Two Invertible Networks for the Matrix Element Method

Theo Heimel
January 2023

Institut für theoretische Physik
Universität Heidelberg

arXiv:2210.00019
Butter, Heimel, Martini, Peitzsch, Plehn

How can we find new physics at the LHC?
Maybe it is hidden in rare processes

$\Downarrow$

Need better analysis techniques!

Traditional analysis

- Hand-crafted observables
- Binned data

$\Downarrow$

Only fraction of information used

Matrix element method

- Based on first principles
- Estimates uncertainties reliably
- Optimal use of information

$\Downarrow$

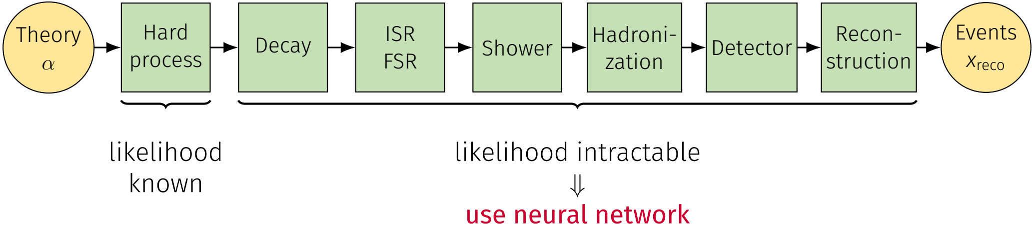
Perfect for processes with few events

Matrix Element Method

- Process with theory parameter α, hard-scattering momenta x_{hard}
- Likelihood at hard-scattering level given by differential cross section

$$p(x_{\text{hard}}|\alpha) = \frac{1}{\sigma(\alpha)} \frac{\mathrm{d}\sigma(\alpha)}{\mathrm{d}x_{\text{hard}}}$$

- Neyman-Pearson lemma $\Longrightarrow$ optimal use of information
- Differential cross section only known analytically at hard-scattering level



Two Invertible Networks for the Matrix Element Method

Normalizing flows

- Random number generators sample from uniform distribution $r \sim u(r)$
- Want to sample from arbitrary distribution $p(x)$
  $\rightarrow$ need function $x = f(r)$ to transform $r \sim u(r)$ to $x \sim p(x)$
- Analytic form of f only known for simple distributions (e.g. Gaussian)
  $\rightarrow$ classical solutions: importance/rejection sampling, VEGAS, ...
- Alternative: Chain of invertible mappings with change of variables formula

$$z_n = f(z_1) = f_{n-1}(\dots f_2(f_1(z_1))\dots)$$

$$p(z_n) = p(z_1)\left|\det \frac{\partial z_1}{\partial z_n}\right| = p(z_1)\prod_{i=2}^{n}\left|\det \frac{\partial z_{i-1}}{\partial z_i}\right|$$

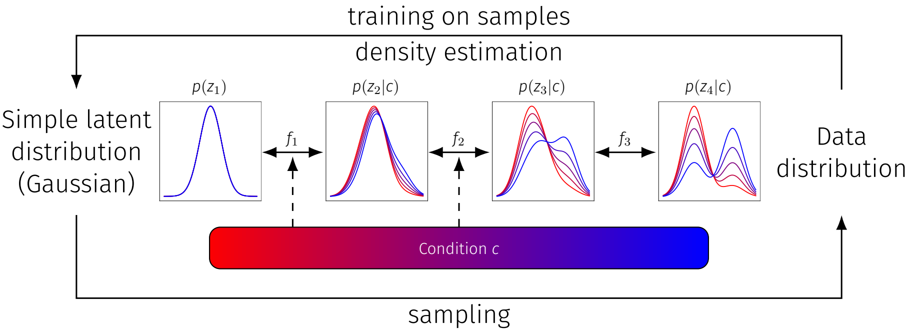
- Add parameter to express conditional distributions

$$z_{i+1} = f_i(z_i; c) \qquad \text{and} \qquad z_i = \bar{f}_i(z_{i+1}; c)$$

Conditional Invertible Neural Networks

- chain of learnable, invertible transformations with tractable Jacobian
  [Ardizzone et al., 1907.02392]
- Train network by maximizing log-likelihood for training dataset

$$\log p(z_n) = \log p(z_1) + \log \det \frac{\partial z_1(z_n; c)}{\partial z_n}$$



Coupling blocks

- Requirements for transformations: Invertible, tractable jacobian, expressive, allow correlations
- Coupling blocks with rotation or permutation R, coupling transformation C
- Simplest: Affine coupling block

  [Dinh et al., 1410.8516]

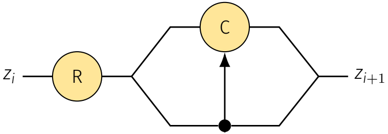
  s, t: fully-connected sub-networks

  (u_1, u_2): input vector split in two

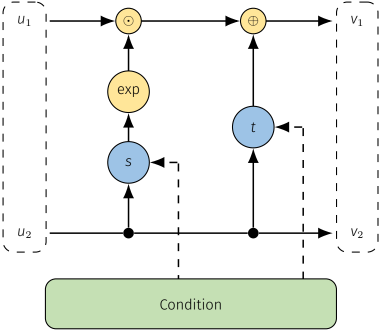
  (v_1, v_2): output vector split in two

  $\rightarrow$ s, t don't have to invertible

  $\rightarrow$ triangular jacobian



Generic coupling block

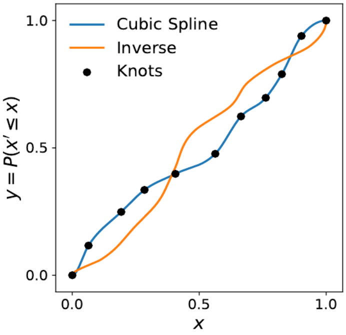


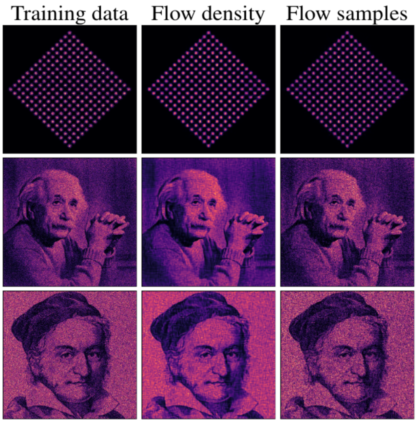
Affine coupling block

Spline coupling blocks

- Disadvantage: Affine transformations are not very expressive
- Better: Spline coupling blocks
  $\rightarrow$ monotonic splines between points given by sub-networks

[Durkan et al., 1906.02145] [Durkan et al., 1906.04032]



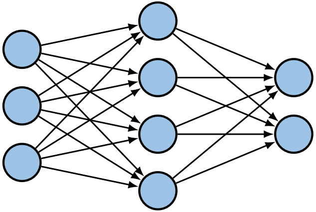


Can learn complex 2D distributions with only two coupling blocks!

Applications in particle physics

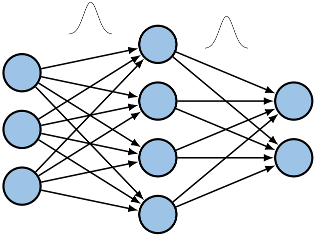
- (c)INNs learn and sample from (conditional) probability distributions
- Useful in physics for
  $\rightarrow$ getting access to otherwise intractable probability distributions
  $\rightarrow$ making sampling more computationally efficient
- Applications include
  $\rightarrow$ event generation [Butter et al., 2110.13632] [Verheyen, 2205.01697]
  $\rightarrow$ importance sampling [Gao et al., 2001.05486] [Heimel et al., 2212.06172]
  $\rightarrow$ detector simulation [Krause, Shih, 2106.05285]
  $\rightarrow$ unfolding [Bellagente et al., 2006.06685]
  $\rightarrow$ Bayesian inference [Butter et al., 2012.09873]
  $\rightarrow$ kinematic reconstruction [Leigh et al., 2207.00664]

Flows with uncertainties



deterministic weights w_i

$\downarrow$



weights $w_i \sim \mathcal{N}(\mu_i, \sigma_i)$

- Quantify training uncertainty with **Bayesian Invertible Neural Networks** (BINN)
  [MacCay, 1995] [Neal, 2012] [Bellagente et al., 2104.04543]
- Simple modification of deterministic network:
  $\rightarrow$ Replace deterministic weights with distribution
  $\rightarrow$ Additional term in loss function
- Extracting uncertainties:
  sample from weight distribution
- Use as generator $\rightarrow$ Histograms with error bars
- Use as density estimator $\rightarrow$ Error on density

Bayesian loss function

- Given a data set $\mathcal{D}$, we want to know (intractable) posterior $p(w|\mathcal{D})$
  $\rightarrow$ approximate with tractable $q_\phi(w)$ (e.g. q Gaussian, $w = (\mu, \sigma)$)
- Choose ϕ to minimize $\mathrm{KL}(q_\phi(w)\,|\,p(w|\mathcal{D}))$
- Rewrite posterior with Bayes' theorem: $p(w|\mathcal{D}) = \frac{p(\mathcal{D}|w)p(w)}{p(\mathcal{D})}$
- Take evidence lower bound (ELBO) for evidence $p(\mathcal{D})$ to get

$$\mathcal{L}_{\text{ELBO}} = \sum_{i=1}^{N} \Big\langle \log p(x_i|w) \Big\rangle_{w \sim q_\phi(w)} - \mathrm{KL}(q_\phi(w), p(w))$$

- Have to choose prior $p(w)$
  $\rightarrow$ sufficiently wide Gaussian for prior-independent results
- Get ensemble of networks by sampling from $q_\phi(w)$

Two Invertible Networks for the Matrix Element Method

MEM at reconstruction level

- Integrate out hard-scattering phase space

$$p(x_{\text{reco}}|\alpha) = \int \mathrm{d}x_{\text{hard}} \; \underbrace{p(x_{\text{hard}}|\alpha)}_{\text{diff. CS}} \; \underbrace{p(x_{\text{reco}}|x_{\text{hard}}, \alpha)}_{\text{estimate with network}}$$

- Need to learn probability distribution $p(x_{\text{reco}}|x_{\text{hard}}, \alpha)$
  In practice: ignore α-dependence and learn $p(x_{\text{reco}}|x_{\text{hard}})$
- Not known analytically $\rightarrow$ learn from data

Solution:
normalizing flow $\rightarrow$ Transfer-cINN

How to compute the integral?

- $|\mathcal{M}|^2$ spans several orders of magnitude
- Narrow distribution from Transfer-cINN

} **Integration challenging**

- Importance sampling with proposal distribution $q(x_{\text{hard}})$

$$p(x_{\text{reco}}|\alpha) = \left\langle \frac{1}{q(x_{\text{hard}})}\, p(x_{\text{hard}}|\alpha)\; p(x_{\text{reco}}|x_{\text{hard}}, \alpha) \right\rangle_{x_{\text{hard}} \sim q(x_{\text{hard}})}$$

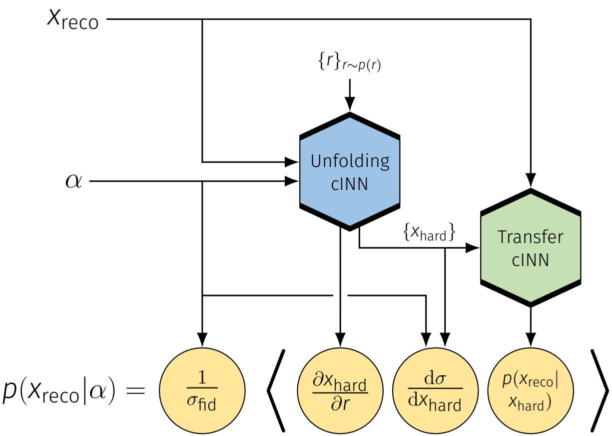
- Bayes' theorem: Integration becomes trivial if

$$x_{\text{hard}} \sim q(x_{\text{hard}}) = p(x_{\text{hard}}|x_{\text{reco}}, \alpha)$$

Solution:
normalizing flow $\rightarrow$ **Unfolding-cINN**

[Bellagente et al., 2006.06685]

Putting it all together



- Training data

$$(\alpha, x_{\text{hard}}, x_{\text{reco}})$$

- Transfer-cINN learns

$$p(x_{\text{reco}}|x_{\text{hard}})$$

$\rightarrow$ transfer function
$\rightarrow$ fast forward simulation

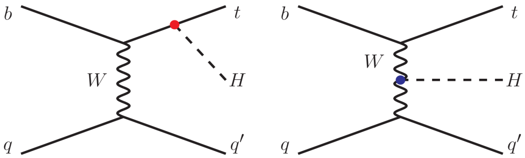
- Unfolding-cINN learns

$$p(x_{\text{hard}}|x_{\text{reco}}, \alpha)$$

$\rightarrow$ phase space sampling

Two Invertible Networks for the Matrix Element Method

LHC process



- Single Higgs production with anomalous non-CP-conserving Higgs coupling

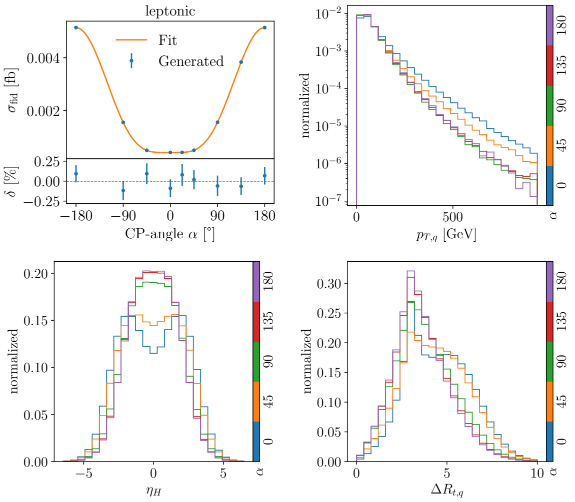
$$\mathcal{L}_{t\bar{t}H} = -\frac{y_t}{\sqrt{2}}\left[\cos\alpha\,\bar{t}t + \frac{2}{3}\mathrm{i}\sin\alpha\,\bar{t}\gamma_5 t\right]H \qquad \text{with } \textbf{CP-angle } \alpha$$

[Artoisenet et al, 1306.6464] [de Aquino, Mawatari, 1307.5607] [Demartin et al, 1504.00611]

- Decays $tHj \to (bW)\ (\gamma\gamma)\ j$. Test on different datasets

Leptonic decay	→	Hadronic decay	→	Hadronic decay + ISR
$(b\mu^+\nu_\mu)\ (\gamma\gamma)\ j$		$(bjj)\ (\gamma\gamma)\ j$		$(bjj)\ (\gamma\gamma)\ j$ + QCD jets
Challenge: reconstruct ν		Challenge: jet combinatorics		Challenge: no clear combinatorics

Why we need the MEM



Around the SM, $\alpha = 0°$:

low total cross section (few events)

$+$

low variation of rate

$+$

kinematic observables still sensitive

$\Downarrow$

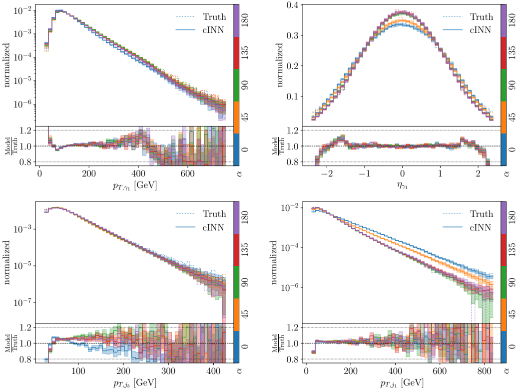
need kinematic observables
to use all available information

$\Downarrow$

ideal use case for MEM

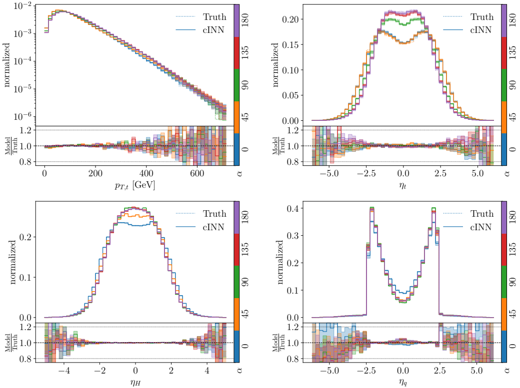
Two Invertible Networks for the Matrix Element Method

Testing the Transfer-cINN



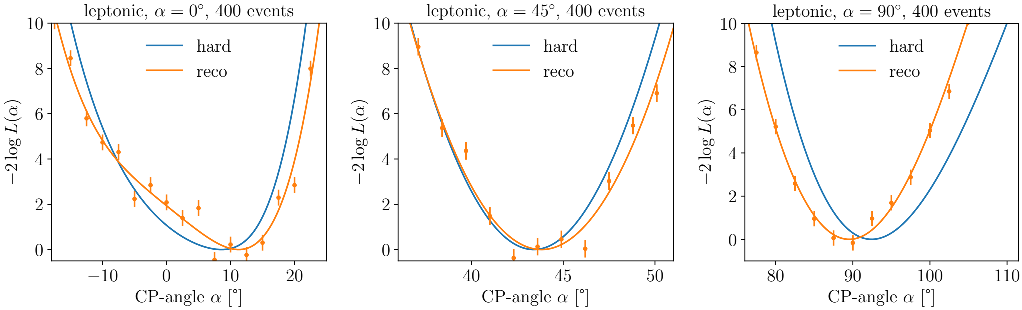
- To test performance:
  Transfer-cINN as forward simulator
- Test dataset: leptonic decay,
  $\alpha = 0°, 45°, 90°, 135°, 180°$
- Histograms at reco-level
- Error bars from Bayesian network
- Good agreement with Truth
- Within BINN errors in bulk
- α-independence valid assumption

Testing the Unfolding-cINN



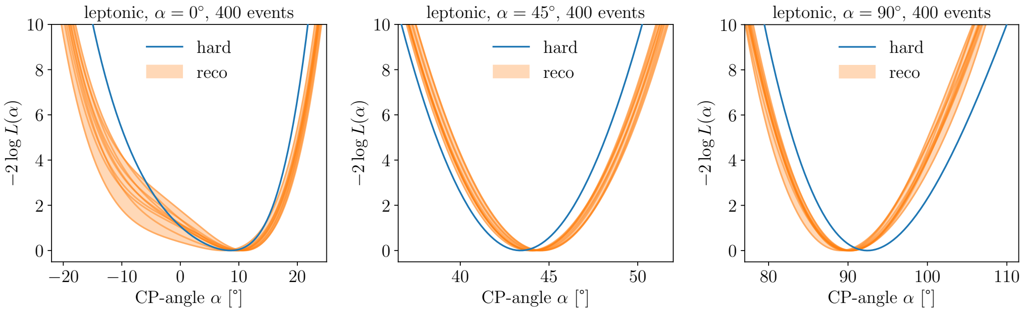
- Unfold each test event once
- Histograms at hard-scattering level
- Error bars from Bayesian network (deterministic Unfolding-cINN used for integration)
- Excellent agreement with Truth

Likelihoods for leptonic decay



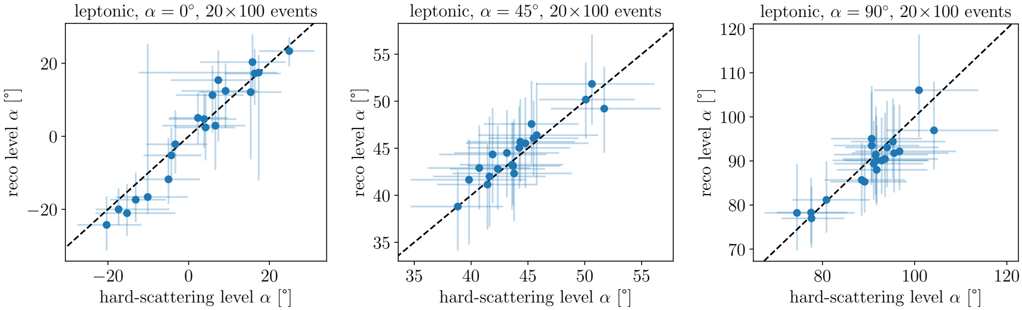
- Deterministic network, $\alpha = 0°, 45°, 90°$, 400 events each
- Extract likelihood for different α, sum events, fit polynomial (orange line)
- Compare to likelihood from hard-scattering data (blue line)
- **Good agreement between hard-scattering and reco-level**
  $\rightarrow$ But how large is the systematic uncertainty from training?

Training uncertainty from BINN



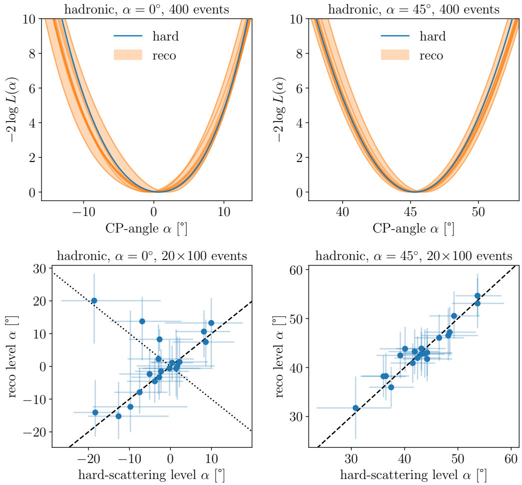
- Extract likelihood for 10 sampled networks
  $\rightarrow$ **estimate of systematic error from training**
- Most challenging around $\alpha = 0°$
  $\rightarrow$ larger statistical and Bayesian uncertainty
- Only uncertainty from finite training data
  $\rightarrow$ lack of expressivity not captured

Calibration checks



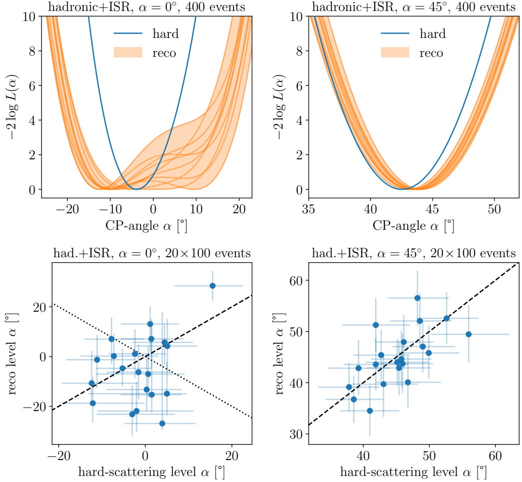
- Minimum and 68% confidence intervals for 20×100 events
- Good correlation betwen reco- and hard-scattering level
- Slight bias can be removed by calibration
- Lagrangian almost symmetric around $\alpha = 0°$
  $\rightarrow$ very asymmetric uncertainties in left panel

Dataset with hadronic decay



- Final state $(bjj)\ (\gamma\gamma)\ j$
  + additional jets from FSR
- Networks must resolve combinatorics
- Variable number of jets
  $\rightarrow$ Unfolding-cINN: zero-padded input
  $\rightarrow$ Transfer-cINN needs fixed dimension
- Almost symmetric around $\alpha = 0°$
  $\rightarrow$ sometimes wrong sign
- Nice correlation between reco- and parton-level

Hadronic decay with ISR



- Final state $(bjj)\ (\gamma\gamma)\ j$
  + additional jets from ISR and FSR
- Can't resolve between relevant jets and ISR jets during reconstruction
  $\rightarrow$ combinatorics more difficult
- Loss of sensitivity around $\alpha = 0°$
- Worse calibration, more bias
- Increased systematic uncertainty captured by Bayesian network

Two Invertible Networks for the Matrix Element Method

Outlook

- Measure fundamental Lagrangian parameters from small numbers of events
- Transfer-cINN: encode QCD and detector effects
- Unfolding-cINN: efficient integration over hard-scattering phase space
- Without ISR: close to hard-scattering truth
- With ISR: worse performance from more challenging combinatorics
- **Promising approach to use more expressive transfer functions without making the MEM computationally intractable**

- Next steps, ideas
  → Use information of additional jets in Transfer-cINN
  → Better handling of ISR
  → Better handling of jet combinatorics
  → Include NLO QCD corrections