Machine Learning for LHC Theory

ACAT 2021

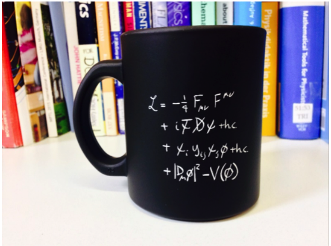
-

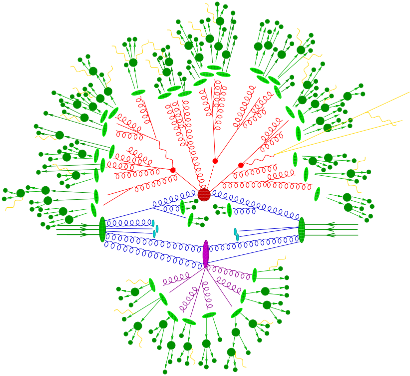
AI Decoded: Towards Sustainable, Diverse, Performant and Effective Scientific Computing

Anja Butter

ITP, Universität Heidelberg

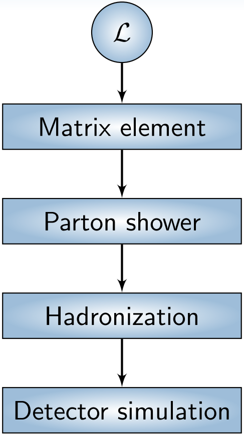
First principle based precision simulations

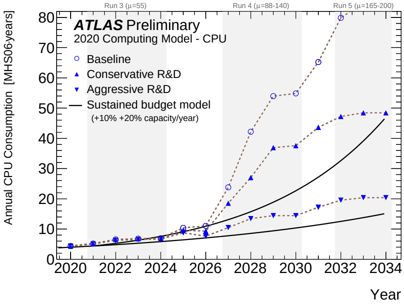




Unique advantage of our field!

Precision simulations with limited resources





Fast evaluation → higher statistics = higher order → **Precision**

Boosting standard event generation...

1. Generate phase space points

2. Calculate event weight

$$w_{event} = f(x_1, Q^2) f(x_2, Q^2) \times \mathcal{M}(x_1, x_2, p_1, \ldots p_n) \times J(p_i(r))^{-1}$$

3. Unweighting via $r > w/w_{\text{max}}$
   $\rightarrow$ optimal for $w \approx 1$

Boosting standard event generation...

Matrix element

$$w_{event} = f(x_1, Q^2) f(x_2, Q^2) \times \mathcal{M}(x_1, x_2, p_1, \ldots p_n) \times J(p_i(r))^{-1}$$

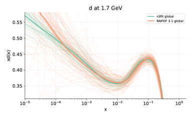
PDF

Phase space mapping

Boosting standard event generation...

Matrix element

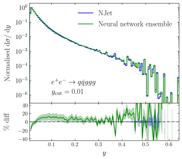
$$w_{event} = f(x_1, Q^2) f(x_2, Q^2) \times \mathcal{M}(x_1, x_2, p_1, \dots p_n) \times J(p_i(r))^{-1}$$



- NNPDF since 2002(!)
- S. Carrazza, J. Cruz-Martinez [1907.05075]

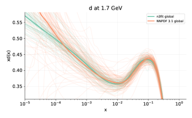
Phase space mapping

Boosting standard event generation...



- Amplitude estimation
- S. Badger, J. Bullock [2002.07516]
- J. Bendavid [1707.00028]

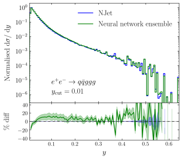
$$w_{event} = f(x_1, Q^2) f(x_2, Q^2) \times \mathcal{M}(x_1, x_2, p_1, \dots p_n) \times J(p_i(r))^{-1}$$



- NNPDF since 2002(!)
- S. Carrazza, J. Cruz-Martinez [1907.05075]

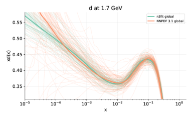
Phase space mapping

Boosting standard event generation...

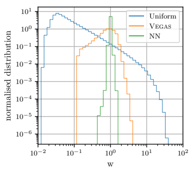


- Amplitude estimation
- S. Badger, J. Bullock [2002.07516]
- J. Bendavid [1707.00028]

$$w_{event} = f(x_1, Q^2) f(x_2, Q^2) \times \mathcal{M}(x_1, x_2, p_1, \dots p_n) \times J(p_i(r))^{-1}$$



- NNPDF since 2002(!)
- S. Carrazza, J. Cruz-Martinez [1907.05075]

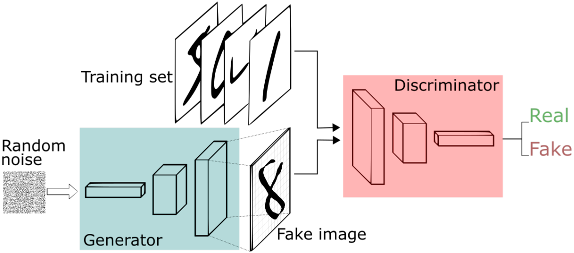


- **Learn phase space distribution**

Event simulation with generative models

1. Generative models for phase space sampling
   - **Control** over phase space density
2. Generative models for event generation
   - **Amplification** beyond training data?
   - Achieve high **precision**
   - Estimate **uncertainties**

Generative Adversarial Networks



Discriminator $[D(x_T) \to 1,\ D(x_G) \to 0]$

$$L_D = \Big\langle -\log D(x) \Big\rangle_{x \sim P_{Truth}} + \Big\langle -\log(1 - D(x)) \Big\rangle_{x \sim P_{Gen}} \to -2\log 0.5$$

Generator $[D(x_G) \to 1]$

$$L_G = \Big\langle -\log D(x) \Big\rangle_{x \sim P_{Gen}}$$

$\Rightarrow$ **New statistically independent samples**

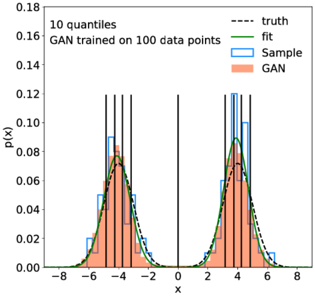
What is the statistical value of GANned events?

A.B., S. Diefenbacher, G. Kasieczka, B. Nachmann, T. Plehn, R. Winterhalder [2008.06545]

- Camel function
- Sample vs. GAN vs. 5 param.-fit

Evaluation on quantiles:

$$\text{MSE}^* = \sum_{j=1}^{N_\text{quant}} \left(p_j - \frac{1}{N_\text{quant}} \right)^2$$



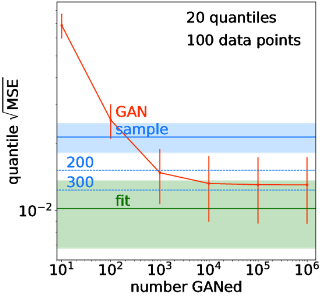
What is the statistical value of GANned events?

A.B., S. Diefenbacher, G. Kasieczka, B. Nachmann, T. Plehn, R. Winterhalder [2008.06545]

- Camel function
- Sample vs. GAN vs. 5 param.-fit

Evaluation on quantiles:

$$\text{MSE}^* = \sum_{j=1}^{N_\text{quant}} \left(p_j - \frac{1}{N_\text{quant}} \right)^2$$



→ Amplification factor 2.5

Sparser data → bigger amplification

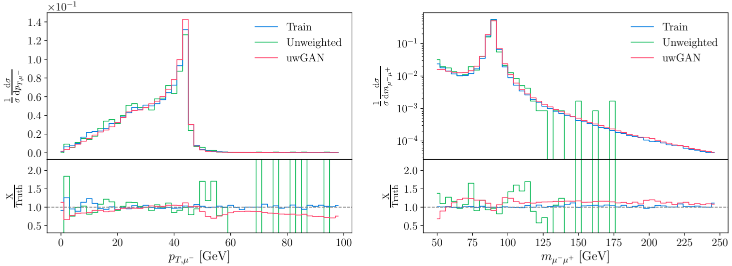
Training on weighted events

M. Backes, AB, T. Plehn, R.Winterhalder [2012.07873]

Low unweighting efficiencies → bottleneck before training

→ Train on weighted events

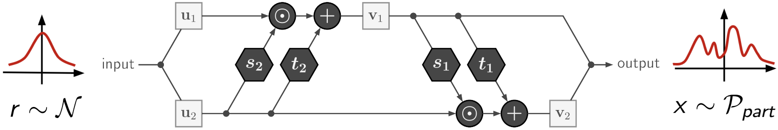
$$\rightarrow L_D = \big\langle -w \log D(x) \big\rangle_{x\sim P_{Truth}} + \big\langle -\log(1-D(x)) \big\rangle_{x\sim P_{Gen}}$$



Populates high energy tails

Large amplification wrt. unweighted data!

Better control with invertible networks



+ Tractable Jacobian

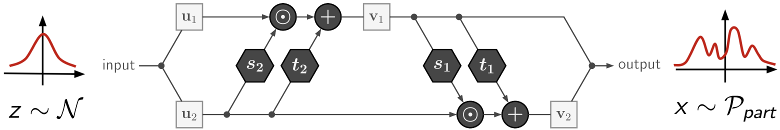
+ Enable correction for perfect precision

+ Fast evaluation in both directions

$$\begin{pmatrix} v_1 \\ v_2 \end{pmatrix} = \begin{pmatrix} u_1 \cdot s_2(u_2) + t_2(u_2) \\ u_2 \end{pmatrix}$$

Training on density

Sherpa [2001.05478, 2001.10028]



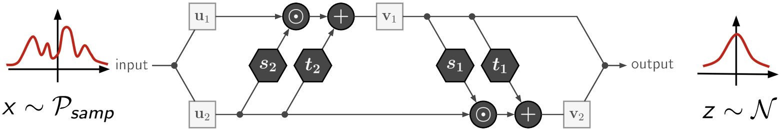
- $z \sim \mathcal{N} \rightarrow$ NN $\rightarrow x \sim p_x$
- $p_x(x) = p_z(z) \cdot J_{\text{NN}}$
- Given target density $t(x)$

$\rightarrow$ Train NN to minimize $\log(p_z(z) \cdot J_{\text{NN}}/t(x))$

- Problem: Calculate $f(x)$ each time

Training on samples

A.B., T. Heimel, S. Hummerich, T. Krebs, T. Plehn, A. Rousselot, S. Vent [arXiv:2110.13632]



- $x \sim p_{\text{samples}} \to$ NN $\to z$

$\to$ Train NN to ensure $z \sim \mathcal{N}$

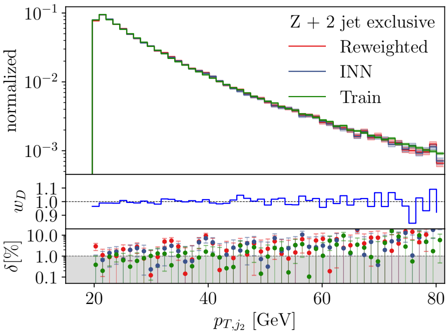
- Loss: Maximize posterior over network weights:

$$
\begin{aligned}
-\log(p(\theta|x)) &= -\log(p(x|\theta)) - \log(p(\theta)) + \text{const.} \\
&= -\log(p(z|\theta)) - \log(J) - \log(p(\theta)) + \text{const.}
\end{aligned}
$$

Naive INN results

Inclusive Z+jets production

- INN easy to train
- Powerful baseline

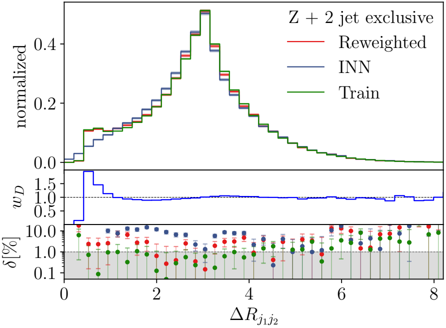


Naive INN results

Inclusive Z+jets production

- INN easy to train
- Powerful baseline

- Challenges:
  - Topological holes
  - Sharp phase space features



How to deal with deviations?

I. Corrections through reweighting

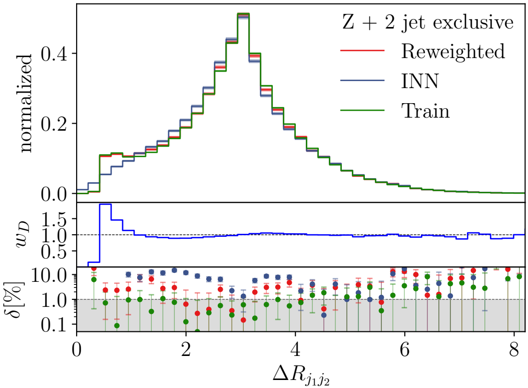
Discriminator

$$\mathcal{L} = -\sum_{x\sim p_{data}} \log(D(x)) - \sum_{x\sim p_{INN}} \log(1 - D(x))$$
$$= -\int \mathrm{d}x \; p_{data}(x)\log(D(x)) + p_{inn}(x)\log(1 - D(x))$$

From variation we obtain

$$0 = \frac{p_{data}(x)}{D(x)} - \frac{p_{inn}(x)}{1 - D(x)}$$
$$\Rightarrow \frac{p_{data}(x)}{p_{inn}(x)} = \frac{D(x)}{1 - D(x)}$$

Reweighting the generated distributions

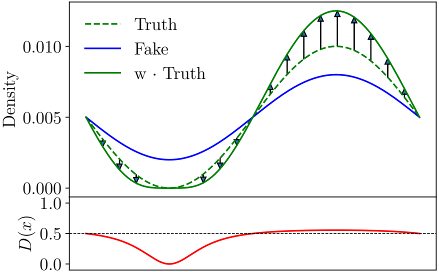


+ Close to perfect distribution after reweighting

- Yields weighted events

II. Discriminator improved training

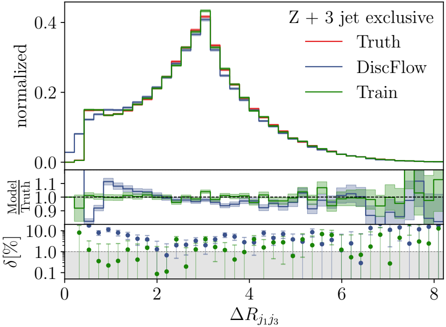
- Include discriminator information to improve training
- Discflow



$$\mathcal{L}_{\text{DiscFlow}} = \sum_{i=1}^{B} w_D(x_i)^\alpha \left(\frac{\psi(x_i; c_i)^2}{2} - \log J(x_i) \right)$$
$$\approx \int dx \; \underbrace{w_D(x)^\alpha P(x)}_{\text{reweighted truth}} \left(\frac{\psi(x; c)^2}{2} - \log J(x) \right)$$

II. Discriminator improved training

- Include discriminator information to improve training
- Discflow

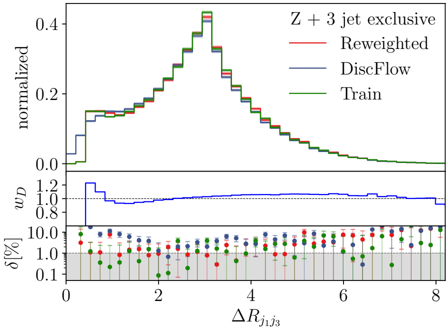


$$\mathcal{L}_{\text{DiscFlow}} = \sum_{i=1}^{B} w_D(x_i)^\alpha \left(\frac{\psi(x_i;c_i)^2}{2} - \log J(x_i) \right)$$

$$\approx \int dx \underbrace{w_D(x)^\alpha P(x)}_{\text{reweighted truth}} \left(\frac{\psi(x;c)^2}{2} - \log J(x) \right)$$

II. Discriminator improved training

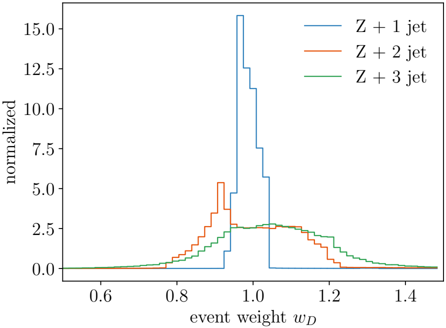
- Include discriminator information to improve training
- Discflow + Reweighting



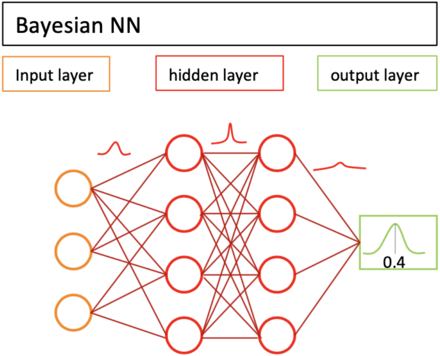
$$\mathcal{L}_{\text{DiscFlow}} = \sum_{i=1}^{B} w_D(x_i)^\alpha \left(\frac{\psi(x_i; c_i)^2}{2} - \log J(x_i) \right)$$

$$\approx \int dx \; \underbrace{w_D(x)^\alpha P(x)}_{\text{reweighted truth}} \left(\frac{\psi(x; c)^2}{2} - \log J(x) \right)$$

Weight distribution after DiscFlow+Reweighting

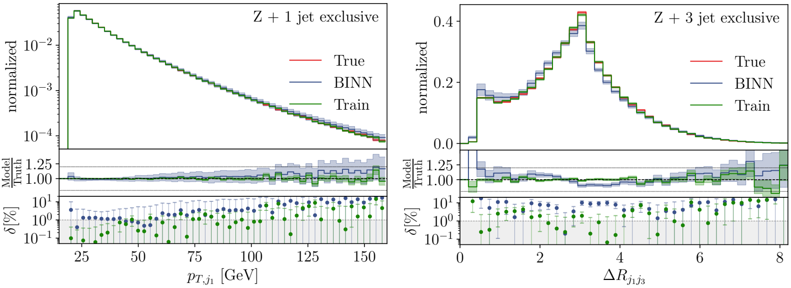


III. Addressing uncertainties



$$\mathcal{L} = \mathcal{L}_{INN} + KL_{prior}$$

BINN results

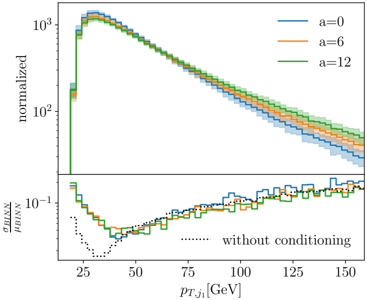


⇒ BINN uncertainty captures convergence of the network ✓

⇒ BINN uncertainty does NOT capture where network fails

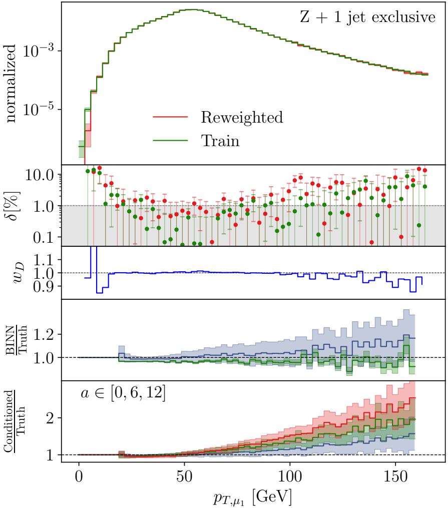
IV. Including external uncertainties through conditioning

$$w = 1 + a \left(\frac{p_{T,j_1} - 15\ \text{GeV}}{100\ \text{GeV}} \right)^2$$

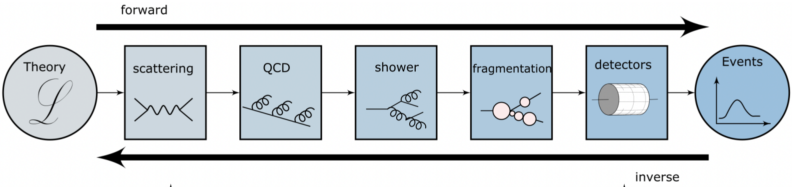


$\rightarrow$ Include prior over α in BINN sampling

Overview on uncertainties



Can we invert the simulation chain?



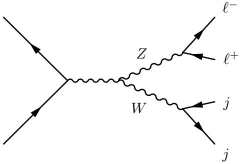
What we
want to know

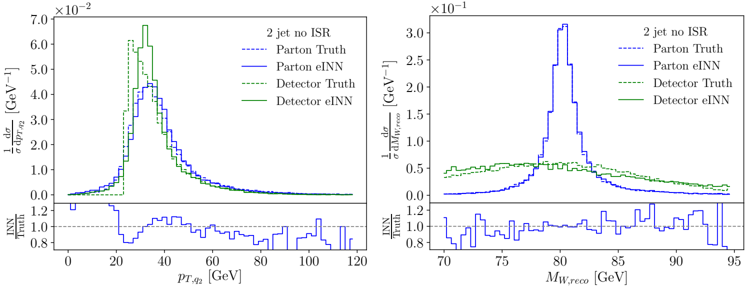
What we
measure or simulate

wish list:
- ☐ multi-dimensional
- ☐ bin independent
- ☐ statistically well defined

Inverting detector effects

$$\begin{pmatrix} x_{part} \\ r_{part} \end{pmatrix} \xleftrightarrow[\leftarrow \text{ unfolding}:\bar{g}]{\text{PYTHIA,DELPHES}:g\rightarrow} \begin{pmatrix} x_{det} \\ r_{det} \end{pmatrix}$$





multi-dimensional ✓ bin independent ✓ statistically well defined ?

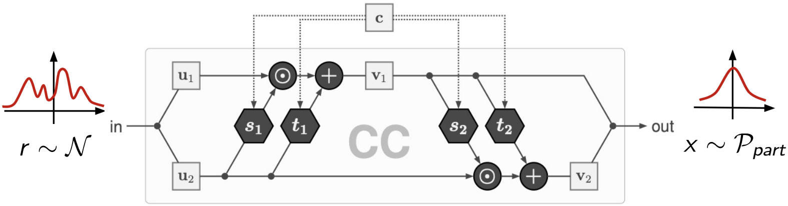
Asking the right question

Given an event x_d, what is the probability distribution at parton level?
$\rightarrow$ event generation conditioned on x_d

$$x_p \xleftrightarrow[\leftarrow \text{ unfolding: } \bar{g}(r, f(x_d))]{g(x_p, f(x_d)) \rightarrow} r$$

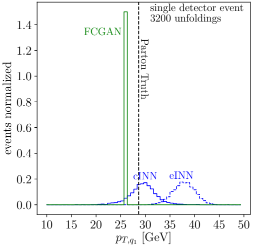
Minimizing the posterior

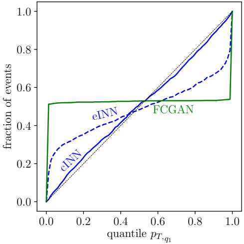
$$L = \left\langle 0.5||\bar{g}(x_p, f(x_d))||_2^2 - \log|J| \right\rangle_{x_p \sim P_p, x_d \sim P_d} - \log p(\theta)$$



Condition INN on detector data [2006.06685]

$$x_p \xleftrightarrow[\leftarrow \text{ unfolding: } \bar{g}(r, f(x_d))]{g(x_p, f(x_d)) \rightarrow} r$$

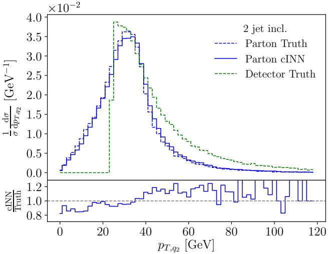




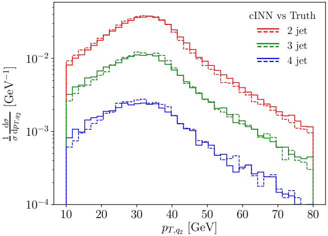
Inverting the full event

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

Train on inclusive dataset



Evaluate
exclusive 2/3/4 jet channels



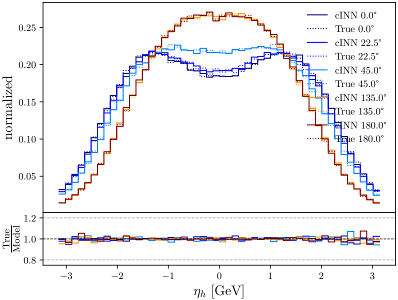
multi-dimensional ✓ bin independent ✓ statistically well defined ✓

Application to MEM

current work in progress with T. Martini, T. Heimel, S. Peitzsch, T. Plehn

- Single top production in association with Higgs
- Measure CP-phase in the top Yukawa coupling

$$\mathcal{L}(\alpha) = \prod_{i=1}^{N} \frac{1}{\sigma_{\text{fid}}(\alpha)} \int d^m z \; \frac{d^m\sigma(\alpha)}{dz_1 \ldots dz_m} \; T(\vec{y}^{(i)}, \vec{z}) \; .$$



We can use ML ...

... to improve precision simulations in forward direction

... to **amplify** underlying statistics

... to achieve **precision** with discriminators

... to estimate the corresponding **uncertainties**

... to **invert** the simulation chain statistically