ν^2-Flows

Fast and improved neutrino reconstruction in multi-neutrino final states with conditional normalizing flows

Matthew Leigh

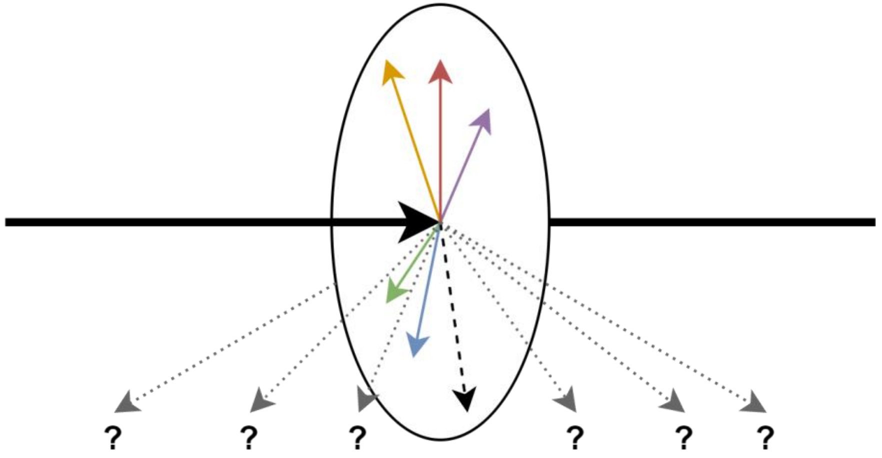
Johnny Raine   Knut Zoch   Tobias Golling

ATLAS EXPERIMENT

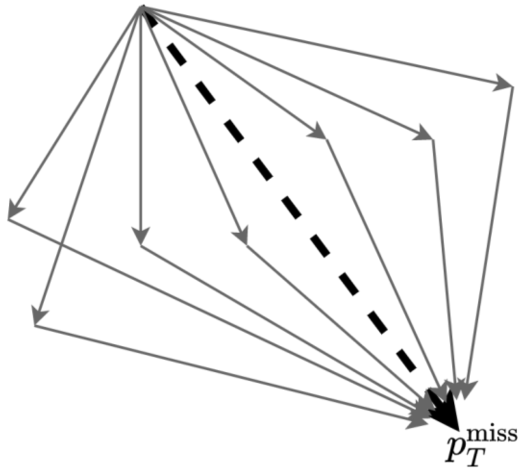
UNIVERSITÉ DE GENÈVE

Problem

- Total neutrino **transverse momenta** is often measured by the experimental proxy $\boldsymbol{p_T^{\text{miss}}}$
- **No information** about the longitudinal momentum
- **Transverse** momenta in final states with more than one neutrino are **under-constrained**



$$\boldsymbol{p_T^{\text{miss}}} = -\sum_{\text{i}\in\text{everything}} \boldsymbol{p_T^i}$$



Our Approach

- Can’t simply train a network like it’s a regression task
- There is missing information which can not be recovered
- Many solutions will still be possible
- But not equally likely

- We use a **conditional normalizing flow**
  to learn the conditional probability density of solutions for each event

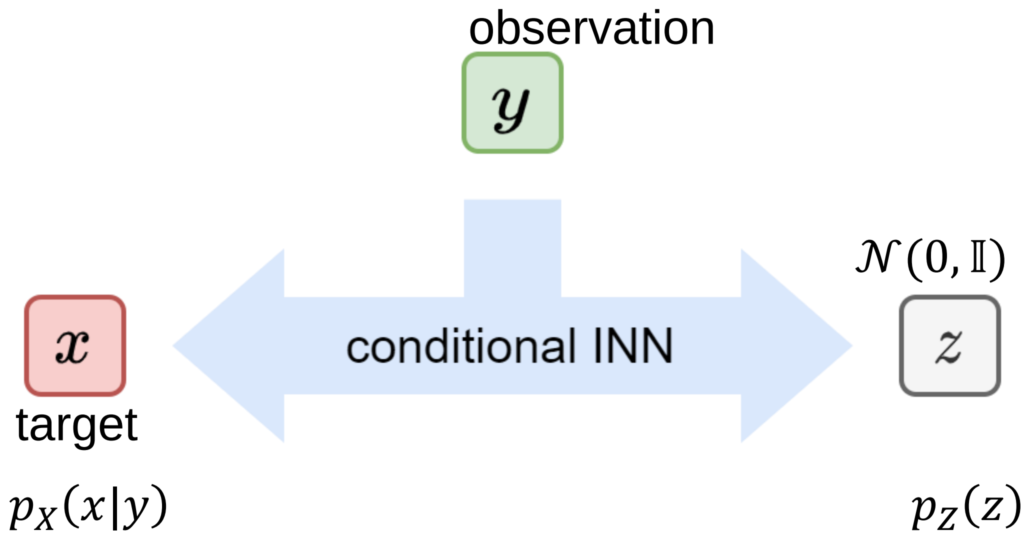
Background

- **Conditional normalising flows** parameterise an invertible map from $\boldsymbol{x}$ to $\boldsymbol{z}$ given $\boldsymbol{y}$ as **conditioning** inputs

- **Training:** Model runs forward for **maximum likelihood** objective

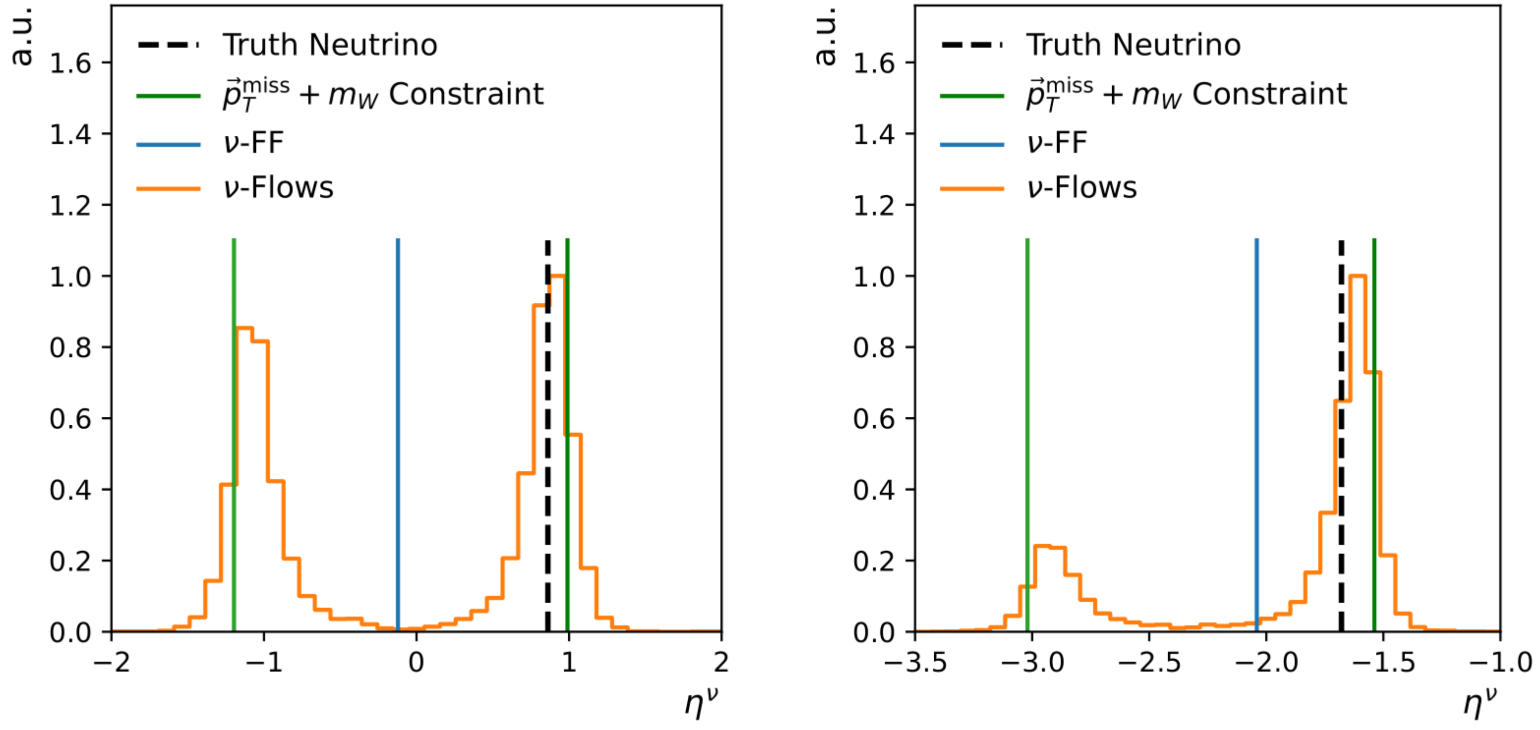
$$
\begin{aligned}
Loss(y, x) &= -\log(p_X(x|y)) \\
&= -\log\left(p_z\big(f_\theta(x|y)\big)\right) - \log\left|\det\big(J(x|y)\big)\right|
\end{aligned}
$$

- **Sampling:** Model runs in reverse giving $p(x|y)$



Previous work

- [Original paper]
- **Single leptonic $t\bar{t}$** decay – Only longitudinal momentum is unknown

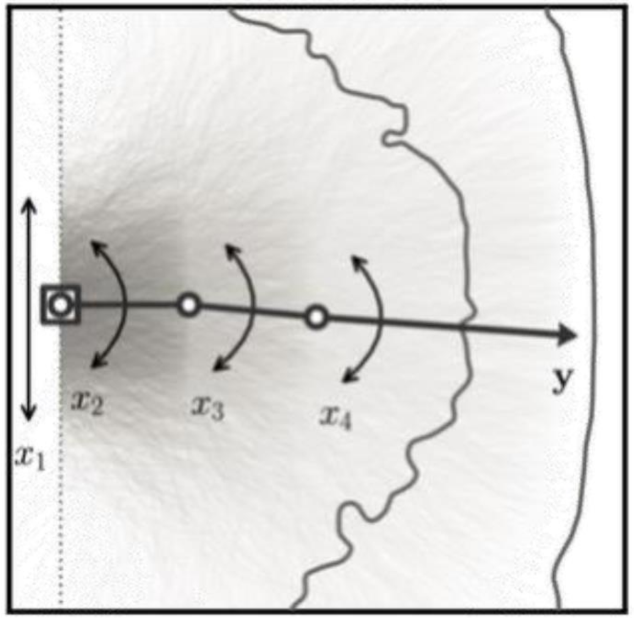


Non-linear Toy Example: Inverse Kinematics

Forward problem

- robot arm with 4 DOF: $x = [x_1, \dots, x_4]$
  - x_1: vertical position of first joint
  - x_2, x_3, x_4: joint angles
  - Gaussian priors: x_1 prefers the center
    x_2, x_3, x_4 prefer to be straight
- observation: hand position $y = [y_1, y_2]$
- geometric arm simulation $y = g(x)$ implicitly defines likelihood $p(y \mid x)$

prior distribution $p(x)$



Ardizzone et al. **"Analyzing inverse problems with invertible neural networks"**, ICLR 2019.

Universität Heidelberg

Visual Learning Lab – Heidelberg Collaboratory for Image Processing (HCI)

Non-linear Toy Example: Inverse Kinematics

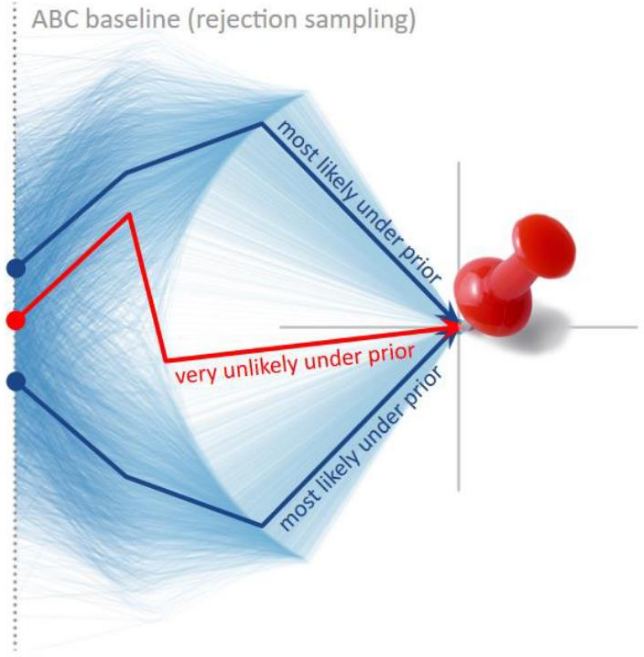
Forward problem

- robot arm with 4 DOF: $x = [x_1, \ldots, x_4]$
  - x_1: vertical position of first joint
  - x_2, x_3, x_4: joint angles
  - Gaussian priors: x_1 prefers the center
    x_2, x_3, x_4 prefer to be straight
- observation: hand position $y = [y_1, y_2]$
- geometric arm simulation $y = g(x)$ implicitly defines likelihood $p(y \mid x)$

INN infers posterior $p(x \mid \hat{y})$ for given hand position $\hat{y}$

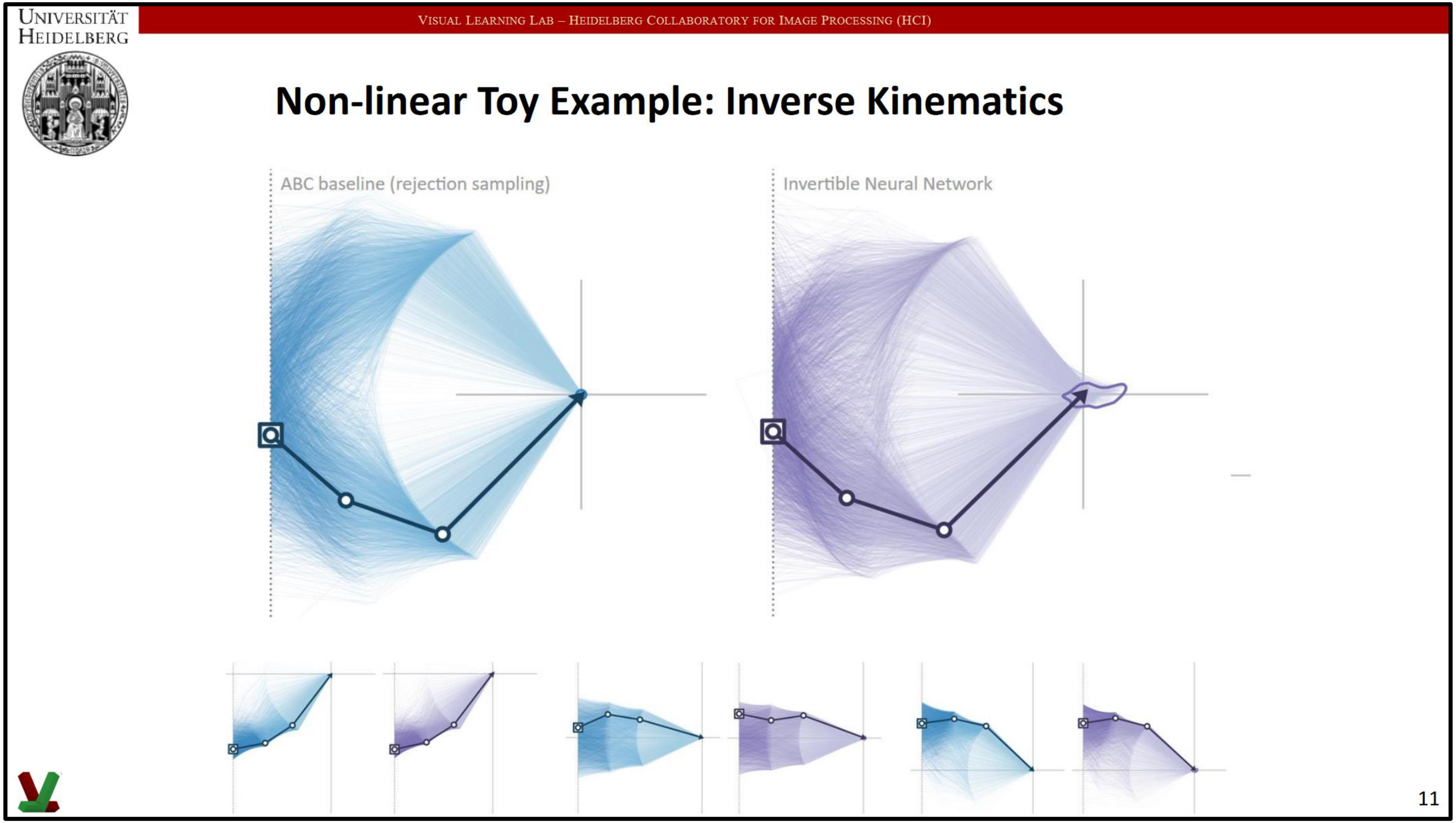
- For $t \in 1, \ldots, T$:
  Sample $z^{(t)} \sim \mathcal{N}(0, \mathbb{I})$ and compute $x^{(t)} = f_\theta(\hat{y}, z^{(t)})$

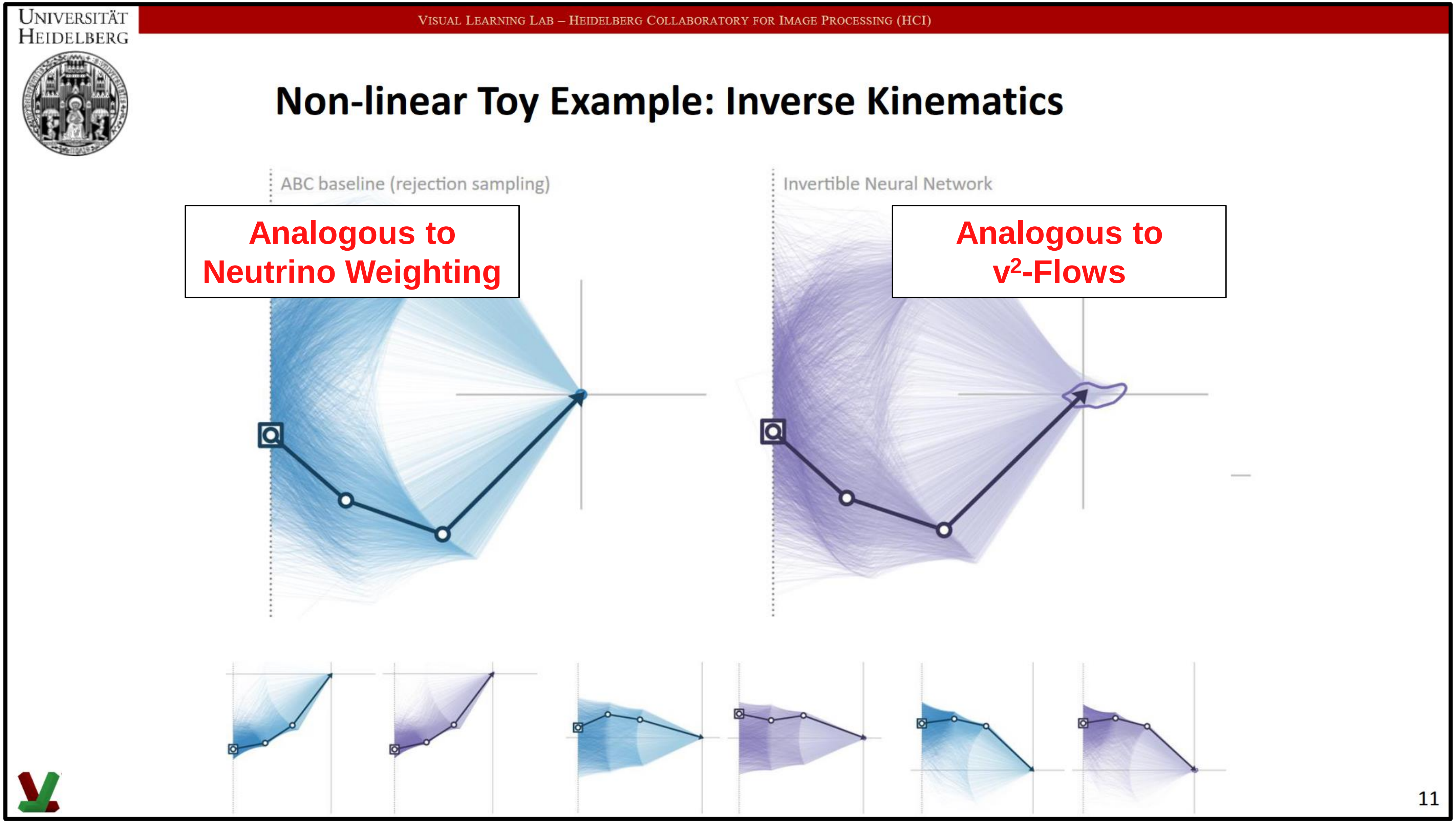
posterior distribution $p^*(x|\hat{y})$



Ardizzone et al. **"Analyzing inverse problems with invertible neural networks"**, ICLR 2019.

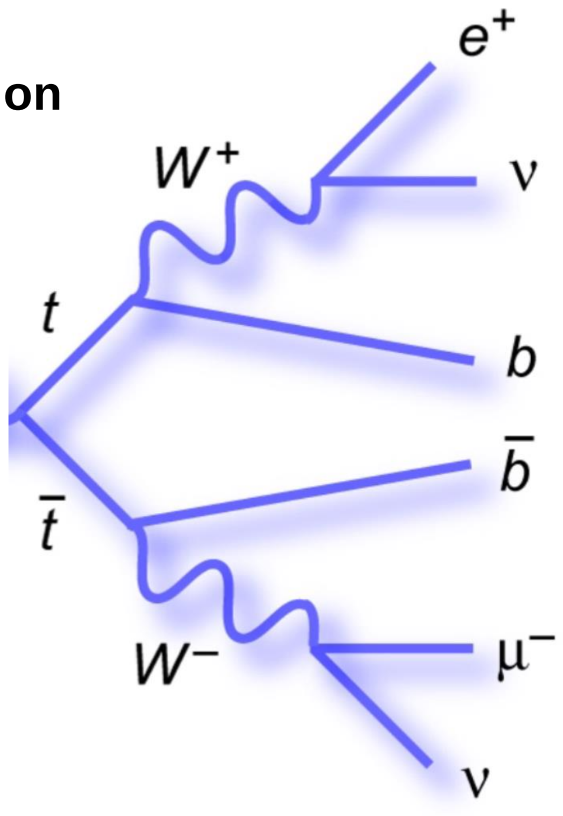
Ullrich Köthe - Heidelberg University - CERN IML workshop - October 2020




Universität Heidelberg
Visual Learning Lab – Heidelberg Collaboratory for Image Processing (HCI)
Non-linear Toy Example: Inverse Kinematics
ABC baseline (rejection sampling)
Invertible Neural Network
Analogous to Neutrino Weighting
Analogous to v²-Flows
11

v^2-Flows Overview

- Goal was to extend our work to **multiple neutrinos** in the final state
- **Dilepton $t\bar{t}$ events** – two neutrinos
- Trained a flow to **generate neutrino candidates** given **event level information**
- Compared to **two reference** methods:
  - Neutrino Weighting
  - Ellipse method



- [ArXiv:2307.02405]
- [Project GitHub]
- [Zenodo Dataset]

Reference Methods

1. Neutrino Weighting: Reference ATLAS analysis

$$(\ell_{1,2} + \nu_{1,2})^2 = m_w^2 = (80.38\,\text{GeV})^2,$$

$$(\ell_{1,2} + \nu_{1,2} + b_{1,2})^2 = m_t^2 = (172.5\,\text{GeV})^2,$$

- Make assumptions on W and top mass
- Iterate through multiple event configurations
  - η^{ν} and $\eta^{\bar{\nu}}$ individually stepped through
  - Jet/lepton association
  - Jet smearing
  - etc
- Solve the mass constraint equations yielding p_{T}^{ν} and $p_{\mathrm{T}}^{\bar{\nu}}$
- Keep solution that yields highest weight

$$w = \exp\left(- \frac{||\vec{p}_{\mathrm{T}}^{\,\text{miss}} - \vec{p}_{\mathrm{T}}^{\,\nu\bar{\nu}}||_2^2}{2\sigma^2} \right)$$

Reference Methods

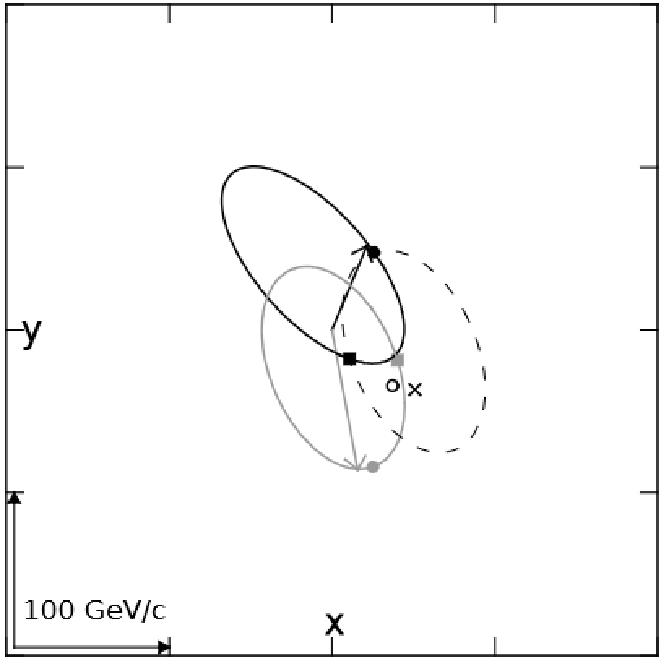
1. Neutrino Weighting: Reference ATLAS analysis

- Make assumptions on W and top mass → **Biases results**
- Iterate through multiple event configurations → **Computationally expensive**
  - η^{ν} and $\eta^{\bar{\nu}}$ individually stepped through → **Discretizes solutions**
  - Jet/lepton association
  - Jet smearing
  - etc
- Solve the mass constraint equations yielding p_{T}^{ν} and $p_{\mathrm{T}}^{\bar{\nu}}$ → **Fails to find solution in approximately 5% of events**
- Keep solution that yields highest weight

Reference Methods

2. Ellipse Method: Betchart et. al.

- Geometric approach to analytically constrain neutrino kinematics
- Requires assumption on W and top mass
- Requires assumption of jet/lepton association
- Solution set for single neutrino defines the surface of an ellipse



Reference Methods

2. Ellipse Method: Betchart et. al.

- Geometric approach to analytically constrain neutrino kinematics
- Requires assumption on W and top mass
- Requires assumption of jet/lepton association
- Solution set for single neutrino defines the surface of an ellipse

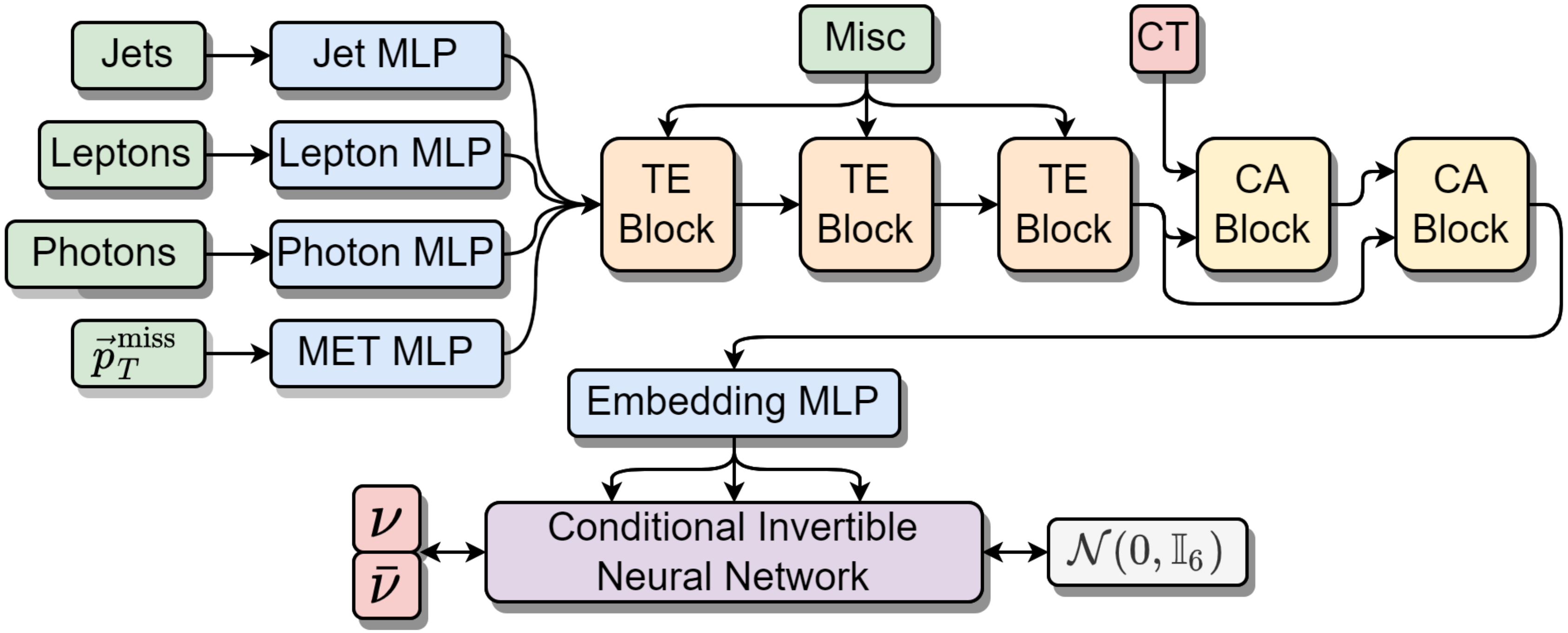
Biases results

Dependent on association method

Fails to find solution in approximately 20% of events

v^2-Flows Overview

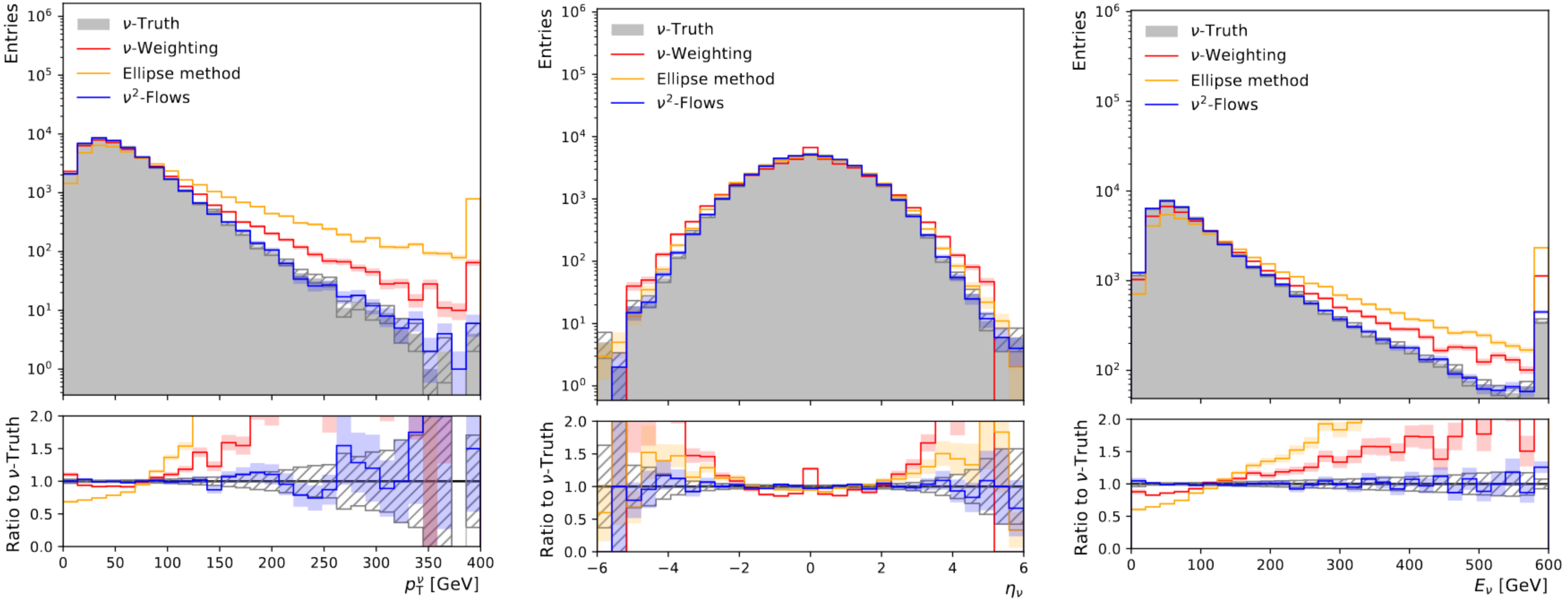
- Feature extraction is performed using a **transformer** with **class attention**
- Each event is represented as a **point cloud** - can scale to any input multiplicity



Results - Kinematics

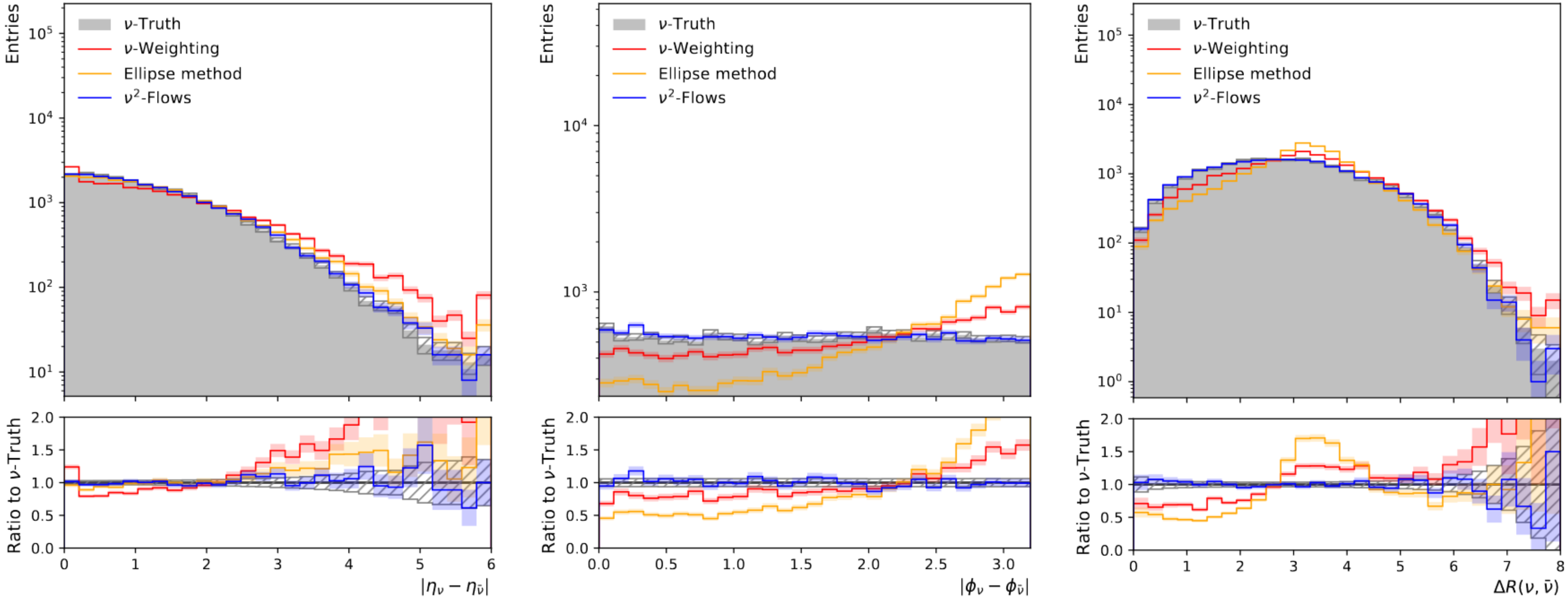
Kinematics

- Neutrino kinematics



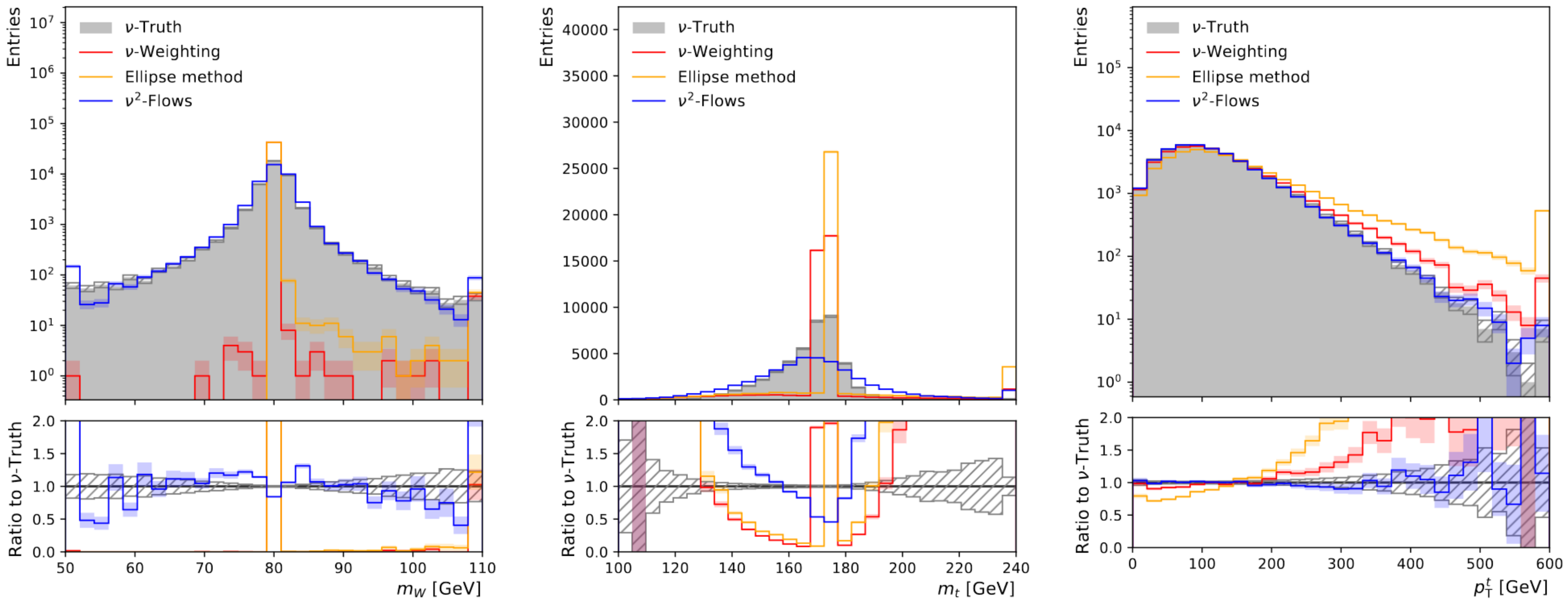
Kinematics

- Neutrino/anti-neutrino separation



Kinematics

- W and top kinematics



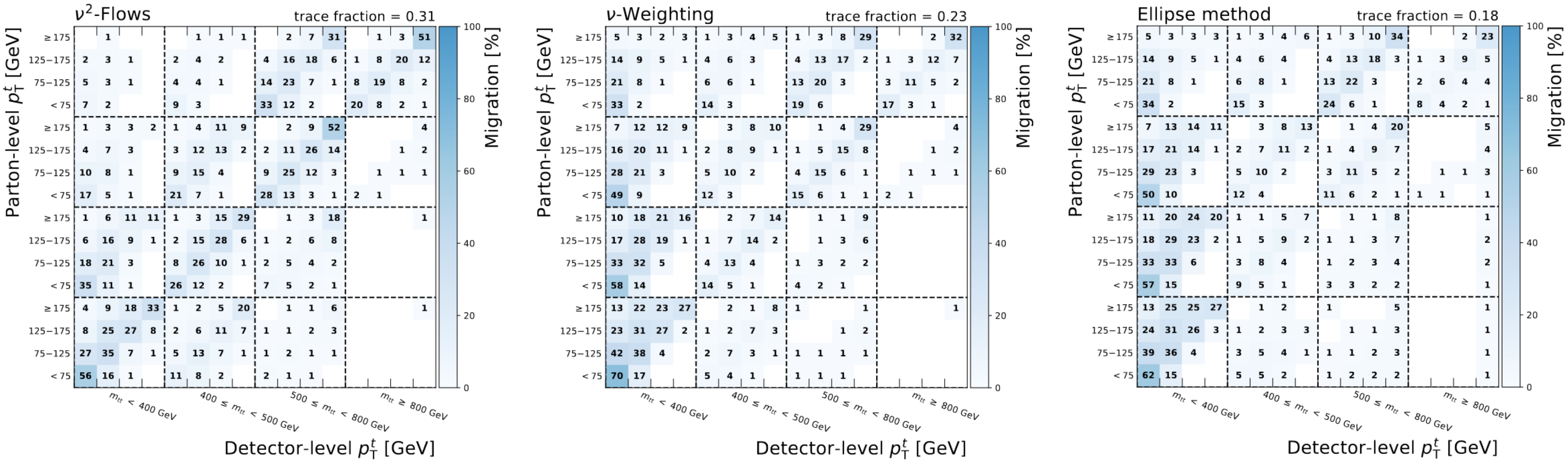
Results - Unfolding

Unfolding

- Evaluate a typical downstream impact of improved neutrino reconstruction
- Performed unfolding following a reference ATLAS analysis
- Double differential measurement of $m_{t\bar{t}}$ and p_{T}^{t}
- Used v^2-Flows and the reference methods to create neutrino candidates

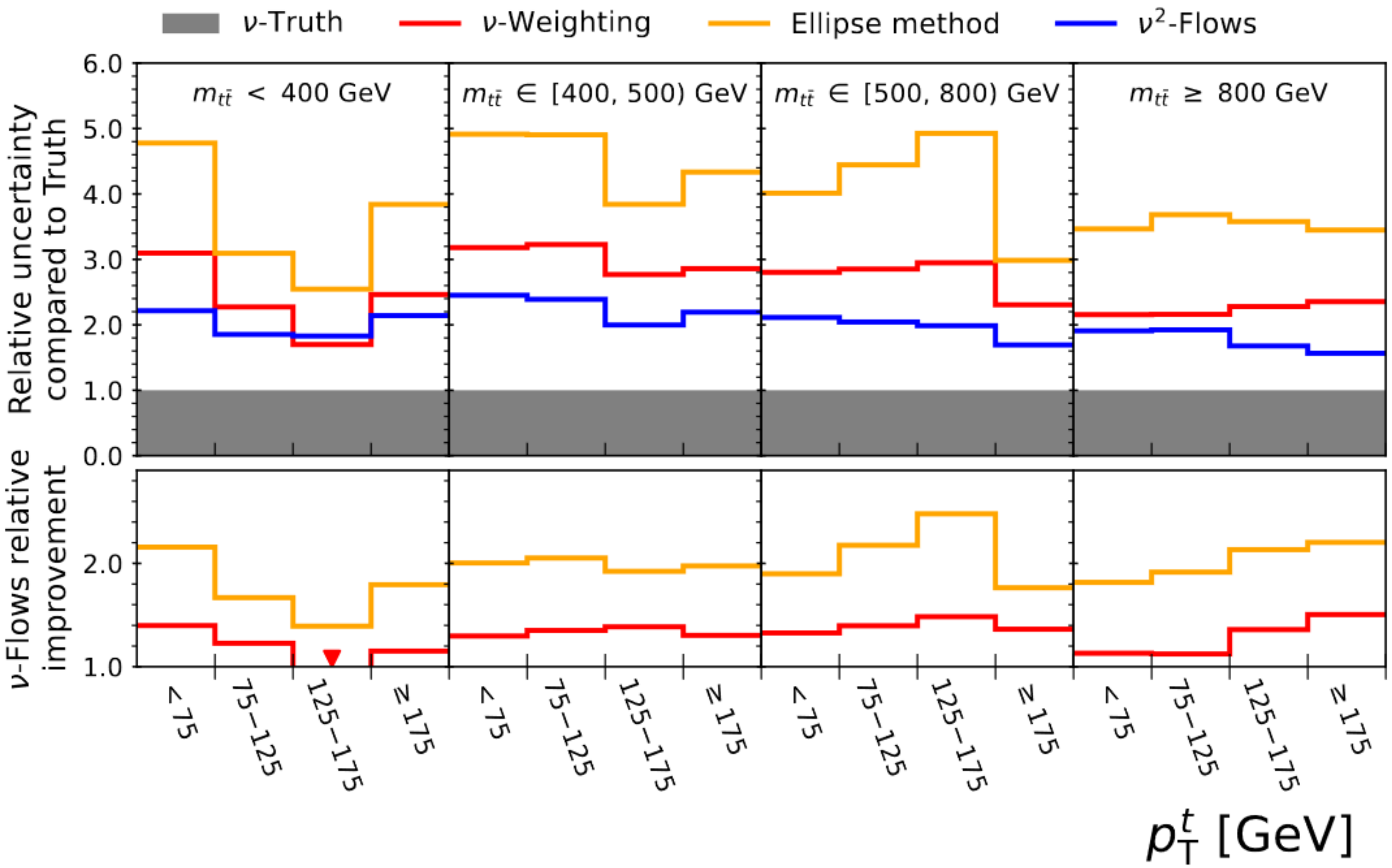
Unfolding

Response matrices



Unfolding

- Performed unfolding with RooUnfold (SVD)
- Looked at the overall uncertainties for each bin of the unfolded distributions



Conclusions

- Have extended our method to work on final states with multiple neutrinos with a more generalizable architecture
- v^2-Flows leads to reduced biases and more accurate event reconstruction compared to reference methods
- More details and results in our latest paper include sensitivity to mass shifts

- Are now developing a tool to run in an Athena environment

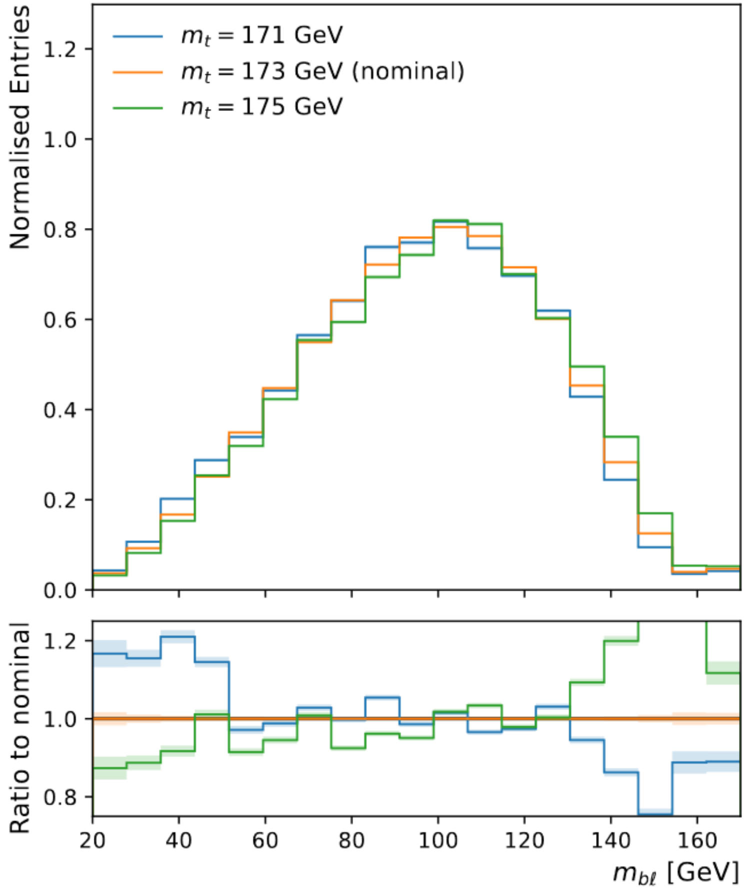
Thank You

Mass Shift

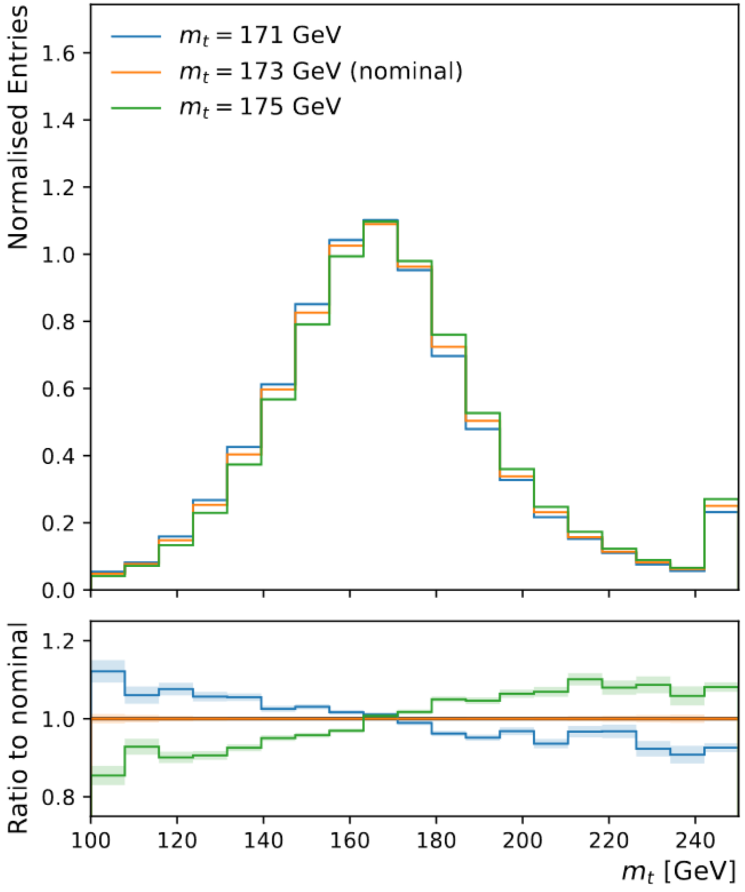
- Unlike the reference methods v^2-Flows makes no hard assumptions on W and top mass
- Would still be implicitly learned from training data
- Wanted to test if the v^2-Flows would be sensitive to a shift in the top mass
- Produced additional test sets with m_t set to 171 and 175 GeV

Mass Shift

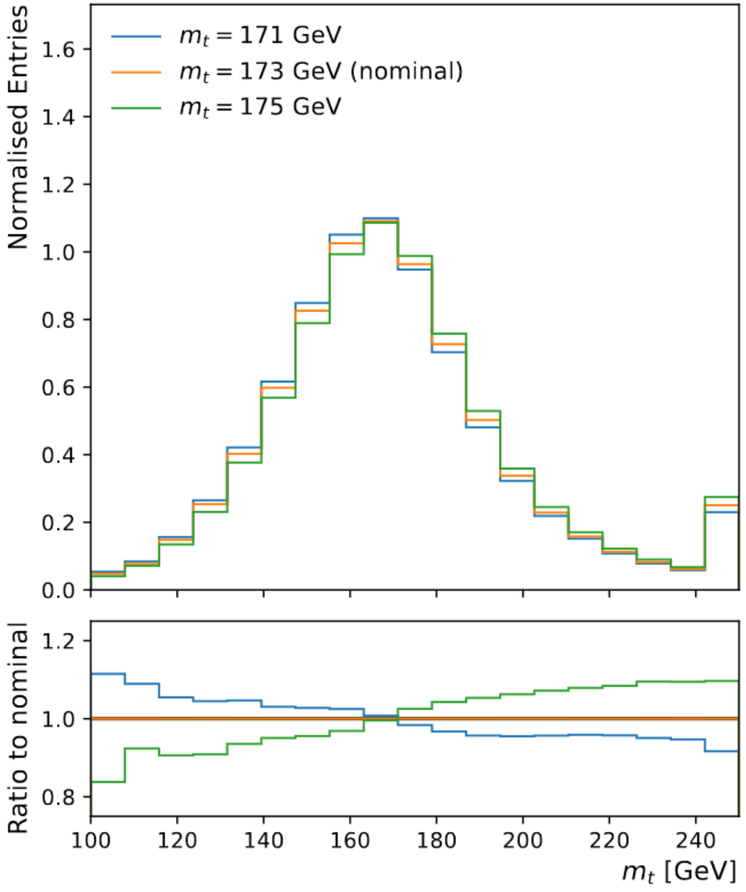
No Neutrino



v²-Flows

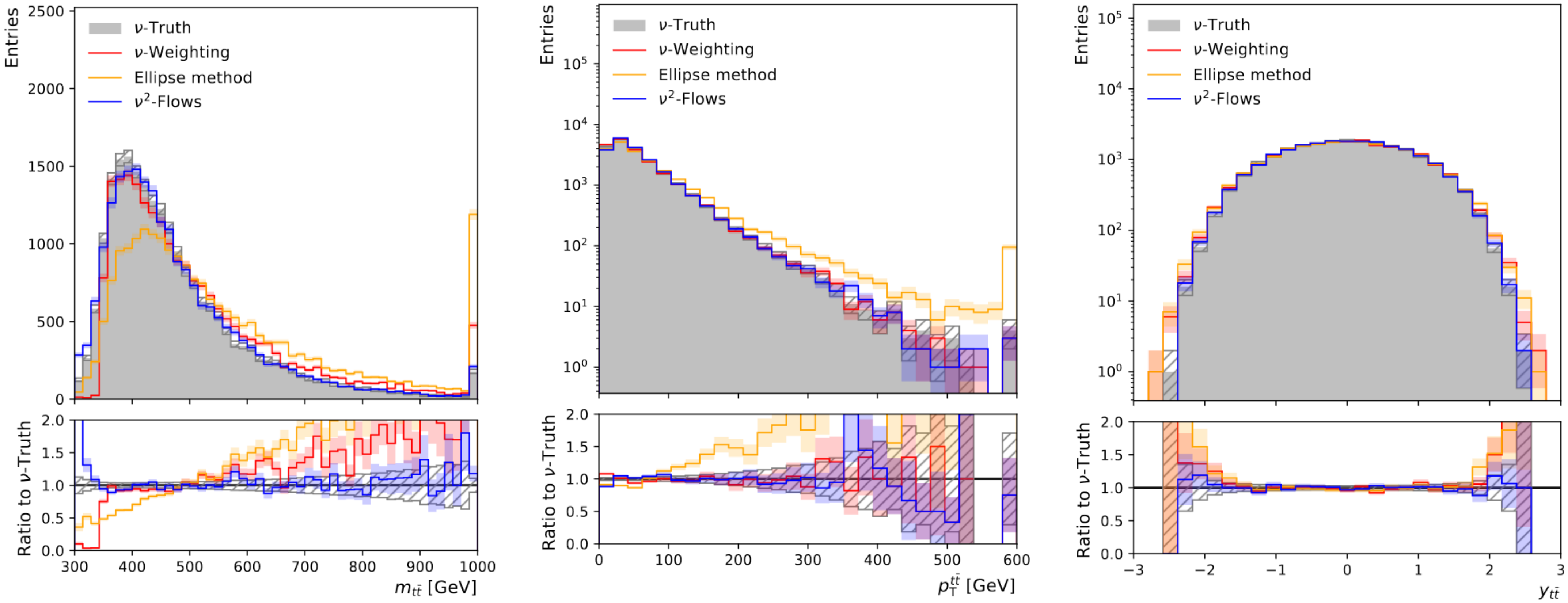


Oversample v²-Flows for smoother templates

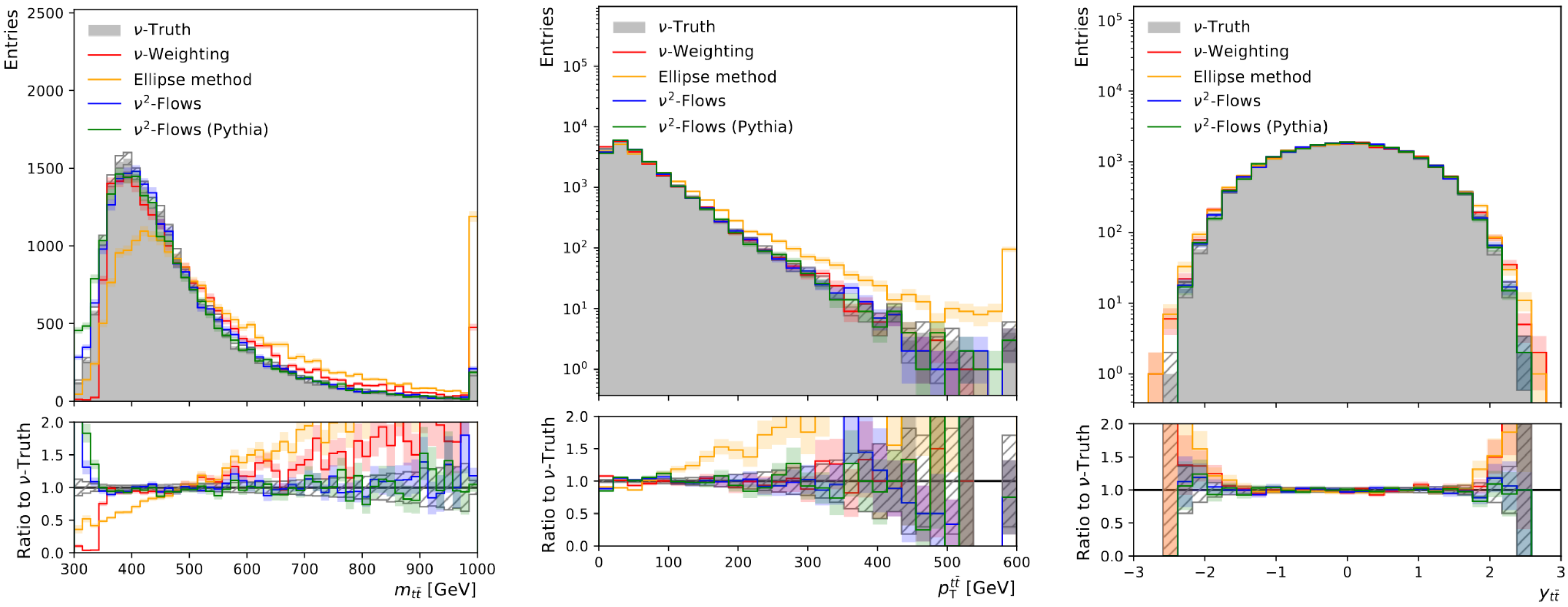


Kinematics

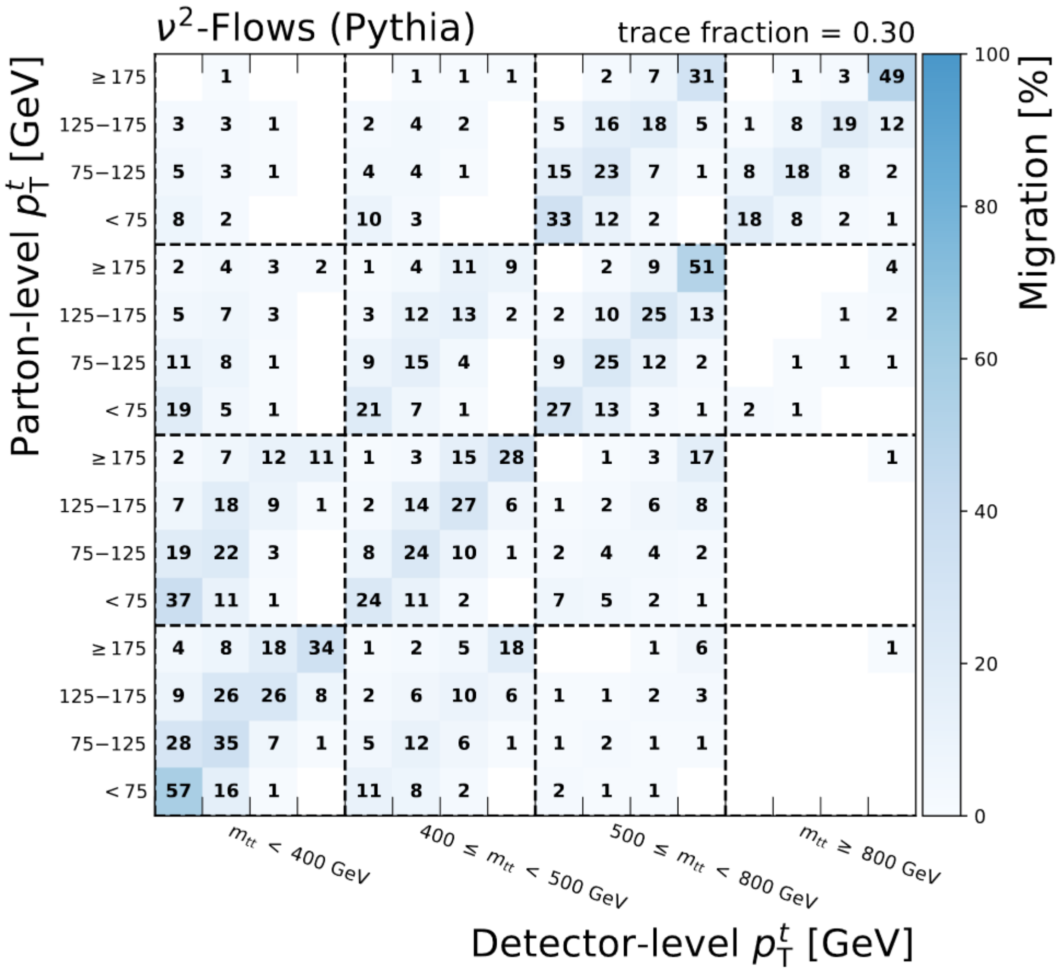
- Full $t\bar{t}$ system



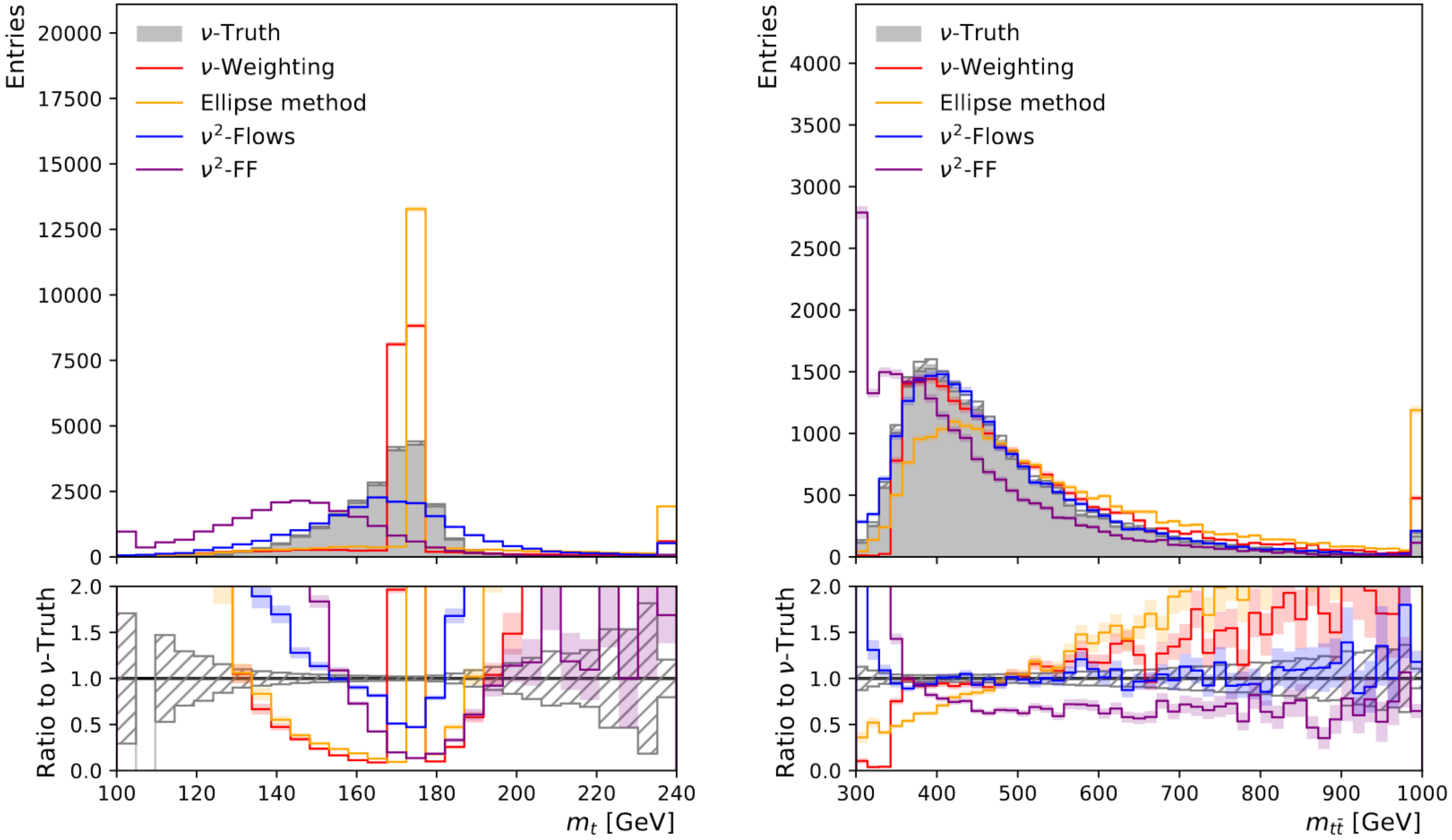
Robustness to Generator



Robustness to Generator



Failure of standard regression

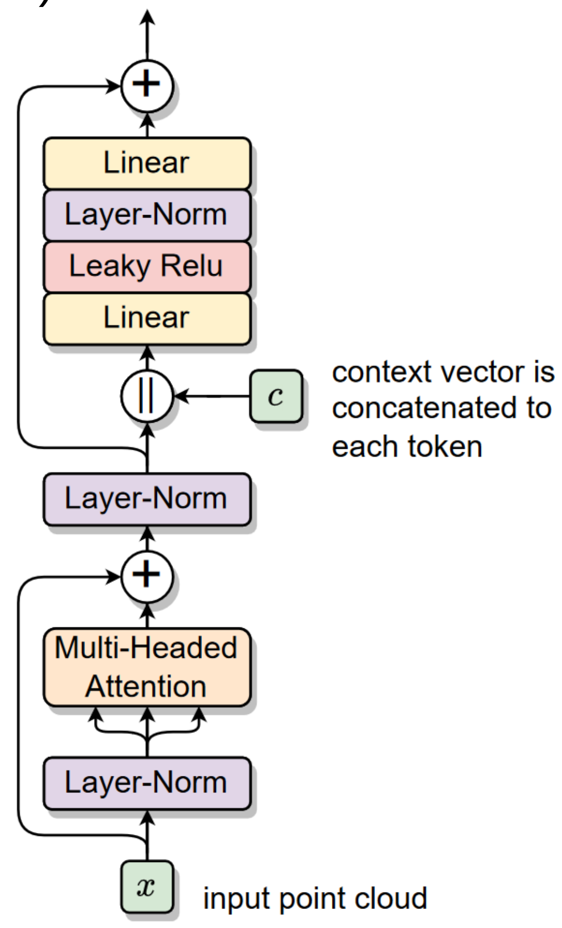


Object Features

Category	Variables	Description
$\vec{p}_{\mathrm{T}}^{\mathrm{miss}}$	p_x^{miss}, p_y^{miss}	Missing transverse momentum 2-vector
Leptons	p_x^{ℓ}, p_y^{ℓ}, p_z^{ℓ}, $\log E^{\ell}$	Lepton momentum 4-vector
	q^{ℓ}	Lepton charge
	ℓ^{flav}	Whether lepton is an electron or muon
Jets	p_x^{j}, p_y^{j}, p_z^{j},$\log E^{j}$	Jet momentum 4-vector
	isB	Whether jet passes b-tagging criteria
Misc	N_{jets}, N_{bjets}	Jet and b-jet multiplicities in the event

Network Architecture

Transformer Encoder (TE) Block



Cross-Attention (CA) Block