ν-Flows: Conditional Neutrino Regression

Matthew Leigh[1], John A. Raine[1⋆] and Tobias Golling[1]

1 University of Geneva, Geneva, Switzerland
⋆ john.raine@unige.ch

August 17, 2022

Abstract

We present ν-Flows, a novel method for restricting the likelihood space of neutrino kinematics in high-energy collider experiments using conditional normalising flows and deep invertible neural networks. This method allows the recovery of the full neutrino momentum which is usually left as a free parameter and permits one to sample neutrino values under a learned conditional likelihood given event observations. We demonstrate the success of ν-Flows in a case study by applying it to simulated semileptonic $t\bar{t}$ events and show that it can lead to more accurate momentum reconstruction, particularly of the longitudinal coordinate. We also show that this has direct benefits in a downstream task of jet association, leading to an improvement of up to a factor of 1.41 compared to conventional methods.

Contents

---

1 Introduction

Collider physics experiments such as those at the Large Hadron Collider (LHC) [1] are at the forefront of studying the fundamental interactions of nature. General purpose detectors such

as ATLAS [2] and CMS [3] are designed to measure nearly all stable particles produced in the high-energy proton-proton collisions. This means that they can be used to probe almost all aspects of the Standard Model of particle physics (SM). Reconstruction of these particles from base detector signals requires sophisticated algorithms and significant computing power. In recent years, deep learning algorithms have attracted significant attention and have been used for both kinematic reconstruction and identification for a wide variety of physics objects in these experiments. Some examples of successful applications include electron identification [4] and jet flavour tagging [5–7]. Advances in deep learning provide exciting new avenues for further improving the reconstruction performance of collider experiments.

Neutrino reconstruction requires a slightly different approach to that of jets and electrons. Neutrinos only couple to the weak nuclear force and typically do not interact with the detector material. They effectively escape from collider experiments without leaving any measurable signal. In the transverse plane perpendicular[1] to the beam pipe, conservation of linear momentum dictates that the momenta of all particles produced by the collision must sum to zero. Missing transverse momentum $\vec{p}_{\mathrm{T}}^{\,\mathrm{miss}}$ serves as an experimental proxy for the net transverse momentum of all undetected particles and it is calculated from the negative vectorial sum of all observed transverse momenta. A significant deviation from zero indicates the presence of at least one undetected particle. There is no such experimental proxy for the longitudinal momentum of undetected particles for proton-proton collisions as the initial momentum of the colliding partons is unknown. In events that produce more than one neutrino, accurate $\vec{p}_{\mathrm{T}}^{\,\mathrm{miss}}$ reconstruction still leaves the individual neutrino kinematics under-constrained.

Many analyses in collider physics investigate processes that involve neutrino production, and these could benefit from knowing the individual kinematics of final-state neutrinos. These processes include many Higgs boson production (VH, ttH) and decay modes ($H \to WW$, $H \to \tau\tau$), as well as decays of the top quark. The top quark decays almost instantaneously, and 99.9% of decays produce a b-quark and a W boson. In approximately one-third of these cases, the W boson decays leptonically, producing a final-state with a neutrino. The top quark is the heaviest particle in the SM which implies that it has the largest coupling to the Higgs boson. The value of its mass m_t has a unique role in the stability of the electroweak vacuum due to its presence in the quadratic term of the Higgs potential [8]. Due to its almost instantaneous decay, it provides us with a unique opportunity to measure the properties of a bare quark. Reconstruction of a leptonically decaying top quark, as opposed to one that decays hadronically, can benefit from a smaller sensitivity to the jet energy calibration and resolution. However, its mass is not directly measurable due to the unknown longitudinal momentum of the neutrino in the final-state. In $t\bar{t}$ decays where both top quarks decay leptonically, the full system is under-constrained, and direct mass measurements for either top quark is impossible.

In addition to neutrinos, many beyond the Standard Model (BSM) theories introduce new weakly interacting massive particles (WIMPS). These are also expected to escape the detector without leaving any directly measurable signal. WIMPS are prime candidates for dark matter, but their presence has yet to be observed in collider physics despite considerable effort [9,10]. Thus, any meaningful restriction of the possible phase space of these undetectable particles would greatly benefit a wide range of both SM studies and BSM searches.

We introduce ν-Flows, a machine learning approach to fully reconstruct the neutrinos produced in collisions from the missing transverse momentum and observed event kinematics. The approach taken in this work is that while many possible momenta values might be possible, they may not all be equally likely. Our method utilises conditional normalising flows [11,12] which exploits the latest developments in deep Bayesian learning to leverage observed infor-

[1] The coordinate system used in this work to describe collider experiment observables follows the convention of the ATLAS collaboration. The x-axis and y-axis lie perpendicular to the beam pipe while the z-axis is parallel. Pseudorapidity is defined as $\eta = -\ln(\tan\frac{\theta}{2})$, where θ is the polar angle.

mation from the final-state and combine it with an inductive bias to restrict the likelihood over the possible neutrino momentum values. By sampling from this likelihood, we obtain plausible estimates of the momenta for each undetected particle, allowing us to reconstruct topologies that involve neutrinos or other weakly interacting neutral particles. We demonstrate the applicability of ν-Flows in a semileptonic $t\bar{t}$ decay which has one neutrino in the final-state. We use estimates of the neutrino kinematics produced by ν-Flows to reconstruct properties of the top quark and compare these to standard methods of neutrino momentum estimation. We also assess the impact of using ν-Flows in the performance of a common downstream task employed in top quark analyses.

The source code[2] and data[3] used for this project are publicly available and can be found online.

2 Method

Estimation of neutrino momenta $\vec{p}^{\,\nu}$ can be framed as an *ill-posed* inverse problem. The forward process in this problem, which describes the transformation from $\vec{p}^{\,\nu}$ and other underlying variables to the observed quantities, is well understood and can be approximated by some stochastic process, such as the Monte Carlo simulations used in collider physics. But the inverse problem, acquiring the $\vec{p}^{\,\nu}$ from our set of observations, is difficult to approximate and the likelihood of the observations can only be implicitly defined by the simulation. This problem is *ill-posed* because there are many situations in collider physics where the individual neutrino kinematics can never be exactly determined, even given perfect reconstruction. An example is the under-constrained two neutrino final-state, where many different combinations of neutrino momenta are viable.

Standard deep learning regression methods collapse both the likelihood and posterior into a point estimate. This is undesirable as it gives no concept of solution diversity or uncertainty and ignores the fact that multiple solutions could exist. A probabilistic approach that can provide the likelihood over a range of viable solutions, rather than collapsing to just one, is required.

One promising method to perform full likelihood inference is to use conditional normalising flows. A normalising flow is a parametric diffeomorphism that defines a map between two probability densities over their respective spaces $f_\theta : X \to Z$. They typically map a complex probability distribution $p_X(x)$ into a simple density $p_Z(z)$ in a latent space with known properties, usually a multivariate normal distribution. These functions are often expressed using invertible neural networks (INNs) which are by design bijective, efficiently invertible, and possess a tractable Jacobian. Efficient density estimation under X is obtained using the change of variables formula

$$p_X(x) = p_z\big(f_\theta(x)\big)\Big|\det\big(J_f(x)\big)\Big|, \tag{1}$$

where $J_f(x)$ is the Jacobian of f_θ evaluated at x. This allows the generation of new data given $p_X(x)$ by sampling from $p_Z(z)$ and applying the inverse of the bijection $f_\theta^{-1}(z)$.

Normalising flows have seen great success in the field of computer vision for unconditional generation [14–16]. Conditional normalising flows use conditional invertible neural networks (cINN) [17] to incorporate contextual information c into the map and lead to expressive conditional densities $p(x|c)$ when training with a maximum (log-)likelihood objective defined by

$$\arg\max_\theta\Big(\log\big(p_X(x|c)\big)\Big) = \arg\max_\theta\Big(\log\big(p_z\big(f_\theta(x|c)\big)\big) + \log\Big|\det\big(J_f(x|c)\big)\Big|\Big). \tag{2}$$

[2] https://github.com/mattcleigh/neutrino_flows
[3] https://doi.org/10.5281/zenodo.6782987 [13]

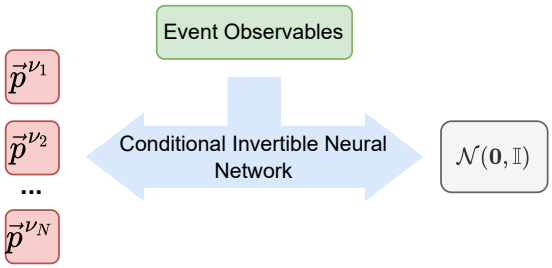


Figure 1: A schematic overview of the cINN used in ν-Flows which predicts the momentum vector of N many neutrinos as a condition of some chosen event observables. The latent density is chosen to be a multivariate normal distribution with $3N$ dimensions, $\mathcal{N}(\mathbf{0}, \mathbb{I})$.

Our method for $\vec{p}^{\,\nu}$ likelihood estimation, called ν-Flows, is built using cINNs. These types of networks have already been used in collider physics, with notable applications including event generation [18], anomaly detection [19–21], density estimation [22], detector unfolding [23], and detector simulation [24, 25]. Our work of attempting to recover the $\vec{p}^{\,\nu}$ from our set of observations falls under detector unfolding and thus it is most similar to Ref [23].

ν-Flows define a map from the combined space of all neutrino momenta to a simple density of equal dimension. To leverage information from the rest of the event, variables from event reconstruction are used as conditional inputs in the cINN. The flow can be trained directly to approximate the full conditional likelihood over the neutrino kinematics by performing gradient ascent on Equation 2. This leads to a rich description of the probability space, effectively allowing degrees of freedom to be recovered with interpretable uncertainties. A simplified diagram of this process is shown in Figure 1.

ν-Flows can be applied to a wide variety of processes involving any number of invisible particles. However, for it to learn a useful likelihood it not only requires the observed information but also underlying assumptions or implicit biases. For example, the assumption of the number of neutrinos or undetectable particles in the event is built into the structure of the cINN. Another necessary assumption is the underlying physical process being studied, which is ingrained into the flow by the composition and properties of the training set. This bias is essential, particularly for reconstructing the longitudinal momentum of the neutrino which can not be recovered directly from measurements. Since the total longitudinal momentum of the colliding particles is unknown, any estimate of the neutrino longitudinal momentum by default agrees with direct measurements. But by choosing a specific process and using its data to train the flow, restrictions on the probability space of momenta beyond the set of all possible solutions are achievable by looking at the viability of these solutions in the context of the assumed process and observed properties of the objects in the event. For each process or assumption, a specific implementation of ν-Flows should be utilised because without leveraging these implicit biases it is not possible to constrain the possible phase space of solutions.

3 Case Study: Semileptonic $t\bar{t}$

In this work, we demonstrate an implementation of ν-Flows designed and applied for semileptonic $t\bar{t}$ decays. Semileptonic $t\bar{t}$ events are produced frequently at the LHC and can be selected with a relatively high signal efficiency. The final-state of this process contains at least two b-jets, two other jets, a lepton, and a single neutrino. The goal is to use ν-Flows to recover the $\vec{p}^{\,\nu}$, allowing us to fully reconstruct the whole $t\bar{t}$ system. Semileptonic $t\bar{t}$ events provide a logical starting point to introduce ν-Flows and benchmark their performance in comparison to

standard techniques, before expanding to other topologies with more neutrinos and additional degrees of freedom.

The nominal approach to estimate $\vec{p}^{\,\nu}$ assumes that the transverse momentum of the neutrino p_T^ν is perfectly captured by $\vec{p}_{\mathrm{T}}^{\,\mathrm{miss}}$ and that the invariant mass of the W boson m_W as measured from its decay products is exactly 80.38 GeV. From the conservation of linear momentum, it follows that the longitudinal component of the neutrino momentum p_z^ν can be expressed as

$$p_z^\nu = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}, \tag{3}$$

where

$$\begin{aligned} a &= (p_z^\ell)^2 - (E^\ell)^2, \\ b &= \alpha p_z^\ell, \\ c &= \frac{\alpha^2}{4} - (E^\ell)^2 (p_{\mathrm{T}}^\nu)^2, \\ \alpha &= m_W^2 - m_\ell^2 + 2(p_x^\ell p_x^\nu + p_y^\ell p_y^\nu). \end{aligned}$$

This approach has several drawbacks. By assuming an exact value for m_W, any results or downstream tasks are biased. It does not consider the possibility for m_W to be off-shell or account for the misidentification, resolution, or mismodelling effects in the lepton or $\vec{p}_{\mathrm{T}}^{\,\mathrm{miss}}$ reconstruction. As a result, the equation may have no real solutions. The convention in this instance is to neglect the imaginary component [26], a step that has little physical motivation. Even in the case where everything is perfectly measured, the equation can yield two real solutions with no preference. The estimate with the smaller magnitude is usually taken. Alternatively, both solutions are considered in any downstream tasks.

In contrast, ν-Flows does not make such hard assumptions. From the composition of the training data, it can learn the width of the m_W distribution and propagate that to a complex distribution over the longitudinal momenta. By providing ν-Flows with additional information from the event it can learn how the resolution of both $\vec{p}_{\mathrm{T}}^{\,\mathrm{miss}}$ and $\vec{p}^{\,\ell}$ further affect this result. With more contextual information, ν-Flows combines observables in a fully probabilistic manner to learn the conditional distribution of possible solutions without collapsing the reconstruction down to singular values. Furthermore, the quadratic solution is only valid for final-states with a single neutrino, while ν-Flows can be scaled to any multiplicity.

3.1 Input Data and Targets

The data used in this work consists of simulated $t\bar{t}$ events with exactly one of the top quarks decaying leptonically into a b-jet and with the $W^\pm$ boson decaying into either (e, ν_e) or (μ, ν_μ), or their corresponding antiparticles [13]. All sets of events are generated from simulated proton-proton collisions at a center-of-mass energy of $\sqrt{s} = 13$ TeV.

Hard interactions are simulated using the MadGraph5_aMC@NLO [27] framework (v3.1.0), with decays of top quarks and W bosons modeled with MadSpin [28]. The mass of the top quark is set to $m_t = 173$ GeV for all events. The event generation is interfaced to Pythia [29] (v8.243) to model parton shower and hadronisation. All steps use the NNPDF2.3LO PDF set [30] with $\alpha_S(m_Z) = 0.130$, as provided by the LHAPDF [31] framework. The detector response is simulated using Delphes [32] (v3.4.2) with a parametrisation that mimics the response of the ATLAS detector [2]. Jets are reconstructed using energy-flow objects and the anti-k_t algorithm [33] in the FastJet implementation [34] with a radius parameter of $R = 0.4$. Jet b-tagging corresponding to an inclusive signal efficiency of 70% is used to identify jets originating from b-quarks. Exactly one reconstructed electron or muon with $p_{\mathrm{T}} > 15$ GeV in

the range $|\eta| < 2.5$ and at least four jets with $p_T > 25\,\text{GeV}$ in the range $|\eta| < 2.5$ are required. Around 600k events are used to train the model and an additional 100k events are used for evaluating performance.

Variables from event reconstruction are used as conditioning inputs to all models presented in this work. These include the kinematics of the signal lepton, kinematics, and b-tagging information of the reconstructed jets (up to 10 sorted by p_T), the $\vec{p}_T^{\text{miss}}$, and additional event observables. The full set of inputs is described in Table 1. The target distribution for the networks is the single neutrino three-momentum vector defined by $\left(p_x^\nu, p_y^\nu, \eta^\nu\right)$. The coordinate system used to represent the momentum of each physics object is individually optimised. We find that using p_x, p_y, and η components alongside the natural logarithm of its energy $\log E^j$ yields the best results. The target density $p_Z(z)$ is chosen to be a standard normal distribution.

Table 1: The different input observables used as conditional variables c in the normalising flow.

Category	Variables	Description
$\vec{p}_T^{\text{miss}}$	p_x^{miss}, p_y^{miss}	Missing transverse momentum 2-vector
Lepton	p_x^ℓ, p_y^ℓ, η^ℓ, $\log E^\ell$	Lepton momentum 4-vector
	ℓ^{flav}	Whether lepton is an electron or muon
Jets	p_x^j, p_y^j, η^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

3.2 cINN Setup

The architecture of the ν-Flows optimised for the neutrino in semileptonic $t\bar{t}$ decays is shown in Figure 2. The conditioning variables c are first passed through a feed-forward (FF) network to ensure that the same high-level features are provided to each of the cINN blocks. In the FF component, a Deep Set [35] is used to extract information from the jets due to its ability to handle varying jet multiplicities while also remaining permutation invariant. The main cINN blocks consist of seven rational-quadratic spline coupling layers [14]. Further details on the specific structure of each module can be found in Appendix A.

The cINN is trained on the objective function in Equation 2 using the Adam optimiser [36] with default β parameters and a batch size of 256. We use a cosine annealing scheduler that cycles the learning rate from zero to 5×10^{-4} and back every 2 epochs. Gradient clipping is essential for stable convergence and a max L2-norm of 5 is used. As a preprocessing step, all conditioning and target variables are independently normalised using the variance and mean of the training set. For cross-validation, 10% of the training is reserved as a holdout set and early stopping is used with a patience parameter of 30 epochs. We use the python packages PyTorch [37] and nflows [38] to construct and train the cINN.

4 Performance

For ν-Flows, two different configurations for conditional neutrino generation are investigated. ν-Flows(sample) represents the case where a single neutrino is sampled per event under the

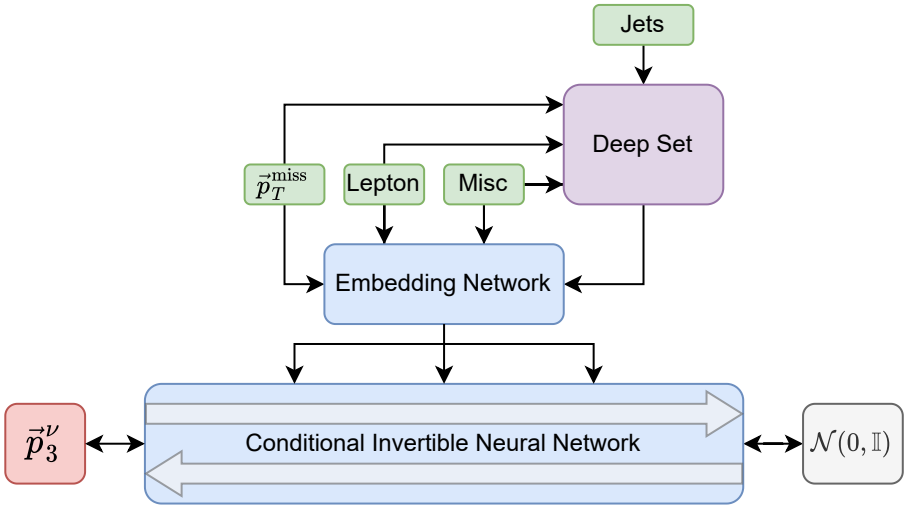


Figure 2: A schema of the ν-Flows for semileptonic $t\bar{t}$. The four classes of conditioning inputs are shown in green and are used as inputs for both the Deep Set and the Embedding Network. There is only one neutrino in the event, so the input and output vectors of the cINN are three-dimensional.

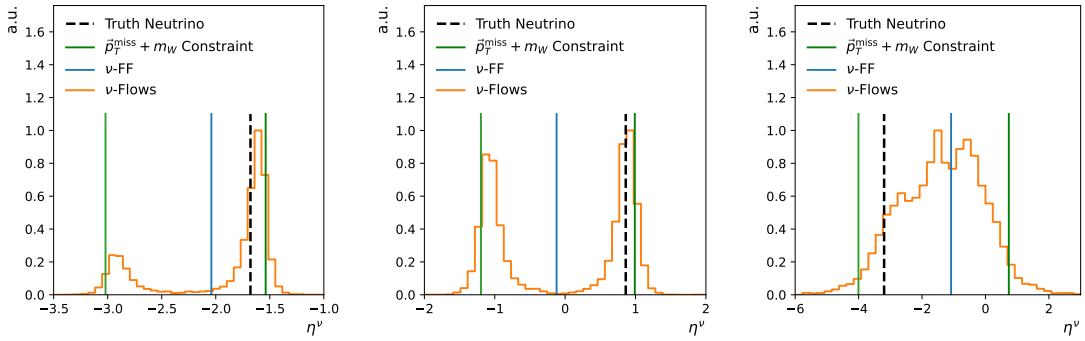
learned conditional likelihood defined by Equation 2. This method of sampling is less biased but suffers from a high variance. As an alternative we also introduce ν-Flows(mode) to stochastically approximate $\arg\max_x p_X(x|c)$. This is done by conditionally generating 256 neutrinos per event and keeping the one with the highest likelihood.

We also train a standard FF regression network that follows the same structure as the FF component of ν-Flows but with a deeper embedding network used to predict the neutrino three-momentum directly. The FF network is trained using the Smooth-L1 loss function [39], with $\vec{p}^{\,\nu}$ as the target variable. We use the same data, optimiser, learning rate scheduler, gradient clipping, and early stopping method as ν-Flows. This method is referred to as ν-FF.

These methods are compared to the current standard approach which combines $\vec{p}_{\text{T}}^{\text{miss}}$ and the solutions defined by Equation 3. As an upper benchmark, we compare all methods to using the true values of the neutrino taken from the simulation. Plots labelled *Truth* refer only to using the true neutrino values, and all other properties, like those of the leptons or the jets, are taken from the reconstructed objects.

To best illustrate the benefits of a probabilistic method such as ν-Flows, Figure 3 shows the reconstruction of the neutrino pseudorapidity for three different samples drawn from the evaluation dataset using the m_W constraint method, ν-FF, and ν-Flows. In Figure 3(a) the true value of η^ν is around -1.70. One of the solutions of Equation 3 is close to the true value and is around -1.55 while the other is significantly further away at -3.05. There is no indication *a priori* which of these two solutions will be closer to the truth and this is one of the main drawbacks of the method. ν-Flows on the other hand provides us with the full likelihood across a range of η^ν values and shows a distribution with two local peaks corresponding to the quadratic solutions. This is worth noting as ν-Flows was able to relearn the kinematic relationship detailed in Equation 3 entirely from data. But unlike the m_W constraint solutions, ν-Flows gives us interpretable uncertainties.

We also trained a version of ν-Flows using quadratic solutions as extra conditioning inputs and observed a slight performance increase. However, we felt that the version which had to relearn this relationship purely from the dataset better demonstrated the power and expressiveness of the method. Furthermore, using ν-Flows without the quadratic solutions also meant the same architecture can be applied to final-states with multiple neutrinos, where the



(a) (b) (c)

Figure 3: The pseudorapidity (η) of three different neutrinos selected from the evaluation dataset. The true values are shown in black. The two solutions from the m_W constraint method are shown in green. The single point estimate using ν-FF is shown in blue. The full conditional likelihood learned by ν-Flows is shown in orange.

quadratic method would be invalid.

Furthermore, for the event represented by 3(a), ν-Flows indicates a preference for one of the possible solutions, with the highest localised cumulative distribution occurring at $\eta^\nu \approx -1.60$, close to the true value. This preference may be due to several reasons. For example, if the Deep Set component of ν-Flows was able to identify which jet corresponds to the b-quark produced by the leptonic top, it would allow further constraints on possible η^ν values. In contrast ν-FF results in a point estimate close to -2.05 which falls between the two peaks, an area of low probability as estimated by ν-Flows. Since ν-FF and ν-Flows share the same information and a large portion of network structure it can be proposed that ν-FF essentially averages the multimodal likelihood which is fully expressed by ν-Flows.

Figure 3(b) shows a similar situation where ν-Flows reproduces the multimodal likelihood as expected by the kinematic constraint but with less of a preference for one solution over the other. Because of this ν-FF results in a point estimate close to the average of the two solutions, resulting in an estimate much closer to $\eta^\nu \approx 0$.

Figure 3(c) shows an event where none of the methods could provide a good estimate for η^ν. It is important to note that the relative width or uncertainty displayed by the likelihood plot of ν-Flows has increased correspondingly. This shows another benefit of this probabilistic approach as it can identify this event as being poorly reconstructed and one can filter it from downstream tasks.

The distribution of the neutrino four-momentum using the different methods for reconstruction are shown in Figure 4. For all coordinates, the distribution of the ν-Flows(sample) is closest to the true momentum distribution. The ν-FF and m_W constraint methods induce a negative bias towards zero. This is most notable for p_z^ν, shown in Figure 4(c), where both methods significantly overestimate the fraction of events close to zero. This results in an underestimation of the energy as shown by Figure 4(d). ν-Flows(mode) also possesses a negative bias in p_z^ν and E^ν, although it is not as significant. There are notable artifacts in the ν-Flows(mode) distributions in the transverse plane which causes a double peak around 20 GeV. This is caused by the shape of the p_x and p_y distributions of the jets and leptons, which due to the cut on p_T also exhibit these double peaks.

Figure 5 shows heatmaps of 2D histograms using coordinates defined by the reconstructed and true p_z^ν. Once again the bias towards zero is apparent in the m_W constraint solutions and in the ν-FF, both with an overestimation at zero. The negative bias in ν-FF is caused by the model often guessing between the two kinematic solutions, as shown by Figure 3. Both

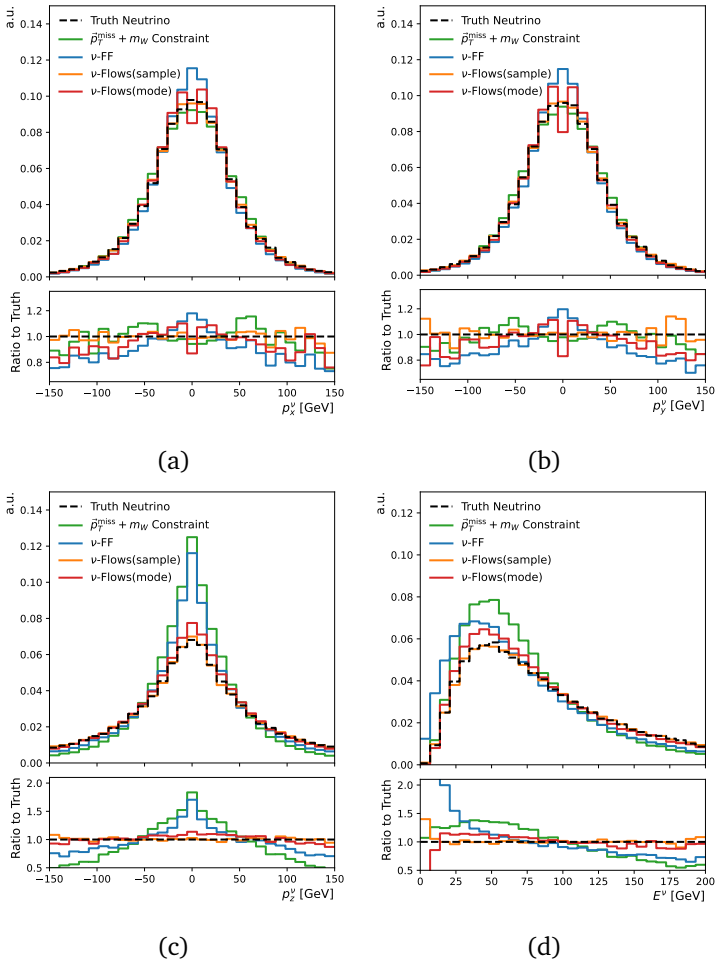


Figure 4: Distributions of each component of the neutrino four-momentum using the different reconstruction methods.

ν-Flows models show a good correlation to *Truth*, however ν-Flows(sample) suffers from too high a variance, showing the drawback in taking a single sample from the learned density. Here ν-Flows(mode) shows good performance with the bulk of events being highly correlated with the true values while also showing no obvious bias.

The reconstructed invariant mass of the leptonic W is shown in Figure 6(a), calculated using the momentum vector of the reconstructed lepton and each estimate of p_z^ν. The distribution using the true neutrino is almost exactly matched by ν-Flows(sample), while ν-Flows(mode) is tightly centered around the mean. ν-FF shows a notable offset of the mean by around 6 GeV. The m_W constraint results in nearly all events having exactly $m_{\ell\nu} = 80.38$ GeV, as expected, and the positive tail arises from events which lead to no real solutions for Equation 3. As is expected, ν-Flows(mode) is biased towards the central value of the m_W since it is estimating the most likely neutrino, which is therefore coupled with the most likely value for m_W.

When looking at the correlation between the reconstructed m_W values and the true values, no correlations are observed for any of the methods. We find that the resolution effects in the $\vec{p}_{\text{T}}^{\text{miss}}$ are enough to destroy all information about the m_W of the event. This observation holds even when using the true value p_z^ν alongside $\vec{p}_{\text{T}}^{\text{miss}}$. It is worth noting that ν-Flows learns the distribution of m_W across the dataset even though it could not specify it on an event-by-event basis. This further demonstrates that it has learned to restrict its predictions of p_z^ν to the true space of possible solutions.

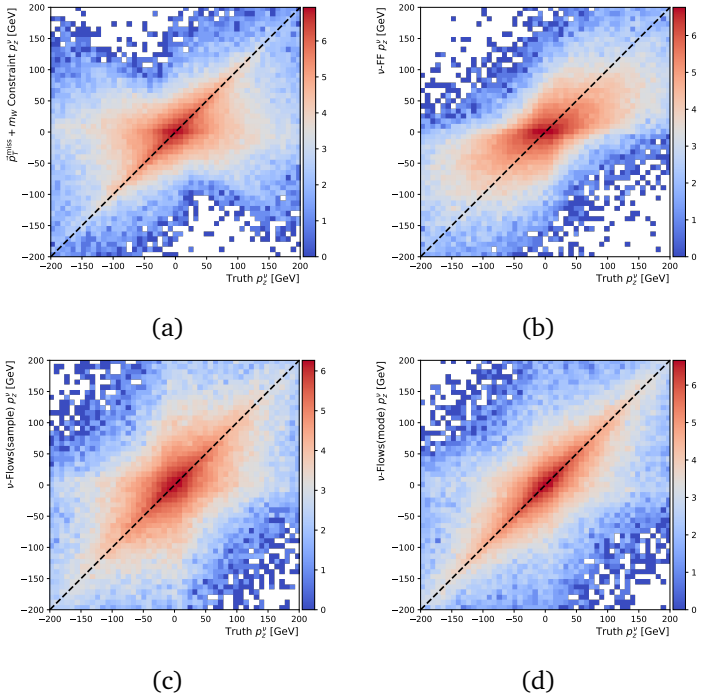
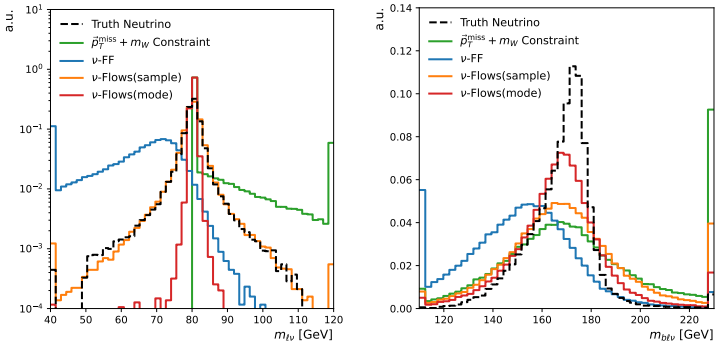


Figure 5: Two dimensional histograms showing the reconstructed versus true p_z^ν using both solutions of the m_w kinematic constraint (a), ν-FF, (b), ν-Flows(sample) (c), and ν-Flows(mode) (d). The diagonal line represents ideal reconstruction. We use a logarithmic scale for the z-axis.

The reconstructed invariant mass of the leptonic top quark is shown in Figure 6(b). The correct b-jet from the leptonically decaying top quark is used in the calculation of the top mass. This is done to highlight the effect of the neutrino reconstruction, and thus only events for which the b-jet is reconstructed are shown. The ν-FF method produces a shifted mass distribution, demonstrating a strong negative bias, with its peak at around 155 GeV. All other methods reduce this bias, but still peak at around 169 GeV, slightly under the simulated top mass of 173 GeV. Notably, the top mass distribution produced when using the true neutrino is negatively skewed while all other distributions are more symmetrical. The m_W constraint method produces the distribution with the largest variance, resulting in a significant number of events with a reconstructed top mass greater than 230 GeV as shown by the overflow bin. The ν-Flows(sample) method reduces this mass variance to around the same level as ν-FF but without the negative shift. The ν-Flows(mode) method further reduces this variance and produces the mass distribution most similar to *Truth*.

χ^2 Jet Association

To assess the impact of ν-Flows in a real downstream task, we investigate its effect on solving the combinatoric assignment of jets to final-state partons in semileptonic $t\bar{t}$ events. Initially, it is unknown which (if any) of the jets that were observed in the event can be associated with the b-quark which was produced alongside the leptonically decaying W boson (b_{lep}). In the final-state of the semileptonic $t\bar{t}$ channel there are four partons originating from the $t\bar{t}$ decay. These are the b-quarks from the leptonically and hadronically decaying top quarks (b_{lep} and b_{had} respectively), as well as the two decay products from the hadronically decaying W boson, q_1 and q_2. Additional jets are also reconstructed from initial state radiation, final-state radiation, and pileup interactions. One of the most common methods used to assign



(a) (b)

Figure 6: Distributions of the invariant mass of the $\ell\nu$ (a) and $b\ell\nu$ (b) systems using different neutrino reconstruction methods. All methods use reconstructed variables for the lepton and jet kinematics and *Truth Neutrino* uses the true neutrino.

the reconstructed jets to each parton is the χ^2 fit [40]. The jet-assignment derived using this method is dependent on the neutrino kinematics and thus it can be used to demonstrate the benefits of having a more accurate neutrino estimate.

It is important to note this is just one of many jet combinatoric solving methods. Another popular approach is KL-Fitter [41] which is similarly dependent on the neutrino momentum. More recent approaches use deep learning to perform the associations [42] and have shown significant performance gains over the χ^2 method. All of these combinatoric techniques should be complemented by ν-Flows, though we demonstrate the potential gains using the χ^2 method as it is already widely used in analyses [40, 43, 44].

In the χ^2 fit method, every possible jet permutation is tested, and the one with the lowest χ^2 value defined by

$$\chi^2 = \frac{(m_W - m_{\ell\nu})^2}{\sigma_{\ell\nu}} + \frac{(m_W - m_{qq})^2}{\sigma_{qq}} + \frac{(m_t - m_{b\ell\nu})^2}{\sigma_{b\ell\nu}} + \frac{(m_t - m_{bqq})^2}{\sigma_{bqq}} \tag{4}$$

is kept. In this work, the σ values are taken from the root mean square error of the relevant mass distributions, using the true jet-assignments, and are derived for each neutrino reconstruction method separately. We perform the χ^2 fit using permutations of up to 9 leading p_{T} ordered jets and record the parton association accuracy for each neutrino reconstruction method.

The b_{lep} matching efficiency has the highest dependence on the neutrino in the χ^2 fit and the association accuracy of the b_{lep} is shown in Table 2. Using estimates from either ν-Flows(sample) or ν-Flows(mode) results in an improved matching efficiency compared to the standard kinematic approach. The χ^2 fit performed with estimates from ν-Flows(mode) instead of the m_W constraint led to an increase in accuracy by a factor of 1.03 for events with four jets and 1.41 for events with nine jets. For events with a low number of jets, few permutations exist, which means that the neutrino term is less likely to have an impact in Equation 4. Therefore, the observed relationship between the performance gained using ν-Flows and the number of jets in the event is expected.

5 Conclusions

We introduce ν-Flows, a probabilistic model for conditional neutrino momentum estimation. We show that in semileptonic $t\bar{t}$ events ν-Flows leads to better overall momentum reconstruc-

Table 2: The fraction of events for which the χ^2 method identified the correct b_{lep} jet using the various neutrino estimation methods. The results are binned by the number of reconstructed jets in the event. Events must first pass a selection requirement where the partons were reconstructed as jets, so a correct permutation was at least possible. This selection did not change the ranking of the methods.

	Number of Jets					
Neutrino Type	4	5	6	7	8	9
Truth Neutrino	0.864	0.753	0.686	0.641	0.611	0.587
$\vec{p}_{\mathrm{T}}^{\mathrm{miss}}$ and m_W Constraint	0.790	0.576	0.476	0.398	0.366	0.286
ν-FF	0.754	0.533	0.410	0.353	0.300	0.302
ν-Flows(sample)	0.803	0.624	0.515	0.457	0.391	0.357
ν-Flows(mode)	**0.813**	**0.664**	**0.575**	**0.508**	**0.481**	**0.405**

tion in comparison to both standard kinematic approaches and deep feed-forward networks. This in turn leads to significant improvements in downstream tasks, as demonstrated using the χ^2 method for solving the jet associations in $t\bar{t}$ events.

To continue this work, we will apply ν-Flows to final-states with more than one neutrino and therefore have under-constrained transverse momenta. A natural extension to the processes studied in this work is the fully leptonic $t\bar{t}$ decay. This will provide some insight into how well ν-Flows scales to higher neutrino multiplicities and target dimensions. Furthermore, we would like to investigate the impact of ν-Flows on other downstream tasks in addition to the χ^2 method. More sophisticated algorithms for jet-assignment that use deep learning [42] have been shown to be very successful and may combine well with ν-Flows.

Additionally, the agreement between the learned conditional likelihood and the distribution of the actual momentum targets needs to be investigated. It is currently unclear if the learned likelihood represents a properly calibrated uncertainty. A further potential utilisation of the full likelihood produced by ν-Flows is that one could reject events where the likelihood was insufficiently constrained. This would be an effective form of event cleaning and a way to filter data to only events where ν-Flows was sufficiently confident in its output. This is another benefit of a fully probabilistic approach which is infeasible with standard supervised regression models.

Acknowledgements

The authors would like to thank Knut Zoch for useful discussions and feedback on the manuscript and for support in producing the datasets for these studies. ML, JR and TG are supported through funding from the SNSF Sinergia grant called Robust Deep Density Models for High-Energy Particle Physics and Solar Flare Analysis (RODEM) with funding number CRSII5_193716. ML is further supported with funding acquired through the Swiss Government Excellence Scholarships for Foreign Scholars.

References

[1] L. R. Evans and P. Bryant, *LHC Machine*, JINST **3**, S08001, doi:10.1088/1748-0221/3/08/S08001.

[2] The ATLAS Collaboration, *The ATLAS Experiment at the CERN Large Hadron Collider*, JINST **3**, S08003, doi:10.1088/1748-0221/3/08/S08003.

[3] The CMS Collaboration, *The cms experiment at the cern lhc*, JINST **3**, S08004, doi:10.1088/1748-0221/3/08/S08004.

[4] The ATLAS Collaboration, *Identification of electrons using a deep neural network in the ATLAS experiment*, ATL-PHYS-PUB-2022-022.

[5] The ATLAS Collaboration, *ATLAS b-jet identification performance and efficiency measurement with $t\bar{t}$ events in pp collisions at $\sqrt{s} = 13$ TeV*, Eur. Phys. J. C **79**(11), 970 (2019), doi:10.1140/epjc/s10052-019-7450-8.

[6] The ATLAS Collaboration, *Graph Neural Network Jet Flavour Tagging with the ATLAS Detector*, ATL-PHYS-PUB-2022-027.

[7] The CMS collaboration, *Deep learning in jet reconstruction at CMS*, Journal of Physics: Conference Series **1085**(4), 042029, doi:10.1088/1742-6596/1085/4/042029.

[8] G. Degrassi *et al.*, *Higgs mass and vacuum stability in the standard model at NNLO*, JHEP **08**, 98, doi:10.1007/jhep08(2012)098.

[9] The ATLAS Collaboration, *Searches for new phenomena in events with two leptons, jets, and missing transverse momentum in 139 fb^{-1} of $\sqrt{s} = 13$ TeV pp collisions with the ATLAS detector*, doi:10.48550/arXiv.2204.13072, 2204.13072.

[10] The CMS Collaboration, *Search for electroweak production of charginos and neutralinos at $\sqrt{s} = 13$ TeV in final-states containing hadronic decays of WW, WZ, or WH and missing transverse momentum*, doi:10.48550/arXiv.2205.15711, 2205.09597.

[11] E. G. Tabak and E. Vanden-Eijnden, *Density estimation by dual ascent of the log-likelihood*, Commun. Math. Sci. **8**(1), 217, doi:10.4310/CMS.2010.v8.n1.a11.

[12] D. Rezende and S. Mohamed, *Variational inference with normalizing flows*, In *Proceedings of The International Conference on Machine Learning*, vol. 37, pp. 1530–1538. PMLR, doi:10.48550/arXiv.1505.05770.

[13] K. Zoch, J. A. Raine, L. Ehrke, D. Sengupta, M. Leigh and T. Golling, *Semileptonic ttbar neutrino regression dataset*, doi:10.5281/zenodo.6782987 (2022).

[14] C. Durkan, A. Bekasov, I. Murray and G. Papamakarios, *Neural spline flows*, In *Proceedings of Advances in Neural Information Processing Systems*, vol. 32, doi:10.48550/arXiv.1906.04032.

[15] D. P. Kingma and P. Dhariwal, *Glow: Generative flow with invertible 1x1 convolutions*, In *Proceedings of Advances in Neural Information Processing Systems*, vol. 31, doi:10.48550/arXiv.1807.03039.

[16] B. Ross and J. Cresswell, *Tractable density estimation on learned manifolds with conformal embedding flows*, In *Proceedings of Advances in Neural Information Processing Systems*, vol. 34, pp. 26635–26648, doi:10.48550/arXiv.2106.05275.

[17] L. Ardizzone, C. Lüth, J. Kruse, C. Rother and U. Köthe, *Guided image generation with conditional invertible neural networks*, doi:10.48550/arXiv.1907.02392, 1907.02392.

[18] B. Stienen and R. Verheyen, *Phase Space Sampling and Inference from Weighted Events with Autoregressive Flows*, SciPost Phys. **10**(2), 38, doi:10.21468/scipostphys.10.2.038.

[19] B. Nachman and D. Shih, *Anomaly detection with density estimation*, Phys. Rev. D **101**, 075042 (2020), doi:10.1103/PhysRevD.101.075042.

[20] A. Hallin, J. Isaacson, Kasieczka *et al.*, *Classifying Anomalies THrough Outer Density Estimation (CATHODE)*, doi:10.48550/ARXIV.2109.00546, 2109.00546.

[21] J. A. Raine, S. Klein, D. Sengupta and T. Golling, *Curtains for your sliding window: Constructing unobserved regions by transforming adjacent intervals*, doi:10.48550/arXiv.2203.09470, 2203.09470.

[22] J. Brehmer and K. Cranmer, *Flows for simultaneous manifold learning and density estimation*, In *Proceedings of Advances in Neural Information Processing Systems*, vol. 33, pp. 442–453, doi:10.48550/arXiv.2003.13913.

[23] M. Bellagente, A. Butter, G. Kasieczka, T. Plehn, A. Rousselot, R. Winterhalder, L. Ardizzone and U. Köthe, *Invertible Networks or Partons to Detector and Back Again*, SciPost Phys. **9**(5), 74, doi:10.21468/scipostphys.9.5.074.

[24] C. Krause and D. Shih, *CaloFlow: Fast and Accurate Generation of Calorimeter Showers with Normalizing Flows*, doi:10.48550/arXiv.2106.05285, 2106.05285.

[25] C. Krause and D. Shih, *CaloFlow II: Even Faster and Still Accurate Generation of Calorimeter Showers with Normalizing Flows*, doi:10.48550/ARXIV.2110.11377, 2110.11377.

[26] M. Grossi, J. Novak, B. Kerševan and D. Rebuzzi, *Comparing traditional and deep-learning techniques of kinematic reconstruction for polarization discrimination in vector boson scattering*, Eur. Phys. J. C **80**(12), 1144, doi:10.1140/epjc/s10052-020-08713-1.

[27] J. Alwall, R. Frederix, S. Frixione, V. Hirschi, F. Maltoni, O. Mattelaer, H.-S. Shao, T. Stelzer, P. Torrielli and M. Zaro, *The automated computation of tree-level and next-to-leading order differential cross sections, and their matching to parton shower simulations*, JHEP **07**, 79, doi:10.1007/jhep07(2014)079.

[28] P. Artoisenet, R. Frederix, O. Mattelaer and R. Rietkerk, *Automatic spin-entangled decays of heavy resonances in monte carlo simulations*, JHEP **03**, 15, doi:10.1007/jhep03(2013)015.

[29] T. Sjöstrand, S. Mrenna and P. Skands, *A brief introduction to pythia 8.1*, Comput. Phys. Commun. **178**, 852, doi:10.1016/j.cpc.2008.01.036.

[30] R. D. Ball, V. Bertone, S. Carrazza, C. S. Deans, L. Del Debbio, S. Forte, A. Guffanti, N. P. Hartland, J. I. Latorre, J. Rojo *et al.*, *Parton distributions with lhc data*, Nucl. Phys. B **867**, 244, doi:10.1016/j.nuclphysb.2012.10.003.

[31] A. Buckley, J. Ferrando, S. Lloyd, K. Nordström, B. Page, M. Rüfenacht, M. Schönherr and G. Watt, *Lhapdf6: parton density access in the lhc precision era*, Eur. Phys. J. C **75**(3), 132, doi:10.1140/epjc/s10052-015-3318-8.

[32] J. de Favereau *et al.*, *DELPHES 3, A modular framework for fast simulation of a generic collider experiment*, JHEP **02**, 057, doi:10.1007/jhep02(2014)057.

[33] M. Cacciari, G. P. Salam and G. Soyez, *The anti-kt jet clustering algorithm*, JHEP **04**, 063, doi:10.1088/1126-6708/2008/04/063.

[34] M. Cacciari, G. P. Salam and G. Soyez, *Fastjet user manual*, Eur. Phys. J. C **72**(3), 1, doi:10.1140/epjc/s10052-012-1896-2.

[35] M. Zaheer, S. Kottur, S. Ravanbakhsh, B. Poczos, R. R. Salakhutdinov and A. J. Smola, *Deep sets*, Proceedings of Advances in Neural Information Processing Systems **30**, doi:10.48550/arXiv.1703.06114.

[36] D. P. Kingma and J. Ba, *Adam: A method for stochastic optimization*, doi:10.48550/arXiv.1412.6980, 1412.6980.

[37] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf *et al.*, *Pytorch: An imperative style, high-performance deep learning library*, In *Proceedings of Advances in Neural Information Processing Systems*, vol. 32, pp. 8024–8035, doi:10.48550/arXiv.1912.01703.

[38] C. Durkan, A. Bekasov, I. Murray and G. Papamakarios, *nflows: normalizing flows in PyTorch*, doi:10.5281/zenodo.4296287.

[39] R. Girshick, *Fast r-cnn*, In *Proceedings of The International Conference on Computer Vision*, pp. 1440–1448, doi:10.48550/arXiv.1504.08083.

[40] The ATLAS Collaboration, *Top-quark mass measurement in the all-hadronic $t\bar{t}$ decay channel at $\sqrt{s} = 8$ TeV with the ATLAS detector. Top-quark mass measurement in the all-hadronic $t\bar{t}$ decay channel at $\sqrt{s} = 8$ TeV with the ATLAS detector*, JHEP **09**, 118, doi:10.1007/jhep09(2017)118.

[41] J. Erdmann, S. Guindon, K. Kroeninger, B. Lemmer, O. Nackenhorst, A. Quadt and P. Stolte, *A likelihood-based reconstruction algorithm for top-quark pairs and the KLFitter framework*, Nucl. Instrum. Meth. A **748**, 18, doi:10.1016/j.nima.2014.02.029.

[42] A. Shmakov, M. J. Fenton, T.-W. Ho, S.-C. Hsu, D. Whiteson and P. Baldi, *Spanet: Generalized permutationless set assignment for particle physics using symmetry preserving attention*, SciPost Phys. **12**(5), 178, doi:10.21468/SciPostPhys.12.5.178.

[43] The ATLAS Collaboration, *Measurements of top-quark pair single- and double-differential cross-sections in the all-hadronic channel in pp collisions at $\sqrt{s} = 13$ TeV using the ATLAS detector*, JHEP **01**, 33, doi:10.1007/jhep01(2021)033.

[44] The ATLAS Collaboration, *Top-quark mass measurement in the all-hadronic $t\bar{t}$ decay channel at $\sqrt{s} = 8$ TeV with the ATLAS detector. Top-quark mass measurement in the all-hadronic $t\bar{t}$ decay channel at $\sqrt{s} = 8$ TeV with the ATLAS detector*, JHEP **09**, 118, doi:10.1007/jhep09(2017)118.

[45] A. L. Maas, A. Y. Hannun, A. Y. Ng *et al.*, *Rectifier nonlinearities improve neural network acoustic models*, In *Proceedings of The International Conference on Machine Learning*, vol. 30, p. 3. PMLR.

[46] J. L. Ba, J. R. Kiros and G. E. Hinton, *Layer normalization*, doi:10.48550/arXiv.1607.06450, 1607.06450.

A Network Structure

Conditional Attention Deep Set

Several different methods for extracting variables from the jet container were studied in the development of ν-Flows, including manually constructing observables, and passing through flattened jet data through a dense network. We found that the Deep Set, specifically with attention pooling, performed considerably better.

Our Deep Set contains three dense networks, the `Feature Net`, the `Attention Net`, and the `Final Net` as shown in Figure 7. The jet variables from Table 1 are passed separately through the `Feature Net` to extract representations per jet f_i, and separately through the `Attention Net` to extract a weight per jet w_i. We then combine these outputs to perform a weighted sum of the representations of the N jets in each event.

$$F = \sum_{i}^{N} w_i \cdot f_i.$$

The result is then passed through the `Final Net` to obtain the extracted features of the entire jet container. Conditional information from the $\vec{p}_{\mathrm{T}}^{\,\mathrm{miss}}$, lepton, and Misc variables are provided to each of the dense networks by concatenating them together with the jet inputs. The `Attention Net` produces a positive definite weight by applying an exponential activation function in the final layer.

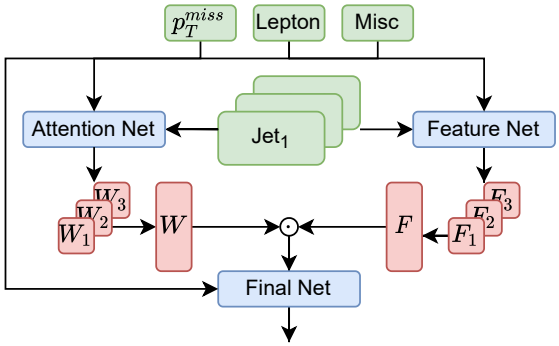


Figure 7: The attention weighted Deep Set for the jet container.

cINN Layer

Many different configurations for the cINN were tested over the course of this work. Combining conditional coupling layers, with rational-quadratic spline transformers [14], and Lower-Upper triangular (LU) decomposed linear layers resulted in the best-observed performance at reconstructing the neutrino three momenta. This block is shown in Figure 8. The cINN is constructed of seven alternating coupling layers. In the very first coupling layer of the flow, we split the neutrino three-momentum by selecting the transverse coordinates for X_A and the longitudinal coordinate for X_B. We then alternate this splitting with each subsequent coupling layer. We found that the masking order did have an impact on the final performance. Conditioning information is provided to the network by concatenating the extracted high-level features from the FF module to the inputs of the `Spline Net`. The python package *nflows* is used to construct the cINN.

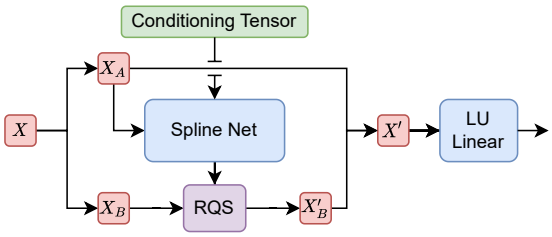


Figure 8: The building blocks of the conditional invertible neural network in ν-Flows.

Dense Network Hyperparameters

The ν-Flows model in Figure 2 contains 5 different types of dense network. The three networks in the Deep Set, an `Embedding Network`, and a `Spline Net` in each layer of the cINN. The hyperparameters were determined by several grid searches using reconstruction performance on a validation set. All dense networks have two hidden layers of 64 nodes each. Each hidden layer applies the LeakyReLU [45] activation function with a slope parameter of 0.1 and Layer-Normalisation [46]. Additive residual connections are used between each hidden layer. Conditional information is injected into the dense networks by concatenating the context tensors to the inputs.

The ν-FF network uses the same structure as the FF component of ν-Flows but with an `Embedding Network` with 4 hidden layers and an output layer with three nodes, corresponding to the neutrino three-momentum.

B Additional Plots

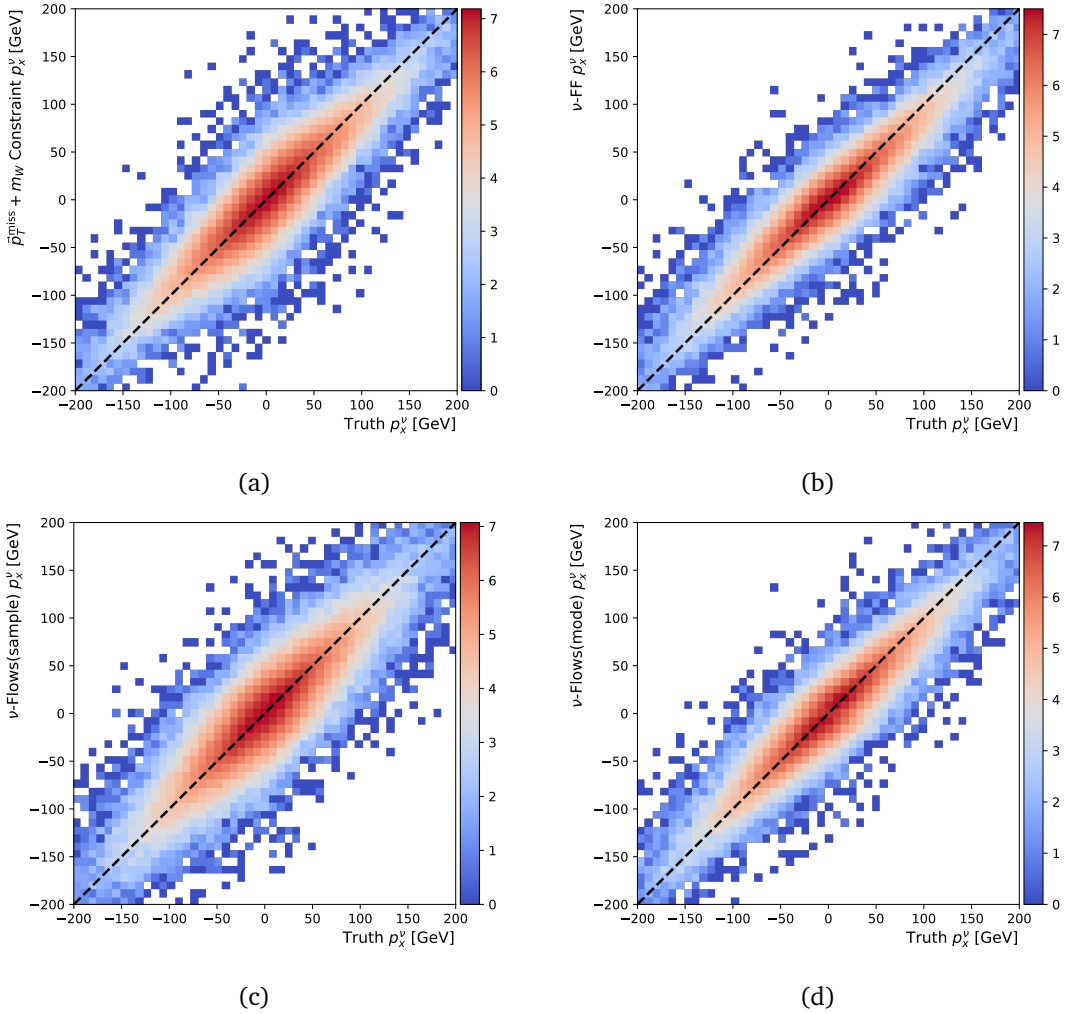


Figure 9: Two dimensional histograms showing the reconstruction performance of p_x^ν using both solutions of the m_w kinematic constraint (a), ν-FF, (b), ν-Flows(sample) (c), and ν-Flows(mode) (d). In each plot, the true value is plotted along the x-axis and the reconstructed value is plotted along the y-axis. The diagonal line represents ideal reconstruction. The z-axis is the natural logarithm of the counts in each bin. The p_y^ν distribution results were virtually identical to these.

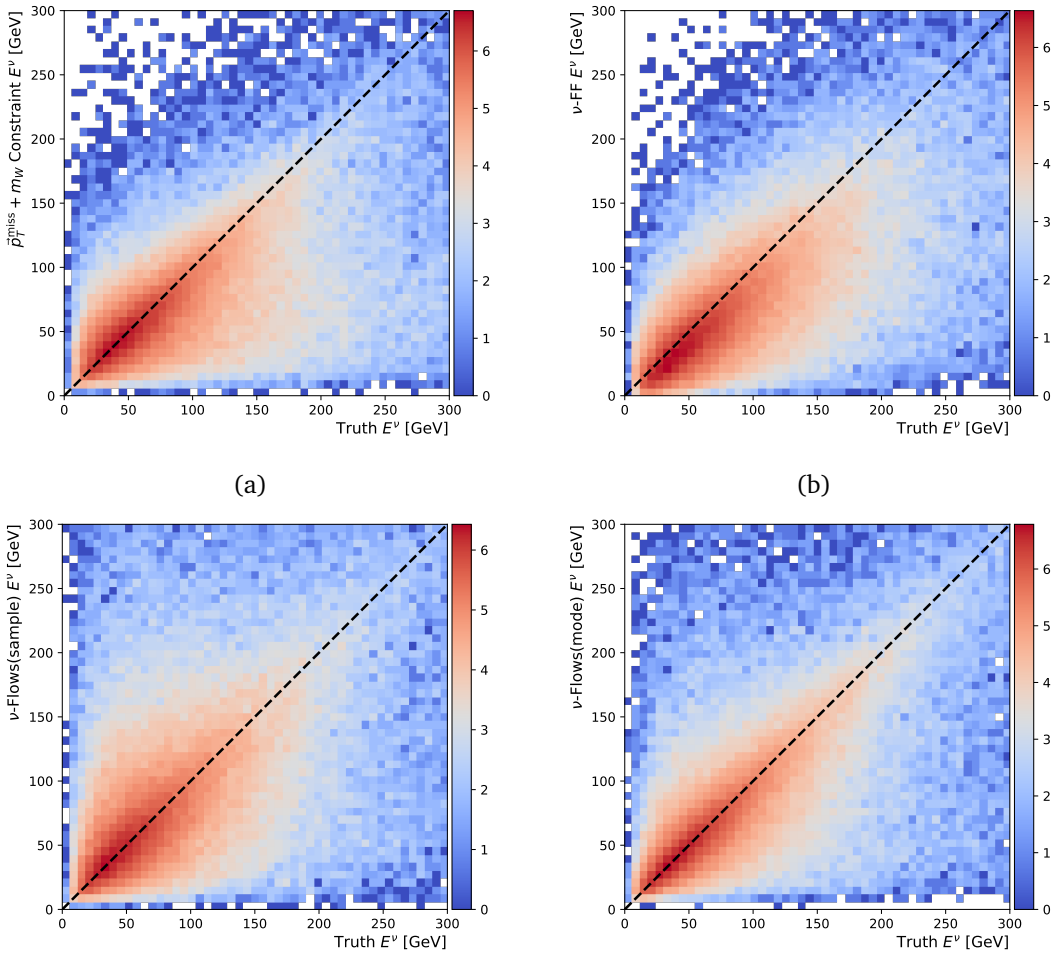


Figure 10: Two dimensional histograms showing the reconstruction performance of the neutrino energy using both solutions of the m_w kinematic constraint (a), ν-FF, (b), ν-Flows(sample) (c), and ν-Flows(mode) (d). In each plot, the true value is plotted along the x-axis and the reconstructed value is plotted along the y-axis. The diagonal line represents ideal reconstruction. The z-axis is the natural logarithm of the counts in each bin.

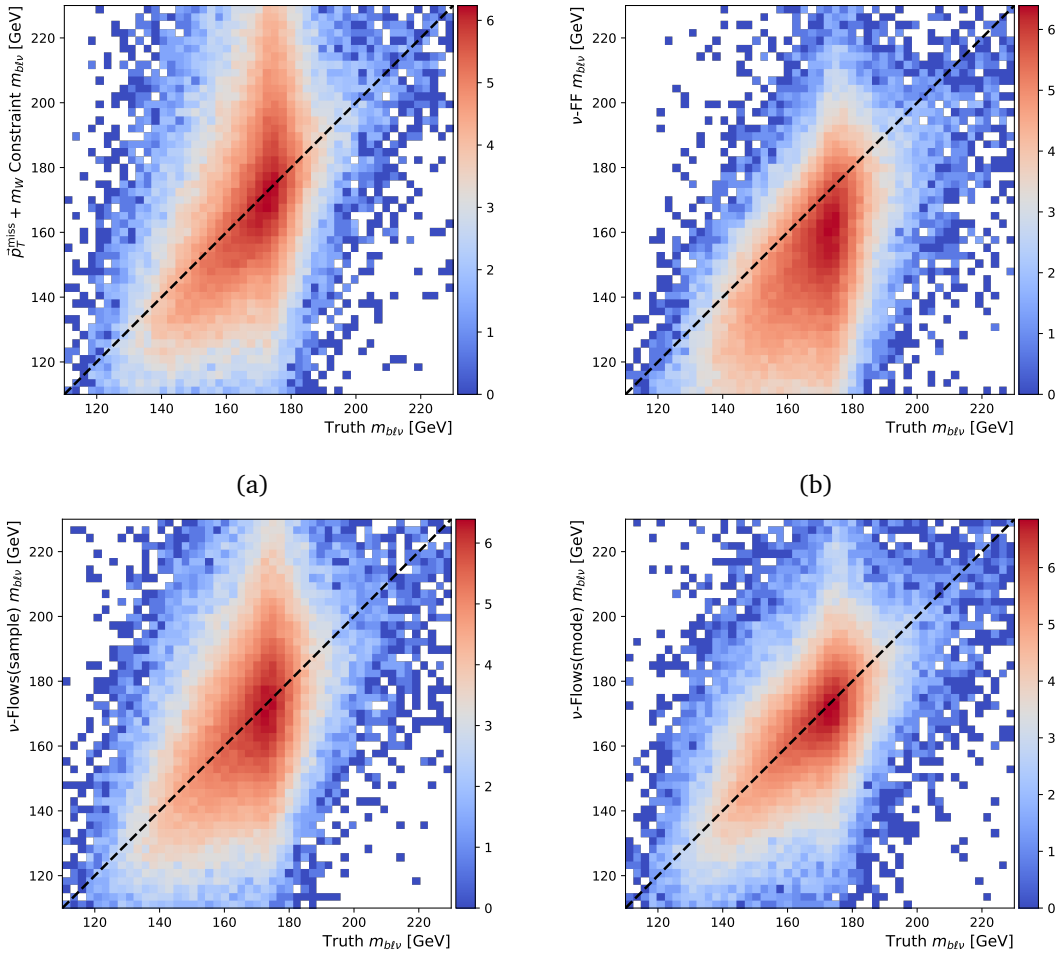


Figure 11: Two dimensional histograms showing the reconstruction performance of the t_{lep} mass using both solutions of the m_w kinematic constraint (a), ν-FF, (b), ν-Flows(sample) (c), and ν-Flows(mode) (d). In each plot, the true value is plotted along the x-axis and the reconstructed along the y-axis. The correct b-jet is used. The diagonal line represents ideal reconstruction. The z-axis is the natural logarithm of the counts in each bin.

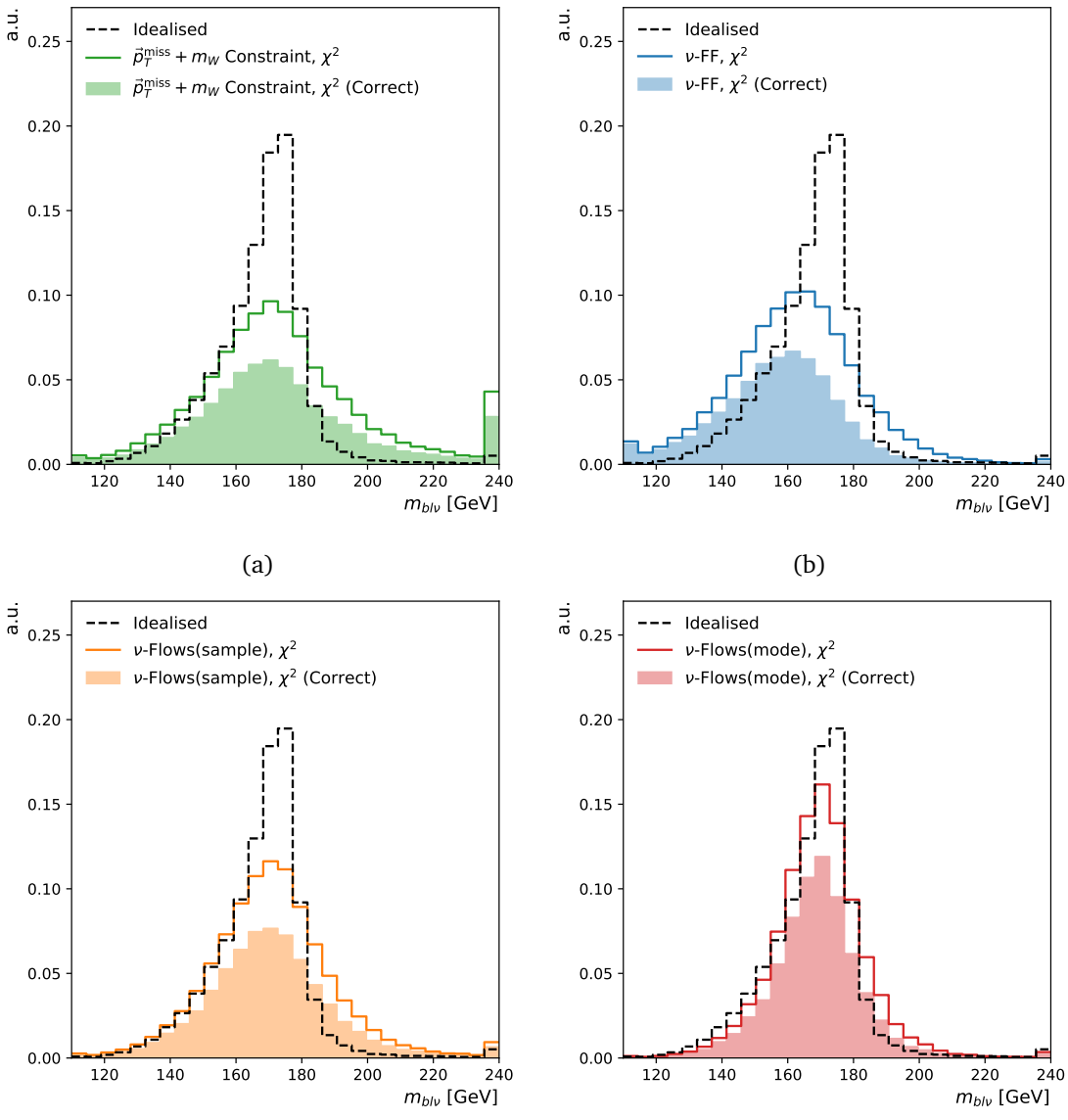


Figure 12: The reconstructed invariant mass of b_{lep} using (a), ν-FF, (b), ν-Flows(sample) (c), and ν-Flows(mode) (d). In each colored plot the b-jet is selected using the χ^2 method. The *Idealised* curve uses both the true neutrino and the correct b-jet. The shaded plots show the subset of data for which the χ^2 method identified the correct b-jet.