Quantifying the energy scale of hadronically decaying tau leptons at the ATLAS detector using a mixture density network (MDN)

Miles Cochran-Branson*
Physics Department, Lawrence University

Quentin Buat†
Physics Department, University of Washington

(Dated: August 27, 2022)

The tau lepton is of great interest in studying the Higgs boson through the decay channel $H \to \tau^+\tau^-$. Experimental limitations, however, prevent us from exploring this decay channel easily. One step in analyzing tau leptons at the ATLAS detector is quantifying their energy scale. In particular, accurate energy scale calibration will help both in understanding the Higgs and tau lepton better. In the following paper, we explore a novel approach to calibrating the tau energy scale (TES) using neural networks. We compare the current method using a boosted regression tree (BRT) with a mixture density network (MDN). We find that the MDN outperforms the current method for tau energy scale calibration while also providing extra information about the quality of prediction for each tau in the detector.

Keywords: High energy physics, Machine Learning, Mixture Density Networks, Higgs, tau lepton

I. INTRODUCTION

The tau lepton, heavy cousin to the electron, is an excellent probe of the Higgs mechanism. Of the leptonic decays of the Higgs boson, the $H \to \tau\tau$ has the highest branching ratio. This makes the tau a natural probe of the Higgs boson coupling to leptons [1]. In run two of pp collision data at $\sqrt{s} = 13\,\text{TeV}$ at the Large Hadron Collider (LHC), there are a predicted 500×10^3 such events [2]. As a result, we have a significant source of data for testing, allowing us to discriminate between high purity and low purity taus. There are, however, many experimental challenges in exploring this decay channel. This paper addresses one of these challenges: calibration of the energy scale of tau leptons in the ATLAS detector.

The tau lepton has a mass of 1.777 GeV. Because of it's large mass, the tau decays nearly instantly in the detector with a proper decay length of 87µm [1]. The first active layer of ATLAS is at 33mm, thus we must look for the decay products of the tau. The tau decays in two primary ways: leptonically and hadronically. Leptonic decays come in the form of $\tau \to \nu_\tau \nu_\ell \ell$ where $\ell = e, \mu$, and hadronic decays come in the form $\tau \to \nu_\tau$hadrons. The hadronic decays have a 65% branching ratio and it is these decays which we will discuss. Hadronic decays of taus primarily involve pions, where the majority include one charged pion (∼50% of all possible tau decays) while the next highest decay channel includes three charged pions (∼15% of all possible tau decays). Therefore, taus look like a jet of pions in the detector with either one or three tracks as shown in Fig. 1. Neutrinos from the decay cannot be reconstructed, thus the only particles visible to us are the pions. We will call the visible decay products of hadronic taus $\tau_{\text{had-vis}}$.

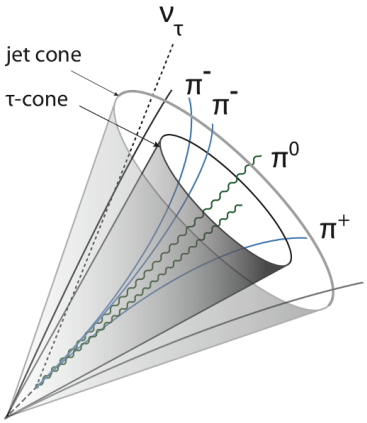

FIG. 1: The decay of a tau lepton to charged and neutral pions looks like a jet in the detector with an inner cone of visible products—the τ cone—and an outer cone of neutrinos—the jet cone.

The ATLAS detector [3] at the LHC is one of the two multi-purpose detectors at CERN. It is a cylindrical detector with several active layers. The first layers in the detector, the tracker, can detect charged particles. A large superconducting solenoid magnet curves the path of particles which can then be detected by the tracker system. Outside these layers are two sets of calorimeters: electronic and hadronic. These can detect most particles including neutral particles. At the very edge of the detector lie muon chambers where tracks of muons can be detected. Muon paths are bent by a superconducting torodial magnet. Only neutrinos cannot be seen by ATLAS, however their presence can be inferred from missing energy in collisions. This is precisely measured in the transverse plane as LHC physics is invariant in this plane.

* mcochranbranson@gmail.com
† https://phys.washington.edu/people/quentin-buat

An important part in understanding how the tau behaves in the detector is quantifying its energy scale. We define the tau energy scale (TES) to be an estimate of the transverse momentum of the tau which we will label $p_T(\tau_{\text{had-vis}})$. The current method for finding this variable takes data from the calorimeter and tracker and combines the estimate from these into a variables called $p_T^{\text{Comb.}}$. As a final step, this data is run through a Boosted Regression Tree (BRT) with a target of $p_T^{\text{truth}}/p_T^{\text{Comb.}}$. The algorithm developed for this purpose is described in detail in [4]. This work has been focused on replacing the BRT with neural networks. Because it has been shown that modern methods of machine learning beat those of old as in [5], we expect that neural networks should perform better than a BRT. Additionally, we use a Mixture Density Network (MDN) to provide us a probabilistic interpretation of our calibration [6]. This gives us the added bonus of being able to quantify the *quality* of our prediction for p_T. Moreover, MDNs have been used with great success both in ATLAS such as [7], and in other fields and industry such as [8–10] where they show excellent performance over old methods of machine learning.

In the following paper, we demonstrate that neural networks beat the current methods for tau energy scale calibration. We also show how, by using the output from a Gaussian mixture model in the neural network, we can extract data on the quality of each prediction of p_T for any event we feed into our network.

II. METHODS

High energy physics (HEP) at the LHC relies heavily on simulation. In particular, to see new physics, we must first simulate what we would like to see such as an $H \to \tau\tau$ event. These simulations are done typically with Monte Carlo particle production simulations such as PYTHIA [11]. Simulated events are then passed through a very sophisticated simulation of the ATLAS detector powered by Geant4 [12–14]. Finally, these signals are digitized and treated as real data. In developing an algorithm for calibration of the tau energy scale, we will work with simulated data. This allows us to perform machine learning with a set of target data determined by true values given from simulation. Once this algorithm is well understood, we will test with the well-understood dataset $Z \to \tau\tau$ before searching for new physics.

The algorithm we are developing uses machine learning and neural networks to describe the p_T spectrum of tau leptons. A neural network is a collection of nodes and hidden layers connected to each other by weights. These components act as a function: they take some input parameters and return an output based on how it has been trained. In order to train a neural network, we provide the network with training data containing expected input data and truth data. The network computes weights connecting nodes between layers by taking the input data and comparing it to the truth. This is done by evaluating a function called the loss function. The network can progress towards a solution to this function by computing its gradient and moving towards a minimum. In our work, we use a mixture density network. In this case, we feed data into a neural network and take the output parameters as inputs to a mixture density model. This gives us a probabilistic interpretation of the network output.

A mixture density model is simply a linear combination of conditional probability densities, i.e., our model looks something like

$$p(t|x) = \sum_{i=1}^{k} \pi_i \, \phi_i(t|x) \tag{1}$$

where $\phi_i(t|x)$ are conditional probability densities with input x and output t, and π_i are mixing parameters satisfying

$$\sum_{i=1}^{k} \pi_i = 1 \quad \text{and} \quad \pi_i \leq 1.$$

We will use Gaussian Mixture Models (GMM) which are simply a linear combination of k Gaussian distributions, i.e.,

$$\phi_k(t|x) = \sum_{i=1}^{k} \pi_i \, \phi_i(t|x; \mu_i, \sigma_i), \tag{2}$$

where μ_i are the means of each Gaussian distribution and σ_i are the standard deviations. We will need to find some global mean and standard deviation in our calculations. These are given by

$$\mu_{\text{global}} = \sum_{i=1}^{k} \pi_i \mu_i, \tag{3}$$

and

$$\sigma^2_{\text{global}} = \sum_{i=1}^{k} \pi_i(\sigma_i^2 + \mu_i^2) - \mu^2_{\text{global}} \tag{4}$$

as determined in [15].

An essential piece in obtaining a probabilistic output from the model is the definition of the loss function. This is extensively covered in [6]. The central principle is to take the residuals of the mixture density network and apply the negative logarithm to this, i.e., our loss function is defined as

$$E = \sum E^q \tag{5}$$

where the contribution from event q is given by

$$E^q = -\ln\left(\sum_{i=1}^{k} \pi_i \, \phi_i(t^q|x^q)\right). \quad (6)$$

In the following section we explain our implementation of this architecture in order to properly calibrate the tau energy scale.

III. NETWORK ARCHITECTURE AND MODELING

In order to quantify the energy spectrum of tau leptons in the ATLAS detector, we use a novel machine learning algorithm built with the `keras` [18] architecture on top of `tensorflow` [19]. These frameworks allow us to easily model with MDNs. The current method of p_T calibration of tau lepton uses standard techniques to estimate the tau energy scale (TES), described in detail in [20]. In particular, a first order estimate of the p_T is given by combining data from the tracker and calorimeter into a variable called $p_T^{\text{Comb.}}$. In our plots we will call this variable *combined*. This variable is then fed into a Boosted Regression Tree (BRT) with a target of $p_T^{\text{Truth}}/p_T^{\text{Comb.}}$. This is done as performing regression on truth data yields a more descriptive model for p_T prediction.

This work is focused on replacing the existing BRT with an MDN in order to improve the performance of the algorithm. Moreover, with an MDN we can obtain an estimate of the quality of each predicted $\tau_{\text{had-vis}}$ candidate

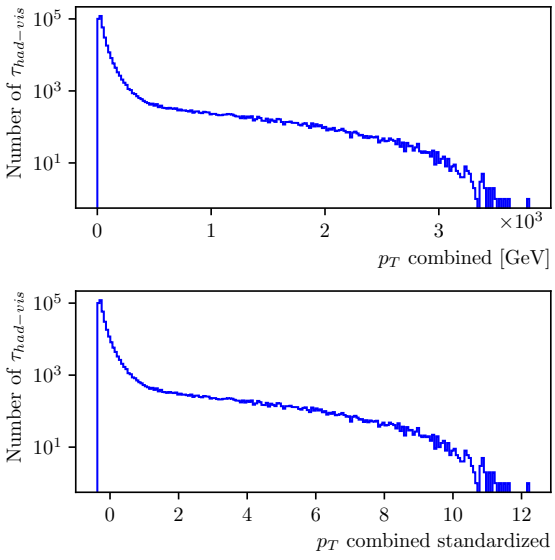

FIG. 2: Example of a variable standardized. Here we show $p_T^{\text{Comb.}}$ before (top panel) and after (bottom panel) standardization. Notice that the shape of the distribution is preserved yet this distribution is shifted and squeezed such that the mean is zero and the standard deviation is one.

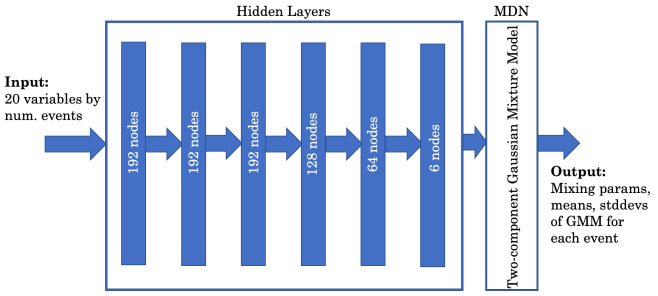

FIG. 3: Network architecture of MDN. The dense neural network contains six hidden layers each with successively fewer nodes. The last six node layer passes its weights to a two-component Gaussian mixture model which in turn gives us a probabilistic interpretation of the network prediction.

via the global standard deviation. Essentially, our network learns both the expected value of p_T as well as the uncertainty in this prediction. Obviously this is hugely advantageous as we can select events based on quality of prediction. Moreover, use of a modern form of machine learning paves the way for future networks to learn directly from data from the tracker and calorimeter instead of using combined estimates.

Variables used for training the MDN are given in Table I. Variables marked as standardized have been modified such that the mean of each is zero and the standard deviation is one via

$$x_{\text{standardized}} = \frac{x_{\text{old}} - \mu}{\sigma} \quad (7)$$

where μ is the mean of the variable and σ the standard deviation. An example of a standardized variable used—p_T combined—is shown in Fig. 2. Variables that are not standardized are either probabilities, only used in the BRT, or are ratios of two variables. For instance, the Neural Network decay mode variables are probabilities and as such not standardized.

The network we chose is shown in Fig. 3. The dense network part of this network is nearly identical to previous attempts at using neural networks for tau energy scale calibration. Preliminary results using recursive neural networks in 2019 show that a neural network has promise. These results, however, did not beat the BRT. We have changed the network used via addition of the Gaussian mixture output layer. The Gaussian mixture model we use has two components, thus we must utilize the methods described above to obtain global mean and standard deviations to get predictions from this model.

In training the model, we performed a coarse scan of hypterparameters. In particular, we scanned for optimal learning rate and batch size evaluated on a) if the network actually produced results and b) how the network compared in performance to the BRT output. For learning rate we scanned the range $[10^{-2}, 10^{-15}]$ and for batch

Variables Used	Only in NN	Only in BRT	Standardized	Description of Variable
μ			x	Average interaction per bunch crossing
$n_{\texttt{vertices}}$			x	Number of nominal vertices in each event
$n_{\texttt{tracks}}$		x		Number of tracks
n_{π^0}		x		Number of neutral pions in event (Reco)
ρ	x		x	Measure of pileup in detector
$\langle\lambda_{\text{center}}\rangle$			x	Distance of the cluster shower center from the calorimeter front face measured along shower axis
$\langle E/V\rangle$			x	Cluster first moment in energy density. E and V represent the energy and volume of each cluster
$\langle\lambda^2\rangle$			x	Cluster second moment
$\langle f_{\text{presampler}}\rangle$				Fraction of cluster energy deposited in barrel and endcap calorimeters.
P_{EM}				Probability of the cluster to be EM-like
$p_T^{\texttt{Comb.}}$			x	Transverse momentum interpolated from calorimetric corrections to energy measurement and TPF reconstruction
$p_T^{\texttt{Calo}}$	x		x	First order calorimeter estimate of TES
$\eta^{\texttt{TPF}}$	x		x	Pseudorapidity from TPF reconstruction
$\eta^{\texttt{Calo.}}$		x		Pseudorapidity from calorimeter reconstruction
γ_π				Relative difference in total energy of the charged pion and total neural pion energy: $\gamma_\pi = (E_{\text{charged}} - E_{\text{neutral}})/(E_{\text{charged}} + E_{\text{neutral}})$
$p_T^{\texttt{EM}}/p_T^{\texttt{LC}}$				Ratio of calorimetric energy at the EM scale to the local hadron calibration transverse momentum p_T^{LC}
$p_T^{\texttt{LC}}/p_T^{\texttt{Comb.}}$				Ratio of the local hadron calibration transverse momentum to $p_T^{\text{Comb.}}$
$p_T^{\texttt{TFP}}/p_T^{\texttt{Comb.}}$				Ratio of the TPF reconstruction transverse momentum to $p_T^{\text{Comb.}}$
`TPF BDT values`		x		BDT values of the TPF algorithm for 1p0n vs 1p1n, 1p1n vs 1pXn, and 3p0n vs 3pXn
`NN Decay Mode Prob Values`	x			NN decay mode probabilities for discriminations 1p0n, 1p1n, 1pXn, 3p0n, and 3pXn

TABLE I: Input variables used in calibrating tau energy scale. Variables used in the current algorithm using a BRT and the proposed algorithm are presented. Standardized variables are transformed such that the mean of the sample is zero and the standard deviation is one. For more information on cluster variables, see [16]. For more information on tau particle flow (TPF) variables see [17].

size we scanned the range $[16, 128]$. A two-dimensional scan was performaed on a three-by-three grid to determine a rough range of values that worked well. Additional tuning was then done variable by variable. We found that a learning rate of 10^{-5} and a batch size of 64 gave the best results. Additionally, extensive work was invested in finding which variables to standardize and which cuts were necessary. Outliers were removed with

$$\begin{aligned} p_T^{\texttt{EM}}/p_T^{\texttt{LC}} &< 25 \\ p_T^{\texttt{LC}}/p_T^{\texttt{Comb.}} &< 25 \\ p_T^{\texttt{TFP}}/p_T^{\texttt{Comb.}} &< 25. \end{aligned}$$

Outliers in the above variables arose from division by numbers very close to zero. Because $p_T^{\text{Comb.}}$ was standardized, many events were shifted close to zero thus creating on the order of 20 outliers removed by these cuts.

Training was done on the simulated dataset $\gamma^* \rightarrow \tau\tau$ which can be found in the ATLAS database.[1] This dataset contains $\sim 1.26 \times 10^7$ events. Each event contains approximately two taus from decay of the γ^*. The network was trained on a third of the available taus of which 20% were used for validation ($\sim 2.50 \times 10^6$ taus) and 80% for training ($\sim 1.25 \times 10^7$ taus). Training usually reached on optimum within 100 epochs. Early stopping was employed with a tolerance of 20 epochs. Total training time was approximately 18 hours on an lxplus-gpu node. Testing was done on the other third of the taus

[1] group.perftau.MC20d_StreamTES.425200.Pythia8EvtGen_A14 NNPDF23LO_Gammatautau_MassWeight_ v3_output.root.

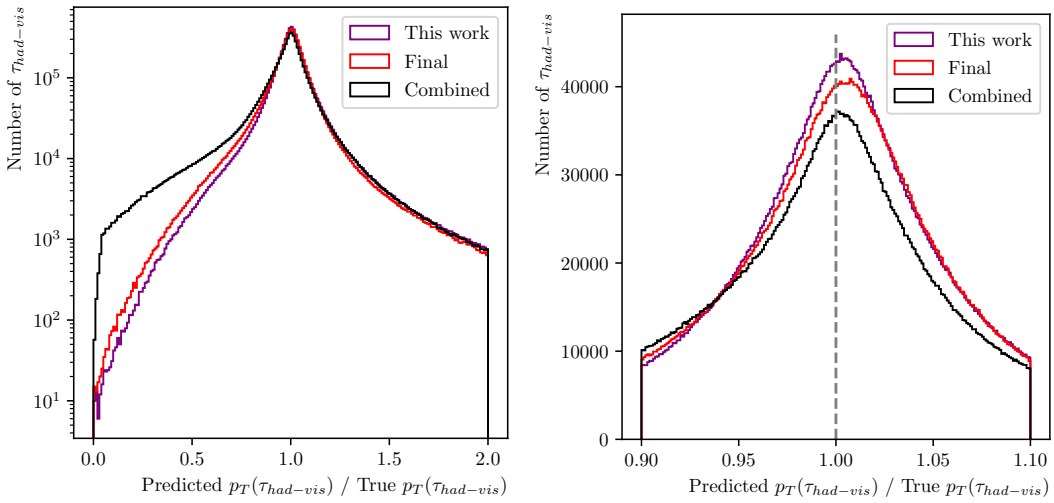

FIG. 4: Response lineshape for all $\tau_{had\text{-}vis}$ candidates. In (a) the plot of the ratio is shown in log scale between zero and two while in (b) we zoom in on the peak, plotting the ratio between 0.9 and 1.1 in linear scale.

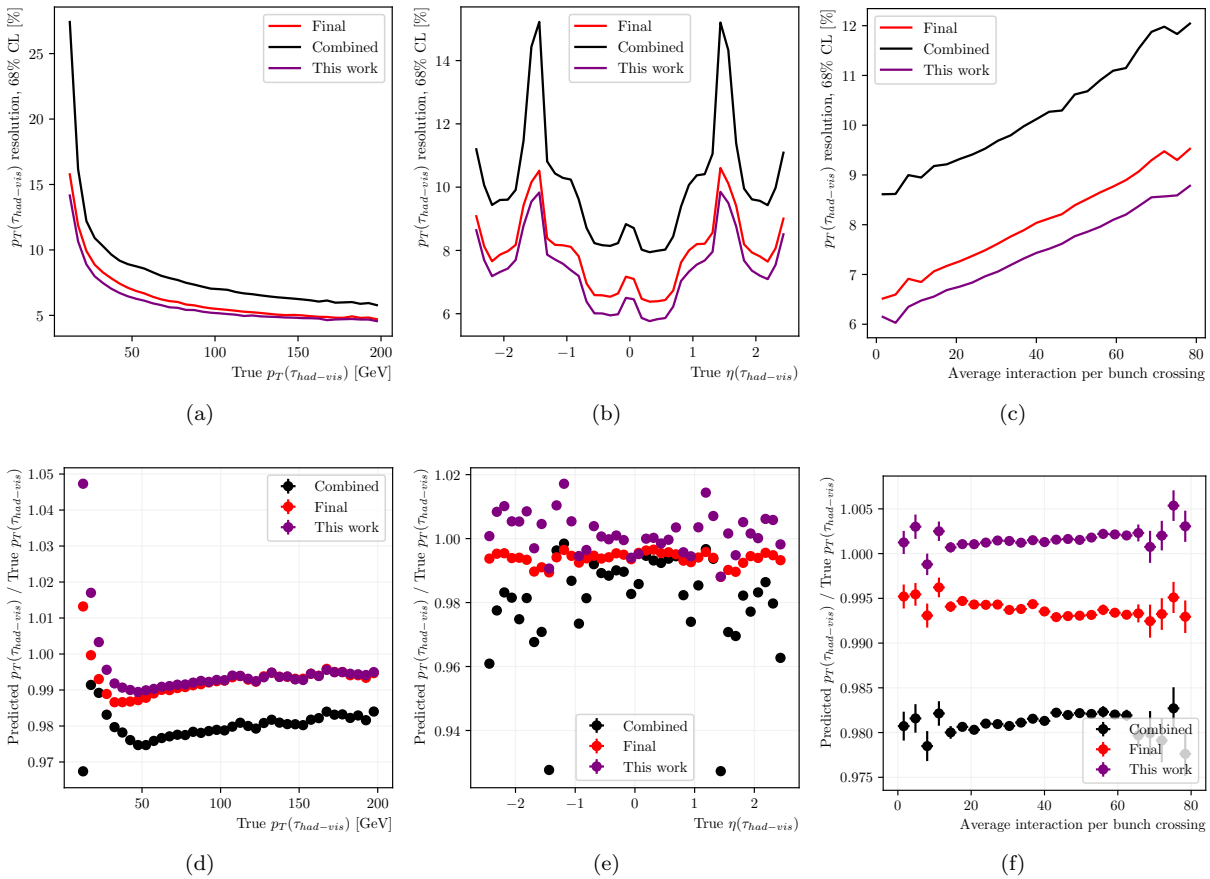

FIG. 5: Resolution and response as a function of p_T, η, and average interaction per bunch crossing. Resolution is determined by taking the width of the plot of the ratio as in Fig. 4 at 68% confidence level. Response is determined by computing the mean of the distribution in Fig. 4. Error bars in response are determined from statistical uncertainties and errors in p_T are determined by the width of each bin.

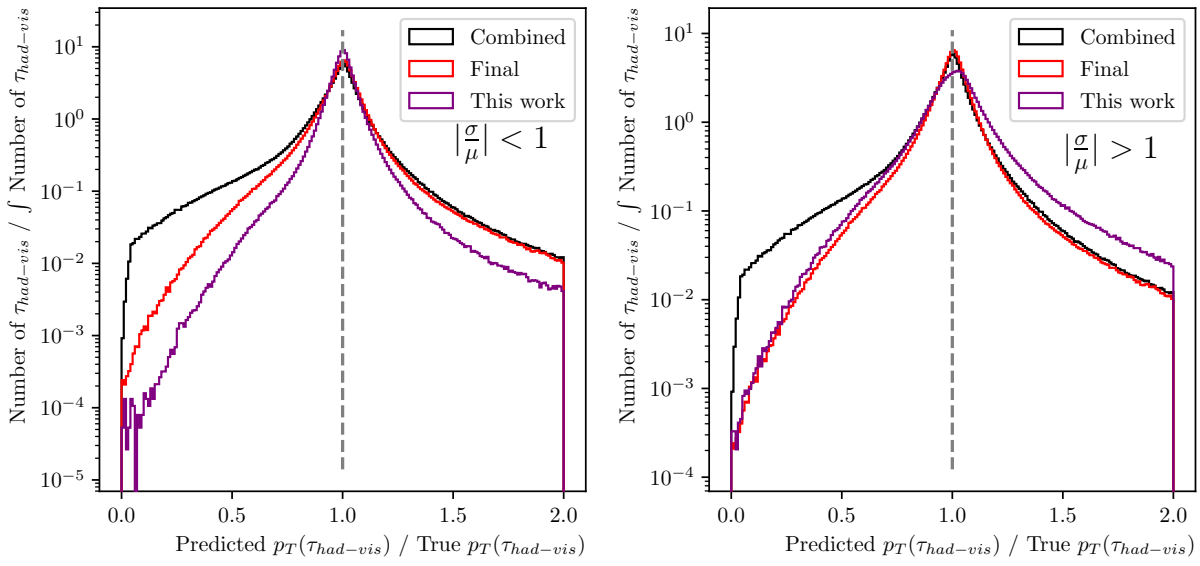

FIG. 6: Response lineshape for better and worse events determined by evaluating $|\sigma/\mu| < 1$ for each $\tau_{\text{had-vis}}$ candidate. The plot in (a) shows better events with a very sharp peak around one which decays quickly, while (b) shows worse events with a very broad peak.

($\sim 6.24 \times 10^6$ taus). The results of this testing are discussed below.

IV. RESULTS

We evaluate the performance of the network on multiple metrics in comparison to the final performance of the current method of TES calibration. In Fig. 4 we show the response of the network in comparison to both the BRT (final, plotted in red) and $p_T^{\text{Comb.}}$ (combined, plotted in black). The response is simply the predicted p_T divided by the truth p_T from simulation. Desirable behavior is a highly peaked distribution that tapers away as steeply as possible from the peak. Although the distribution is not required to peak at one, this is easier to interpret than a distribution which is systematically shifted from one. Fig. 4 shows most of the response distribution. From this plot we can clearly see that the MDN beats the final as the distribution is more highly peaked around one. Moreover, the width of the distribution at 68% confidence level is 8.7% for final while the MDN produces a curve of width 8.3%.

In order to better understand the performance of the model we can view this distribution in two other ways. First, we define the *resolution* as the width of the response distribution in Fig. 4 at 68% confidence level. We can then evaluate the resolution curve as a function of true p_T, true η, and average interaction per bunch crossing. This is done by working with the response distribution in bins of p_T, η, and average interaction per bunch crossing. Desirable performance is to have as thin a peak as possible for all bins, thus we want the resolution curve to be as low as possible. We plot these curves in Fig. 5(a,b,c). As shown in these curves, performance of the MDN is better than the BRT as a function of all variables plotted.

We also take a closer look at the performance by plotting the mean of the response curve in Fig. 4 as a function of true p_T, true η, and average interaction per bunch crossing as shown in Fig. 5(d,e,f). Error bars in response are determined from statistical uncertainties while error bars in the x-variable indicate the width of each bin. Desirable behavior for this plot is to be as close to one as possible. As a function of p_T, the MDN beats Final except at very low p_T—the first bin in this plot contains the p_T range $[10, 15]$ GeV. After this bin, response is comparable to or better than final. Response as a function of η and average interaction per bunch crossing performs better than final. Thus, we find that the MDN does indeed perform better than the BRT.

As mentioned previously, one advantage of using an MDN is that for each prediction by the network we have some information about the quality of this prediction. This is obtained by using the global standard deviation computed by taking the square root of Eqn. 4. We can then find better events by evaluating the inequality

$$\left|\frac{\sigma}{\mu}\right| < 1. \tag{8}$$

It follows that worse events will have $|\sigma/\mu| > 1$. We can

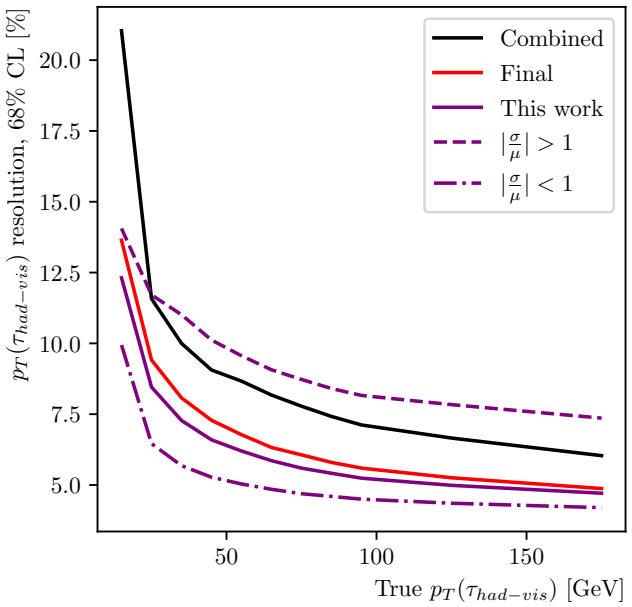

FIG. 7: Resolution plot with splitting of better and worse events evaluated by $|\sigma/\mu| < 1$. Better events have very low resolution, while worse events have resolution above combined variables.

visualize this splitting by looking at the response curves for better and worse events as shown in Fig. 6. In Fig. 6(a) we have plotted the better events in comparison to final and combined. Of the better events, the width of the MDN response at 68% confidence level has a width of 5.7%. In contrast, the worse events shown in Fig. 6(b) have a width at 68% confidence level of 13.5%. For comparison, the width of final response is 8.7%. Notice that in both cases this distribution is normalized so the integral of the histogram is one in order to better show comparison to final and combined.

We can further examine this splitting by plotting the resolution as a function of true p_T for better and worse events. This is shown in Fig. 7. Here we can clearly see the splitting in resolution between better and worse events. For $|\sigma/\mu| < 1$ the resolution curve is significantly lower than final ranging from about 3% below final in the low p_T regime to 1% below final at higher p_T. Events with $|\sigma/\mu| > 1$ perform worse compared to final except in the low p_T region ranging from <1% worse than Final to slightly greater than 2% above Final.

V. DISCUSSION AND CONCLUSIONS

We have shown that the novel mixture density network for calibration of the tau energy scale performs better than the current method which uses a boosted regression tree. We have used multiple metrics to evaluate both networks. Plots of response, and plots of resolution and response as a function of true p_T, true η, and average interaction per bunch crossing, demonstrate significant performance gains in using an MDN over a BRT. Moreover, with an MDN we are able to obtain an estimate of the quality for each predicted $\tau_{\text{had-vis}}$ candidate. This allows us to split events by better and worse performance by evaluating $|\sigma/\mu|$ where σ is the global standard deviation given by the mixture model, and μ is the global mean. The ability to split events can be very valuable. For instance, we can see where in the detector we perform worse in estimating the energy scale of taus. Moreover, in future analysis, as a result of having such a wealth of tau data, we can even split the data from an $H \to \tau\tau$ analysis into better and worse $\tau_{\text{had-vis}}$ candidates.

Currently, a final round of validation is being implemented in order to merge this work with the ATLAS software. This will provide all collaborators access to our algorithm and will allow them to harvest the extra piece of information provided by the global standard deviation given by our network. While the use of a prediction of uncertainty for each p_T estimate of $\tau_{\text{had-vis}}$ candidates is most certainly useful, the precise utility of this has yet to be fully realized.

Because we have shown that an MDN outperforms a BRT, we can now begin to explore the possibility of removing combined variables from the workflow. This means building a more complex network to learn directly from tracker and calorimeter data. A preliminary exploration of this seemed to yield promising results but did not approach the performance achieved above. The utility of exploring such an algorithm becomes apparent when considering the TES calibration workflow. Notice that in order to obtain an estimate of the TES, we first must obtain estimates from several other algorithms such as the tau particle flow algorithm for predicting tracks in the detector. This means, with each new simulated dataset, parameters from all of these algorithms must be re-tuned. If we can train a network without these parameters we will have a more efficient workflow and may even be able to gain better performance.

ACKNOWLEDGMENTS

We wish to thank the University of Washington Institute of Nuclear Theory (INT) and the National Science Foundation (NSF) for funding this research. Additionally, I would like to thank Quentin Buat for his invaluable mentorship and support throughout the UW REU program.

Appendix: Supplemental material

For more information and additional plots, see the following presentations of this work:

- For more detailed plots on performance, refer to the tau working group plenary meeting presenta-

tion [21]. Slides can be found on the CERN indico at this link.

- For a discussion of machine learning techniques used, see the EPE-ML plenary meeting [22]. Slides and a recording of the talk can be found on the CERN indico at this link.
- For a broad overview of the project, and an introduction to the basics of machine learning see the REU final presentation [23]. Slides can be found on the University of Washington REU website.

Code from our analysis can be found on GitHub in qbaut/taunet.

[1] ATLAS Collaboration, Evidence for the Higgs-boson Yukawa coupling to tau leptons with the ATLAS detector, Journal of High Energy Physics **2015**, 117 (2015), arXiv:1501.04943 [hep-ex].

[2] ATLAS Collaboration, Measurements of Higgs boson production cross-sections in the $H \to \tau^+\tau^-$ decay channel in pp collisions at $\sqrt{s} = 13\,\text{TeV}$ with the ATLAS detector (2022), arXiv:2201.08269 [hep-ex].

[3] ATLAS Collaboration, The ATLAS Experiment at the CERN Large Hadron Collider, Journal of Instrumentation **3** (08), S08003.

[4] ATLAS Collaboration, *Measurement of the Tau Lepton Reconstruction and Identification Performance in the ATLAS Experiment Using pp Collisions at* $\sqrt{s} =$ 13 TeV, Tech. Rep. ATLAS-CONF-2017-029 (ATLAS-COM-CONF-2017-015, 2017).

[5] ATLAS Collaboration, Identification of hadronic tau lepton decays using neural networks in the ATLAS experiment, ATLAS PUB Note **ATL-PHYS-PUB-2019-033** (2019).

[6] C. M. Bishop, *Mixture Density Networks*, Technical NCRG/94/004 (Aston University, Birmingham, U.K., 1994).

[7] Elham E. Khoda and ATLAS Collaboration, ATLAS pixel cluster splitting using Mixture Density Networks, in *Proceedings of 7th Annual Conference on Large Hadron Collider Physics — PoS(LHCP2019)* (Sissa Medialab, Puebla, Mexico, 2019) p. 009.

[8] A. E. Lovell, A. T. Mohan, T. M. Sprouse, and M. R. Mumpower, Nuclear masses learned from a probabilistic neural network, Physical Review C **106**, 014305 (2022).

[9] H. Zen and A. Senior, Deep mixture density networks for acoustic modeling in statistical parametric speech synthesis, in *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (2014) pp. 3844–3848.

[10] V. Nilsson, H. Gruselius, T. Zhang, G. De Kerf, and M. Claessens, Probabilistic dose prediction using mixture density networks for automated radiation therapy treatment planning, Physics in Medicine & Biology **66**, 055003 (2021).

[11] T. Sjöstrand, S. Mrenna, and P. Skands, A Brief Introduction to PYTHIA 8.1, Computer Physics Communications **178**, 852 (2008), arXiv:0710.3820 [hep-ph].

[12] S. Agostinelli *et al.*, Geant4—a simulation toolkit, Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment **506**, 250 (2003).

[13] J. Allison *et al.*, Geant4 developments and applications, IEEE Transactions on Nuclear Science **53**, 270 (2006).

[14] J. Allison and others, Recent developments in Geant4, Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment **835**, 186 (2016).

[15] L. Trailovic and L. Y. Pao, Variance Estimation and Ranking of Gaussian Mixture Distributions in Target Tracking Applications, in *Proc. IEEE Conf. Decision and Control 2002, Las Vegas, NV* (2002).

[16] ATLAS Collaboration, Local Hadronic Calibration, ATLAS PUB Note **ATL-LARG-PUB-2009-001** (2009).

[17] ATLAS Collaboration, Reconstruction of hadronic decay products of tau leptons with the ATLAS experiment, The European Physical Journal C **76**, 295 (2016), arXiv:1512.05955 [hep-ex].

[18] F. Chollet *et al.*, Keras (2015).

[19] M. Abadi *et al.*, TensorFlow: Large-scale machine learning on heterogeneous systems (2015).

[20] ATLAS Collaboration, Reconstruction, Energy Calibration, and Identification of Hadronically Decaying Tau Leptons in the ATLAS Experiment for Run-2 of the LHC, ATLAS PUB Note **ATL-PHYS-PUB-2015-045** (2015).

[21] Miles Cochran-Branson and Quentin Buat, TES determination using neural networks (2022).

[22] Miles Cochran-Branson and Quentin Buat, Tau energy scale calibration at ATLAS using neural networks (22).

[23] Miles Cochran-Branson and Quentin Buat, Tau energy scale calibration at ATLAS using machine learning (2022).