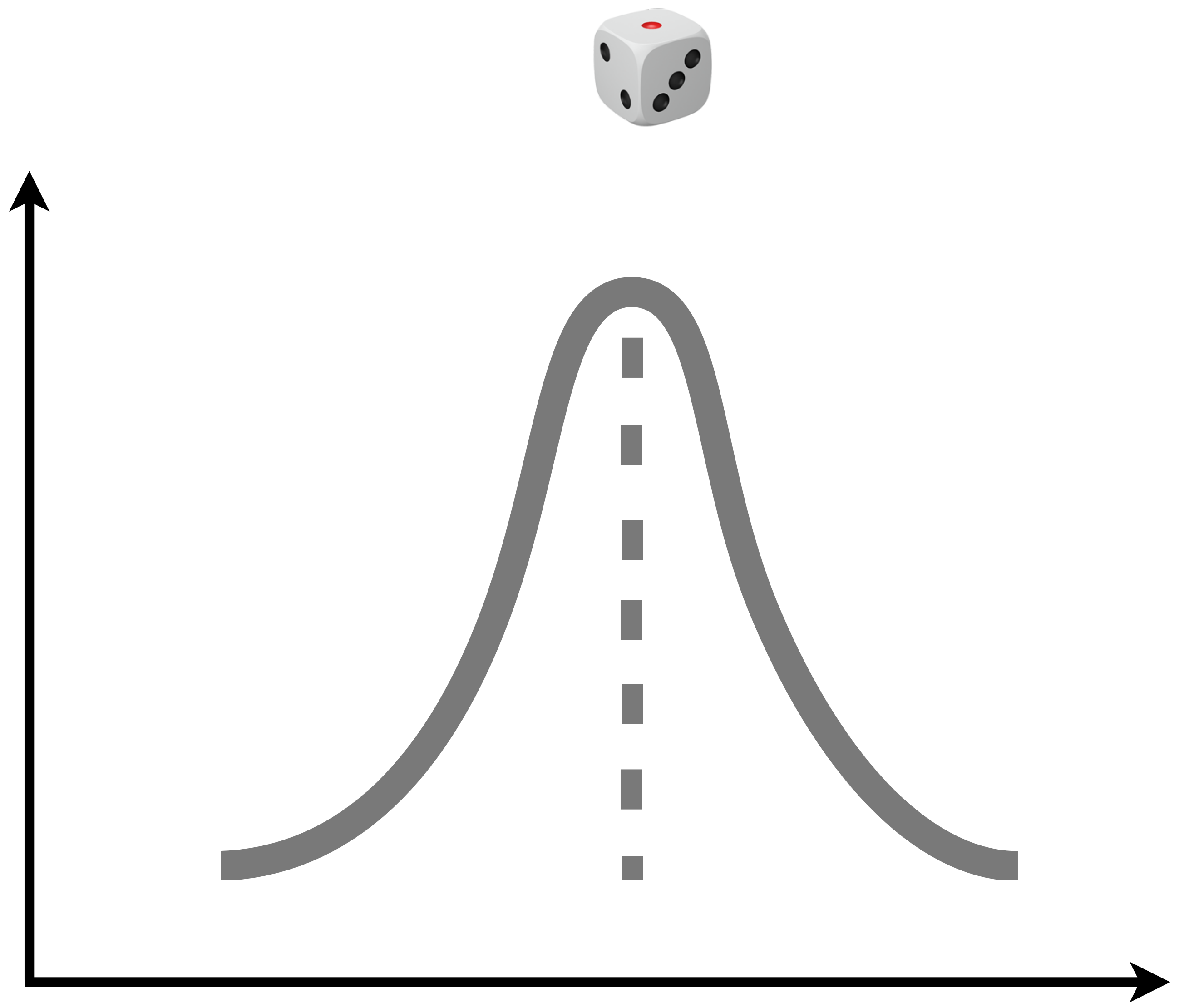
?

Latent Space

$z \sim \mathcal{N}(0,1)$

👍🏽 Easy to sample from



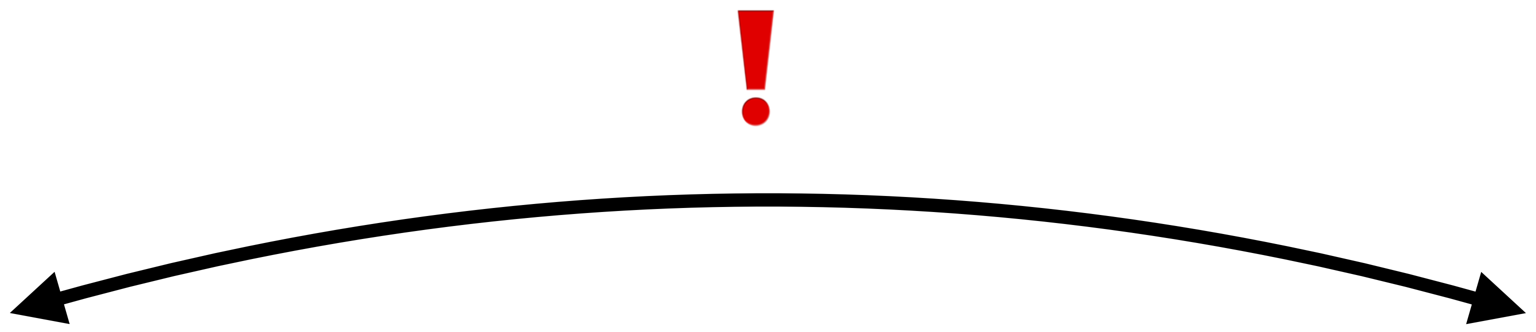


How to be generative

Phase Space

$x \sim p(x)$

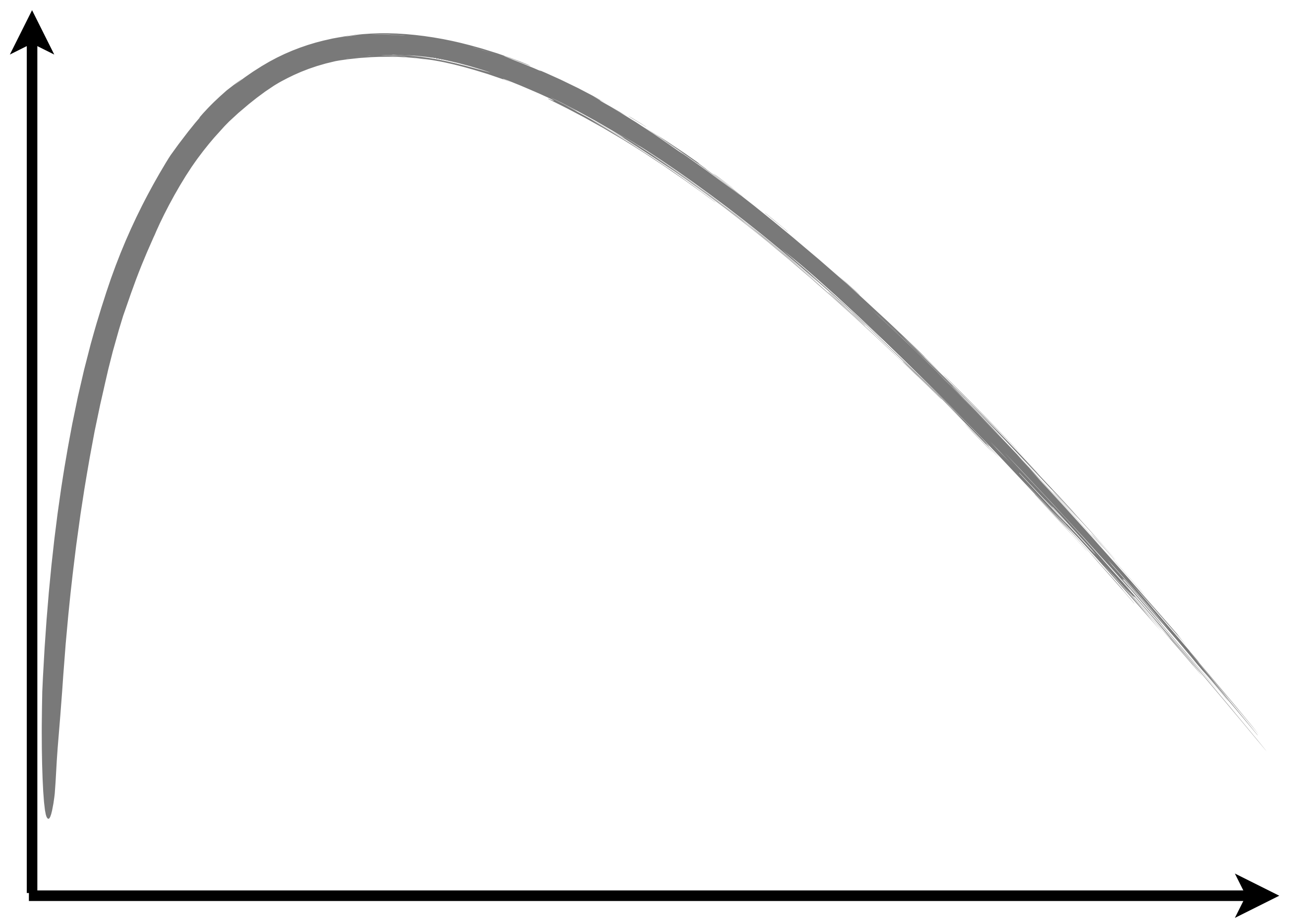
👎 Difficult to sample from



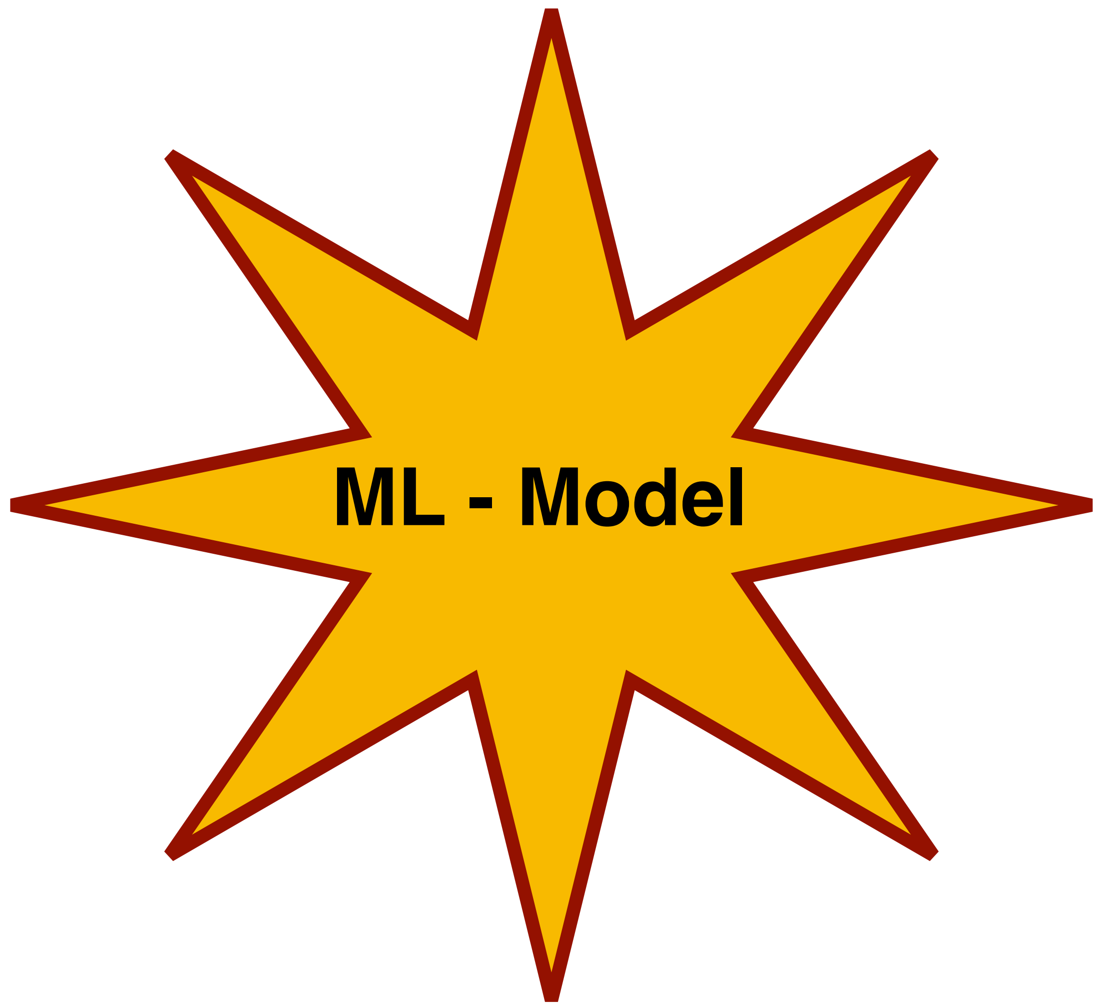
Latent Space

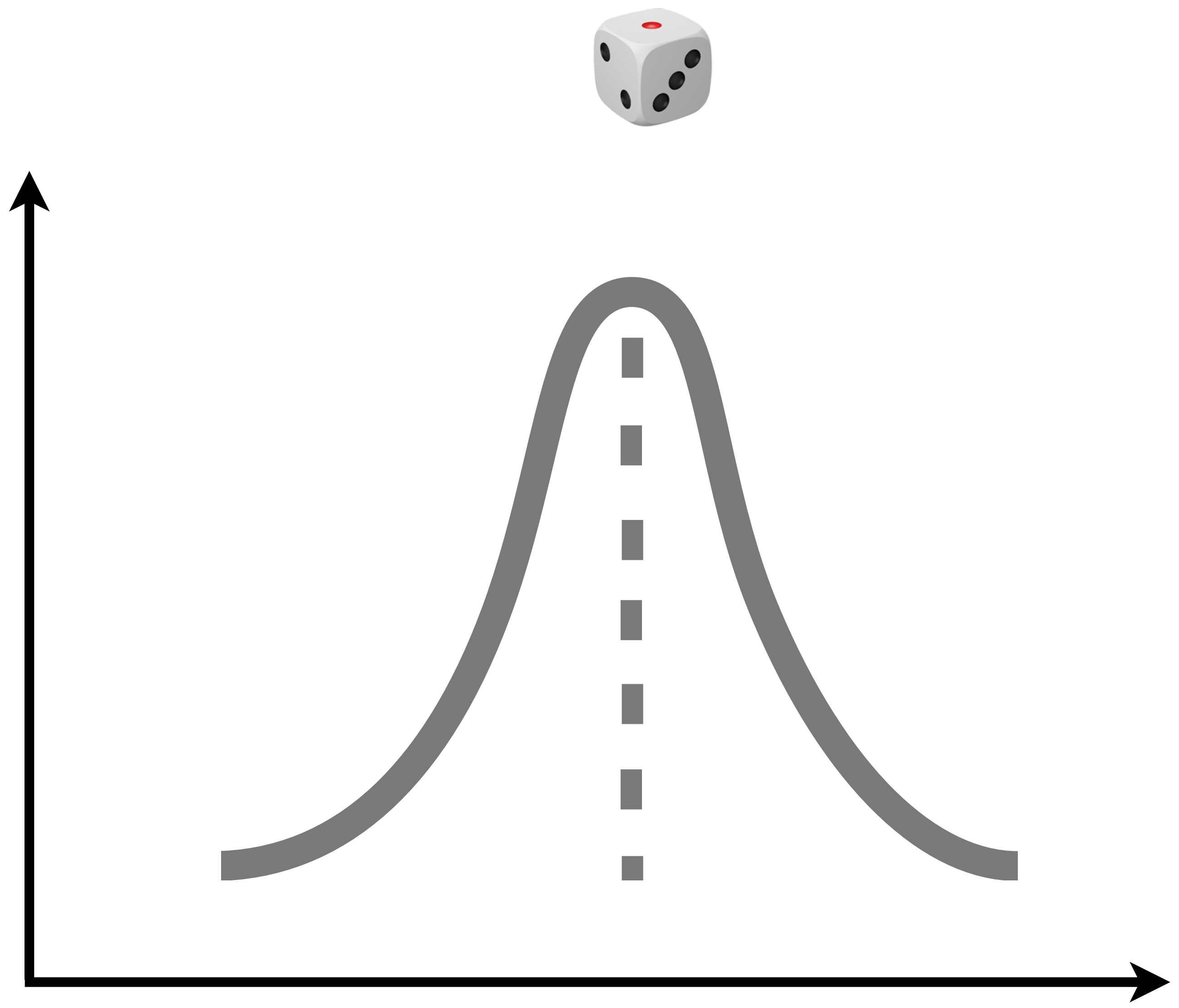
$z \sim \mathcal{N}(0,1)$

👍 Easy to sample from



Observable

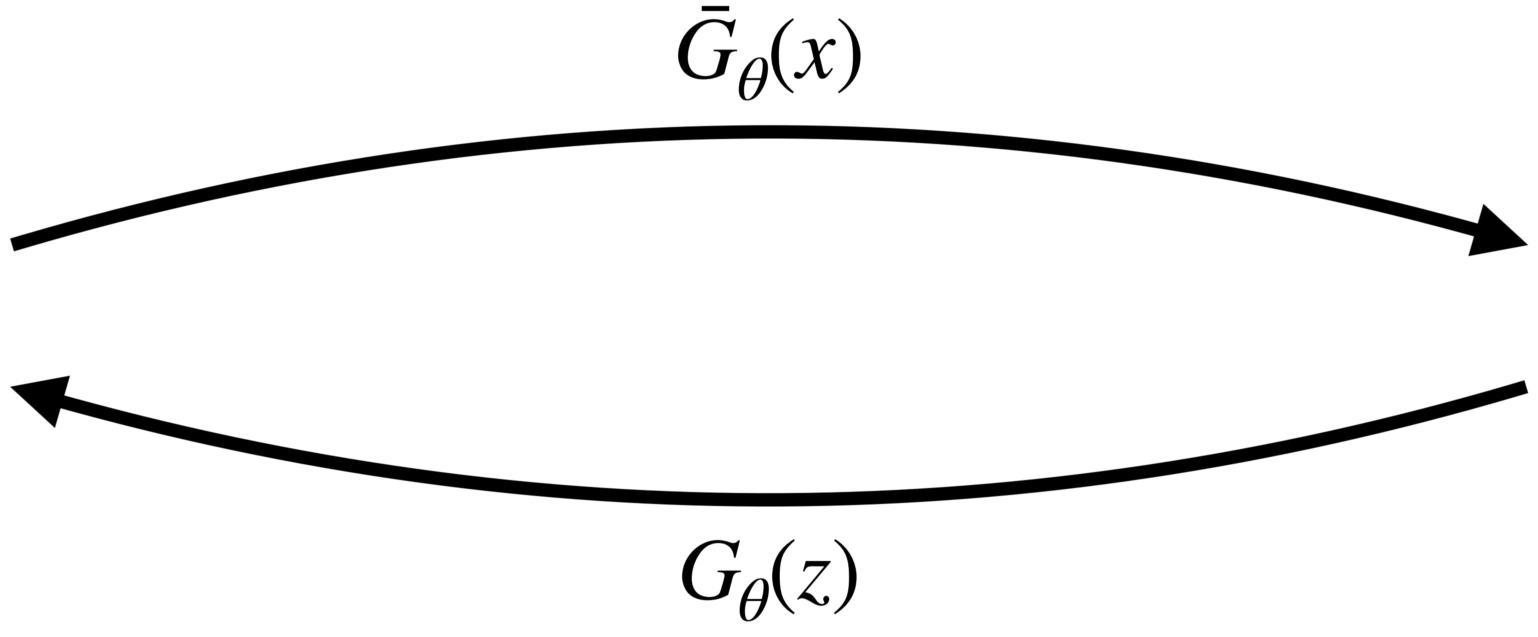




Invertible Neural Networks (INNs)

Phase Space

$x \sim p(x)$



Latent Space

$z \sim \mathcal{N}(0,1)$

Bijective mapping $G_\theta(x)$

$$p_\theta(x) = p(z)\frac{dz}{dx} = p(z)\left|\frac{\partial \bar{G}_\theta(x)}{\partial x}\right|$$

$$\mathscr{L}_{INN} = -\log p_\theta(x) = -\log p(\bar{G}_\theta(x)) - log\left|\frac{\partial \bar{G}_\theta(x)}{\partial x}\right|$$

Diffusion Models

Phase Space

$x_0 \sim p(x_0)$

Latent Space

$x_T \sim p(x_T)(= \mathcal{N}(0,1))$

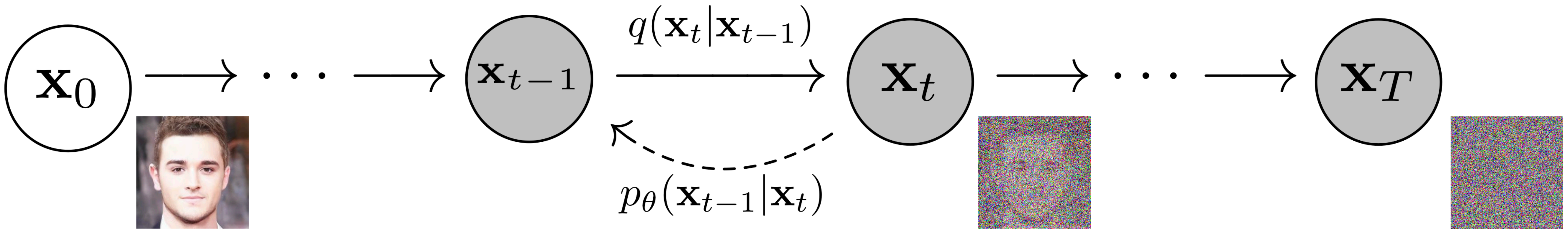
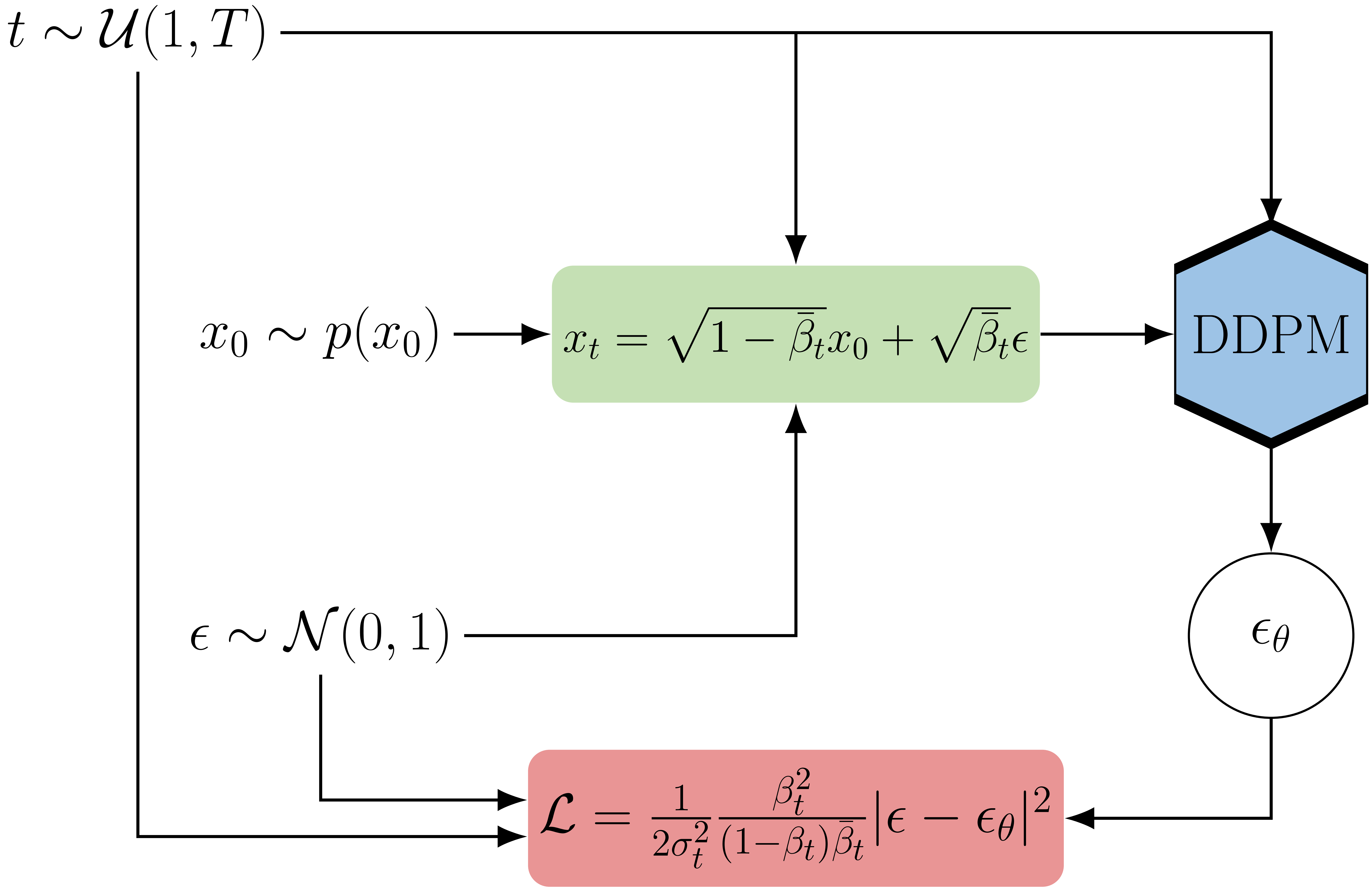
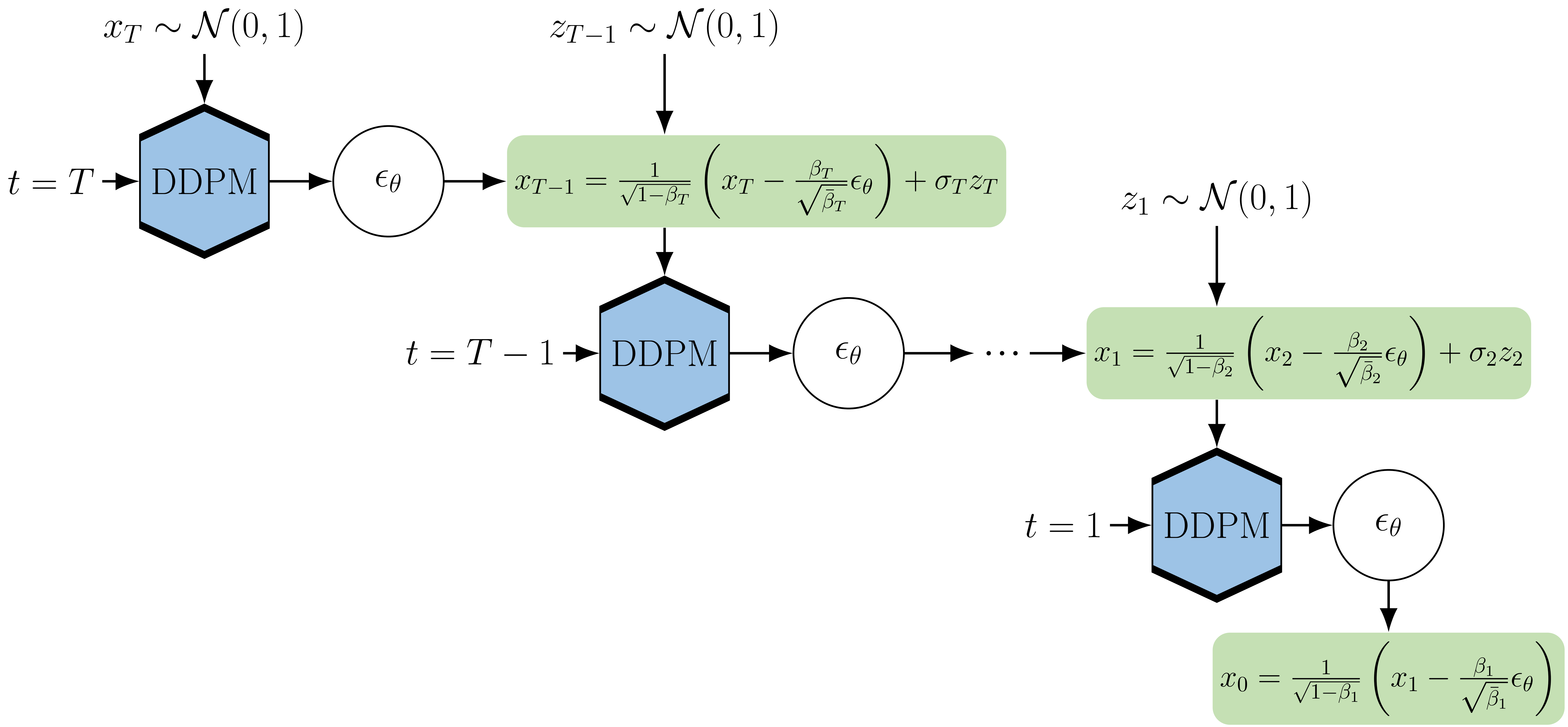


Figure from J.Ho et al.: arXiv:2006.11239

Diffusion Models (DDPM)



Diffusion Models (DDPM)



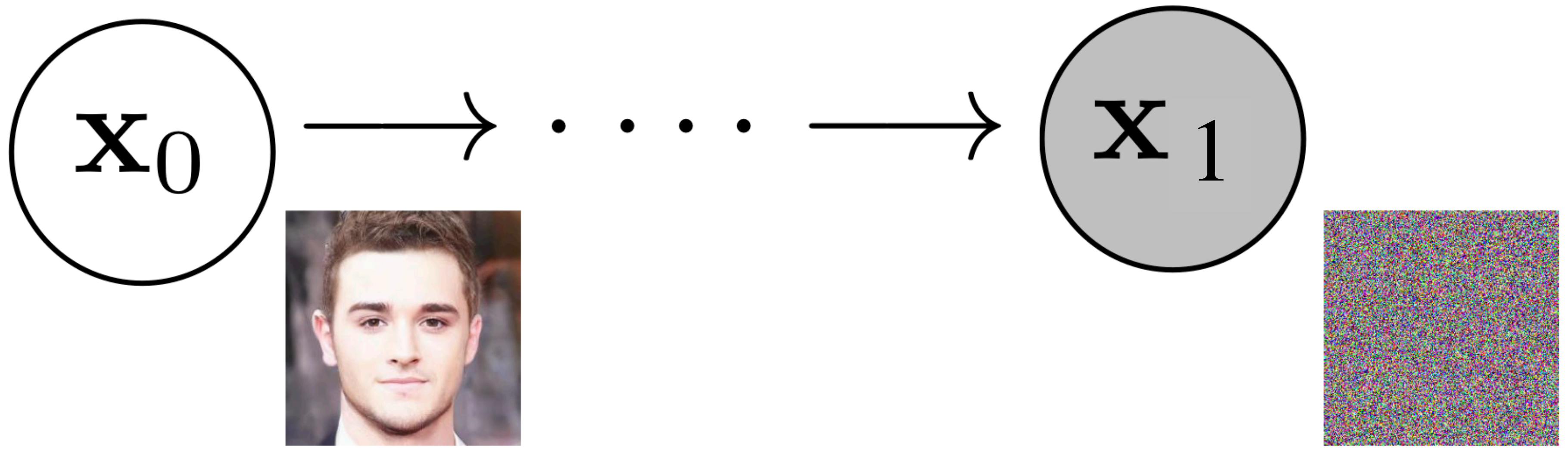
Diffusion Models (CFM)

Phase Space

$x_0 \sim p(x_0)$

Latent Space

$x_1 \sim p(x_1)(= \mathcal{N}(0,1))$

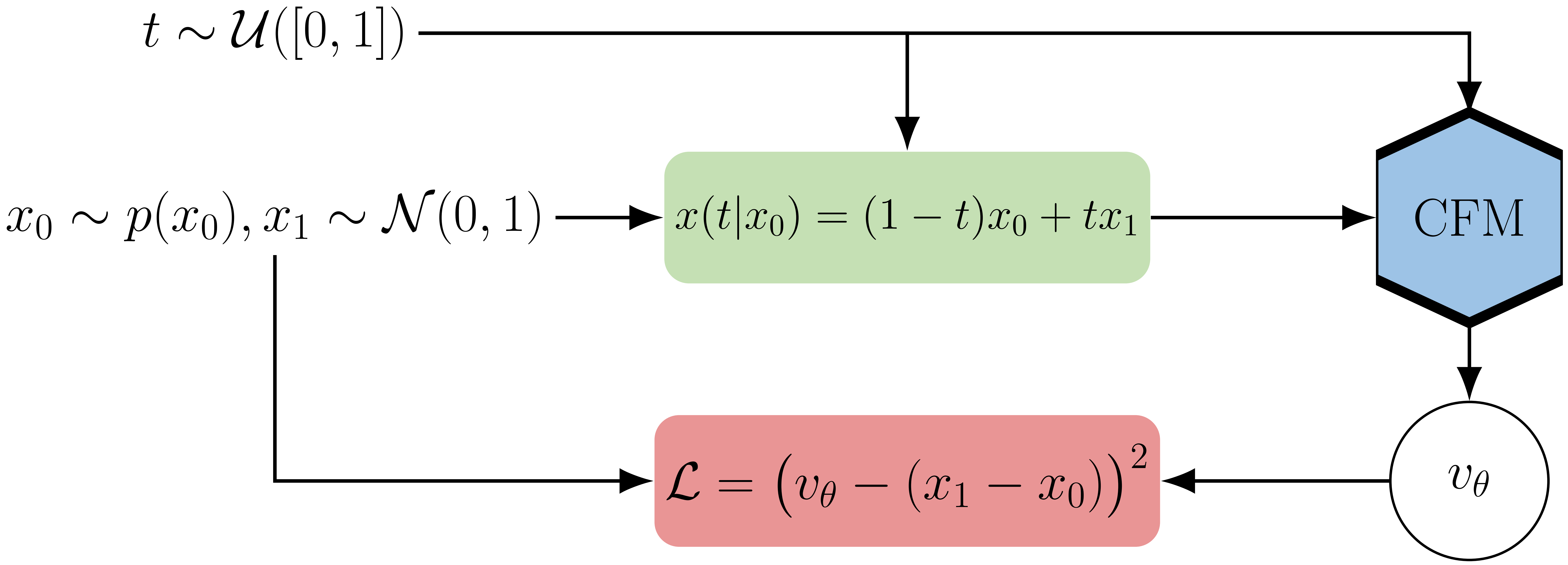


Continuous time evolution **t**

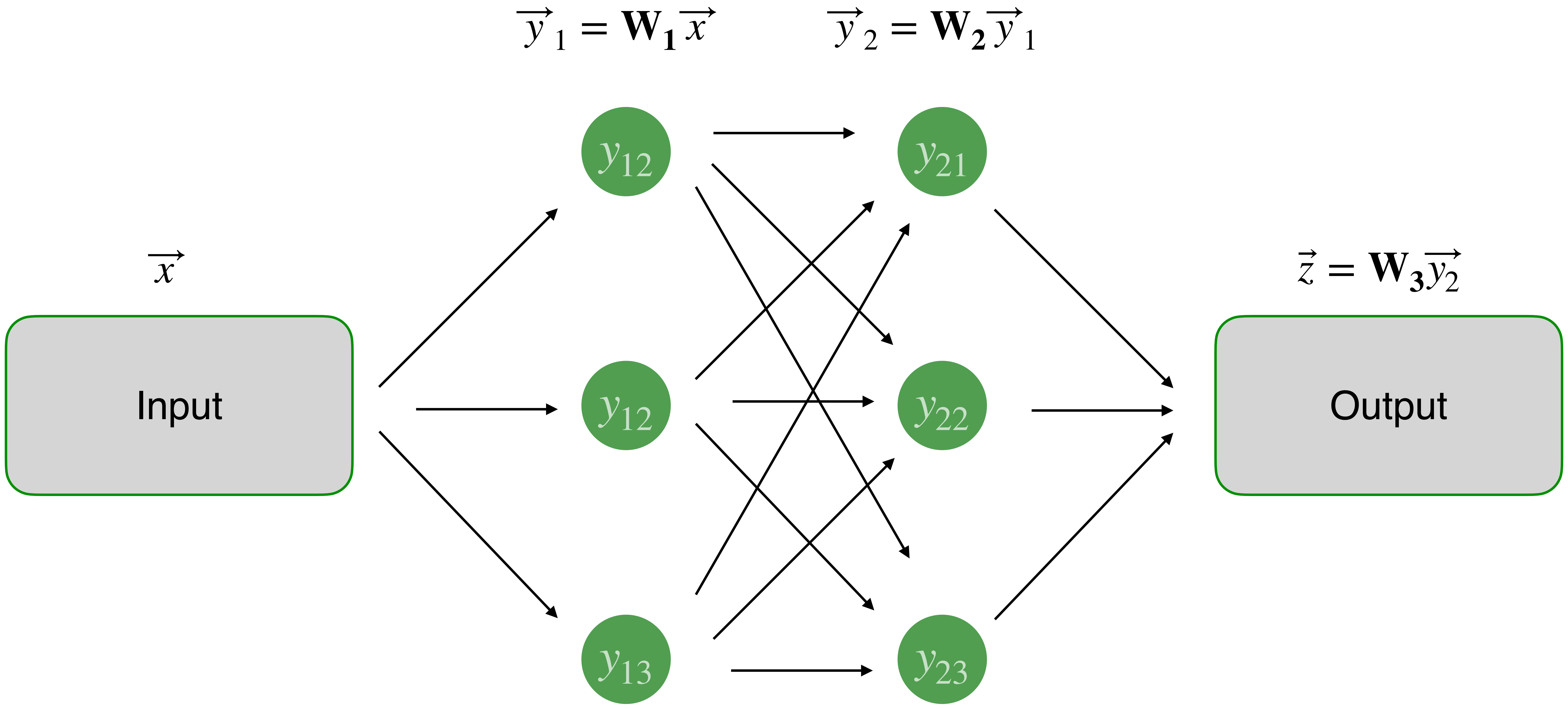
Evolution governed by $\frac{dx}{dt}$

Figure from J.Ho et al.: arXiv:2006.11239

Diffusion Models (CFM)

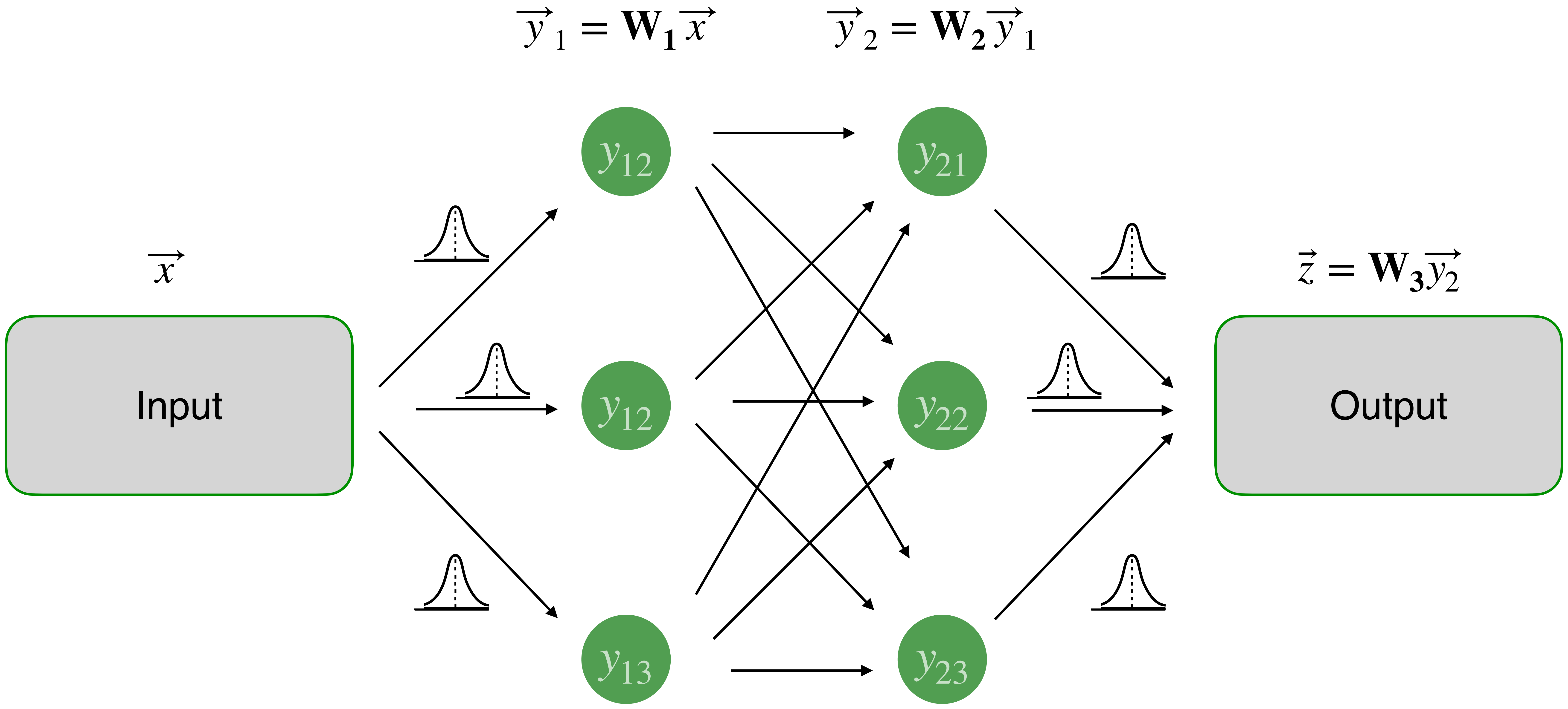


What about uncertainties?



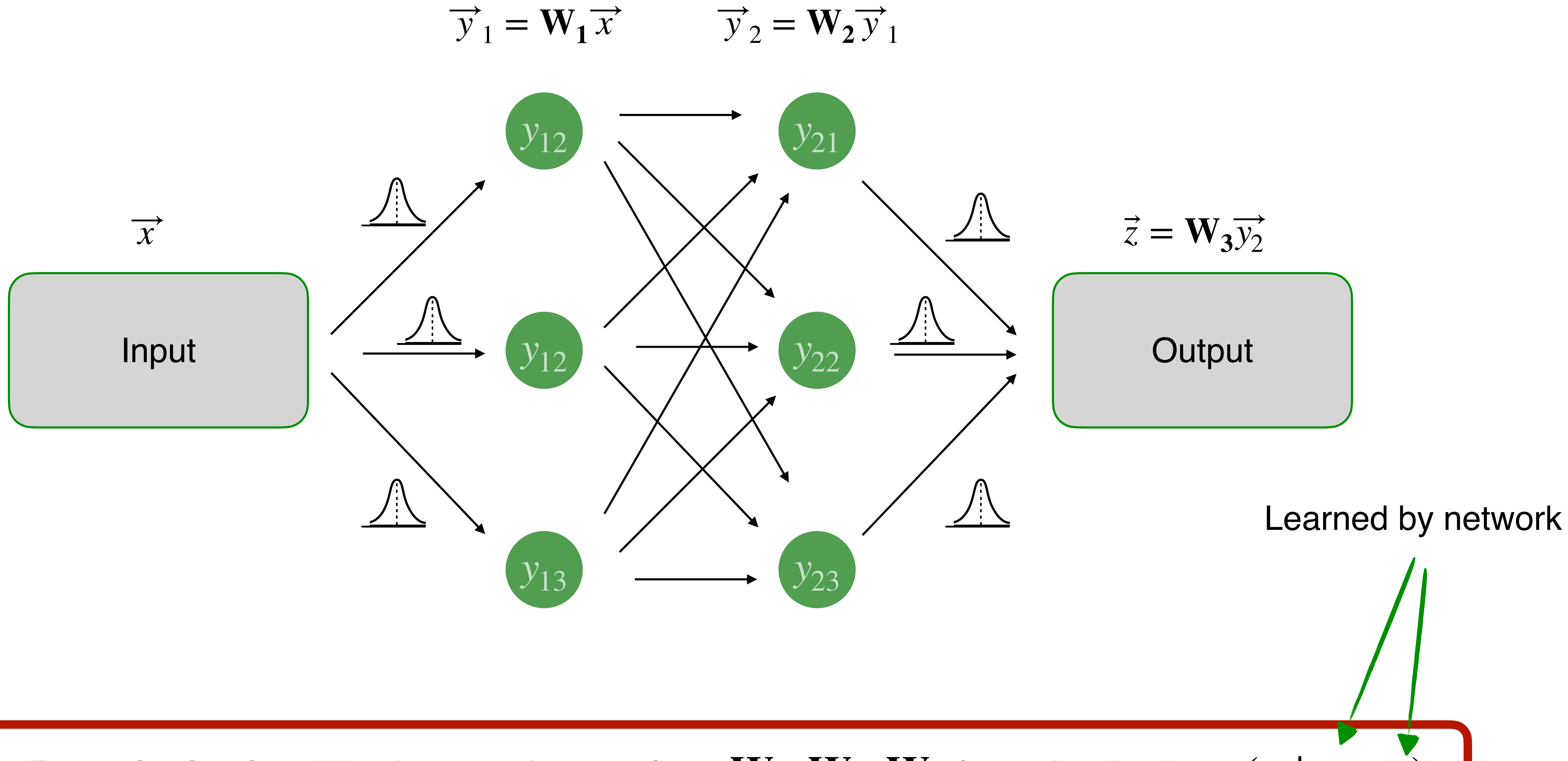
Once training is done: $\mathbf{W_1}, \mathbf{W_2}, \mathbf{W_3}$ fixed (*"Network output is deterministic"*)

What about uncertainties?



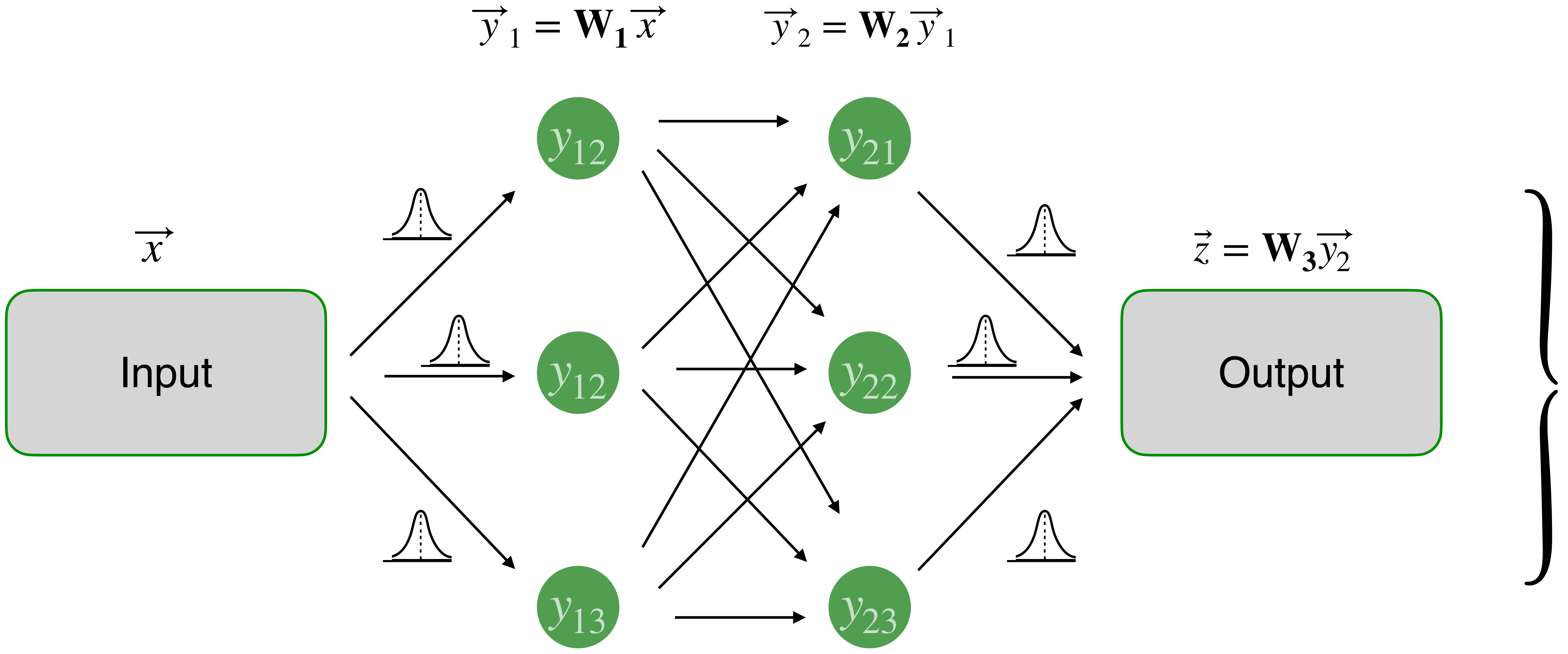
Bayesianization: We draw each entry from $\mathbf{W_1}, \mathbf{W_2}, \mathbf{W_3}$ from distribution $q(w \,|\, \mu_\phi, \sigma_\phi)$

What about uncertainties?



Bayesianization: We draw each entry from $\mathbf{W_1}, \mathbf{W_2}, \mathbf{W_3}$ from distribution $q(w \mid \mu_\phi, \sigma_\phi)$

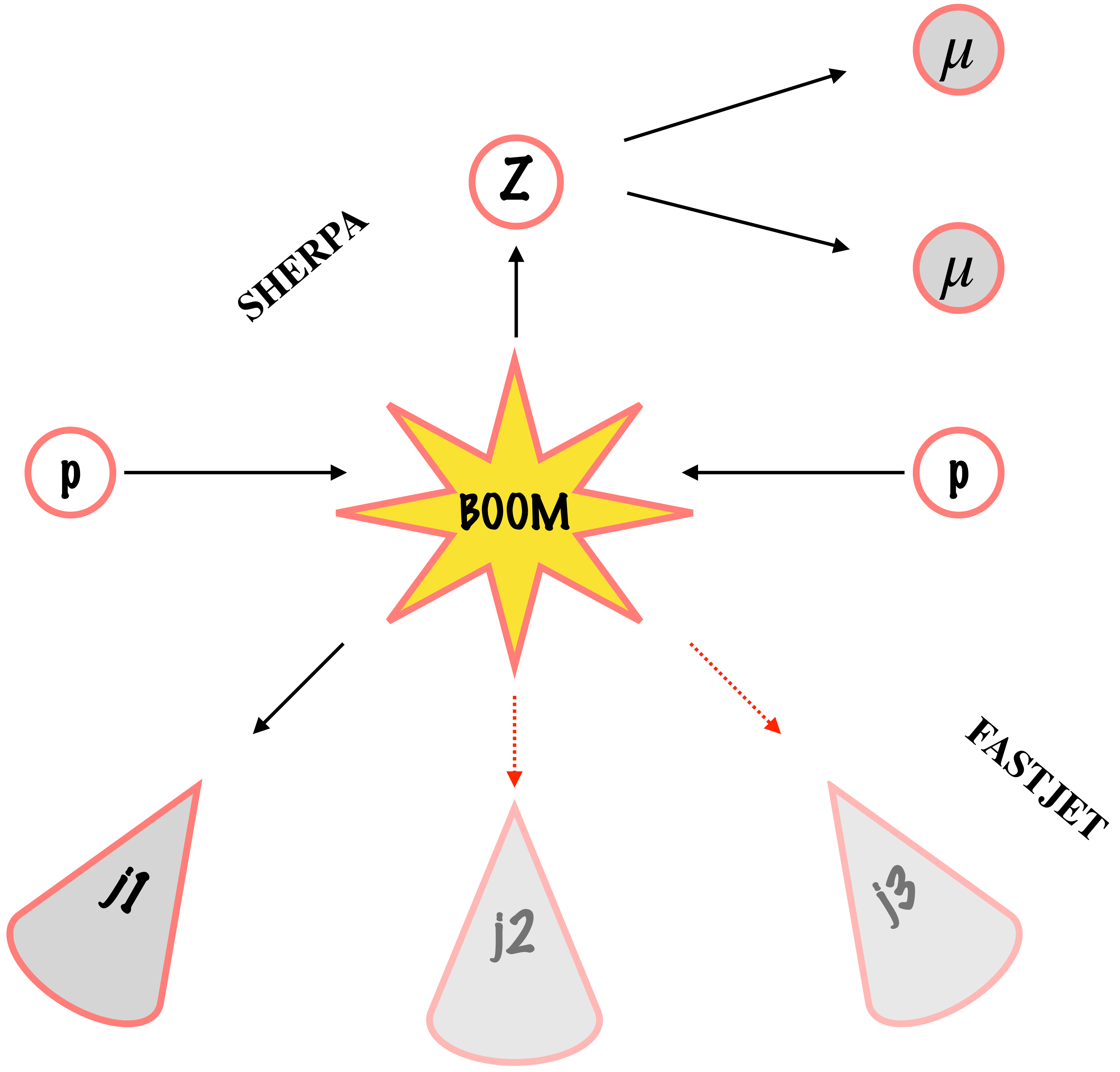
What about uncertainties?



$$\langle \vec{z} \rangle = \frac{1}{N} \sum_i \vec{z}_i$$

$$\sigma^2_{pred} = \frac{1}{N} \sum_{i=1}^{N} \left(\langle \vec{z} \rangle - \vec{z}_i \right)^2$$

Concrete Application — LHC



<u>Z ($\rightarrow \mu\mu$) + jets:</u>

3 - 5 final state particles (including jets)

12 - 20 dimensional phase space

<u>Smart preprocessing:</u>

Global Phase Shift

Drop muon masses

$\rightarrow$ reduces phase space to 9 - 17 dimensions

To be precise

Previous studies showed: INNs can reach precision benchmark

Percent level precision (comparable to statistical uncertainty)

Uncertainty well defined

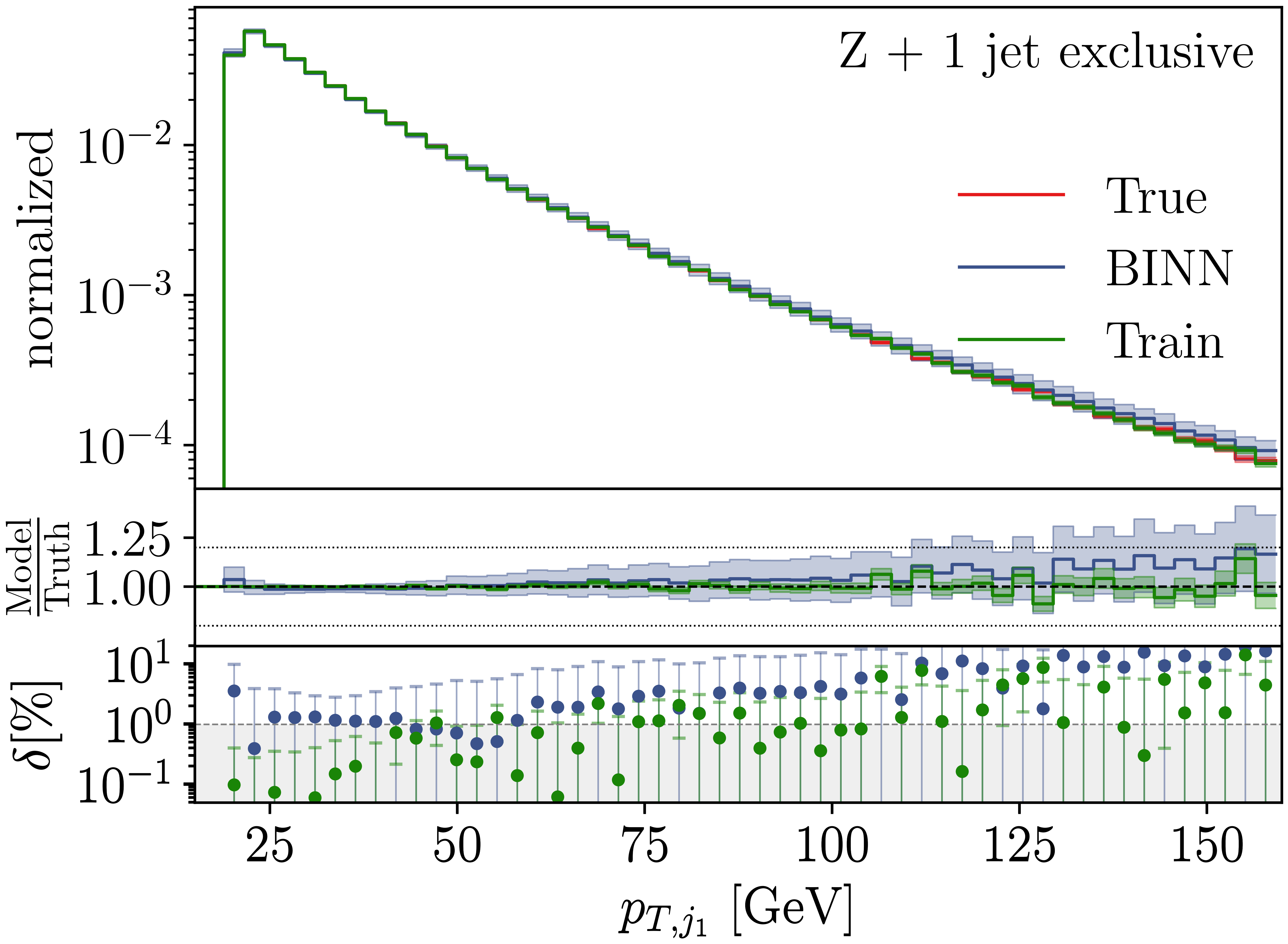


Figure from A. Butter et al.: arXiv:2110.13632

To be precise

Previous studies showed: INNs can reach precision benchmark

Percent level precision (comparable to statistical uncertainty)

Uncertainty well defined

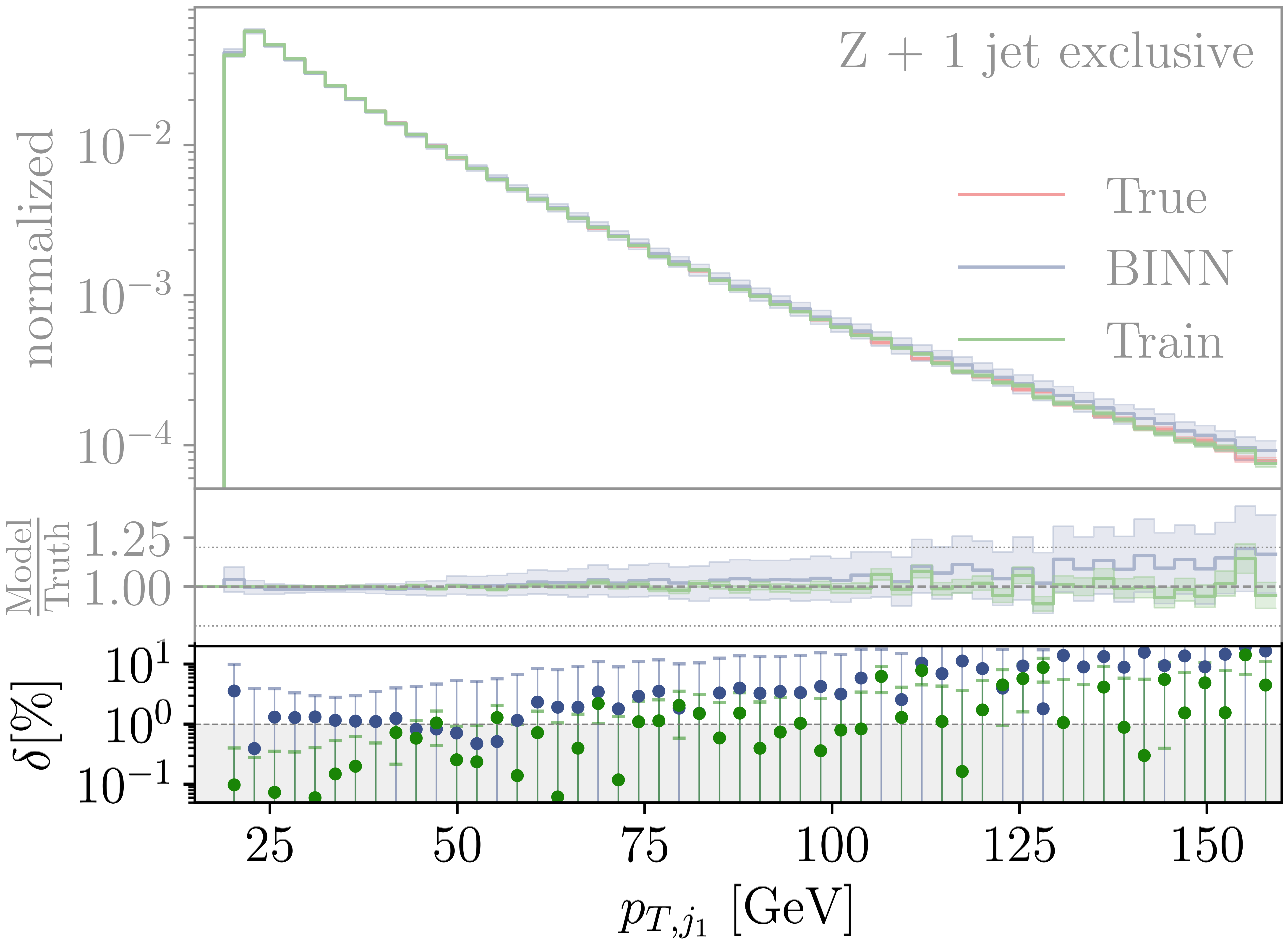


Figure from A. Butter et al.: arXiv:2110.13632

To be precise

Previous studies showed: INNs can reach precision benchmark

Percent level precision (comparable to statistical uncertainty)

Uncertainty well defined

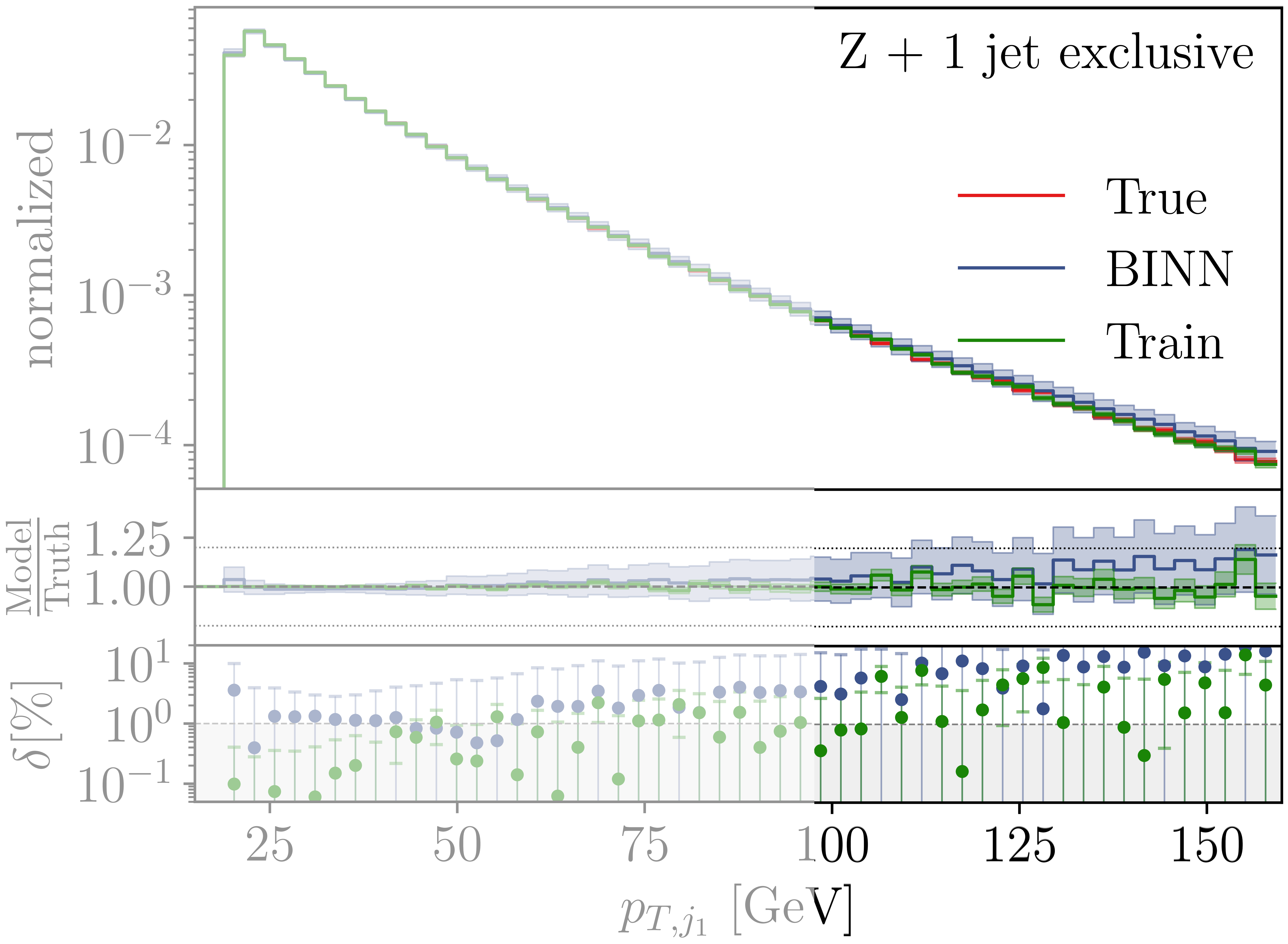


Figure from A. Butter et al.: arXiv:2110.13632

To be precise

Previous studies showed: INNs can reach precision benchmark

Percent level precision (comparable to statistical uncertainty)

Uncertainty well defined

<u>BUT:</u> only in one dimensional marginal distributions

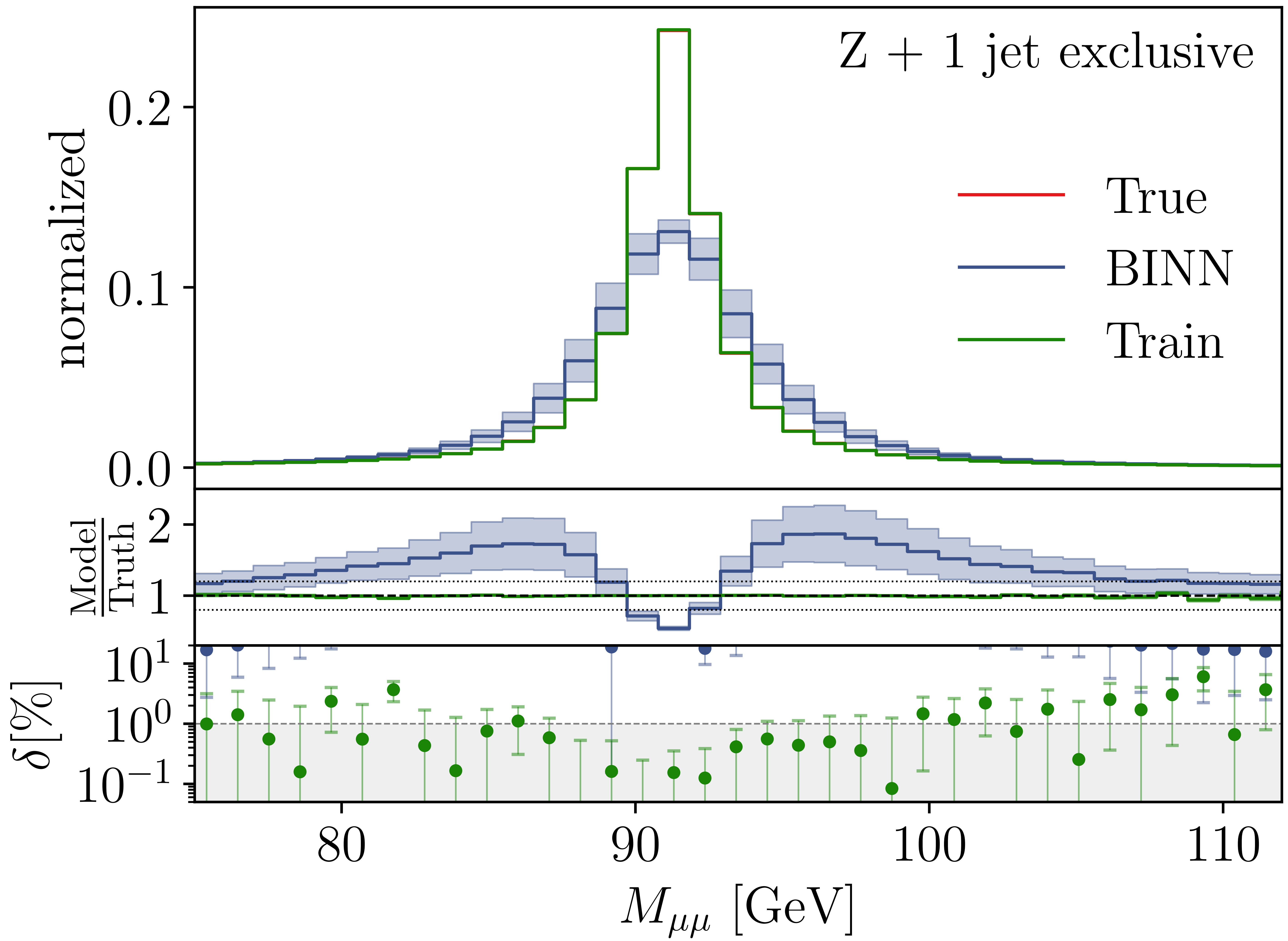


Figure from A. Butter et al.: arXiv:2110.13632

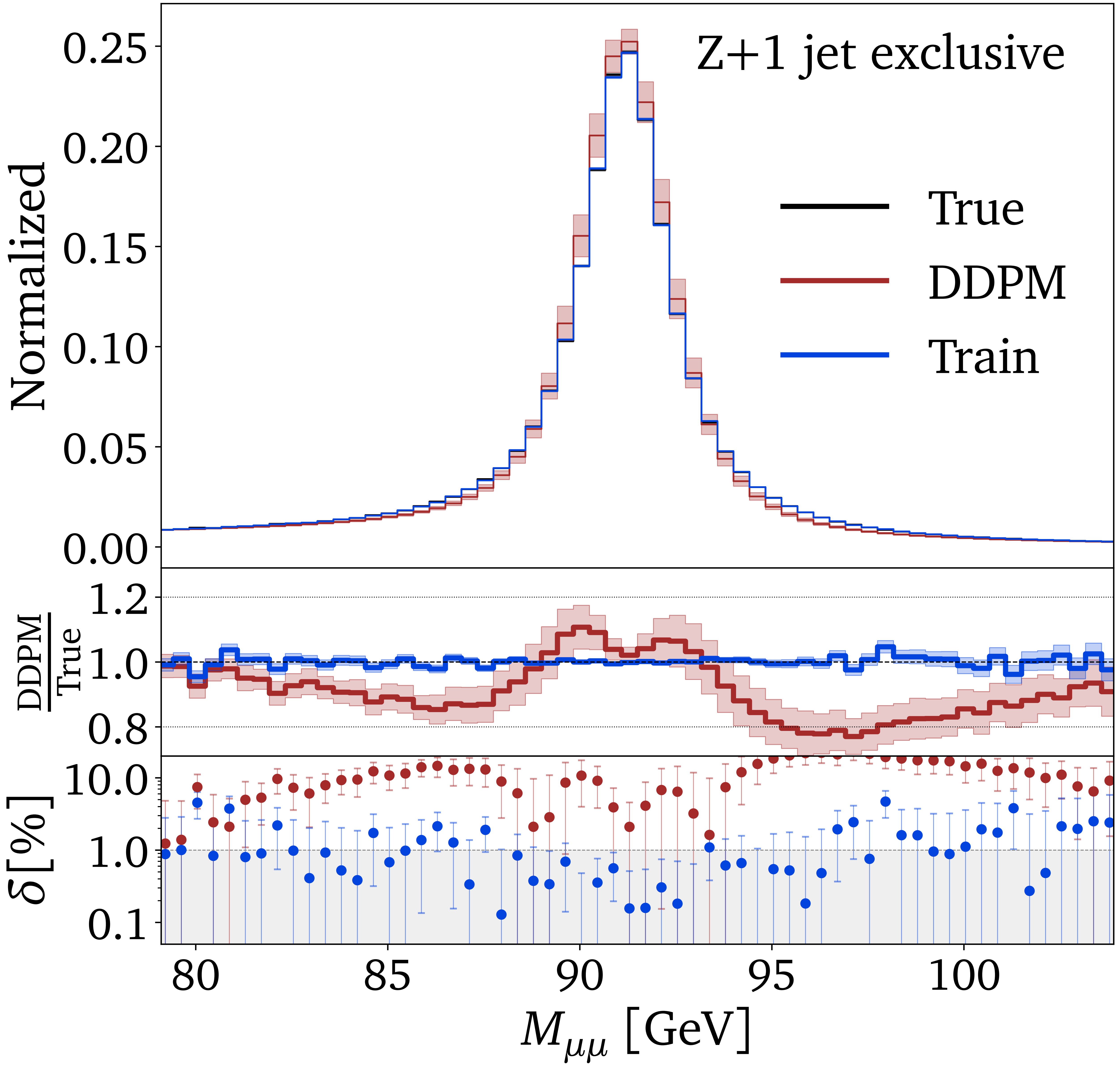
To be precise

Previous studies showed: INNs can reach precision benchmark

Percent level precision (comparable to statistical uncertainty)

Uncertainty well defined

<u>BUT:</u> only in one dimensional marginal distributions



→ **Diffusion Models surpass precision**

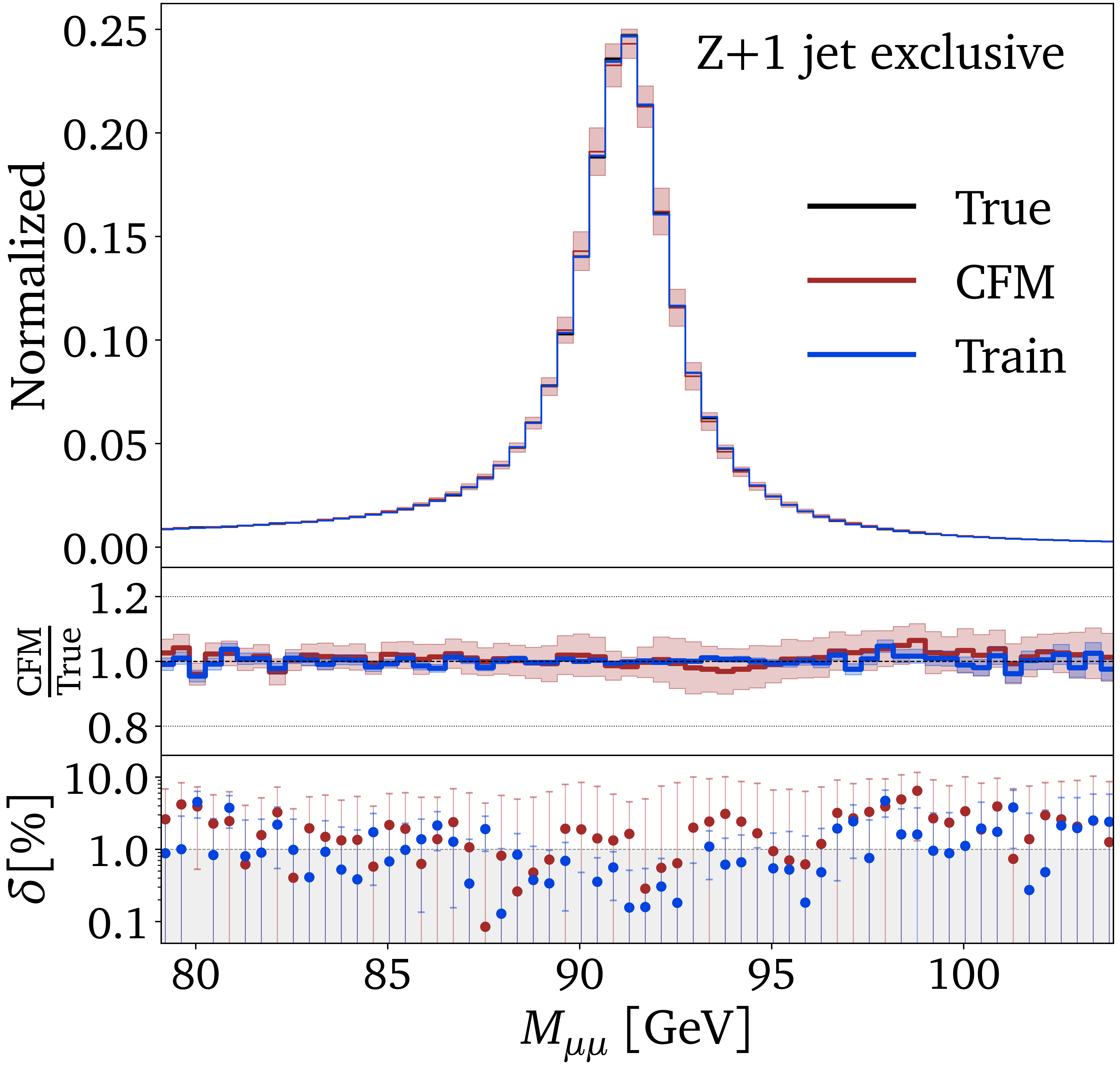
To be precise

Previous studies showed: INNs can reach precision benchmark

Percent level precision (comparable to statistical uncertainty)

Uncertainty well defined

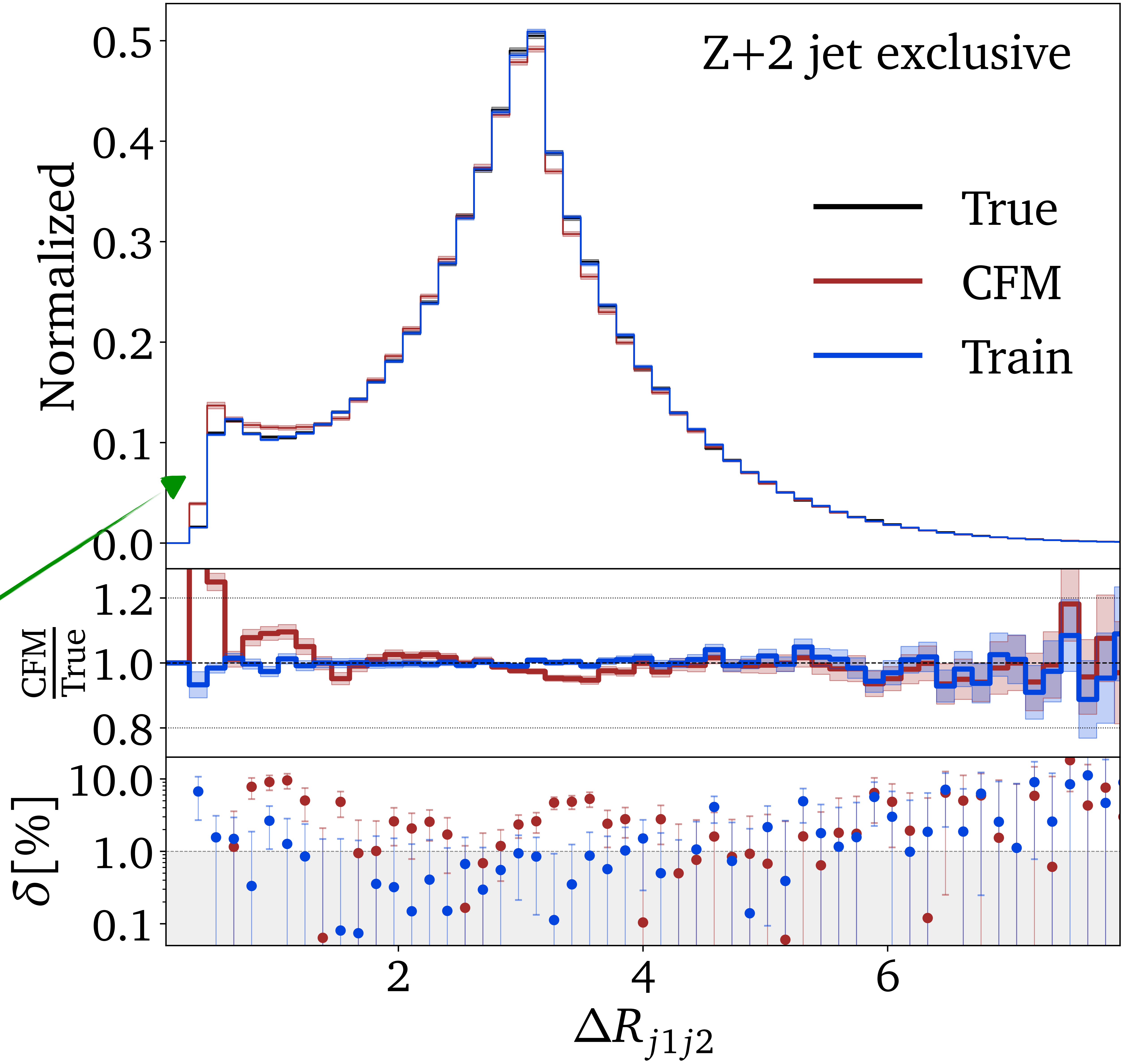
<u>BUT:</u> only in one dimensional marginal distributions



Diffusion Models surpass precision

To be precise

For all non-autoregressiv models: Difficult is to learn **sharp cuts** in correlations

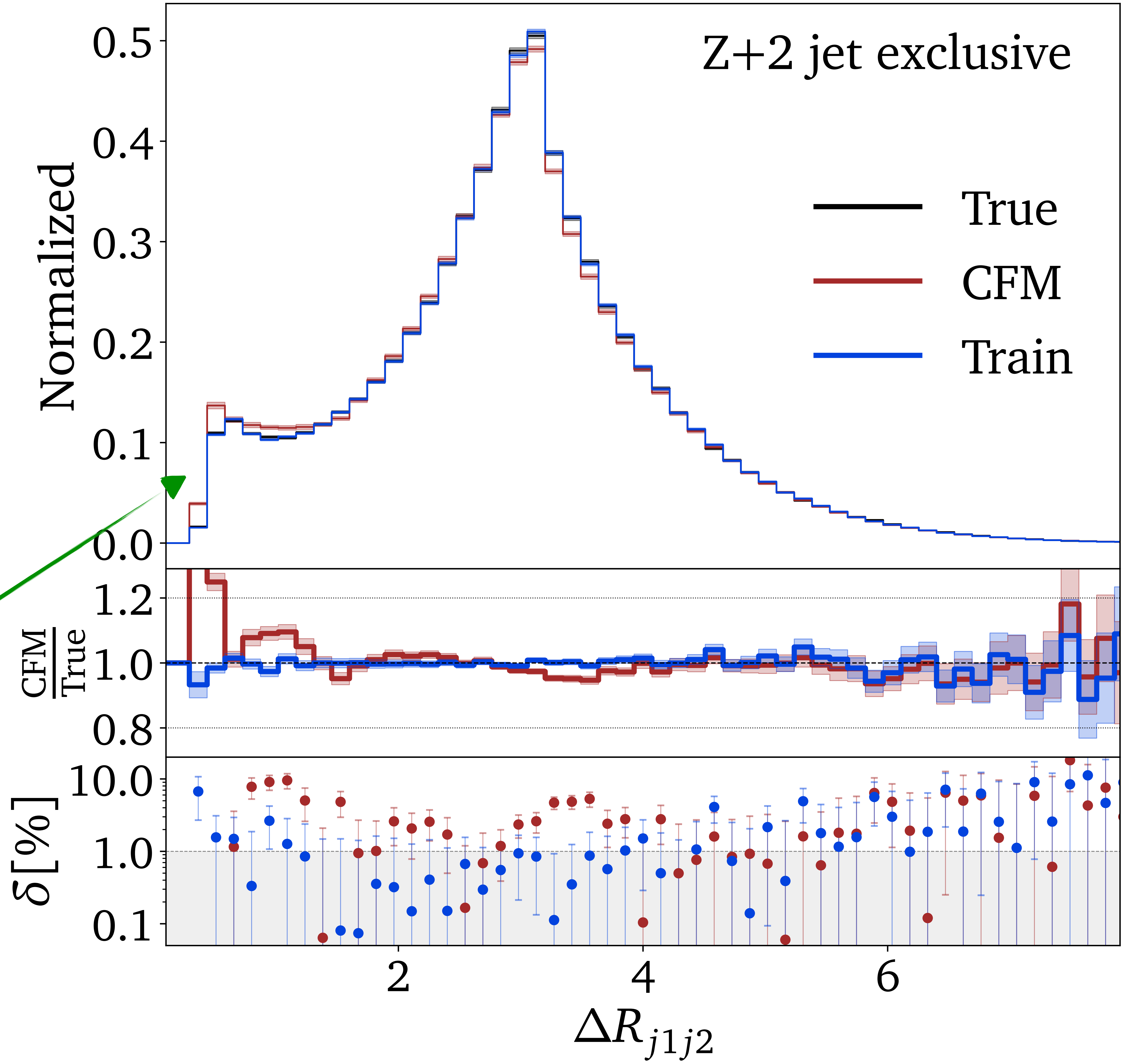


Some tricks already applied

To be precise

For all non-autoregressiv models: Difficult is to learn **sharp cuts** in correlations

Could an autoregressiv model help?



Some tricks already applied

And now what?

New, “hyped” ML-Models can compete with current benchmark

All of them come with their own advantages and disadvantages

A lot of on-going research (generation speed up, precision, etc.)

→ Diffusion models show potential to be applied to particle physics tasks

Backup

Diffusion Models (DDPM)

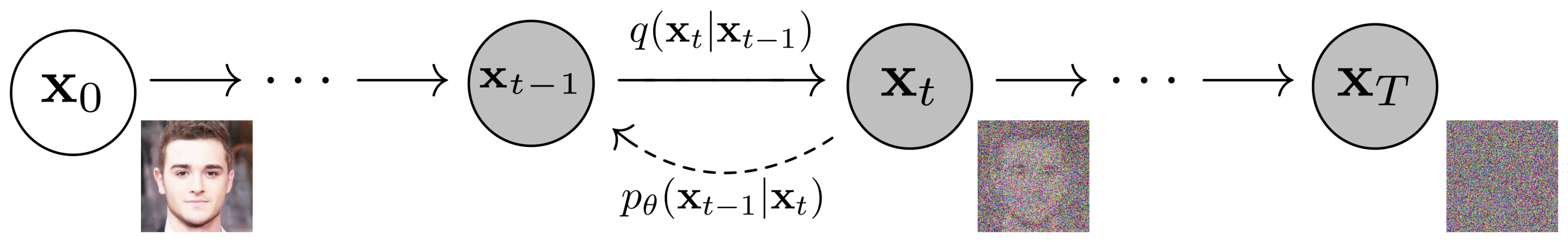
Forwards

Phase Space

$x_0 \sim p(x_0)$

Latent Space

$x_T \sim p(x_T)(= \mathcal{N}(0,1))$



$$q(x_1, \ldots, x_T | x_0) = \prod_{t=1}^{T} q(x_t | x_{t-1})$$

$$q(x_t | x_{t-1}) = \mathcal{N}(x_t; \sqrt{1-\beta_t} x_{t-1}, \beta_t)$$

where β_t follows noise scheduler

Figure from J.Ho et al.: arXiv:2006.11239

Diffusion Models (DDPM)

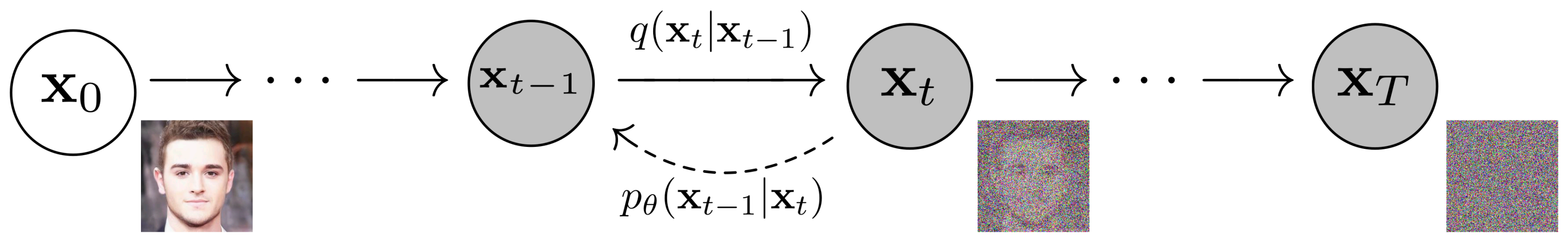
Phase Space

$x_0 \sim p(x_0)$

Backwards

Latent Space

$x_T \sim p(x_T)(= \mathcal{N}(0,1))$



To reverse: $q(x_{t-1} | x_t) = \dfrac{q(x_t | x_{t-1}) q(x_{t-1})}{q(x_t)}$

…but we don't know $q(x_t)$ & $q(x_{t-1})$ 🙎‍♀️ 🤷‍♀️

Figure from J.Ho et al.: arXiv:2006.11239

Diffusion Models (DDPM)

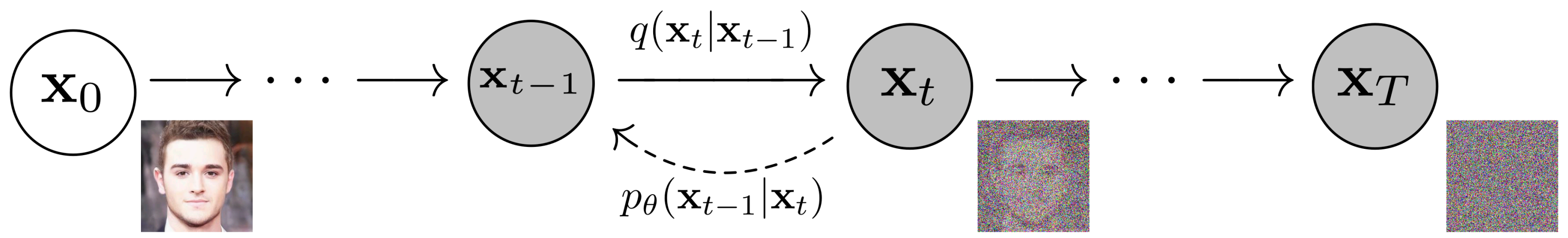
Phase Space

$x_0 \sim p(x_0)$

Backwards

Latent Space

$x_T \sim p(x_T)(= \mathcal{N}(0,1))$



Instead, learn: $p_\theta(x_{t-1} \,|\, x_t) = \mathcal{N}(x_{t-1}; \mu_\theta(x_t, t), \sigma_\theta^2(x_t, t))$

$$p(x_0, \ldots, x_T \,|\, \theta) = p(x_T) \prod_{t=1}^{T} p_\theta(x_{t-1} \,|\, x_t)$$

Figure from J.Ho et al.: arXiv:2006.11239

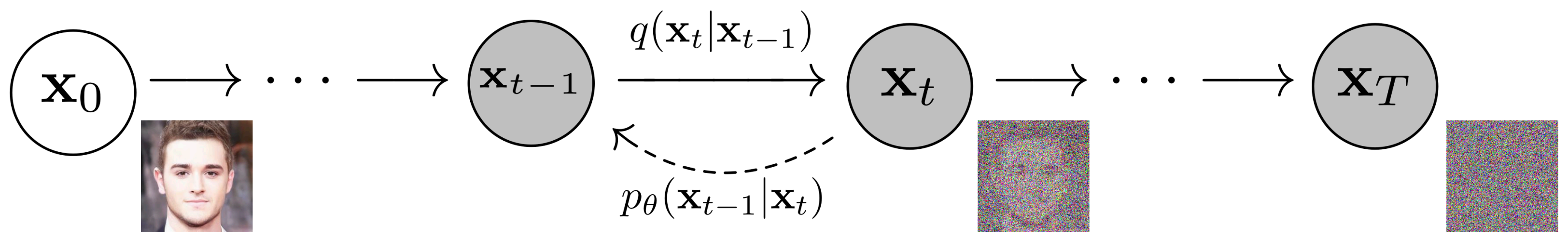
Diffusion Models (DDPM)

Phase Space

$x_0 \sim p(x_0)$

Latent Space

$x_T \sim p(x_T)(= \mathcal{N}(0,1))$



$$\mathscr{L}_{DDPM} = -\log p_\theta(x_0) \approx \frac{1}{2\sigma_t^2}\frac{\beta_t^2}{(1-\beta_t)\bar{\beta}_t}\,|\,\epsilon(t) - \epsilon_\theta(t)\,|^2$$

Figure from J.Ho et al.: arXiv:2006.11239

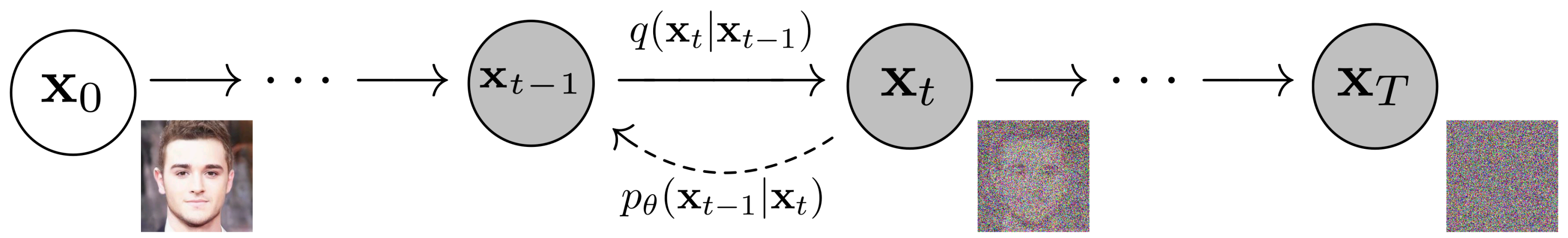
Diffusion Models (DDPM)

Phase Space

$x_0 \sim p(x_0)$

Latent Space

$x_T \sim p(x_T)(= \mathcal{N}(0,1))$



$$\mathscr{L}_{DDPM} = -\log p_\theta(x_0) \approx \frac{1}{2\sigma_t^2}\frac{\beta_t^2}{(1-\beta_t)\bar{\beta}_t}\,|\,\epsilon(t) - \epsilon_\theta(t)\,|^2$$

 :

- Reparametrization
- Estimation
- …

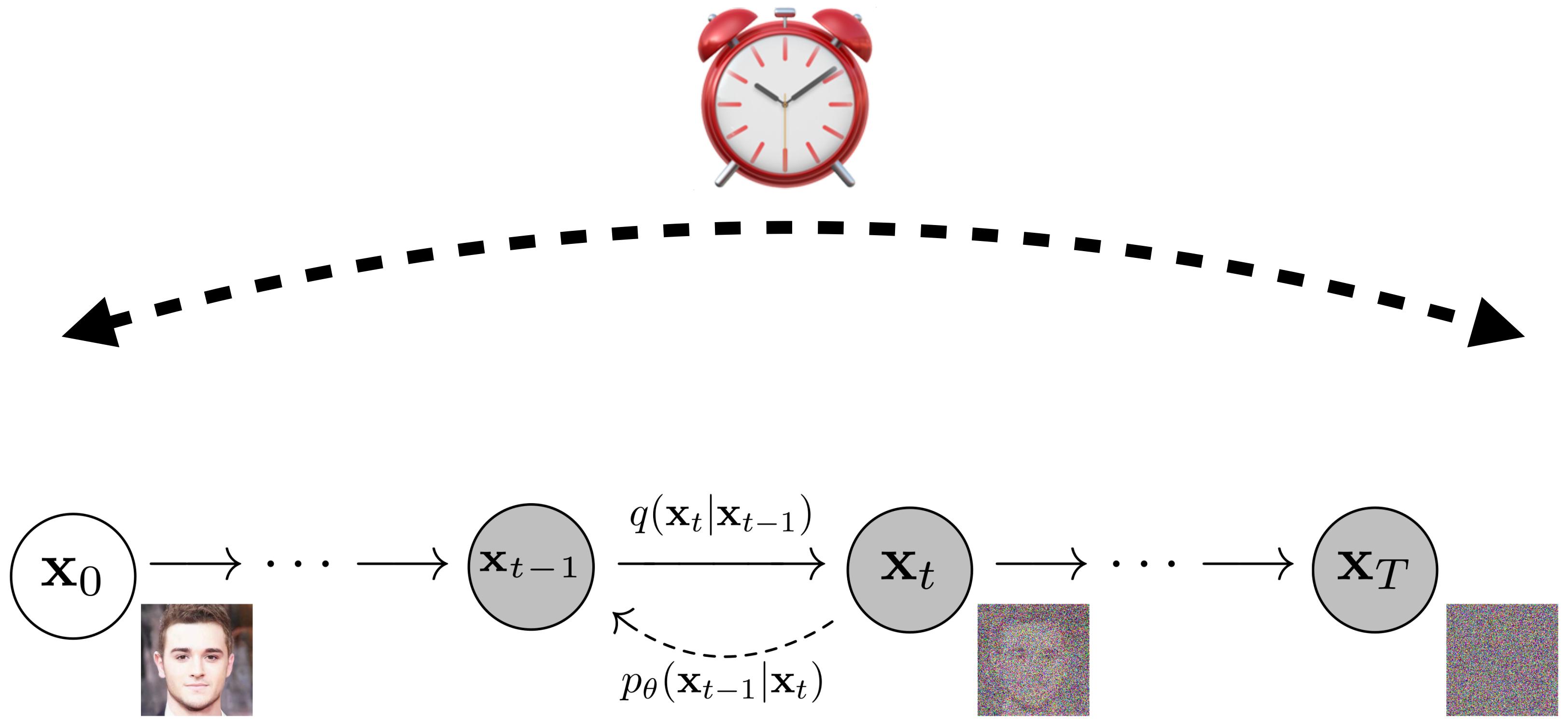
Predicted and actual noise added at time t

Figure from J.Ho et al.: arXiv:2006.11239

Diffusion Models (DDPM)

Phase Space

$x_0 \sim p(x_0)$



Latent Space

$x_T \sim p(x_T)(= \mathcal{N}(0,1))$

$$\mathscr{L}_{DDPM} = -\log p_\theta(x_0) \approx \frac{1}{2\sigma_t^2}\frac{\beta_t^2}{(1-\beta_t)\bar{\beta}_t}|\epsilon(t) - \epsilon_\theta(t)|^2$$

Denoising

Figure from J.Ho et al.: arXiv:2006.11239

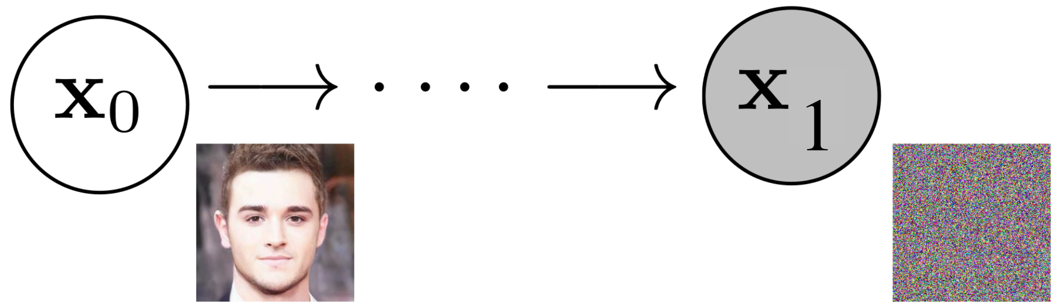
Diffusion Models (CFM)

Phase Space

$x_0 \sim p(x_0)$

Latent Space

$x_1 \sim p(x_1)(= \mathcal{N}(0,1))$



1. Connect x_0 with x_1 by a linear trajectory: $x(t\,|\,x_0) = (1-t)x_0 + tx_1$

2. Derive: $\frac{d}{dt}x(t\,|\,x_0) = (x_1 - x_0) \equiv v_\theta(x_1, x_0)$

3. Solve for x_0 (ODE): $x_0 = x_t - \int_0^1 v_\theta(x(t), t)dt$

Figure from J.Ho et al.: arXiv:2006.11239

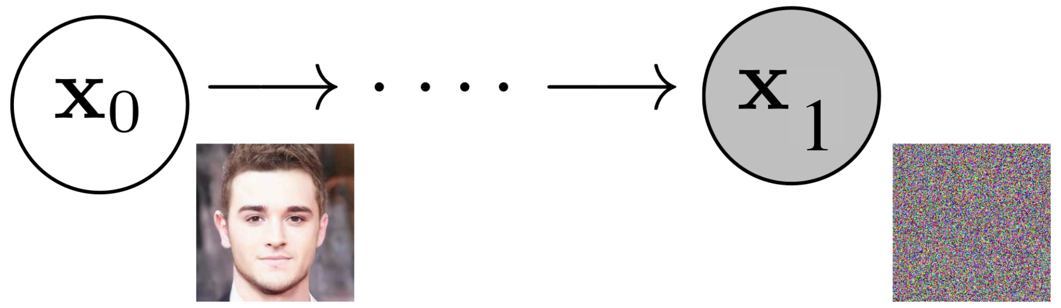
Diffusion Models (CFM)

Phase Space

$x_0 \sim p(x_0)$

Latent Space

$x_1 \sim p(x_1)(=\mathcal{N}(0,1))$



1. Connect x_0 with x_1 by a linear trajectory: $x(t\,|\,x_0) = (1-t)x_0 + tx_1$

2. Derive: $\frac{d}{dt}x(t\,|\,x_0) = (x_1 - x_0) \equiv v_\theta(x_1, x_0)$

3. Solve for x_0 (ODE): $x_0 = x_t - \int_0^1 v_\theta(x(t), t)\,dt$

Figure from J.Ho et al.: arXiv:2006.11239

How to Bayesianize

1. Replace each linear layer with a *Bayesian* layer

2. Add additional regularisation term to loss