Improvements to AngryTops: a Machine Learning Approach to Top Quark Kinematics Reconstruction

AngryTops Project Summer 2020 Report

Brandon Tan Chin Hian and Haolin Liu

Supervisor: Pekka Sinervo

Institution: University of Toronto

Contents

1 Introduction

1.1 Top Quark Production at the LHC

The top quark is the heaviest fundamental particle at 175 GeV. Its large mass allows it to play a special role in the standard model and top quark physics provides key insight into strong interactions and physics at the electroweak scale. At the LHC, top quarks are produced in pairs through collisions of pp pairs at 13 TeV and observed through detector signals of decay products. The production of top quarks result in 6 partons in the final state, in one of three decay channels involving leptons and jets from quarks and gluons. The challenge is to map detected momenta and reconstruct the kinematics of the top quark pairs in these events. Current reconstruction algorithms include χ^2-fit and KLFitter, which use statistical methods to predict most likely event processes. Improvements in top quark reconstruction are key to understanding top quark processes and properties and precision measurements of the production process.

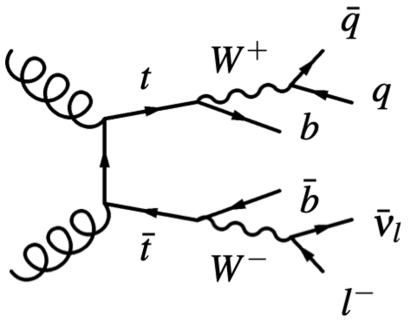


Figure 1: An example of a leading order semileptonic decay process for top quark production resulting from gluon scattering.

1.2 Motivation

Current approaches of reconstruction based on likelihood methods such as χ^2-fit rely on objective functions and combinatorial methods to determine the most likely event history for each top quark event. The complexities of top quark event kinematics and limited detector resolutions make the reconstruction of top quark momenta difficult. Existing methods are often susceptible to resolution and event selection limitations. In particular, cuts to the dataset and detector resolutions limit the accuracy and type of events that algorithmic approaches can predict. Additionally, especially in semileptonic lepton+jets channels, missing momenta from leptons can provide a challenge to reconstruction. We propose improvements to AngryTops, a machine learning approach to reconstructing top quark momenta in pair production events. The network approach provides probabilistic reconstructions independent of objective functions, potentially improving upon the accuracy and overcoming limitations of existing methods. In this project, we develop improvements to AngryTops, a machine learning suite that reconstructs top quark production and decay event kinematics trained on Monte Carlo event samples. We aim to improve the accuracy and optimize performance of the network to a level comparable to those of existing algorithmic approaches as well as present new ways of representing predictions.

1.3 Previous Work

Previous work provided an implementation for AngryTops, a machine learning approach to top quark reconstruction that predicts ttbar event histories using response of the detector. AngryTops predicts top, bottom, and W-boson daughter momenta in the lepton+jets channel, in which one top quark

decays fully hadronically while the other semileptonically. The network is trained and tested on events generated using Monte Carlo event generators for events generated in pp collisions at 13 TeV. 200 million events were generated and leading order events, parton showering, and detector effects are simulated using MadGraph5, PYTHIA8, and Delphes3 respectively. Events are cut on the transverse momentum (p_T) and pseudorapidity η of the particles where we select only the events with p_T greater than 20 GeV and $|\eta|$ greater than 2.5. After these cuts, roughly 5 million events are left for training and testing of the network.
Previous work experimented with a LSTM's BDLSTM's, FFNN's, and CNN's and have found that CNN's perform best in terms of accuracy and training speed. Representations of input and output data various coordinate systems such as Cartesian (in terms of $x-, y-, z-$ momenta, ie. p_x, p_y, p_z) and polar (in terms of transverse momentum, pseudorapidity, and azimuthal angle, ie. p_T, η, ϕ) were explored. Additionally, feature scaling and additional training hyperparameters were tested. The current network runs with a CNN architecture with minmax scaling and performs comparably well with both representations. The current architecture is summarized in the diagram below.

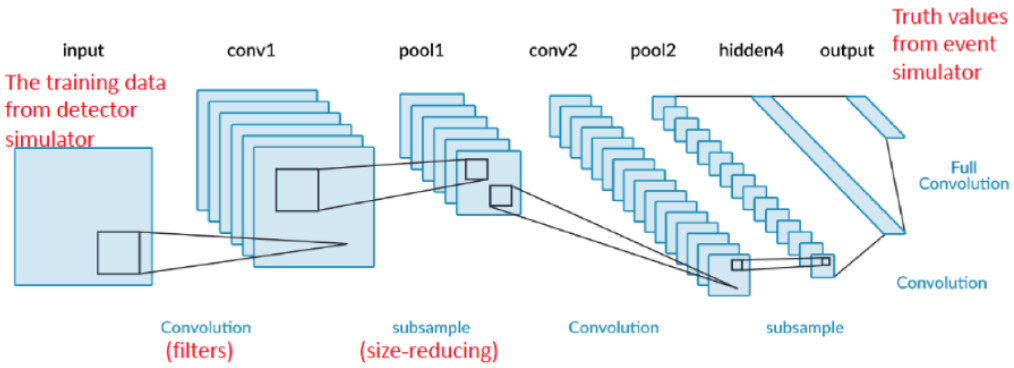


Figure 2: The convolutional neural network

The network consists of 6 convolutional layers, 2 pooling layers, and 5 dense layers and is trained using an Adam Optimizer with a learning rate of 10^{-5} and MSE loss function. The input to the network consists of a 6×6 matrix containing the momenta, mass, b-tagging state of each jet in addition to the momenta, time of flight, missing transverse energy, missing azimuthal energy of the muon. The output of the network is a 6×3 matrix containing the momenta of the bottom quark, W-boson, and top quark from the hadronic and semileptnoic quark decays. The accuracy of the output of the network is comparable to those of benchmark reconstruction methods based on χ^2-fit. The network and χ^2-fit perform similarly on reconstructing top quark kinematics. AngryTops improves upon χ^2-fit for some p_T and η variables, while χ^2-fit significantly outperforms AngryTops on ϕ variables.

2 Goals

Our main goal in this project was to optimize the network performance in making predictions of the various kinematic variables for each parton. In particular, while the existing network performs reasonably well on η and p_T variables, ϕ variables are comparatively less accurately predicted in some partons. The predictions of the network demonstrates a structure in some ϕ variables, in contrast to the flat distributions in the truth data. Previous works has suggested a dependence on data representation for prediction structure and quality. Additionally, various training parameters such as number of epochs, data scaling, and loss functions can be further optimized and explored. To this

end, we aim to explore the effect of representations of input and output data and experiment with a a multitude of training parameters such as data scaling, the number of epochs for the training, and custom loss functions. Modifications on the dataset that is fed into the network were also made in an attempt to help the network produce better predictions. Such modifications comprise augmenting the dataset, flipping the sign of the pseudorapidity and making cuts on the transverse momenta of the jets (as well as the truth particles).
Additionally, while previous works experimented with network types and structures, current outputs suggest that results may still be improved with the implementation of a more suitable network structure. As such, we also aim to experiment with types of neural network as well as examine the structure of the current architecture. As a way of measuring the network performance, we have also introduced histograms which indicate the distribution of the closest jets to the bottom quark and the W boson, or both.

3 Improvements Investigated

Previous investigations suggest that both the network architecture and configurations and training parameters can be further optimized to improve the quality of predictions. We explore an RNN architecture for the network and compare its performance to that of the CNN. Additionally, we experiment with various training parameters and examine their impact on predictions for the CNN. For training, the input and output data is rescaled to a consistent range. We compare two data rescaling methods: minmax and standard. The first rescales data taking into account the entire range of the kinematic variable, and the second normalizes each variable to a mean of 0 and a standard deviation 1. Another training parameter we experimented with is the number of epochs, which indicates the number of times the dataset is fed through the network. While we train the network at a standard length of 15 epochs for usual sessions, we also test epoch numbers between 50 to 100 to estimate the number of epochs at which loss values stabilize an thus the optimum of epochs for training for the CNN.
Another parameter we explore is the representation of the input and output momenta of the network. While previous work performed training in both Cartesian (p_x, p_y, p_z) and polar coordinates (p_T, η, ϕ), we provide a rigorous comparison of the representations trained on the CNN. One potential cause of the poor prediction of ϕ variables is the failure to account for the continuity of the ϕ variable at $-\pi$ and π during training when trained in *pxpypz* coordinates. We attempt to implement a custom loss function that into takes account the wraparound for ϕ when computing the loss component in ϕ. In addition to these, we also examine various other training parameters such as batch size and metrics.
Although the χ^2 values give an approximate indication of the goodness of fit between the predicted and truth data, χ^2 comparisons become unreliable at large values, and in particular when the sizes of the two fits being compared differ. We develop a closest jet matching algorithm identifying the closest jets to each hadronic b and W and leptonic b as an additional measure of network performance, and use this to make cuts on the dataset as well to improve the quality of training and predictions.
Lastly, we investigate ways to increase the size of the dataset through data augmentation, where we develop ways to augment data by rotating and flipping each event along the beam axis. We describe the results and discuss implications in the following section.

4 Results and Discussion

4.1 RNN Architecture

We construct an RNN architecture as a preliminary experimentation with other network architectures. The network consists of 6 RNN layers and a dense output layer with 1 input time step and 3 output time steps. The number of time steps is reflective of the interpretation of the input jet momenta as a single event and the chronology of the decay, where in reconstruction of the b, W, and t kinematic variables,

the predictions of the W and t depend on the output from prior steps. The architecture is trained and tested on the same dataset using the same scaling, loss function, and training hyperparameters.

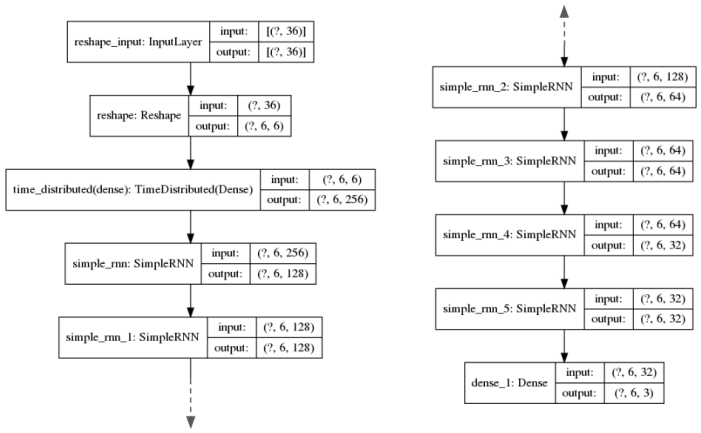


Figure 3: The RNN model

Preliminary results show that the RNN performs notably worse than the existing CNN architecture except in some t kinematic variables. This is likely due to the structure of the RNN network, where results from the predictions in the b and W timesteps can be used to improve predictions of t variables. Adjustments to the network to improve predictions in the b and W variables should be investigated. Additionally, the RNN trains at roughly 7000s per epoch, significantly slower than the 2000s per epoch rate of the CNN. The RNN needs to be further optimized, both in terms of the structure and the hyperparameters, in order to perform at a comparative accuracy to the CNN and existing χ^2 reconstruction methods.

4.2 Training parameters

4.2.1 Data Scaling

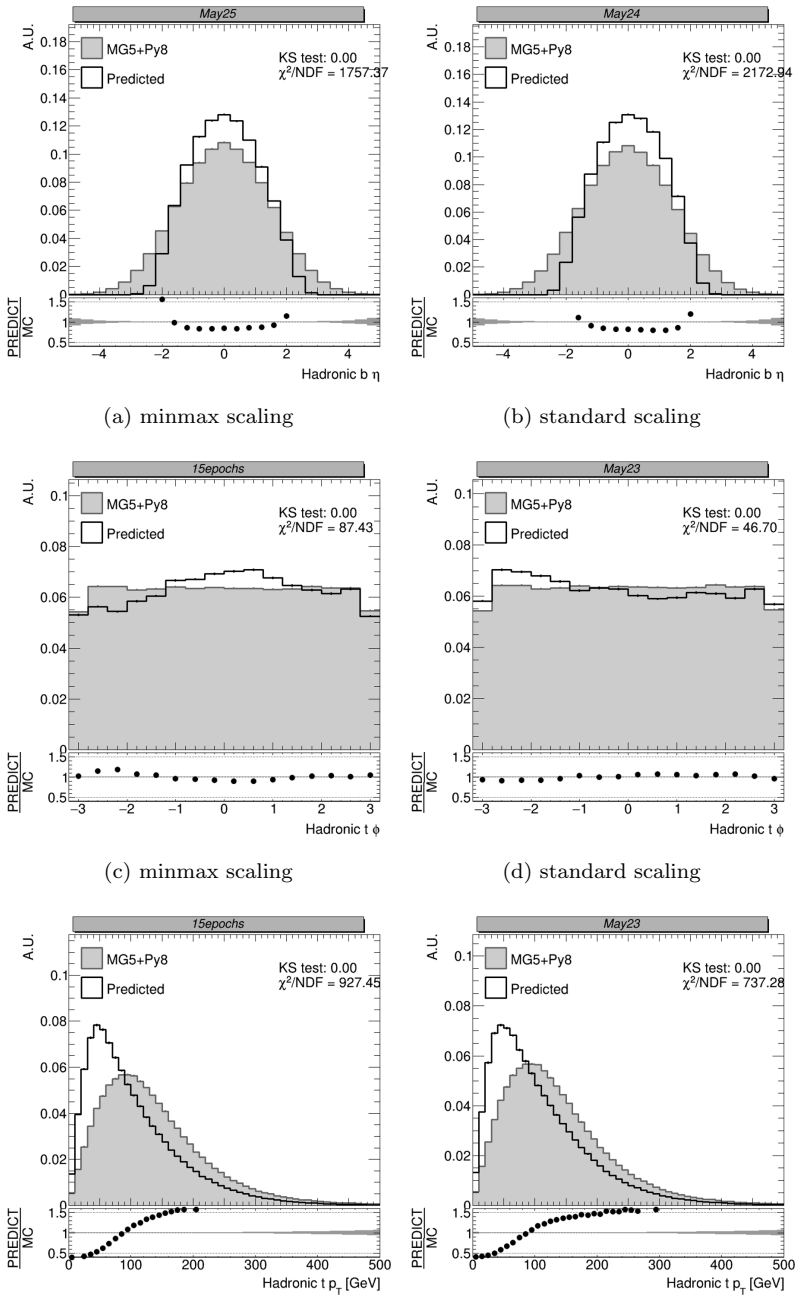
For neural networks such as the CNN or the RNN, the weights are usually small random values that are updated by the optimizer as a response to error estimates on the dataset, so it is necessary to scale the inputs and outputs of the network. We have experimented with two data scaling methods. The first is min-max scaling, which normalizes data between -1 and 1 and is given by the equation below. Here x_{scaled} is the scaled value, x is an arbitrary entry in the original data, and x_{min} and x_{max} are the minimum and maximum of the dataset respectively:

$$x_{scaled} = \frac{x - x_{min}}{x_{max} - x_{min}}$$

The other method is standard normalization, given by the equation below, where in this case, $\overline{x}$ is the mean value of the dataset and σ is the standard deviation of the dataset.

$$x_{scaled} = \frac{x - \overline{x}}{\sigma}$$

We have found that the network performs very similarly with both data-scaling methods. Below are some distributions of the pseudorapidity of the bottom quark and the ϕ and p_T of the hadronic top quark. Visually-speaking, there is no striking difference in the predictions. Though the $\frac{\chi^2}{NDF}$ values are different, they do not present sufficient evidence to pursue one scaling method other the other. Ultimately, we ran most of our trainings using the minmax scaling.



(a) minmax scaling

(b) standard scaling

(c) minmax scaling

(d) standard scaling

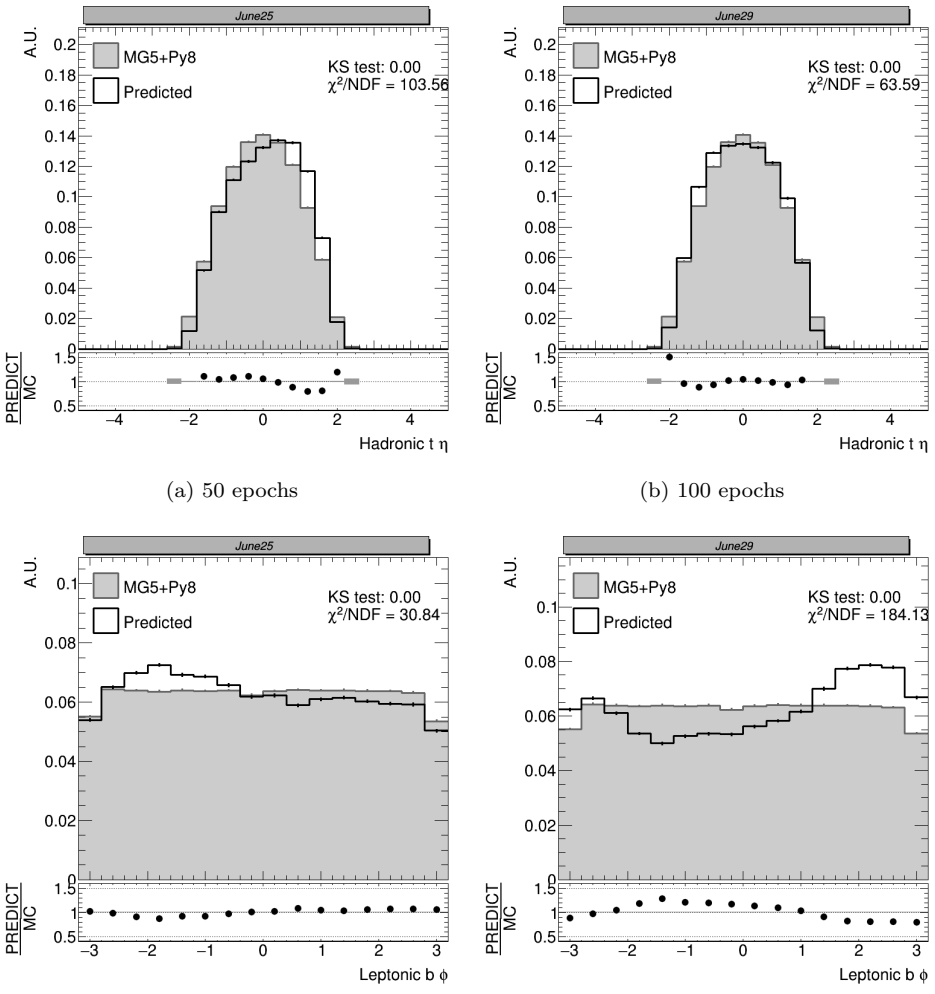
(e) minmax scaling

(f) standard scaling

Figure 4: Comparison of minmax scaling and standard scaling

4.2.2 Number of Epochs

In machine learning terms, an epoch is when the whole dataset had the opportunity to pass through the neural network and update the network's internal parameters. For AngryTops, 90% of the dataset is used for training and 10% is for validation. During the latter, the loss function values are computed. Our aim here was to determine the number of epochs that would optimize the loss function. In previous works, training were run on a maximum of 50 epochs, whereby the loss function values still seemed to decrease. We therefore experimented on 75 to 100 epochs, checking whether the loss and validation values still were going down. Our models are built in such a way that training would be stopped whenever the validation values started to increase. However, we also tried to let the training run without stopping to judge whether the training should indeed be stopped when the validation goes up or if the network would still be able to make better predictions irrespective of whether validation increases.



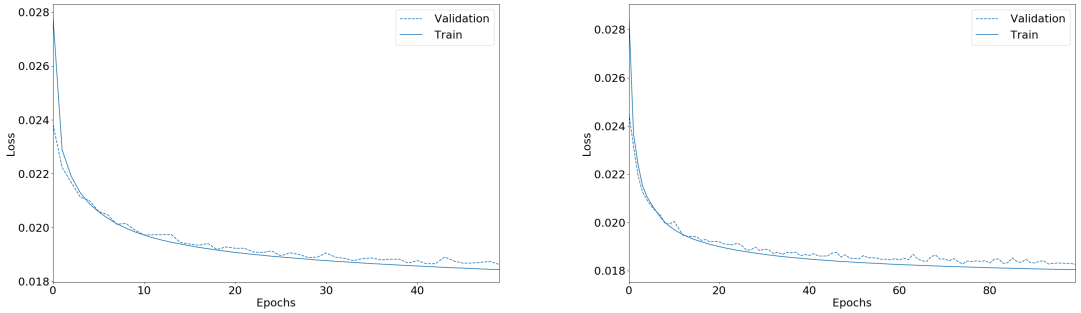
(a) 50 epochs

(b) 100 epochs

(c) 50 epcohs

(d) 100 epochs

Figure 5: 50 epochs vs. 100 epochs prediction plots



(a) 50 epochs

(b) 100 epochs

Figure 6: Loss vs. Number of Epochs plots

As it can be seen in Figure 4, the predictions seem to improve for some kinematic variable(Hadronic t η) and worsen for others(Leptonic b ϕ) when the number of epochs for the training is increased. In Figure 5, the training losses seem to be monotonically decreasing while the validation losses experience some fluctuations. They do not present concrete evidence of 'overtraining'. This is certainly an interesting outcome and perhaps necessitates further investigation in the future, especially in figuring out what is the ideal number of epochs for training in such machine learning problems.

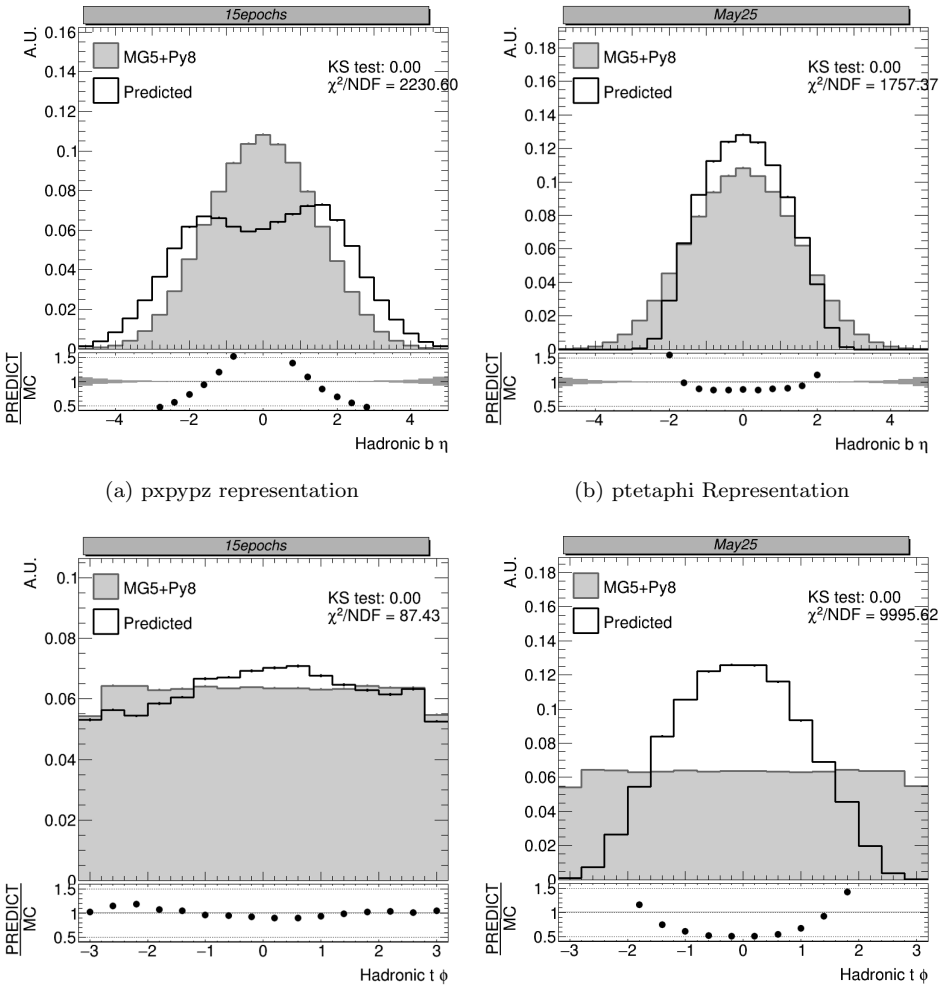
4.2.3 Representations

The dataset that we use for training comprises data in both p_x, p_y, p_z representation and the p_T, η, ϕ representation. Both representations contain the same information about the muon, jets and truth particles, since they are related by a simple conversion outlined below. Our investigation here consisted of working out which representation helps the network produce better predictions for the kinematic variables of the partons.

p_x, p_y, and p_z are the Cartesian representation of the particles' momenta. We assume that the z-direction is along the axis of the proton beam. p_T is thus the transverse momentum, ϕ is the azimuthal angle and θ is the polar angle. Notably, the spatial coordinate η is far more convenient than θ for particle physics. Somewhat analogous to spherical coordinates, θ ranges from 0 to π and is also called the scattering angle, i.e. the angle with which the particle's momentum deviates from the proton beam axis. As such, when θ is $\frac{\pi}{2}$, η is zero and effectively, when θ approaches zero, η approaches positive/negative infinities. For proton-proton collisions where the momenta are expected to have similar values, η tends to be symmetric about zero.

$$p_T = \sqrt{p_x^2 + p_y^2}$$
$$\eta = -ln(tan(\frac{\theta}{2}))$$
$$\phi = \arctan(\frac{p_y}{p_x})$$
$$p_x = p_T cos(\phi)$$
$$p_y = p_T sin(\phi)$$
$$p_z = p_T sinh(\eta)$$

For the CNN model, the two different representations produce stark differences for some kinematic variables.



(a) pxpypz representation

(b) ptetaphi Representation

(c) pxpypz Representation

(d) ptetaphi Repressentation

Figure 7: Prediction plots demonstrating the major differences between the representations

Figure 6(a) and 6(b) show that the ptetaphi representation does a much better job at predicting the η distribution of the bottom quark than its counterpart, while Figure 6(c) and 6(d) yield the opposite diagnosis. This time, the ptetaphi representation gives a far worse prediction for the hadronic top ϕ, and these distributions are recurrent for all the other partons. This bell-shaped ϕ distribution has been an intriguing issue that we have attempted to solve, mainly with the use of a custom loss function.

4.2.4 Custom Loss Function

Previously, all trainings were done with a Mean Squared Error (M.S.E) loss function. For reasons such as obtaining valuable insight on whether there are other available loss functions which would help the network perform better and solving predictions issues that comes with using an M.S.E loss,

we decided to investigate the use of the custom loss function that is in the AngryTops package and attempt to improve it.

$$M.S.E = \frac{1}{N}\sum_{n=1}^{N}(y_n - y_n^p)^2$$

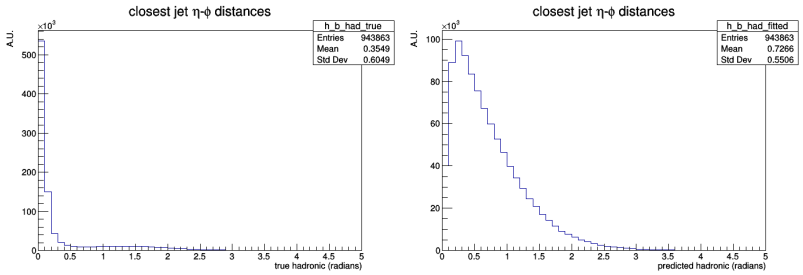
As the name suggests, the M.S.E computes the loss by finding the squared difference for each y_n, the n^{th} truth value and y_n^p, the n_{th} predicted value and summing these values before finally dividing by N, the number of predictions. Challenges we have encountered included understanding the nature of the input to the loss. As a means to prevent a suspected ϕ wrap-around issue when using the CNN model and the ptetaphi representation, we also attempted to implement some code that would solve this. Since we were not sure what the input to the loss was, we wrote separate custom loss scripts: one which assumes that the input is in the Cartesian representation of the momentum (where we would have to convert the truth and predicted values into the ptetaphi representation and calculate the loss values), and one which assumes the input is in the ptetaphi representation already, where only the loss values calculations and the wrap-around fix would be present. Unfortunately, though the second option gave 'better' predictions, they were too faulty to assume we found a proper fix. Future investigation needs to be made on that.

4.3 Closest Jet Matching

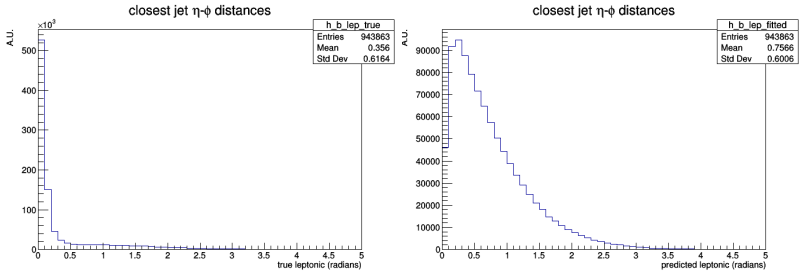
In each $t\bar{t}$ event, each b quark results in a single jet and the hadronic W boson decays to give two daughter jets observed in the detector response. The kinematic variables predicted by the network should match closely to the corresponding jet momenta for each parton. As an additional measure of network performance, we introduce a jet matching algorithm, where for each b and W, the closest daughter jets in $\eta - \phi$ space are identified. This is done for a single jet for each b and for the sum of two jets for each W. For each parton-jet pair, the distance in $\eta - \phi$ space is taken as

$$R = \sqrt{(\Delta\eta)^2 + (\Delta\phi)^2}.$$

Here $\Delta\eta = |\eta_p - \eta_j|$ and $\Delta\phi = \min\{|\Delta\phi_p - \Delta\phi_j|, 2\pi - |\Delta\phi_p - \Delta\phi_j|\}$ where η_p, ϕ_p are the η and ϕ momentum variables of the parton and η_j, ϕ_j those of the jet and a minimum function is taken to account for the wraparound nature of the ϕ variable. This distance is calculated between a b and each jet and between the hadronic W and the 4-vector sum jet for each possible jet pair combination for every event. The distributions of the smallest-distance for the hadronic b and W and leptonic b are produced below.

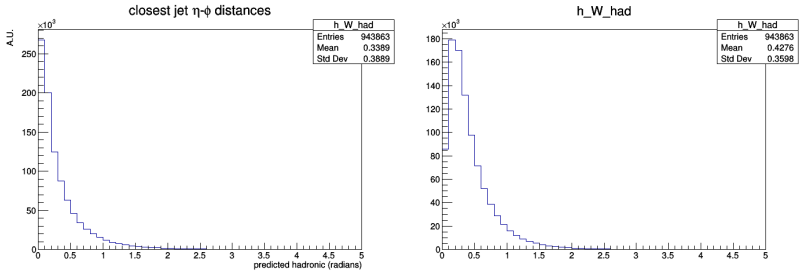


(a) hadronic true (b) hadronic predicted

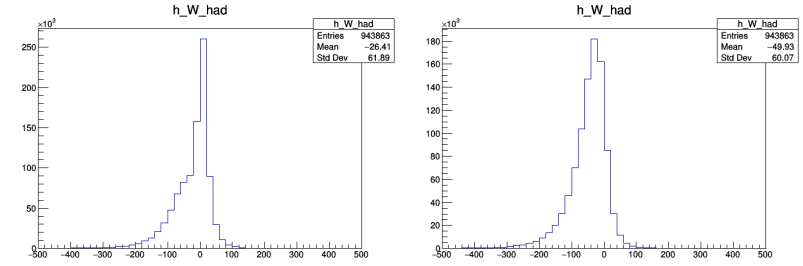


(c) leptonic true (d) leptonic predicted

Figure 8: Distributions of distances to closest jets for leptonic and hadronic b's



(a) hadronic true $\eta - \phi$ (b) hadronic predicted $\eta - \phi$



(c) hadronic true p_T (d) hadronic predicted p_T

Figure 9: Distributions of distances to closest jets for hadronic W's

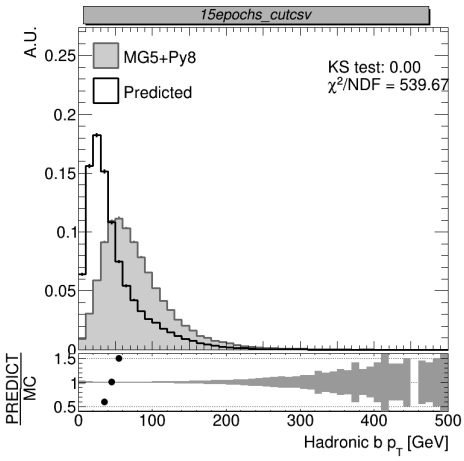
From the distributions of the truth data, it is evident that the the identified closest jets lie largely within 0.4 of the parton for b's and 1.0 for W's. For the b's in particular, the truth distribution has a large peak at 0 with a width of around 0.4 and a long tail extending beyond 2.5. This suggests that

only a portion of the events have a jet matched to the b, with the rest missing information about the jets or suffering inaccuracies due to detector flaws. In comparison, the distributions for the matching done on predicted data have a much wider peak slightly offset from 0. The proximity of the peak close to 0 suggests that most predicted variables can in large be identified with the relevant jets and that most predictions are consistent with input data, although the larger widths suggest that the quality of the predictions can still be improved. For the hadronic W bosons, the difference in p_T between each W and its closest daughter jets is also determined according to the equation

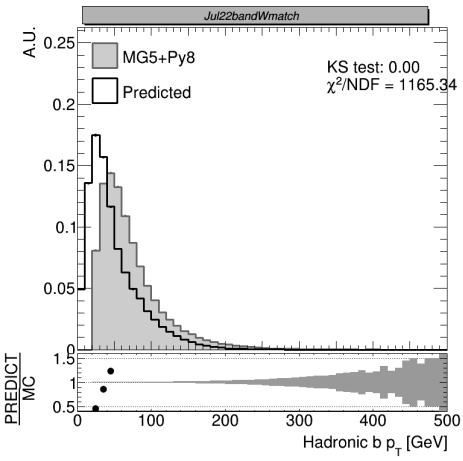
$$\Delta p_T = p_{T_W} - p_{T_j}.$$

The asymmetry in the p_T distributions suggests quark boosting in the production production process, which may influence predictions of the network.

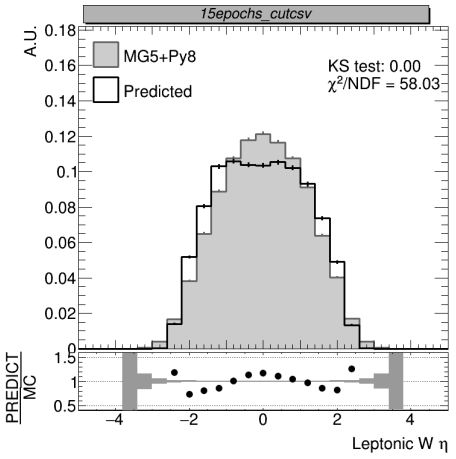
The lack of matched jets in some events suggest that some events are missing information and cannot be accurately predicted by the network. We have thus used jet matching as another criterion to make cuts on the dataset, where we require the minimum $\eta - \phi$ distance to be less than 0.4 for b's and 1.0 for W's for accepted events. We term such events as having matched jets for the b's and W. We trained the network on datasets where only b-matching is done, only W-matching is done, and where both b-matching and W-matching are done. After cuts were performed, roughly 4 million, 2 million, and 1.5 million events are left for the b-, W-, and b and W-matched events respectively. We found that training on the b and W matched dataset provided a notable improvement. In particular, the p_T, η, ϕ variables all exhibited improvements in accuracy and training on the cut dataset appeared to have fixed the asymmetry in top and W η. One consideration for network performance is the significant reduction in the number of events for training and testing due to the cuts; direct comparisons of prediction quality cannot be made to uncut datasets and predictions have potential for further improvement with a larger dataset comparable to the original. Addionally, the performance of the network trained on the cut dataset on uncut data is yet to be assessed. Both of these points are areas of exploration for future investigations.



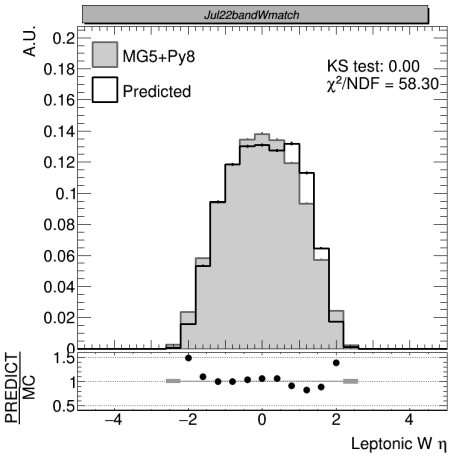
(a) hadronic b p_T uncut



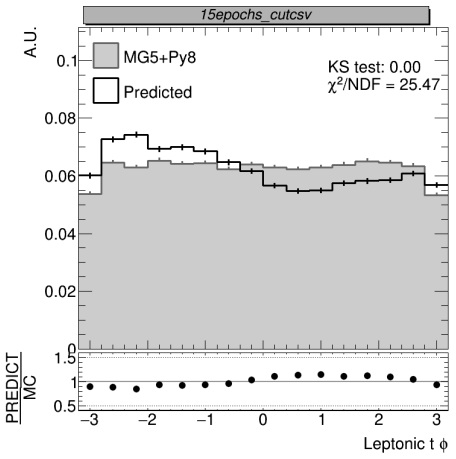
(b) hadronic b p_T cut



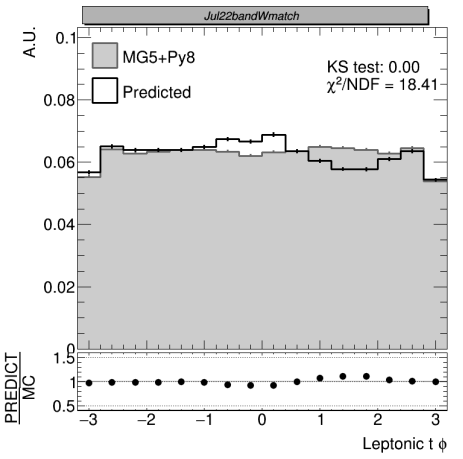
(c) leptonic W η uncut



(d) leptonic W η cut



(e) leptonic t ϕ uncut



(f) leptonic t ϕ cut

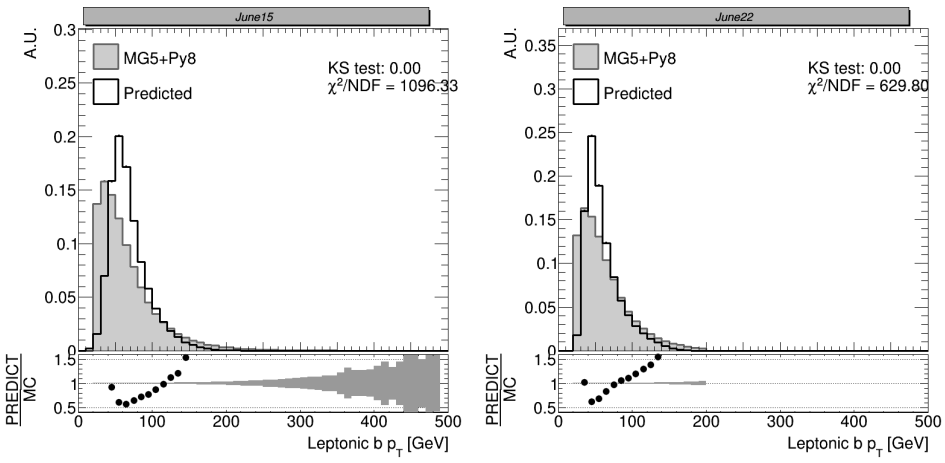
Figure 10: Comparison of prediction plots for some kinematic variables for uncut and b and W matched datasets trained on 1509381 events

4.4 Cuts On The Dataset

As outlined in the Previous Work section, the initial cuts on the dataset comprised rejecting events with jets whose p_T are less than 20 GeV or events with $|\eta| > 2.5$. For our investigation, we also made the same cuts on the truth particles. Since we hypothesized that the scaling and range of p_T possibly affects the quality of predictions, we made additional cuts, rejecting events whose jets and truth particles' p_T are greater than 200 GeV. Since few events lie in this region, the cuts made were logical and allowed us to see the effect of the scaling on a much smaller range of p_T in the dataset.

$$\Delta = \sqrt{(\eta_{jet} - \eta_{truth})^2 + (\phi_{jet} - \phi_{truth})^2}$$

Ultimately, this proved unsuccessful. We've only seem modest improvement as seen in Figure 7 below. Though the χ^2 values have a big difference, this is attributed to the differing sizes of the validation dataset and they are therefore not comparable.



(a) Usual Dataset

(b) Tuncated Dataset

Figure 11: Modest improvements can been seen in the leptonic b p_T and this is recurrent for all other partons

4.5 Data Augmentation

We expect that with a greater sample size for the dataset, the better the network could perform in predicting the kinematic variables. Formerly and for much of our own training runs, our dataset comprised 4 to 5.5 million events depending on the cuts we made.

There are two ways in which we can artificially increase the size of the dataset. First, for each event, we can rotate the muon, jets and truth particles' Lorentz vectors about the z-axis by an arbitrary angle ϕ. The second data augmentation method consists of flipping the sign of the pseudorapidity η. Importantly, these new events are all possible at the LHC and are thus appropriate data to feed into the network.

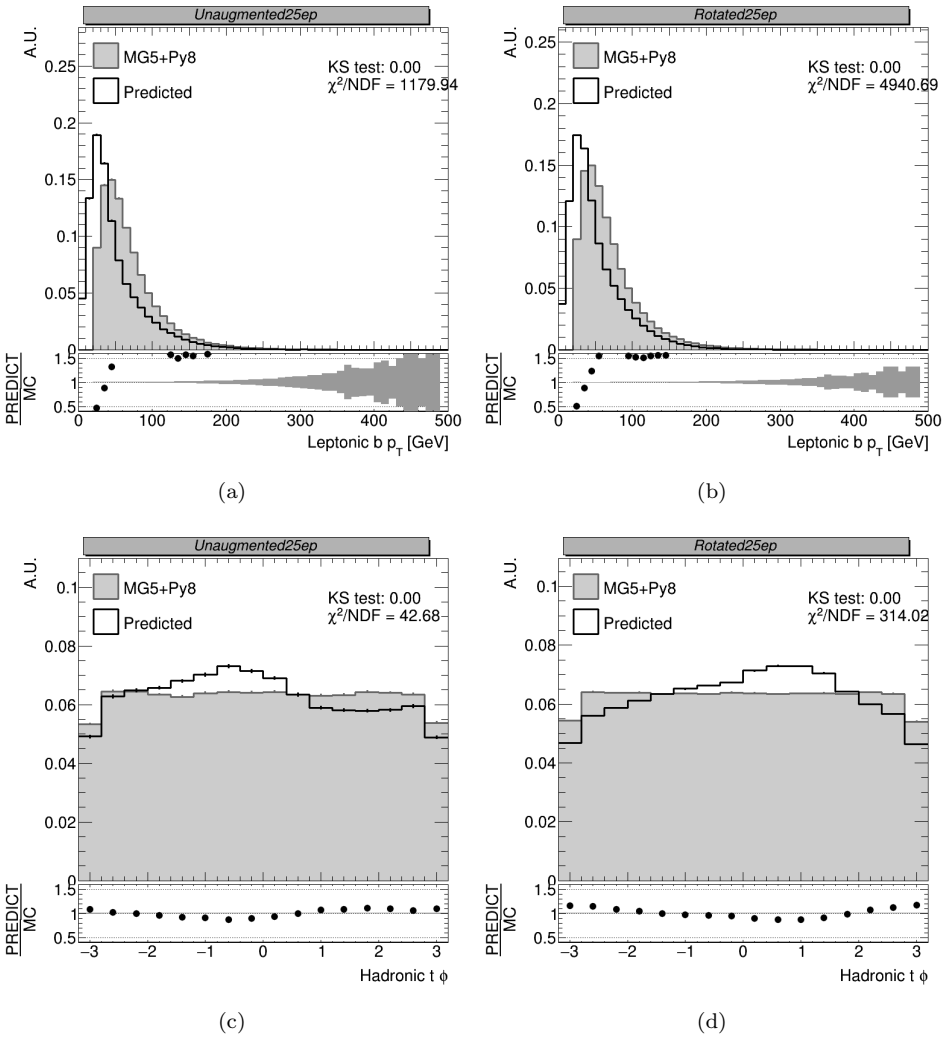
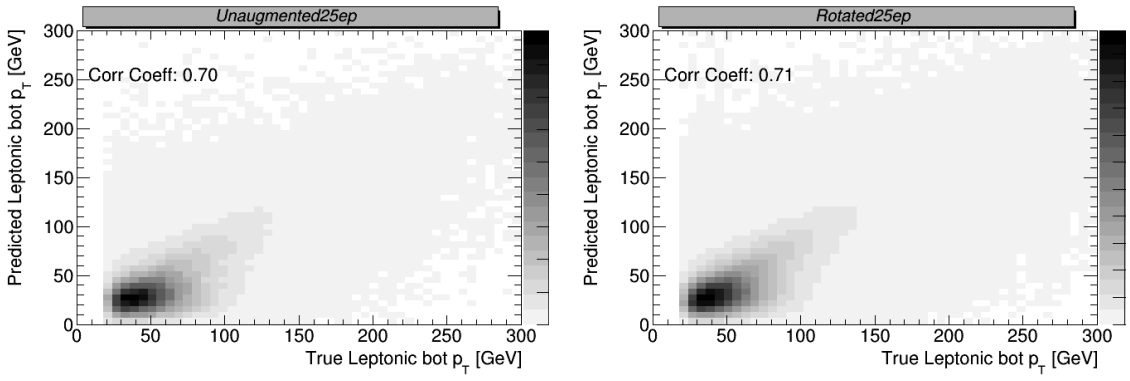


Figure 12: Prediction plots with (right) and without (left) data augmentation

The data augmentation performed in the above plots was a 5-times ϕ-rotation about the z-axis, which effectively increase the number of events to somewhere around 19 million events. Since the size of the validation dataset is again largely different, the χ^2 are not comparable. There seems to be a marginal improvement, if any, in the predictions. The correlation plots confirms the marginal improvement of the results. Figure 9 shows that the correlation coefficient increase by a a hundredth with data augmentation.



(a) (b)

Figure 13: Correlation plots of the leptonic b p_T with(right) and without(left) data augmentation

5 Conclusion

5.1 Takeaways

We investigated a new RNN architecture and several training parameters and configurations. The CNN performed significantly better than the RNN network in almost all kinematic variables except for those of the t. We found that data representation and number of epochs have a significant effect on the quality of predictions, while data scaling and batch size has minimal effect on network performance. We conclude that the network performs best when trained with minmax scaling, Cartesian representaion, and at around 15 epochs. The custom loss function is still being developed but shows promise for improving upon the currently implemented M.S.E. and resolving the ϕ wraparound consideration.

The newly implemented closest jets matching algorithm provided an additional metric for assessing network prediction accuracy that improves upon the χ^2 test as it is independent of the size of the dataset and can be used even when predicted and truth data differ significantly. Using this metric, we see that the current network performs fairly well with the greater proportion of predicted events kinematics consistent with input jets. The closest jets matching algorithm was also used as another metric to select events for the dataset and the network trained on the new dataset performs significant better in almost all variables than when trained on the uncut.

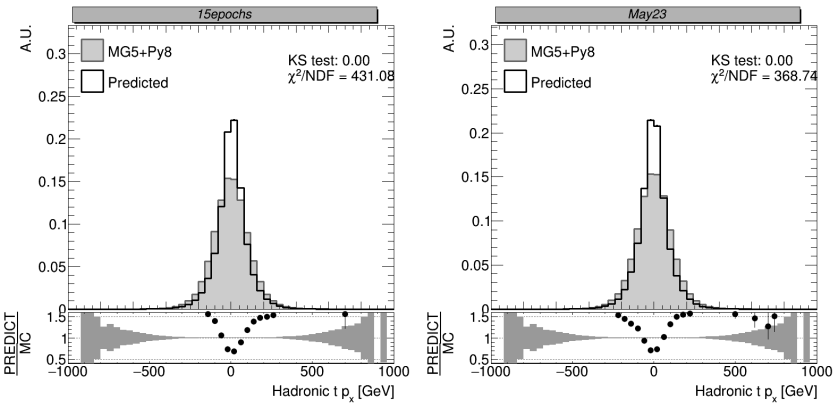
Current data augmentation on ϕ rotation with a factor of 5 did not seem to produce any significant improvements in the quality of the predictions when performed on the cut dataset. Additional investigations into larger augmentation factors or other data generation methods may provide further insight.

5.2 Future Works

Possible future points of investigation are exploring additional network architectures and structure for the current network, testing a larger augmentation factors or generating additional events, implementing a proper custom loss function to resolve the ϕ wraparound issue, exploring additional network parameters such as learning rate and batch size, and developing more ways of measuring the effectiveness of predictions.

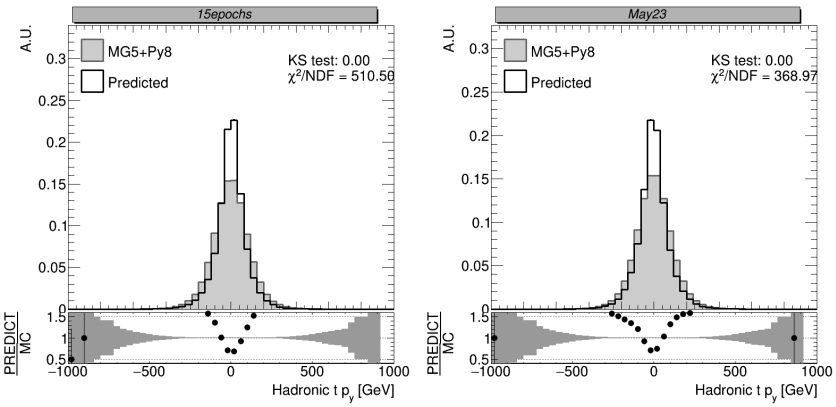
6 Appendix

In this section, we include other relevant figures from the different sections referred in the report.



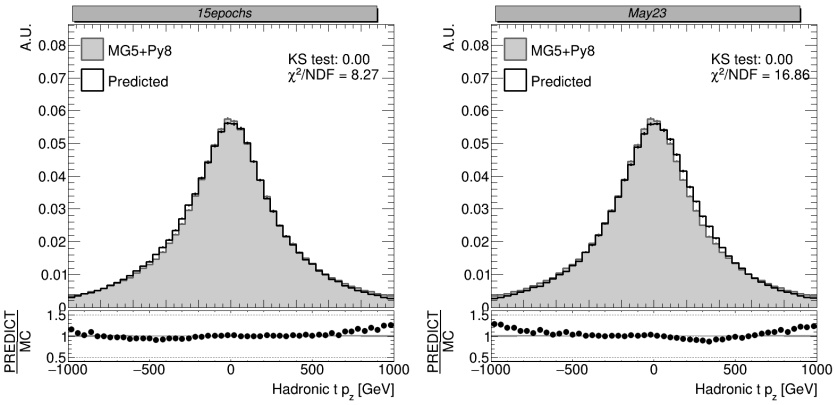
(a) minmax scaling

(b) standard scaling



(c) minmax scaling

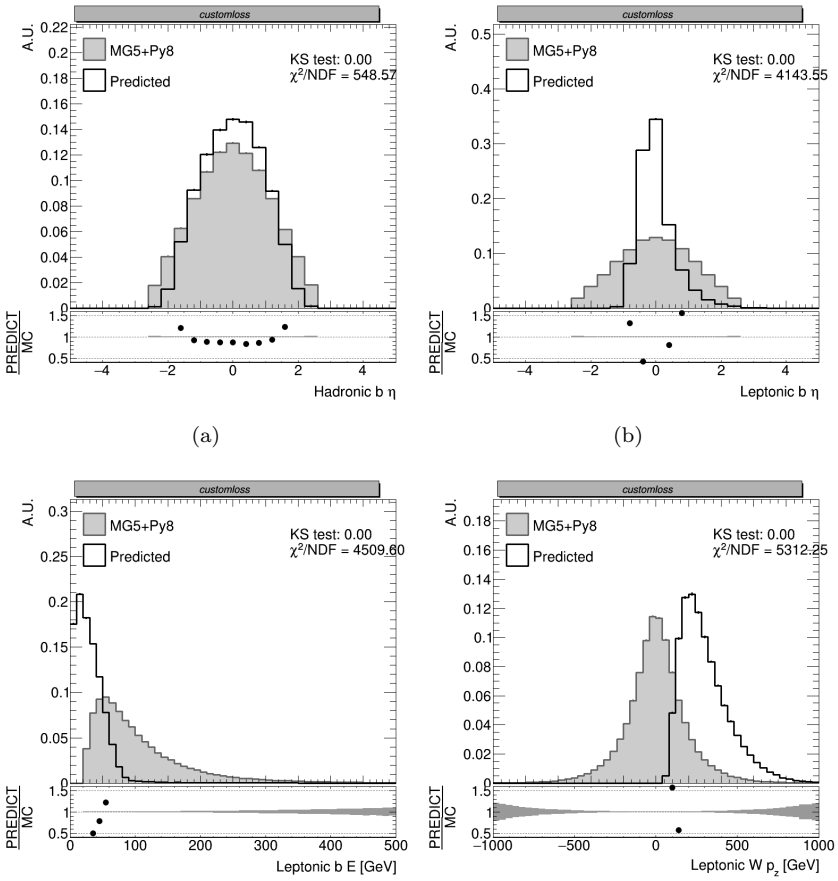
(d) standard scaling



(e) minmax scaling

(f) standard scaling

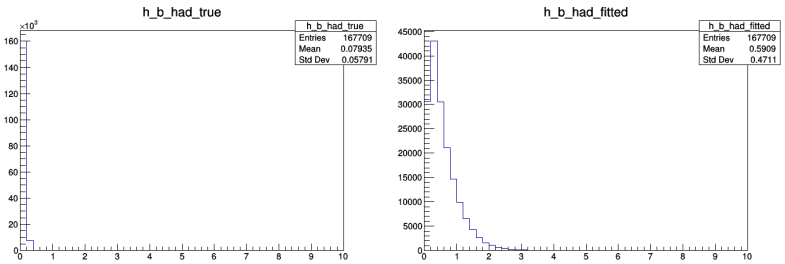
Figure 14: Comparison of minmax scaling and standard scaling for other kinematic variables of the top quark



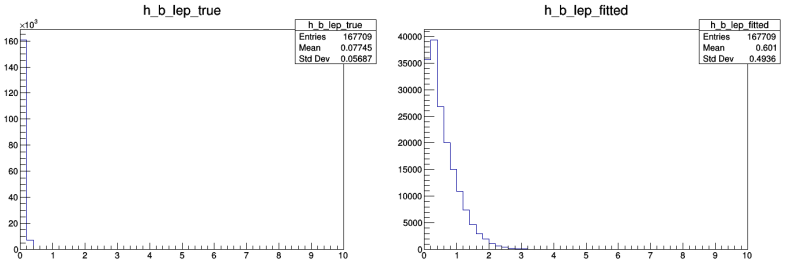
(a) (b)

(c) (d)

Figure 15: Sample plots for the custom loss function. We see that in (a), we obtain a fairly good distribution for the hadronic b η but (b), (c) and (d) show that the code is still buggy/not fully understood properly. If this is fixed in the future, we could get further insights into the performance of the neural network.

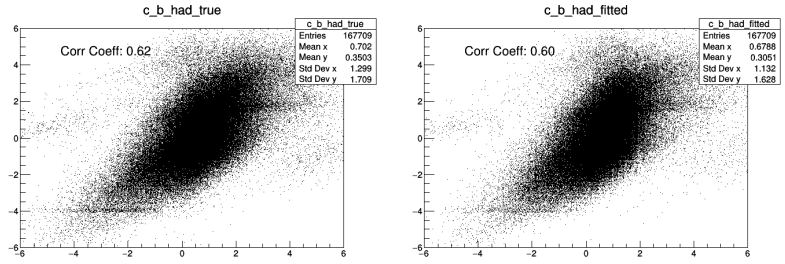


(a) hadronic true (b) hadronic predicted

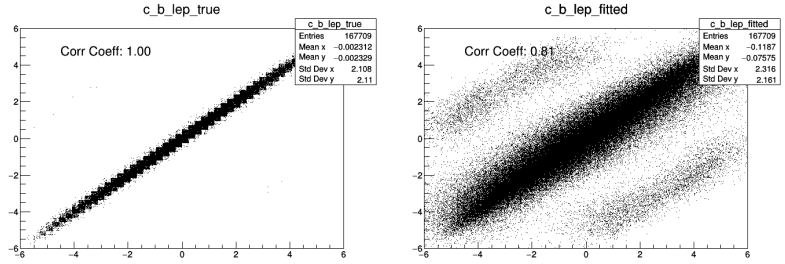


(c) leptonic true (d) leptonic predicted

Figure 16: Distributions of distances to closest jets for leptonic and hadronic b's for b and W matched data

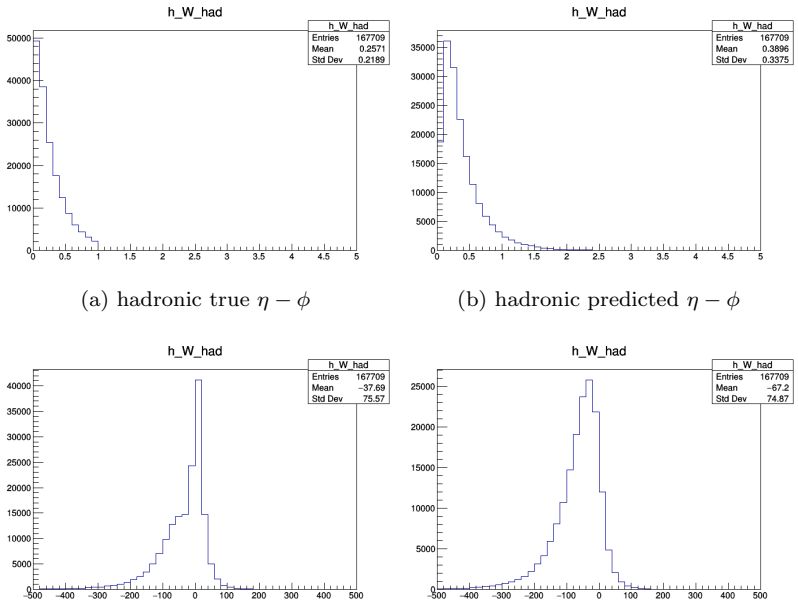


(a) hadronic true (b) hadronic predicted



(c) leptonic true (d) leptonic predicted

Figure 17: Correlation plots of distances to closest jets for leptonic and hadronic b's for b and W matched data



(a) hadronic true $\eta - \phi$

(b) hadronic predicted $\eta - \phi$

(c) hadronic true p_T

(d) hadronic predicted p_T

Figure 18: Distributions of distances to closest jets for leptonic and hadronic W's for b and W matched data

7 References

[1] Claudio Campagnari and Melissa Franklin. The discovery of the top quark. *Reviews of Modern Physics*, 69(1):137–211, 1997. https://cdn.journals.aps.org/files/RevModPhys.69.137.pdf.

[2] Parton shower monte carlo event generators. `http://www.scholarpedia.org/article/Parton_shower_Monte_Carlo_event_generators`

[3] Pseudo-rapidity, azimuthal angle, and transverse momentum. `https://www-cdf.fnal.gov/physics/new/qcd/ue_escan/etaphi.html`