Machine Learning: Science and Technology

A detailed study of interpretability of deep neural network based top taggers

MACHINE
LEARNING
Science and Technology

PAPER

OPEN ACCESS

RECEIVED
21 February 2023

REVISED
14 June 2023

ACCEPTED FOR PUBLICATION
21 June 2023

PUBLISHED
11 July 2023

A detailed study of interpretability of deep neural network based top taggers

Ayush Khot, Mark S Neubauer and Avik Roy*

Department of Physics & National Center for Supercomputing Applications (NCSA), University of Illinois at Urbana-Champaign, Urbana, IL, 61801, United States of America

* Author to whom any correspondence should be addressed.

E-mail: avroy@illinois.edu

Keywords: explainable AI, interpretable machine learning, jet classification with deep learning

Abstract

Recent developments in the methods of explainable artificial intelligence (XAI) allow researchers to explore the inner workings of deep neural networks (DNNs), revealing crucial information about input–output relationships and realizing how data connects with machine learning models. In this paper we explore interpretability of DNN models designed to identify jets coming from top quark decay in high energy proton–proton collisions at the Large Hadron Collider. We review a subset of existing top tagger models and explore different quantitative methods to identify which features play the most important roles in identifying the top jets. We also investigate how and why feature importance varies across different XAI metrics, how correlations among features impact their explainability, and how latent space representations encode information as well as correlate with physically meaningful quantities. Our studies uncover some major pitfalls of existing XAI methods and illustrate how they can be overcome to obtain consistent and meaningful interpretation of these models. We additionally illustrate the activity of hidden layers as neural activation pattern diagrams and demonstrate how they can be used to understand how DNNs relay information across the layers and how this understanding can help to make such models significantly simpler by allowing effective model reoptimization and hyperparameter tuning. These studies not only facilitate a methodological approach to interpreting models but also unveil new insights about what these models learn. Incorporating these observations into augmented model design, we propose the particle flow interaction network model and demonstrate how interpretability-inspired model augmentation can improve top tagging performance.

1. Introduction

Machine learning (ML) models are ubiquitous in experimental high energy physics (HEP). With an ever increasing volume of data coupled with complex detector phenomenology, these models are useful to find meaningful information from these large datasets. Over time, ML models have grown in complexity and simpler regression and classification models have been replaced by intricate and deep neural networks (DNNs). Owing to their intractably large number of trainable parameters and arbitrarily complex non-linear nature, DNNs have often been treated as *black boxes*. It has always been challenging to understand how different input features contribute to the network's computational process and how the inter-connected neural pathways convey information. In recent years, advances in *explainable* artificial intelligence (XAI) [1] have made it possible to build intelligible relationship between an AI model's inputs, architecture, and predictions [2–4]. While some methods remain model agnostic, a substantial subset of these methods have been developed to infer interpretability of computer vision models where an intuitive reasoning can be extracted from human-annotated datasets to validate XAI techniques. However, in other data structures such as large tabular data or relational data constructs like graphs, use of XAI methods are still quite novel [5, 6].

In recent times, XAI has been successful in learning the underlying physics of a number of problems in high energy detectors [7], including parton showers at the Large Hadron Collider (LHC) [8] and jet reconstruction using particle flow algorithms [9].

One of the major applications of ML in the field of HEP is classification of jets, which is referred to as *jet tagging*. Jets represent hadronic showers observed as conical spray of particles originating from quarks and gluons produced in the high energy collisions at a collider experiment like the LHC. Identifying jets that originate from decay products of a particle such as the top quark (t) and being able to separate them from other jet categories, such as jets originating from the quantum chromodynamics (QCD) background, is an important challenge in many physics analyses. Traditional top tagging algorithms based on kinematic features of jets and clustering of jet constituents (see [10–13] for example) have been used in particle phenomenology research as well as by the ATLAS and CMS experiments and their predecessors. In Run 1 physics analyses, using data collected at center-of-mass energies of 7 and 8 TeV, these top tagging algorithms along with low-complexity statistical models like decision trees took the center stage in dealing with top tagging [14–16]. However, owing to their superior performance, models based on DNNs started becoming popular in Run 2 at a higher center-of-mass energy of 13 TeV [17, 18].

For top quarks produced with large momenta, the decay products can be packed close to one another and be reconstructed as a single jet. For such *boosted* jets, top tagging can be particularly challenging and require a better analysis of *jet substructures*, a collection of constituents and their derivative properties that can offer better discrimination between jet classes. DNNs have proven to be useful to exploit the jet substructure properties in performing jet classification. A wide variety of deep learning models have been developed to optimize top tagging [19–33]. A comprehensive review and comparison of many of these models is given in [34]. Some of these models have exploited DNN's capacity to approximate arbitrary non-linear functions [35] and their huge success with problems in the field of computer vision, other models have been inspired by the underlying physics information like jet clustering history [22], physical symmetries [23] and physics-inspired feature engineering [27]. These efforts have inspired novel model architectures and feature engineering by creating or augmenting input feature spaces with physically meaningful quantities [27, 36, 37].

The rich history of physics-inspired model development makes the problem of top tagging an excellent playground to better understand the modern XAI tools themselves. This allows us to traverse a rare two-way bridge in exploring the relationship between data and models—our physics knowledge will allow us to better understand the inner workings of modern XAI tools and perfect them while those improved tools would allow us to take a deeper look at the models— paving ways for analyzing and reoptimizing them. As it has been pointed out in [38], such insights into explainability of DNN-based models are important to validate them, to make them reliable and reusable. Additionally, the broader scope of uncertainty quantification in association with ML models relies on developing robust explanations [39] and in the field of HEP for problems like top tagging will require dedicated understanding of how robust as well as interpretable these models are [40].

Yet another remarkable application of interpretability is to understand how the model conveys information and in doing so, which parts of a DNN most actively engage in forward propagation of information. Such studies could be useful to understand and reoptimize model complexity. Given DNNs have shown remarkable success in jet and event classification, recent work has placed emphasis on developing DNN-enabled field programmable gate arrays (FPGAs) for trigger-level applications at the LHC [41–43]. As resource consumption and latency of FPGAs directly depend on the size of the network to be implemented, it is definitely easier to embed simpler networks on these devices. Hence, methods that allow interpreting a network's response patterns as well as provide critical insights about model optimization without compromising its performance can greatly benefit these new budding fields of ML applications, especially for online event selection and jet tagging at current and future high energy colliders.

Application of state-of-the-art explainability techniques for interpreting jet tagger models is receiving more attention recently [37, 44–46] and has been demonstrated to be successful in identifying feature importance for models like the interaction network (IN) [47]. In this paper, we study the interpretability of a subset of existing ML-based top tagging models. The models we have chosen use multi-layer perceptrons (MLPs) as underlying neural architecture. Choosing simpler neural architecture allows us to elucidate the applicability and limitations of existing XAI methods and develop new tools to examine them without convoluting these efforts with the complexity of larger models or unorthodox data structures. To compare our results for different models as well as with existing benchmarks in published literature, we use the dataset developed by the authors of [23] and later used in the top tagger model review in [34]. The models explored in this paper along with the dataset have been reviewed in section 2. The model hyperparameters explained in this section will constitute the *baseline* model in each category. Variants of each model are studied to better understand their interpretability where the underlying architecture remains the same but model hyperparameters, input features, or data preprocessing might be changed. Section 3 reviews modern XAI methods that we will use in investigating the explainability of top tagger models. In section 4, we analyze the results of applying XAI methods on different top tagger models. Based on the insights from these XAI studies, we propose an interpretability-inspired novel and better-performing tog tagger model in section 5. Section 6 summarizes our findings and illustrates new dimensions to explore in the conjunction of XAI and HEP.

2. Review of top tagging dataset and models

The dataset used in this paper has been used to perform model benchmarking studies in [34] and publicly available at [48]. This dataset consists of 1 million top (signal) jets and 1 million QCD (background) jets generated with PYTHIA8 [49] with its default tune at 14 TeV center of mass energy for proton–proton collisions. The detector simulation was performed with DELPHES [50] and jets were reconstructed using the $anti-k_t$ algorithm [51] with a jet radius of $R = 0.8$ using FASTJET [52]. Only jets with transverse momenta within the range of 550 and 650 GeV are considered. For each jet, the dataset contains the four momenta of up to 200 constituents with zero-padded entries for missing constituents. The dataset is divided into training, validation, and testing sets with a 6:2:2 split. Some characteristic jet features from a random subsample of the training data are shown in figure 1.

In this paper we consider three different NN-based models for top tagging. Given the tagger distinguishes between two jet classes, minimizing the standard binary cross-entropy loss has been used as the training objective for all models. The training is done using the ADAM optimizer with minibatches. All networks showed comparable performance with different batchsizes. The architecture, hyperparameters, and data preprocessing for each of the baseline models is summarized below-

- **TopoDNN** [17, 19]: The simplest top tagging model we consider is a fully connected MLP trained with transverse momentum (p_T), azimuthal angle (ϕ), and pseudorapidity (η) of the 30 most energetic particles. Usually referred to as TopoDNN, this model represents a quintessential MLP network. Although TopoDNN is outperformed by many other ML-models for top tagging, its simple architecture allows us to explore different XAI metrics, their limitations, and the best practices to overcome them. Since MLPs are still widely used in HEP for a wide variety of applications, our studies of modern XAI for this model will also illustrate the best practices to interpret the input–output relations for such models.

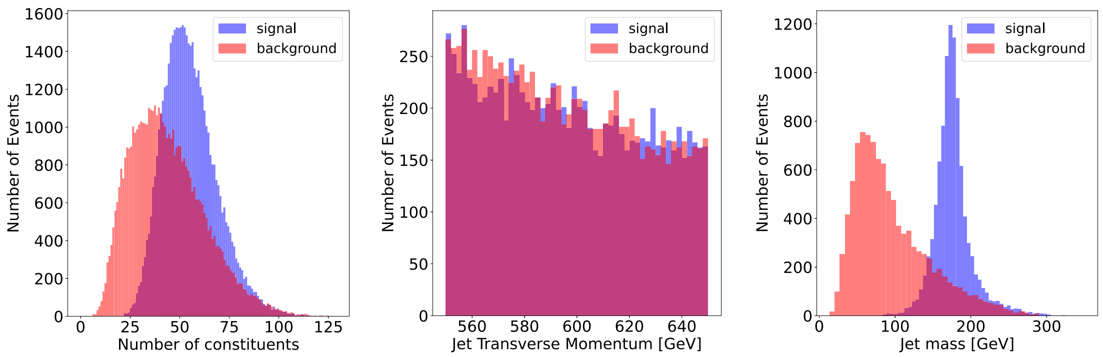
  TopoDNN is trained on preprocessed data where (i) the jet is rotated on the $\eta - \phi$ plane to have the most energetic component aligned along the central coordinate (0,0), (ii) the second most energetic component falls along the negative-ϕ axis, and (iii) all momenta are scaled with an arbitrarily chosen factor of 1/1700. The transformations (i) and (ii) take advantage of the underlying Lorentz invariance of collider physics and (iii) converts the momenta into unitless quantities and scales them down to a numerical range comparable to those of η, ϕ quantities. The baseline model is constructed with 4 hidden layers with 300, 102, 12, and 6 nodes respectively and RELU activation function. The output layer consists of a single node which is converted by the sigmoid function to represent the probability of the jet being classified as a signal jet.
- **Multi-body *N*-subjettiness (MB*N*S)** [20, 21]: Top-tagging with N-subjettiness variables uses an MLP as the underlying trainable architecture. However, the input to the network is different from the usual kinematic variables. It uses the multi-body N-subjettiness variables [53], defined as

  $$\tau_n^{(\beta)} = \frac{1}{p_{T,J}} \sum_i p_{T,i} \min\left\{\Delta R_{1i}^{\beta}, \Delta R_{2i}^{\beta}, \dots, \Delta R_{ni}^{\beta}\right\} \tag{1}$$

  where $p_{T,J}$ and $p_{T,i}$ represent the transverse momenta of the jet and its ith constituent and ΔR_{ki} is the distance between the kth jet axis and the ith particle constituent. The n jet axes chosen for calculating $\tau_n^{(\beta)}$ are obtained using the k_t algorithm [54] with E-scheme recombination [55]. Figure 2 shows the distribution of some of the τ variables for QCD and top jets. The input to MBNS tagger is the set of subjettiness variables

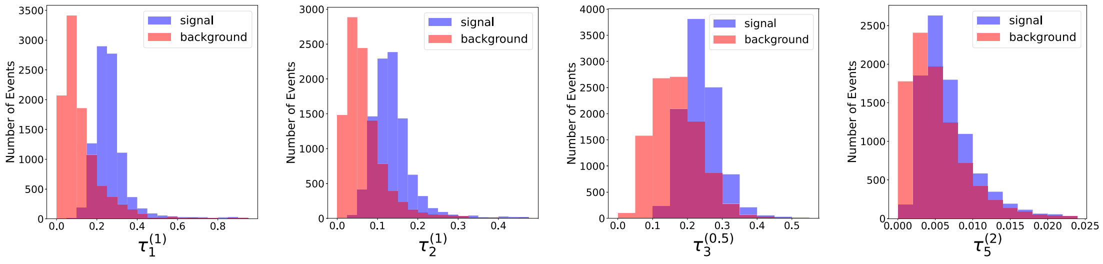
  $$\left\{\tau_1^{(0.5)}, \tau_1^{(1)}, \tau_1^{(2)}, \tau_2^{(0.5)}, \tau_2^{(1)}, \tau_2^{(2)}, \dots, \tau_{N-2}^{(0.5)}, \tau_{N-2}^{(1)}, \tau_{N-2}^{(2)}, \tau_{N-1}^{(1)}, \tau_{N-1}^{(2)}\right\} \bigcup \{p_{T,J}, m_J\} \tag{2}$$

  where, besides the subjettiness variables, the jet p_T and jet mass (m_J) variables are used as inputs to provide a kinematic scale for the jet event. However, the latter inputs are scaled by a factor of 1/1000 to mitigate the several orders of magnitude gap between their numerical range and those of the τs. In our work we consider the MB8S model- an MLP consisting of 4 hidden layers with (200, 200, 50, 50) nodes respectively. The RELU activation function is used for the hidden layers. The output layer consists of 2 nodes transformed by the



(a) (b) (c)

Figure 1. Distribution of (a) number of constituent particles, (b) jet transverse momentum ($p_{T,J}$), and (c) jet mass (m_J) for background QCD jets and signal top jets.



(a) (b) (c) (d)

Figure 2. Distribution of (a) $\tau_1^{(1)}$, (b) $\tau_2^{(1)}$, (c) $\tau_3^{(0.5)}$, and (d) $\tau_5^{(2)}$ for background and signal jets.

SoftMax function to respectively represent the probabilities of the jet being classified as a background or signal jet.

Particle flow network (PFN) [24]: PFNs are built following the deep set [56] architecture, inherently making the network invariant under permutation of particle constituents. The model implements the following relation

$$\mathrm{PFN} = F\left(\sum_{i=0}^{N-1} \Phi\left(p_i\right)\right) \tag{3}$$

where F and Φ represent non-linear functions implemented as trainable NNs, N is the number of jet constituents and p_i is the four momentum of the ith jet constituent. The Φ network learns to create a latent embedding of each constituent particle. These latent embeddings are summed over for all constituent particles of a jet creating jet-level latent embeddings, making the network invariant under permutation of particle constituents. Finally, the jet-level embeddings are passed on to the F network which performs the jet classification.

We train the network with the p_T, η, ϕ of jet constituents as input. As a part of data preprocessing, we standardized the constituents' η and ϕ by subtracting the jet's η and ϕ. Also, the p_T values of the jet constituents are scaled by the inverse of sum of constituent p_Ts, i.e. $^1/\sum_i p_{T,i}$. The Φ network is implemented as an MLP with 3 layers of 100, 100, and 256 nodes respectively. Each layer is followed by a ReLU activation layer. The output layer of Φ represents a 256 dimensional latent space of jet representation. The F network consists of 3 hidden layers with 100 nodes per layer with ReLU activations. The output consists of two nodes transformed by the SoftMax operation to represent the probabilities corresponding to each jet class.

3. Interpretability of ML models: tools and methods

While a number of XAI techniques have been developed in ML literature, how any one of these methods actually *explain* an ML model can actually be quite different from the others [57–59]. Often we find the XAI techniques producing diverging explanations, making it challenging to rely on these methods. We investigate a number of these techniques and compare the corresponding results to understand what may contribute to their divergence. Here, we summarize the methods that we are going to use to explore the interpretability of top tagger models.

- **The ΔAUC method:** Identifying feature importance has been an important part of studying classification models [60]. In standard feature selection tasks, a reasonable subset of the features that excels in some model performance metric is chosen. Although it is conceptually different from feature ranking in post-hoc model interpretation, many interpretation metrics also rely on identifying feature importance with a simpler surrogate model which is trained to minimize a model's performance loss [61]. One of the most useful model analysis tools for binary classification is the region operator characteristic (ROC) curve, and the area under the curve (AUC) serves as a scalar metric for evaluating model performance. ROC-AUC based feature ranking has been widely promoted in ML literature [57, 62, 63]. We adapt those same principles for our model interpretation studies. One straightforward way of evaluating a feature's contribution in making predictions is to investigate the change in model's performance in terms of the ROC-AUC score when a particular feature is masked from the input—by replacing it with a population-wide average value or a zero value, whichever is contextually relevant to the model's relationship with the training dataset.
- **Shapely additive explanations (SHAP) [64]:** The SHAP scores represent a game theoretic approach in identifying the importance of difference features. For each instance of the dataset, the input features are assigned an additive score that determines to what extent a particular feature contributes to the classifier prediction. An average model prediction is determined by replacing each feature by its population average and then individual features are added back to the model to find their impact on leading the prediction towards the optimal value. For each feature, the SHAP score is determined by evaluating the average contribution of adding the feature over all possible feature subsets defined without that feature. Given that evaluating exact SHAP scores require iterating over 2^n sets of feature combinations for each data instance with n features, several simplifying assumptions are made to reduce the computational complexity. In our work, we use the kernel SHAP method—a model agnostic approach to obtain local explanations similar to the LIME framework [65] and obtained by generating random samples around the data point and performing a mean-squared-error-minimizing linear regression over the samples to evaluate the SHAP scores. In order to avoid any overfitting in obtaining the SHAP score, the number of samples in each model were chosen to be at least twice as many as the number of input features
- **Layerwise relevance propagation (LRP) [66, 67]:** The LRP technique propagates the classification score predicted by the network backwards through the layers of the network and attributes a partial relevance score to each input. The backpropagation of LRP scores in an MLP network is obtained by the following relation-

$$r_j^{(n)} = \sum_k \frac{a_j^{(n)} w_{jk:n}}{\sum_m a_m^{(n)} w_{mk:n}} r_k^{(n+1)} \tag{4}$$

where $a_j^{(n)}$ and $r_j^{(n)}$ are the activation and relevance scores of the jth node in nth layer and $w_{jk:n}$ is the weight that determines the contribution of the jth activation in the nth layer to the kth node in layer $n+1$. The relevance score of the final layer is the same as the network outputs. The inputs to the network are identified as the 0th layer and the relevance scores assigned to them are denoted as $r_j^{(0)}$. The original LRP method has been developed for simple MLP networks. Variants of this method have been explored to propagate relevance across convolutional neural networks [68] and graph neural networks [69]. While the basic LRP rule in equation (4) conserves the total relevance score i.e. the classifier network's output, based on the distribution of weights and activations, relevance scores can become unbounded when $\sum_k a_j^{(n)} w_{jk:n} \to 0$. To overcome this, the LRP rule is modified to treat positive and negative weights asymmetrically. We use the so called LRP-γ rule defined as-

$$r_j^{(n)} = \sum_k \frac{a_j^{(n)} \left(w_{jk:n} + \gamma w_{jk:n}^{+}\right)}{\sum_m a_m^{(n)} \left(w_{mk:n} + \gamma w_{mk:n}^{+}\right)} r_k^{(n+1)} \tag{5}$$

where $w^{+} = w \cdot \Theta(w)$, Θ being the Heaviside step function and γ is a regularization parameter. In our representation, we always present the normalized relevance scores so that $\sum_j r_j^{(0)} = 1$.

Table 1. Comparison of different XAI methods in terms of their implementation heuristics. We consider a method scalable if the complexity of its implementation grows at most by linear order. A local explanation refers to explanation metrics assigned to individual features for a given data point while a global explanation refers to explanation metrics assigned to individual features for the entire dataset.

	ΔAUC	SHAP	LRP	RNA/NAP
Scalability in input dimension	✘	✘	✓	✓
Local explanation	✘	✓	✓	✘
Global explanation	✓	✓	✓	✓
Requires forward propagation	✓	✓	✓	✓
Requires backward propagation	✘	✘	✓	✘
Susceptible to spurious correlations	✓	✓	✓	✘
Addresses model complexity	✘	✘	✘	✓
Requires retraining	✘	✘	✘	✘

- **Neural activation pattern (NAP) diagrams** [44, 45]**:** While the aforementioned methods help identify the importance of features for a trained NN model, the NAP diagrams visualize the information propagation pathways through the network's architecture. The NAP diagram visualizes the relative neural activation (RNA) score, defined as-

$$\mathrm{RNA}(j,k;\mathcal{S}) = \frac{\sum_{i=1}^{N} a_{j,k}(s_i)}{\max_j \sum_{i=1}^{N} a_{j,k}(s_i)} \tag{6}$$

where $\mathcal{S} = \{s_1, s_2 \ldots s_N\}$ represents a set of samples over which the RNA score is evaluated. The quantity $a_{j,k}(s_i)$ is the activation of jth neuron in the kth layer when the input to the network is s_i. When summed over all the samples in the evaluation set $\mathcal{S}$, this represents the cumulative neural response of a node, which is normalized with respect to the largest cumulative neural response in the same layer to obtain the RNA score. Hence, in each layer, there will be at least one node with an RNA score of 1. Since the neurons are activated with ReLU activation in the models we consider, the RNA score will be strictly non-negative, and $\leqslant 1$. In a qualitative way, we are trying to see which neurons most actively engage to obtain the predictions from our models. Since the MLPs in our models consist of only Dense layers, each layer takes all the activations from the previous layer as inputs. As all nodes within a given layer are subject to the same set of inputs, we can reliably estimate how strongly they perceive and transfer that information to the next layer by looking at their activation values. For the same reason, we normalize the cumulative activation of a node with respect to the largest aggregate in the same layer.

The NAP diagram is obtained by presenting the RNA scores for the different layers of the model as a two dimensional heatmap where along the horizontal axis lies the different activation layers of the network and the vertical axis represents the different nodes in those activation layers. NAP diagrams illustrate the relative activity level of different nodes within each layer and hence can demonstrate the sparsity of the model's activity.

The methods that we have explored so far adopt widely different approaches to understanding different aspects of an NN. A summary of their properties is given in table 1. In our analyses, we use models with $\mathcal{O}(10-100)$ inputs, so scalability is not a major bottleneck for application of these methods. On the other hand, although some of these methods allow exploring local explanations, i.e. explanations for individual data samples, we concern our studies with global explanations alone.

4. Model interpretability for top taggers

Ideally, we expect an XAI metric to correctly identify features that the NN consider most important. Hence, any post-hoc feature ranking XAI method should ideally identify the same set of features though their relative rankings can moderately vary. However, there is no straightforward correlation among XAI methods introduced in the previous section. In the following subsections, we first investigate and validate these methods in the context of simpler TopoDNN and MB8S models. Building on the insights obtained from these studies, we use these tools to interpret the PFN model and investigate its latent space representation.

4.1. TopoDNN

Since TopoDNN is arguably the simplest NN-based model to perform top tagging, it is perhaps the most ideal model to investigate different aspects of XAI. Given that correlation among features has been demonstrated to be an important aspect of identifying feature rankings in classical ML [70] as well as modern XAI methods, we start by examining the pairwise Pearson correlation coefficient for a subset of the input features for background and signal jets in figure 3. The correlation matrices for both jet categories are mostly sparse except for some large anti-correlations between $p_{t,0}$, the transverse momentum of the most energetic jet constituent, and that of some of the low energy constituents. Given the dataset has been generated within a limited jet p_T range, such anti-correlations are expected—the higher the energy of the most energetic constituent, the lower the energy of the remaining constituents. Given that the numerical range of the p_t of lower energy constituents is typically much smaller than that of the highest energy constituent, we can expect the impact of their anti-correlations with $p_{t,0}$ on the NN's performance to be rather small. We can indeed verify that in figures 4(a)–(c) where we identify the important features for the TopoDNN model using the ΔAUC (figure 4(a)) and SHAP scores (figures 4(b) and (c)).

The ΔAUC score cannot independently identify the features that contribute to identification of signal and background jets. By evaluating the SHAP scores for subsets of the dataset that only contain one kind of jets, one can identify the features that most dominantly contribute to identification of the corresponding jet class. However, as shown in figures 4(b) and (c), there is a significant overlap between the features that are identified as the most important ones for both jet categories. Unlike computer vision models that deal with image or videos as input data and importance distribution for different images can vary based on which pixels carry the most relevant information, the same feature can contribute equally importantly for different classes in models with tabular data. Why the network treats the same set of variables as important becomes clearer upon inspecting the distribution of some of these preprocessed features as shown in figures 4(d)–(g). ϕ_1, ϕ_2, η_2 all show strong classification characteristics, and given these variables are either loosely correlated or almost uncorrelated as shown in figure 3, they all can independently contribute to the network's ability to tell apart the different jet classes. On the other hand, $p_{t,0}$ by itself is a modest discriminator and hence identified as having a modest impact on the model's performance. This has also been verified by training a variant of the TopoDNN model that excludes the $p_{t,0}$ variable and as shown in table A1, performs almost equally as well as the baseline model.

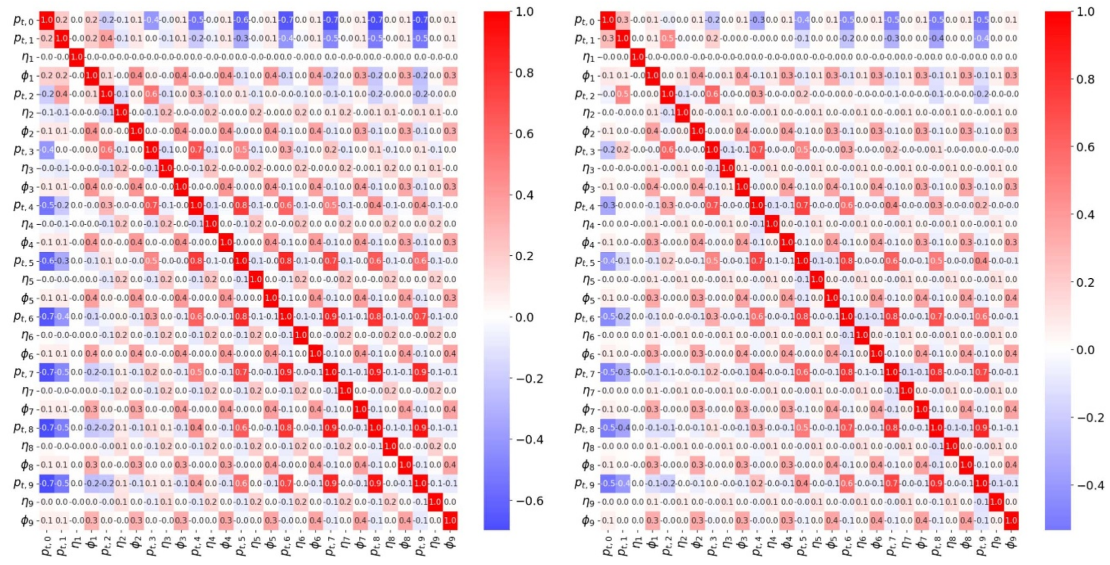
However, we see a stark difference in the distribution of the relevance scores (figure 5) among different features, obtained from the LRP method, when compared to other feature ranking metrics we have considered so far. Unlike SHAP or ΔAUC scores, a subset of the p_t variables have the largest relevance scores for both jet categories. While the most highly ranked features from the previous two methods show strong discriminating characteristics, some of the highly ranked features from the LRP method show very little discriminating capacity. This difference can be understood from the nature of these ranking methods. Both ΔAUC and SHAP calculate the model's deviation from the *mean behavior*, i.e. qualitatively, they both represent how much information is obtained from inclusion of the true value of a feature instead of using the population mean as a feature mask. On the other hand, LRP calculates the feature's cumulative relevance, which additively includes the relevance scores attributed to each feature's mean behavior. Assuming $\vec{x} = \{x_i\}$ be a sample jet event taken from the set of events X and $\vec{x}_{\setminus k} = \vec{x}\setminus\{x_k\}\bigcup\{\mathbf{E}(X_k)\}$ be the event set where we mask the kth input feature by replacing it with its mean value, the linear order behavior of the NN can be approximated using the deep taylor decomposition formalism [71]-

$$f(\vec{x}) \approx f(\vec{x}_{\setminus k}) + \frac{\partial f}{\partial x_k}(x_k - \bar{x}_k) \tag{7}$$

where $f(\vec{x})$ represents the output of the NN before the final SIGMOID activation. Noting that the relevance scores additively distribute the functional output among the different inputs, i.e. $f(\vec{x}) = \sum_i r(x_i)$ and $f(\vec{x}_{\setminus k}) = \sum_{i\neq k} r(x_i) + r(\bar{x}_k)$, we can rewrite equation (7) as,

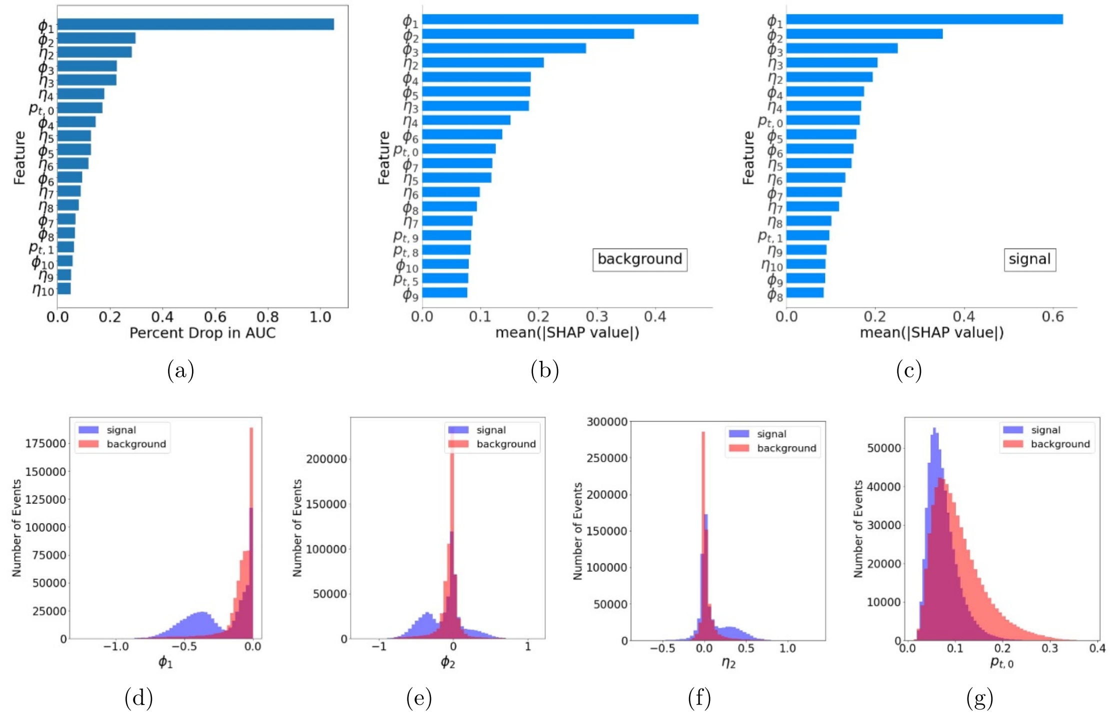
$$\sum_i r(x_i) \approx \sum_{i\neq k} r(x_i) + r(\bar{x}_k) + \frac{\partial f}{\partial x_k}(x_k - \bar{x}_k) \tag{8}$$

We define $\delta r_k = f(\vec{x}) - f(\vec{x}_{\setminus k}) \approx \frac{\partial f}{\partial x_k}(x_k - \bar{x}_k)$ as the *differential relevance score* attributed to the corresponding feature. When the features are loosely correlated, collecting terms with equivalent indices in equation (8) and ignoring higher order effects, we can write $r(x_k) \approx r(\bar{x}_k) + \delta r_k$ where we denote $r(\bar{x}_k)$ as the *mean-behavior relevance score*. Figure 6(a) show the absolute mean behavior relevance scores of different features and the relative size and distribution of the relevance scores are very similar to what we can see in figure 5. This explains that a large contribution of the LRP scores actually comes from the mean behavior relevances, and



(a) (b)

Figure 3. Feature correlation matrix for p_T, η, and ϕ of the 10 highest energy constituents of (a) background QCD jets and (b) signal top jets.



(a) (b) (c)

(d) (e) (f) (g)

Figure 4. Feature ranking obtained from the (a) ΔAUC scores, (b) SHAP scores for background QCD jets, and (c) SHAP scores for signal top jets. Only the 20 highest-ranked features are shown. Figures (d)–(g) show the distributions of the inputs ϕ_1, ϕ_2, η_2, and $p_{t,0}$ for two jet categories after performing the preprocessing described in section 2.

has very little to do with the network's ability to distinguish different jet types. In fact, many of the angular variables that are regarded as highly important by ΔAUC and SHAP methods also show large differential relevance, as shown in the distributions of mean absolute differential relevance (MAD relevance) scores (normalized with respect to relevance scores from the baseline model) in figures 6(b) and (c).

Now we turn our focus to examine the behavior of the internal architecture of the model with NAP diagrams. As discussed in section 3, NAP diagrams plot the RNA scores of different nodes of the activation layers within the network. Figure 7 shows the 2D map of RNA scores for QCD and top jets where the RNA scores of the former are plotted as negative values to allow simultaneous visualization. It can be readily

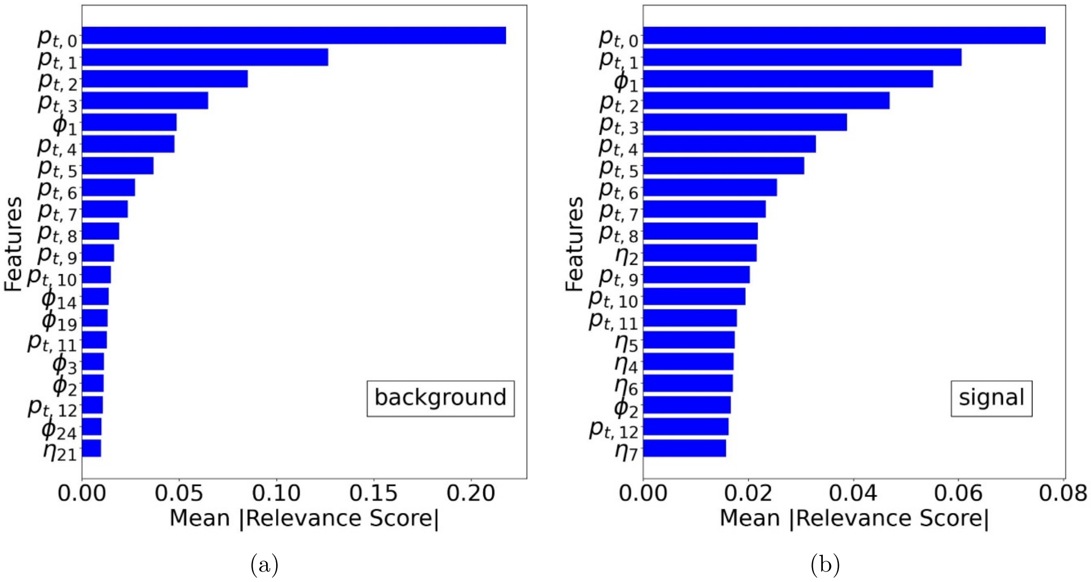


Figure 5. Distribution of the relevance scores obtained from LRP for (a) background QCD jets and (b) signal top jets. Only the 20 highest-ranked features are shown for each jet category.

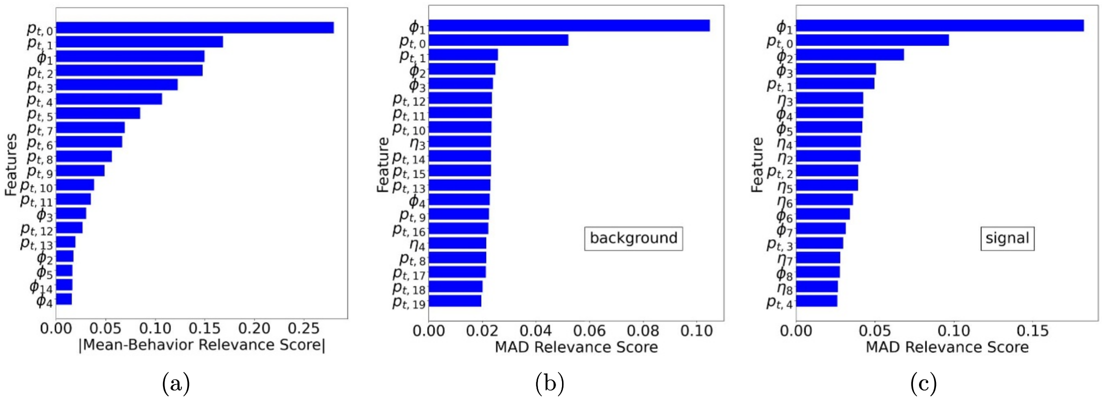


Figure 6. Distribution of the (a) absolute mean-behavior relevance score, (b) MAD relevance score for background QCD jets, and (c) MAD relevance score signal top jets. Only the 20 highest-ranked features are shown.

understood that the network in figure 7 is quite *sparse*, i.e. most nodes show relatively smaller activations. We can heuristically quantify *sparsity* of the network by the fractional number of hidden activation nodes with an RNA score less than a given threshold. We arbitrarily choose this threshold to be 0.2 and find that the network's sparsity measures for background and signal jet categories are 0.86 and 0.76 respectively, giving an overall sparsity measure of 0.70. This implies that about 70% of the nodes show a cumulative activity level of less than 20% compared to that of the most active node in the corresponding layer. We also see in figure 7 that the most active nodes for different jet categories are almost completely disentangled by the time information propagates to the third layer. Large sparsity of the network and early disentanglement of jet categories indicate the network's complexity can be reduced without any noticeable compromise in its performance. To demonstrate this, we have trained variants of the TopoDNN model with lesser complexity by simultaneously reducing the depth and width of the MLP model. As shown in table A1, these simplified models perform almost equally well while the model complexity is significantly reduced.

4.2. MB8S

Although the underlying architecture of the MB8S model is an MLP, there is stark difference among the input features. Unlike the input to the TopoDNN model explored in section 4.1, the inputs to the MB8S model are highly correlated for both jet categories (figure 8). Such large correlations among features can make it hard to distinguish whether a feature truly conveys independent discriminating characteristics or if a feature is deemed important by a model simply because it is correlated with another feature. Training an NN

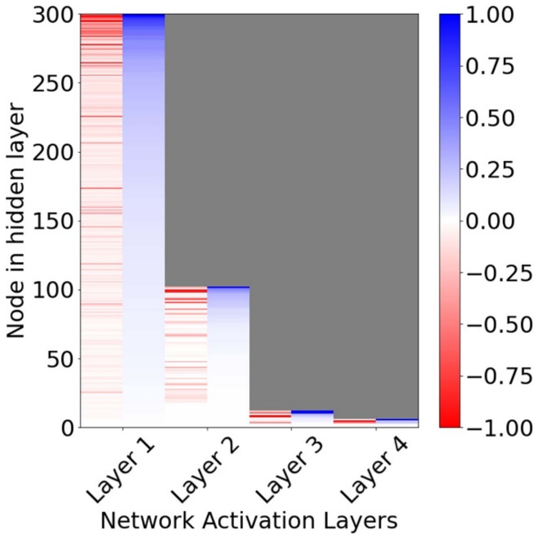
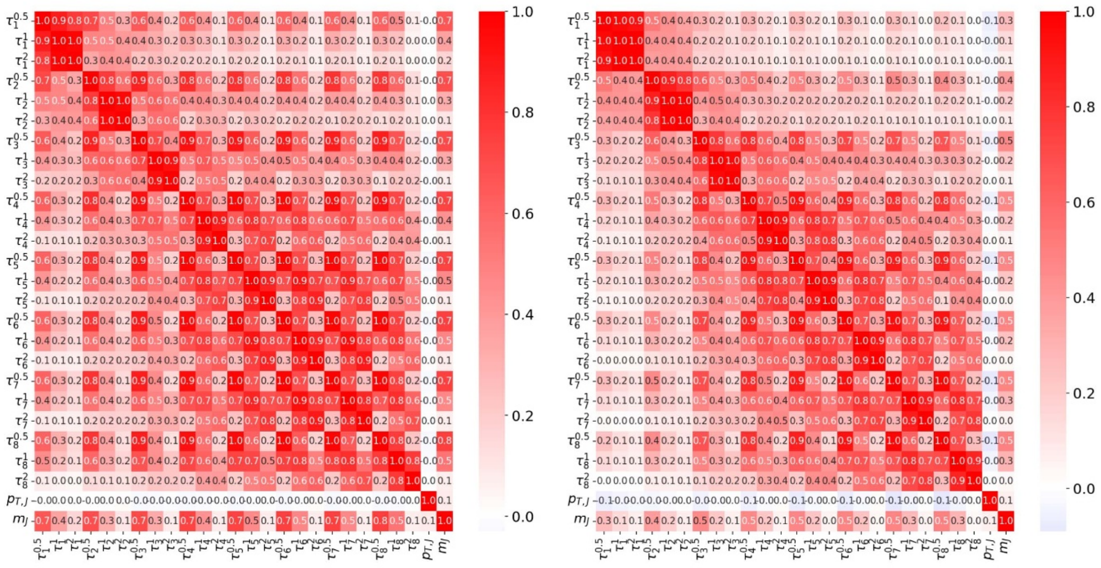


Figure 7. NAP diagram for TopoDNN model visualizing a 2D map of RNA scores for different nodes of the activation layers. To simultaneously visualize the scores for QCD and top jets, we project the RNA scores of the former as negative values. The nodes in each layer are ordered according to their RNA scores for the top jets.



(a)

(b)

Figure 8. Feature correlation matrix for (a) background QCD jets and (b) signal top jets for the different input features for the MB8S network.

with correlated feature inputs can contribute to increased model complexity [72], overfitting [73], and obscure its interpretability [74]. The MB8S network has been trained with DropOut layers [75] with a dropout rate of 0.2 (0.1) for the first (final) two hidden layers to protect it from the problem of overfitting. With such large correlations among input features, we want to differentiate between two aspects of a feature's importance- its independent contribution to a network's decision making process and its deemed importance in a certain instance of a trained model because of its correlations with other features.

Figure 9(a) shows the distribution of ΔAUC score for the top 10 features. When compared with the SHAP score distributions for background and signal jets in figures 9(b) and (c), the sets of top ranking features for these different methods show significant overlap but their raking shows some noticeable difference, especially for the jet mass feature. As shown in figure 1(c), the distribution of jet mass is very different for the two types of jets and naturally, it is expected to be a strong discriminator. This is also reflected in the distribution of SHAP scores. But the ΔAUC ranking places the jet mass variable at a lower ranking compared to subjettiness variables $\tau_1^{(2)}$ and $\tau_2^{(2)}$. As shown in figure 8, these variables demonstrate

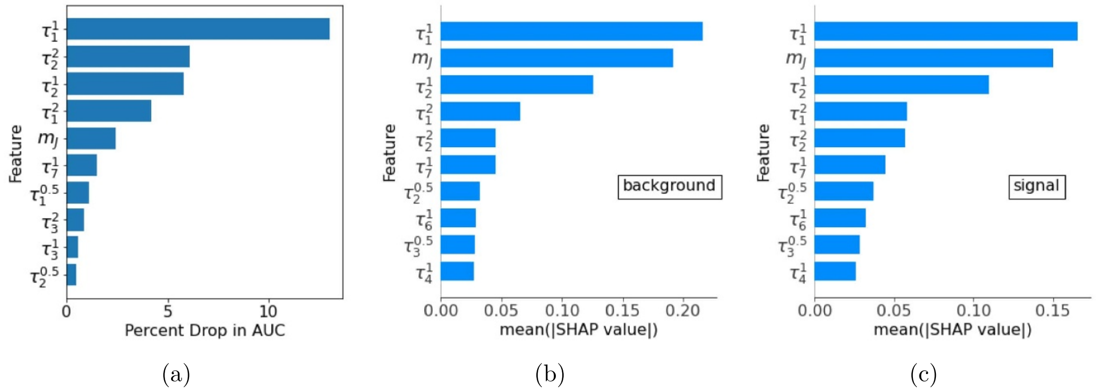


Figure 9. Feature ranking obtained from the (a) ΔAUC scores, (b) SHAP scores for background QCD jets, and (c) SHAP scores for signal top jets for the MB8S network. Only the 10 highest-ranked features are shown.

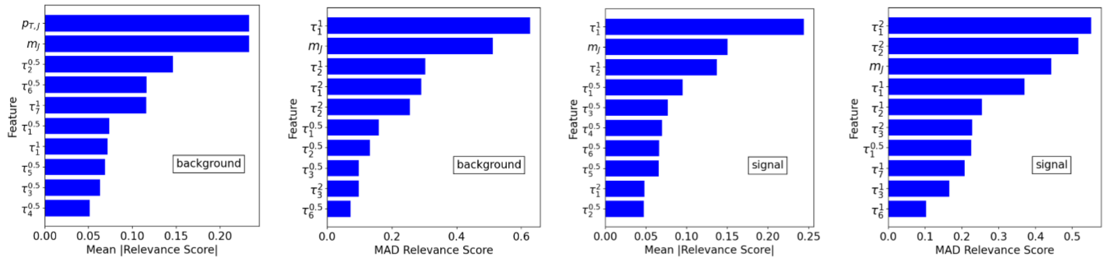
almost 100% correlation with $\tau_1^{(1)}$ and $\tau_2^{(1)}$ respectively, both of which are identified as top ranking variables in both ΔAUC and SHAP rankings and have strong discriminative distributions as shown in figures 2(a) and (b). We do not expect a one-to-one correspondence between the feature rankings from ΔAUC and SHAP scores. Nevertheless, having a lower rank for jet mass compared to variables that display almost perfect correlations with other strong discriminators naturally intrigues the question whether it is overshadowing the importance of variables that actually adds new information to the classifier. It has been observed that the inclusion of jet mass significantly improves the performance of the MBNS taggers [20]. Hence, the jet mass distribution has the ability to better contribute to a network's decision-making process compared to other highly correlated subjettiness variables.

In order to investigate whether the highly correlated variables can actually independently contribute to the network's performance, we train a variant of the MB8S network where only the $\tau_x^{(1)}$ variables are included, along with the mass and transverse momentum of jets. This choice is inspired from the block-diagonal concentration of pairwise correlations in figure 8. This alternate network has almost identical performance as compared to the baseline MB8S network as shown in table A1. The feature rankings via ΔAUC and SHAP scores for this network also consistently identify $\tau_1^{(1)}, \tau_2^{(1)}$, and jet mass as the most important features for this classification model. These top ranking features display relatively weaker correlations among themselves (correlation coefficients $\leqslant$0.4) and hence, can contribute new information to the classifier's decision making process. Moreover, since the two networks demonstrate almost equivalent performance, the highly correlated subjettiness variables only marginally impact the network's performance. The relatively high ΔAUC score attributed to $\tau_1^{(2)}$ and $\tau_2^{(2)}$ in figure 9(a) must be resulting from strong feature correlations.

This, however, does not imply that these features are unimportant for the particular instance of the trained network we investigated. On the contrary, the large ΔAUC scores associated with these variables indicate that the trained MB8S model depends on these correlations for proper inference. But the large importance associated with these variables should not be interpreted as their independent contribution to jet classification.

Next we turn our attention to relevance scores attributed to different input features by the LRP method. From our studies of the LRP method for the TopoDNN model, we know the relevance scores can be an unreliable measure in identifying how important a feature really is. We can see that pattern repeated for the MB8S network too. Figures 10(a) and (c) show the mean absolute relevance scores attributed to different features for background and signal jets. LRP attributes a large relevance score to $p_{T,J}$, the transverse momentum of the jet for the background jets. However, this variable is one of the least expressive features for the network. It has almost no correlation with other features, and has a very similar distribution for both jet types. Hence, assigning this feature a very large relevance score definitely raises some concerns about the reliability of the feature ranking obtained from LRP. The distribution of MAD relevance scores in figure 10(b) gives a more appropriate distribution for feature importances. For the top jets, the MAD relevance distribution (figure 10(d)) identifies $\tau_1^{(2)}, \tau_2^{(2)}, m_J$ as the most important features. $\tau_1^{(2)}, \tau_2^{(2)}$ also ranked high in the ΔAUC metric and we have explained how their importance primarily stems from their correlations with other expressive input features to the network.

To demonstrate how the different hidden activation layers contribute to information propagation for the two jet categories, we show the 2D map of the RNA scores of different nodes in figure 11. Although the



(a) (b) (c) (d)

Figure 10. Distribution of (a) relevance scores for background QCD, (b) MAD relevance scores for background QCD, (c) relevance scores for signal top jets, and (d) MAD relevance scores for signal top jets for the MB8S network. Only the 10 highest-ranked features are shown for each jet category.

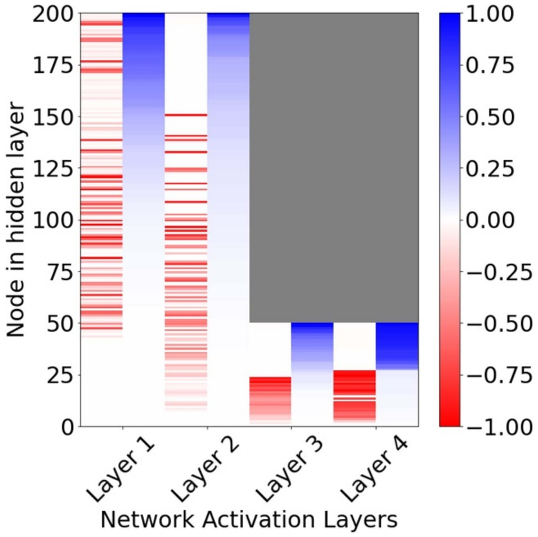


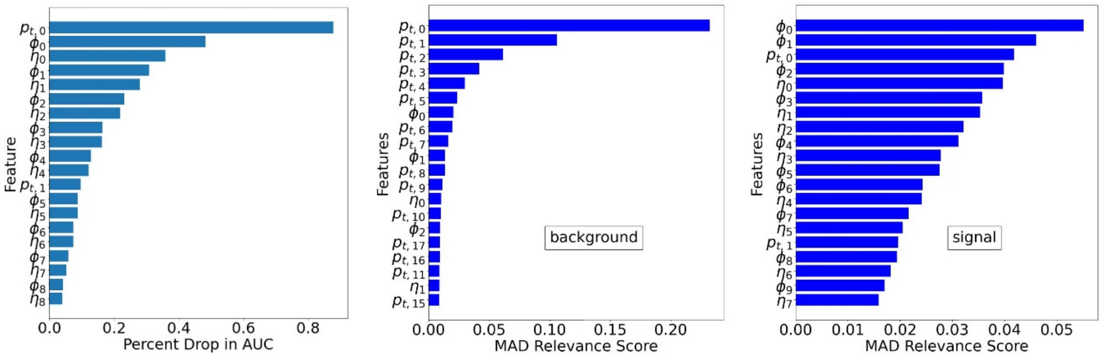
Figure 11. NAP diagram for MB8S model visualizing a 2D map of RNA scores for different nodes of the activation layers. To simultaneously visualize the scores for QCD and top jets, we project the RNA scores of the former as negative values. The nodes in each layer are ordered according to their RNA scores for the top jets.

network appears to be relatively sparse for different jet categories, with a sparsity measure of 0.74 and 0.64 for background and signal jets measured with respect to a threshold of 0.2 on RNA scores, different nodes are most strongly activated for signal and jet categories. As a result, the network's overall sparsity becomes 0.44. However, we can already see that the nodes that are most strongly activated by the two jet categories are almost completely disentangled at Layer 2. This indicates that the network might be simplified by choosing a shallower network and indeed verified in table A1 where we see that models trained without the final two layers have almost identical performance metrics.

4.3. PFN

While the previously analyzed TopoDNN and MB8S models both employed a single MLP to perform the jet classification, PFN utilizes a deep set topology that linearly combines particle-level neural embeddings to obtain a jet level latent space, which eventually is used to train a second MLP to learn the classification task.

We start by examining the feature ranking from the ΔAUC and MAD relevance scores in figure 12. We note that the ranking of features obtained from the ΔAUC metric is somewhat different from the ranking of the MAD relevance scores for the two jet categories. For instance, the azimuthal angle of the most energetic jet constituent, ϕ_0 appears with a low MAD relevance score for background QCD jets while appearing as the top-ranked feature for the signal jets while also reporting a relatively large ΔAUC value. On the other hand, the relative transverse momentum of that same constituent has the largest MAD relevance score for background jets as well as the largest ΔAUC score while being ranked after ϕ_0, ϕ_1 for the signal jets. This difference in MAD relevance ranking for the two jet classes is somewhat unlike what we have seen for the classifier models previously investigated. These deviations in relative rankings are understood by examining the distributions of the corresponding features. Note that in linear order, the differential relevance



(a) (b) (c)

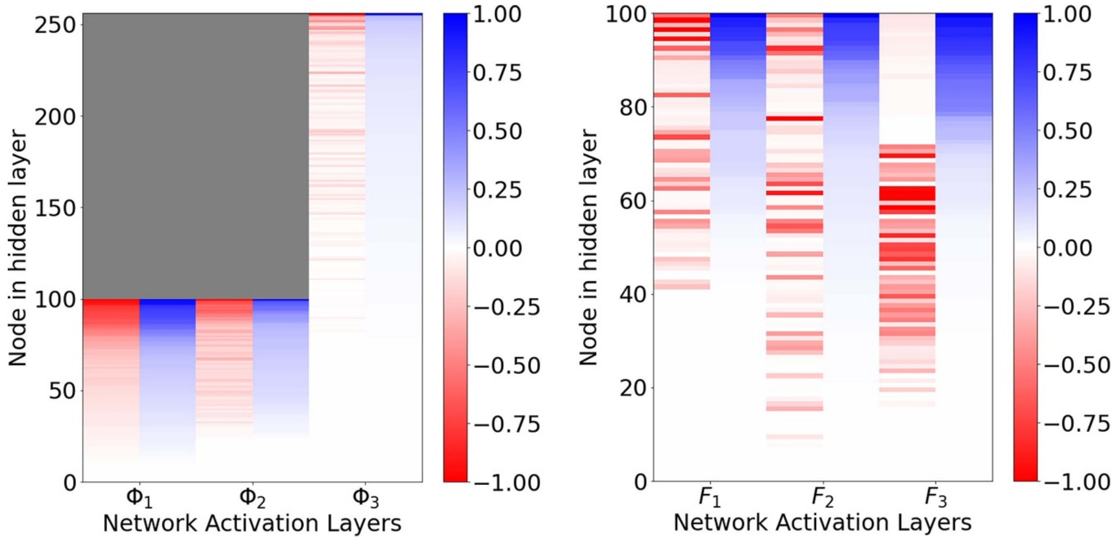
Figure 12. Feature ranking obtained from the (a) ΔAUC scores, (b) MAD relevance scores for background QCD jets, and (c) MAD relevance scores for signal top jets for the PFN network. Only the 20 highest-ranked features are shown.

$\delta r_k \propto x_k - \bar{x}_k$. Hence, larger deviations from the mean can lead to larger differential relevance scores. The distribution of $\phi_0 - \bar{\phi}_0$ is sharply peaked at zero for background jets. Hence, it is reasonable for this variable to have a low MAD relevance score for background jets. On the other hand, the same variable shows long-tailed distributions for the signal jets and hence, has a larger impact on its MAD relevance score. Similarly, the distribution of $p_{t,0} - \bar{p}_{t,0}$ has long tails for both jet classes with a wider spread for the background jets. As a result, this variable shows up high in ranking for both jet classes, while being ranked higher for background jets compared to signal jets.

The TopoDNN network is also trained on a similar set of inputs and hence, it is naturally expected to see noticeable overlap in the set of important features for these two networks. However, the input data preprocessing is very different for these two networks which may result into significant differences on how these variables are treated by the corresponding models. TopoDNN always centers the most energetic constituent along the origin in the (η, ϕ) plane while using a fixed scale normalization for the p_t variables. Hence, the ϕ_0 variable is trivially set to zero for all jets and it does not appear in TopoDNN's list of top-ranked variables. While such differences make it difficult to obtain a one-to-one comparison between the two sets of features rankings, some common conclusions can be drawn from both models. For instance, distributions pertaining to the highest energy constituents are (trivially) more important than the lower energy constituents. We also see that angular distributions of these constituents play a more decisive role in determining the jet class for the top jets. On the other hand, the distributions of their transverse momenta have a larger impact in classifying the background jets.

The particle level embeddings obtained by the Φ network are summed over to obtain a latent space representation of the jet. Characterization of latent spaces has been a topic of general interest in many areas of ML application. For instance, disentangling semantic features of images via latent spaces in variational autoencoders (VAEs) [76] and its variants [77–80] has been widely studied in modern ML literature. In the context of collider physics, how latent spaces embed information and can be used as effective candidates for anomaly detection and bump hunting has been studied [24, 30, 81]. The proponents of PFN performed detailed studies showing how the latent space representation forms discernible contours in the (η, ϕ) plane of jet image representations. While such studies are useful to divulge geometric features of the latent space configuration, interpreting how they actually contribute to the network's decision making process, especially for large latent spaces with $\mathcal{O}(100)$-dimensions, remains unexplored.

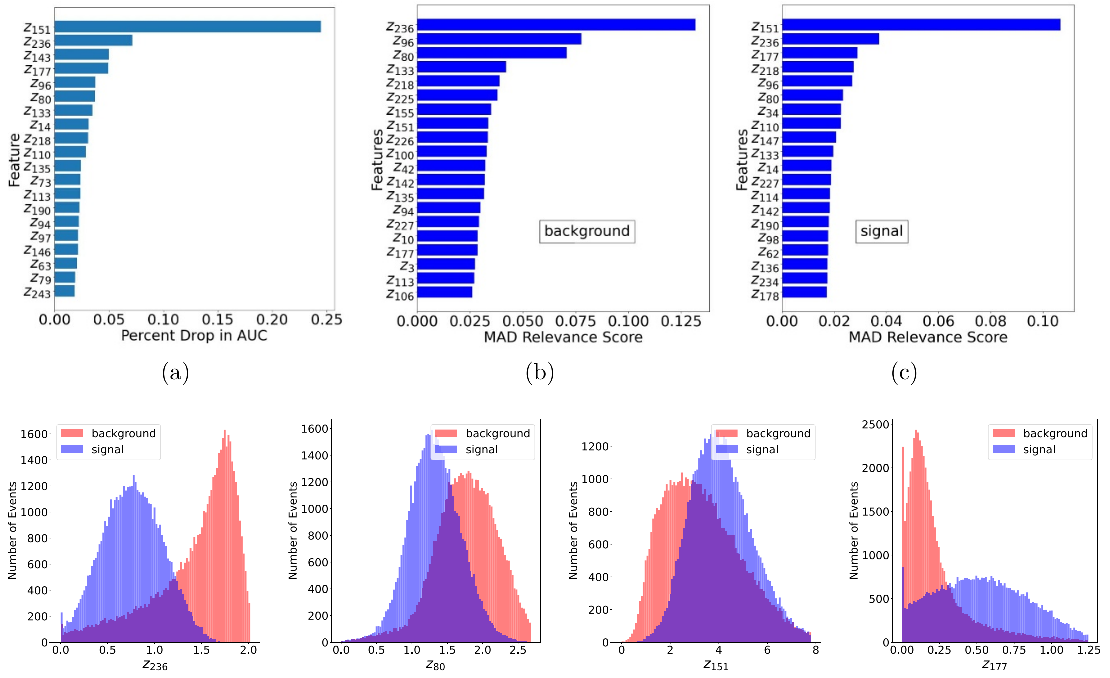
Since the PFN network is trained to obtain classification scores in two disentangled dimensions in the very last layer of F network, it is only expected that the information propagation pathways for the different types of jets will show some level of disentanglement within the hidden layers of the networks. This is indeed verified by the NAP diagrams shown in figure 13. These NAP diagrams reveal some crucial insights. Firstly, we clearly see that the activity level of different nodes in the final layer of the Φ network for background and signal jets is very similar. It implies that the network embeds the jet-level information in the same latent subspace. Secondly, the latent space appears to be very sparse and we indeed found that many of the latent space variables are identically zero for all events in both jet categories. A third observation is that the F network effectively learns to disentangle the representation of jet classes only at the third hidden layer (figure 13(b)). But the first layer of the F network is quite sparse with RNA scores for almost 40% of the nodes being very close to zero for both jet classes. With these observations in mind, we retrained variant networks with latent space dimensions of 64 and 32 while reducing the number of nodes in the first hidden layer of F and still got comparable performance (table A1).



(a)

(b)

Figure 13. NAP diagrams for (a) Φ and (b) F networks for the PFN model showing the RNA scores of different nodes in these networks for the two jet classes. To simultaneously visualize the scores for QCD and top jets, we project the RNA scores for the QCD jets as negative values. The nodes in each layer are ordered according to their RNA scores for the top jets.



(a)

(b)

(c)

(d)

(e)

(f)

(g)

Figure 14. Figures in the top row show the latent space feature rankings obtained from the (a) ΔAUC scores, (b) MAD relevance scores for background QCD jets, and (c) MAD relevance scores for signal top jets. Only the 20 highest-ranked features are shown. Figures in the bottom row show the distribution of some of the highest-ranked latent space features for both jet categories.

Given the sparsity of the latent space representation, we expect that only a small subset of these latent features will actually have a strong contribution towards the decision making process of F in the baseline model. The ranking of different latent features using the ΔAUC score and MAD relevance scores are shown in figures 14(a)–(c). There are noticeable overlaps among the features that rank high with these methods, though the actual sequence of latent variables, understandably, shows some differences. We show the distributions of some of these latent space embeddings in figures 14(d)–(g).

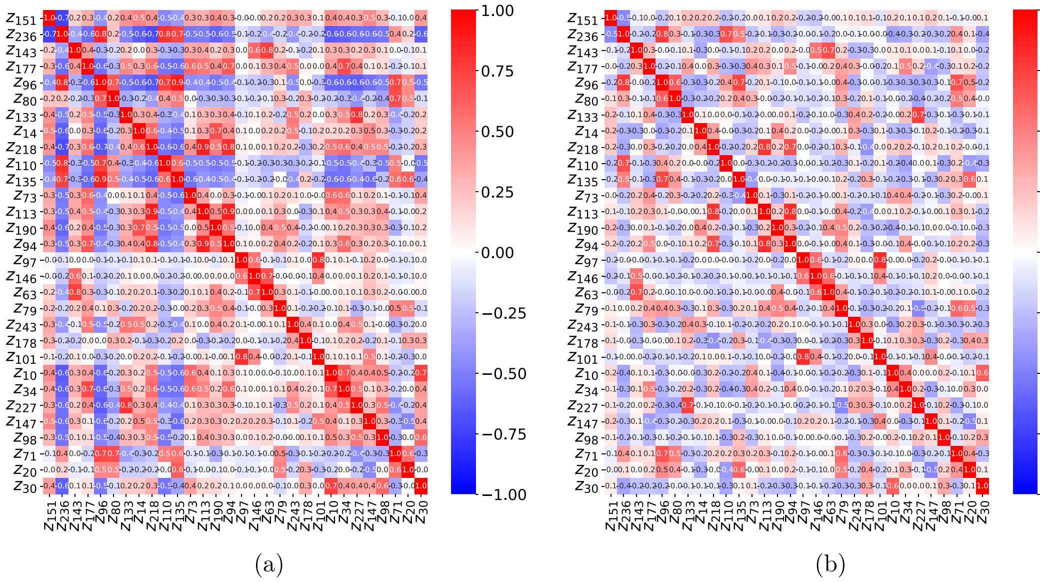
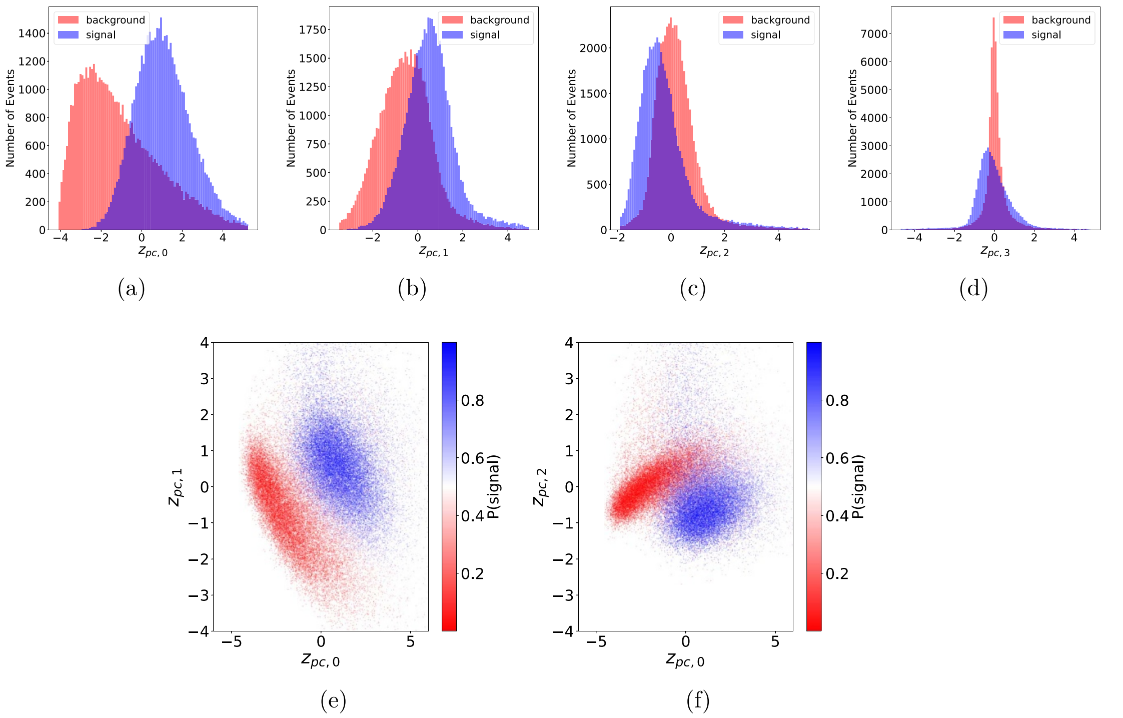


Figure 15. Correlation matrix for latent space features for (a) background QCD jets and (b) signal top jets. Only the top 30 latent features obtained from the ΔAUC metric are shown.

These distributions highlight a stark contrast between the latent space representation for PFN when compared to those studied in the context of generative models like VAEs. The latter class of models are known to provide semantic disentanglement in their latent spaces creating a clear separation in latent space dimensions that account for variabilities in input distributions. For instance, VAEs trained on the popular celebrity portrait dataset CelebA [82] have successfully demonstrated disentanglement of semantics like age, gender, skin tone, hairstyle etc [77] in the latent space. In the context of the top tagging dataset, it is the jet kinematics that is embedded in the latent space. As seen from the NAP diagram in figure 13(b) and the latent space feature distributions in figures 14(d)–(g), the PFN latent space does not provide any disentanglement between the jet classes. In fact, for models like PFN that are trained to learn the jet classes and not the distribution of training dataset, the model has no additional incentive in disentangling the jet classes. Rather, PFN learns to embed the information regarding jet classes in correlations among latent features. This can be seen from the latent space correlation matrices of the two jet classes shown in figure 15. The pairwise feature correlations are quite different for the two jet classes, creating a clearer context for the classifier network F to obtain the desired jet classification.

To illustrate the nature of the jet-class-identifying characteristics in the correlations among the latent features, we examine the distribution of variances in the datasets using principal component analysis (PCA) [83]. PCA performs a linear transformation on these features to obtain a set of orthogonal feature spaces with no cross-correlation among the transformed features. Since the size of the latent space is large but sparse, we select the top-ranking subset of latent features so that simultaneously masking each latent feature in the remaining subset causes at most 1% drop in the AUC score from the test data. For our baseline PFN model, this requires choosing 95 of the 256 latent features. We found that 99% of the observed variance in the test data was described by the top 37 principal components. We show the distribution of the top four components in figures 16(a)–(d). We can readily see how these PCA-transformed latent features can differentiate between the two jet classes in figures 16(e) and (f).

Having examined the disentanglement between the jet classes by the principal components of the latent space, it is also instructive to investigate the physical nature of the latent space learned by the network. While it is neither trivial nor obvious for neural networks to learn about features that bear meaning to humans, it has been seen that latent space networks can occasionally learn about physical variables [30]. In case of the PFN, since the latent space dimensions are highly correlated with each other, we chose to study the correlation between the principal components and jet features like jet mass, the number of constituents, and the subjettiness variables which, as shown in figures 1 and 2, can have moderate to strong discriminative power. We found that the first principal component, $z_{pc,0}$ (figure 16(a)) shows a strong correlation with jet mass for both jet categories with correlation coefficients being 0.82 and 0.64 for background and signal jets respectively. $z_{pc,0}$ also shows strong correlations with the number of jet constituents. $z_{pc,1}$ and $z_{pc,2}$ showed moderate to strong correlations with the subjettiness variables $\tau_1^{(1)}$ and $\tau_2^{(1)}$, implying the PFN model also learns to somewhat reconstruct distributions similar to these variables.



(a) (b) (c) (d)

(e) (f)

Figure 16. Distributions of some of the principal components for background QCD jets and signal top jets (top row) and pairwise distributions for some of these components as a function of the predicted signal class probability (bottom row).

5. Interpretability inspires: the particle flow interaction network (PFIN)

The performance metrics of PFN are found to be very similar to that of the MB8S network. Our studies suggest that PFN learns to loosely reconstruct some of the expressive jet features in its latent space. However, what constrains PFN's performance can be understood by examining the construction of the latent space. The PFN latent space is constructed by linearly combining the individual particle-level embeddings from the Φ network. Such linear combinations constrain the latent space's ability to learn any inter-particle interaction. The particle-level embeddings obtained from the Φ network do not take into account the ensemble of particles constituting the jets. As a result, the network learns to create per-particle embeddings by emphasizing particle-level features that are known to have moderate-to-strong expressive distributions (figure 12). Hence, we can expect a noticeable improvement in PFN's performance if the latent space can be augmented with interaction-level representations. In fact, modern architectures known to outperform the PFN model for top tagging take inter-particle interactions in some form into account.

Inspired by our observations regarding feature importance and latent space distributions for PFN as well as the trend in building modern architectures for top tagging, we propose an augmentation of PFN by including an IN [30, 47] to demonstrate how particle-level interactions allow for better-performing models.

The dataflow for the proposed PFIN model is shown in figure 17. The interactions are modeled in PFIN by constructing a fully connected undirected graph with $N_{pp} = \frac{P(P-1)}{2}$ edges where P is the maximum number of constituent particles the network is trained with. Each particle is represented with N_p features. For our purpose, we use $N_p = 3$ with (p_t, η, ϕ) for each particle with the same preprocessing used for PFN. Each edge is initially represented with $2N_P$ features by concatenating the individual particle-level features. This node-to-edge level feature construction is facilitated by a couple of interaction matrices of size $P \times N_{pp}$ called R_R and R_S. For $P = 4$, these matrices are constructed in the following manner:

$$\left(\frac{R_R}{R_S}\right) = \begin{array}{c} \\ P0 \\ P1 \\ P2 \\ P3 \\ P0 \\ P1 \\ P2 \\ P3 \end{array} \begin{array}{cccccc} (0,1) & (0,2) & (0,3) & (1,2) & (1,3) & (2,3) \\ \end{array} \left(\begin{array}{cccccc} 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 \end{array}\right) \tag{9}$$

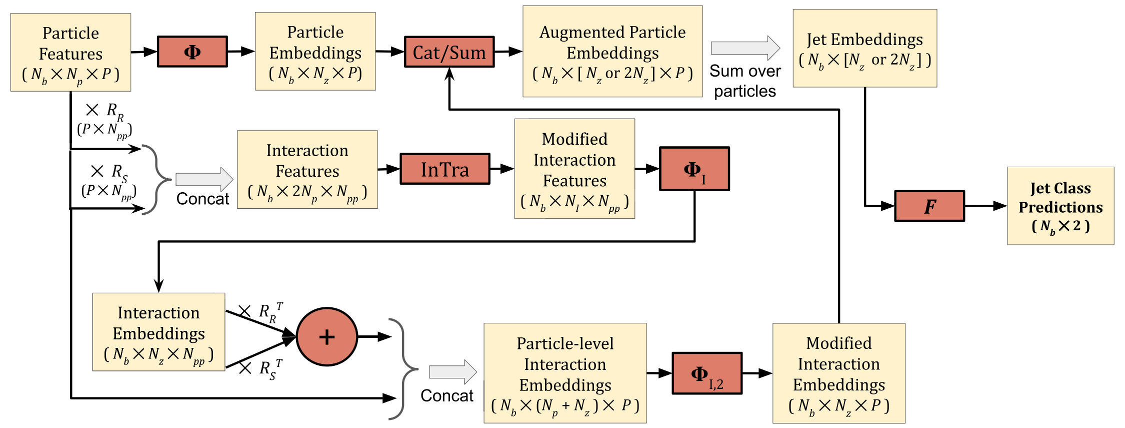

Particle Features (N_b × N_p × P)
Φ
Particle Embeddings (N_b × N_z × P)
Cat/Sum
Augmented Particle Embeddings (N_b × [N_z or 2N_z] × P)
Sum over particles
Jet Embeddings (N_b × [N_z or 2N_z])
× R_R (P × N_pp)
× R_S (P × N_pp)
Concat
Interaction Features (N_b × 2N_p × N_pp)
InTra
Modified Interaction Features (N_b × N_I × N_pp)
Φ_I
F
Jet Class Predictions (N_b × 2)
Interaction Embeddings (N_b × N_z × N_pp)
× R_R^T
× R_S^T
+
Concat
Particle-level Interaction Embeddings (N_b × (N_p + N_z) × P)
Φ_I,2
Modified Interaction Embeddings (N_b × N_z × P)

Figure 17. Model architecture and data flow for the PFIN model. N_b represents the batch size. The `InTra` block computes the pairwise particle interaction features given in equation (10). The `Cat/Sum` block creates the augmented particle embeddings by either concatenating or summing the outputs of $\Phi_{I,2}$ with Φ. $\Phi, \Phi_I, \Phi_{I,2}, F$ networks are implemented as fully connected MLPs with ReLU activation.

where we have labeled the rows with the particle ID and each column label (i,j) represents which particles are connected by this edge. The edge-level features are transformed by the interaction transformation (`InTra`) block to calculate a $N_I = 4$ dimensional representation for each edge by calculating the physics-inspired quantities [27, 33]: $\ln\Delta, \ln k_T, \ln z, \ln m^2$ where

$$\begin{aligned}
\Delta &= \sqrt{(\eta_1-\eta_2)^2+(\phi_1-\phi_2)^2}\\
k_T &= \min\left(p_{t,1},p_{t,2}\right)\Delta\\
z &= \frac{\min\left(p_{t,1},p_{t,2}\right)}{p_{t,1}+p_{t,2}}\\
m^2 &= (E_1+E_2)^2 - ||\vec{p}_1+\vec{p}_2||^2.
\end{aligned} \tag{10}$$

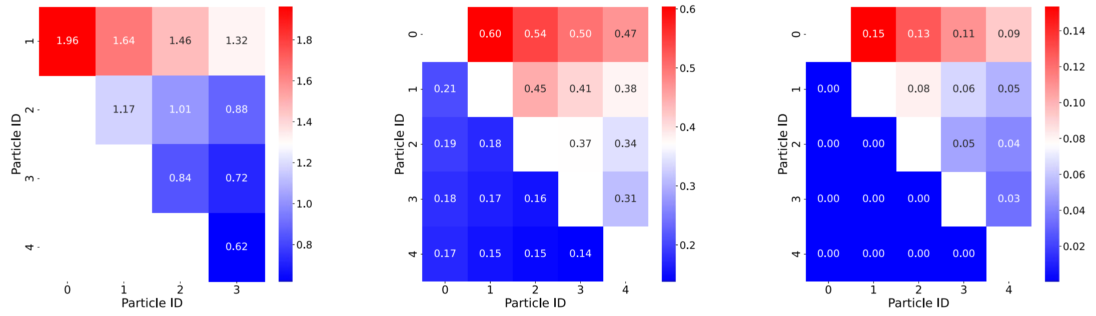
The subscripts 1 and 2 represent the two particles associated with the edge and each quantity in the aforementioned relations represents its unpreprocessed value. Given these quantities are symmetric with respect to the particles, the actual ordering of the particles does not impact PFIN's dataflow, maintaining the permutation-invariant property of PFN. These interaction features are transformed into N_z dimensional interaction embeddings by the trainable Φ_I network. These embeddings are propagated back to particle level using the interaction matrices and only those interactions are considered where both constituents are present. These particle-level interaction embeddings are concatenated with the original particle features and further transformed into N_z dimensional modified per-particle interaction embeddings via a trainable $\Phi_{I,2}$ network. These embeddings are concatenated or summed with per-particle embedding from PFN's Φ network to obtain augmented particle embeddings. These augmented features are then summed over its constituents to obtain the jet-level latent space. Finally, the F network obtains jet class probabilities for each of the jet class based on these jet-level latent space features.

The model hyperparameters and performance metrics for our implementation of PFIN are given in table 2. The choices of numbers of nodes in the hidden layers of Φ and F as well as the size of the latent space were inspired from the study of RNA scores and NAP diagrams for PFN. This allows us to keep the increase in number of trainable parameters manageably small. Both versions of PFIN show notable improvement in performance outperforming both PFN and IN models. PFIN outperforms Lorentz group equivariant neural network (LGN) [29] and its performance is comparable to those of ParticleNet and ResNeXt models while requiring a significantly smaller number of parameters, providing much faster training and model convergence.

PFIN allows us to explore the impact of pairwise particle interaction on jet classification. We calculate the ΔAUC and MAD relevance score for each pair of particles by masking the corresponding input to the Φ_I network and calculating the deviation in model prediction with respect to the baseline model's result. We additionally calculate the deviation in the model's jet class prediction probabilities. The results for the PFIN network with summation used for augmented particle embeddings are shown in figure 18. The pairwise particle interactions play a particularly important role in identifying the signal jets. The mean deviations in the background jet class probabilities are barely impacted by masking interaction features. However, for the signal jets, this impact is found to be rather large, the mean prediction probability is reduced by almost 15%

Table 2. Model hyperparameters and performance metrics for the PFIN model. (s) and (c) respectively represent architectures where per-particle interaction embeddings from $\Phi_{I,2}$ are summed and concatenated with the particle embeddings from Φ. The background rejection rate $1/\epsilon_B$ is evaluated at a signal efficiency of 30%.

Model hyperparameters	
Number of constituents, P	60
Nodes in Φ network	$(100, 100, 64)$
Nodes in Φ_I network	$(128, 128, 64)$
Nodes in $\Phi_{I,2}$ network	$(128, 128, 64)$
Nodes in F network	$(64, 100, 100)$
Latent space dimension	64 (s), 128 (c)
Number of parameters	97k (s), 101k (c)
Performance metrics	
ROC-AUC	0.9839 (s), 0.9838 (c)
Accuracy	0.937 (s), 0.937 (c)
Background rejection rate ($1/\epsilon_B$)	1041 (s), 1030 (c)



(a) (b) (c)

Figure 18. (a) ΔAUC, (b) normalized MAD relevance score, and (c) mean deviation in jet class prediction probabilities for the PFIN model (with summation being used for particle level feature augmentation) for pairwise masking of particle interaction features for the five most energetic constituents. Particle constituents are arranged in decreasing order of their energies. For (b) and (c), the upper and lower diagonal entries represent the corresponding scores for the signal and background jet classes respectively.

when the interaction between the two most energetic jets is masked. This clearly outlines the importance of particle interactions in detecting signal jets and consequently, explains the improvement observed in model performance.

Similar to what we observed for the PFN model, the jet class information is found to be embodied in the distribution of correlations among the latent space features. Hence, we further investigated PFIN's latent space by performing PCA and one crucial observation is a stronger correlation between jet mass and the top principal component of the PFIN latent space. These correlation coefficients were found to be close to 90% for both jet classes for both variants of the PFIN model. The other principal components also showed moderately improved correlations with the subjettiness variables and the number of jet constituents. As a result, the latent jet features allow construction of notably more expressive distributions, contributing to the observed improvement in jet tagging performance.

6. Conclusion

This paper presents a comprehensive study of the interpretability of DNN based top tagger models. Our work has unveiled a number of important aspects regarding how these models connect with the corresponding datasets. We have observed intriguing inconsistencies in feature ranking from different ranking methods, especially how the LRP method can produce a relevance distribution that can lead to a misleading interpretation of feature importance. Modifying LRP to obtain *differential* relevance scores has been found to be more consistent with other approaches in XAI. Furthermore, explainability metrics need to be carefully studied and understood for models trained with highly correlated input features since, as we show with the MB8S model, interpretability of AI models can be obscured in such cases. Our investigation suggests that models learn to embed jet class information in correlations among latent space embeddings and can learn to mimic distributions that closely resemble physical jet features. On the other hand, RNA scores and NAP diagrams can lead to an effective understanding of how information is propagated through different layers of a network and can lead to efficient model reoptimization strategies. Using NAP diagrams to reduce network complexity can be especially beneficial when multiple variants of the same network are required to be trained, e.g. to quantify uncertainties on event classification scores from systematic variations or adapt an architecture to learn jet classification in different phase spaces. RNA scores and NAP diagrams also open the possibility of incorporating these methods to obtain *in-situ* model optimization during training.

Observations regarding feature importance and model sparsity can also lead to better model building, as we demonstrate with the PFIN model which learns to take advantage of individual particle-level feature embeddings as well as pairwise particle interactions to noticeably improve over the PFN model. PFIN's performance is better than or comparable to a number of other novel models while its implementation needs a smaller number of parameters, ensuring faster training and quicker convergence. Studying the impact of pairwise particle interactions on PFIN using ΔAUC and MAD relevance scores reveals how these interactions can play an important role to identify top jets.

This work establishes a methodological paradigm to demonstrate the usage of XAI tools for day-to-day applications of DNNs and MLPs in HEP. Many modern HEP analyses heavily rely on DNNs not only for jet tagging and object reconstruction, but also for event classification and identification of signal events in search for physics beyond the standard model. Our work lays the foundation of exploring quantitative and robust explanations for such models. Compared to many of the existing tools to find feature importance, calculating ΔAUC and MAD relevance scores is much faster and can produce equally reliable explanations for the model performance. While these methods may require further sophistication to be applicable with other data structures, we expect them to provide satisfactory performance for most classification problems relying on tabular data. Building on the results and observations presented in this work, our future work will take a deeper look into adapting these novel tools and methods for more general data types and interpreting novel architectures like graph nets and transformers in the context of top tagging and more general jet classification scenarios.

Data availability statement

The data that support the findings of this study are openly available at the following URL/DOI: https://github.com/FAIR4HEP/xAI4toptagger/.

Acknowledgments

We would like to thank Volodymyr Kindratenko and the Center for Artificial Intelligence Innovation at the NCSA for support through our affiliation. We would like to thank Javier Duarte for useful discussions, comments, and suggestions. We are grateful to Gregor Kasieczka and Huilin Qu for pointing us to important resources and code repositories. Comments and suggestions from Sung Hak Lim, Ranit Das, Gregor Kasieczka, and David Shih helped us identify and rectify some inconsistencies in the previous version of our code. This research is part of the Delta research computing project, which is supported by the National Science Foundation (Award OCI 2005572), and the State of Illinois. Delta is a joint effort of the University of Illinois at Urbana-Champaign and its National Center for Supercomputing Applications. This work utilizes resources supported by the National Science Foundation's Major Research Instrumentation program, Grant 1725729, as well as the University of Illinois at Urbana-Champaign. This work was supported by the FAIR Data program of the U.S. Department of Energy, Office of Science, Advanced Scientific Computing Research, under Contract Number DE-SC0021258 and by the U.S. Department of Energy, Office of Science, High Energy Physics, under Contract Number DE-SC0023365.

Appendix A. Performance of baseline and variant models

Table A1. Performance and sparsity of the baseline and model variants for different model architectures. The background rejection rate $1/\epsilon_B$ is evaluated at a signal efficiency of 30%.

Architecture	Description	Params	AUC	Acc	$1/\epsilon_B$	Sparsity
TopoDNN	*Baseline*	59k	0.971	0.916	278	0.705
	Trained without $p_{T,0}$	59k	0.970	0.914	267	0.852
	Hidden layers: $(240, 80, 10)$	42k	0.972	0.916	309	0.818
	Hidden layers: $(120, 40, 6)$	16k	0.972	0.916	305	0.584
MB8S	*Baseline*	57k	0.980	0.928	796	0.426
	Trained without jet p_T	57k	0.980	0.928	775	0.444
	Trained with $\{\tau_x^{(1)}\}\bigcup\{p_{T,J}, m_J\}$	55k	0.976	0.921	516	0.404
	Hidden layers: $(200, 200, 50)$	55k	0.980	0.928	816	0.416
	Hidden layers: $(200, 200)$	45k	0.980	0.928	775	0.452
PFN	*Baseline*	82k	0.980	0.928	699	0.811 (Φ), 0.530 (F)
	$\Phi\ : (100, 100, 64), F : (64, 100, 100)$	38k	0.978	0.925	653	0.617 (Φ), 0.439 (F)
	$\Phi\ : (100, 64, 32), F : (64, 100, 100)$	28k	0.978	0.924	603	0.576 (Φ), 0.436 (F)
PFIN	*Baseline (concatenation)*	101k	0.984	0.937	1030	0.178 (Φ), 0.584 (Φ_I) 0.675 $(\Phi_{I,2})$, 0.705 (F)
	Baseline (summation)	97k	0.984	0.937	1041	0.208 (Φ), 0.625 (Φ_I) 0.70 $(\Phi_{I,2})$, 0.712 (F)

ORCID iDs

Ayush Khot https://orcid.org/0000-0002-1077-016X
Mark S Neubauer https://orcid.org/0000-0001-8434-9274
Avik Roy https://orcid.org/0000-0002-0116-1012

References

[1] Miller T 2019 Explanation in artificial intelligence: insights from the social sciences *Artif. Intell.* **267** 1

[2] Gunning D, Stefik M, Choi J, Miller T, Stumpf S and Yang G-Z 2019 XAI—explainable artificial intelligence *Sci. Robot.* **4** eaay7120

[3] Linardatos P, Papastefanopoulos V and Kotsiantis S 2020 Explainable AI: a review of machine learning interpretability methods *Entropy* **23** 18

[4] Vilone G and Longo L 2020 Explainable artificial intelligence: a systematic review (arXiv:2006.00093)

[5] Sahakyan M, Aung Z and Rahwan T 2021 Explainable artificial intelligence for tabular data: a survey *IEEE Access* **9** 135392

[6] Yuan H, Yu H, Gui S and Ji S 2022 Explainability in graph neural networks: a taxonomic survey *IEEE Trans. Pattern Anal. Mach. Intell.* 45 1–19

[7] Turvill D, Barnby L, Yuan B and Zahir A 2020 A survey of interpretability of machine learning in accelerator-based high energy physics *2020 IEEE/ACM Int. Conf. on Big Data Computing, Applications and Technologies (BDCAT)* pp 77–86 (https://doi.org/10.1109/BDCAT50828.2020.00025)

[8] Lai Y S, Neill D, Płoskoń M and Ringer F 2022 Explainable machine learning of the underlying physics of high-energy particle collisions *Phys. Lett.* B **829** 137055

[9] Mokhtar F, Kansal R, Diaz D, Duarte J, Pata J, Pierini M and Vlimant J-R 2021 Explaining machine-learned particle-flow reconstruction *Proc. 35th Conf. on Neural Information Processing Systems* (arXiv:2111.12840)

[10] Kaplan D E, Rehermann K, Schwartz M D and Tweedie B 2008 Top tagging: a method for identifying boosted hadronically decaying top quarks *Phys. Rev. Lett.* **101** 142001

[11] Almeida L G, Lee S J, Perez G, Sung I and Virzi J 2009 Top quark jets at the LHC *Phys. Rev.* D **79** 074012

[12] Almeida L G, Lee S J, Perez G, Sterman G and Sung I 2010 Template overlap method for massive jets *Phys. Rev.* D **82** 054034

[13] Plehn T and Spannowsky M 2012 Top tagging *J. Phys. G: Nucl. Part. Phys.* **39** 083001

[14] Aad G (The ATLAS Collaboration) 2016 Identification of high transverse momentum top quarks in pp collisions at $\sqrt{s} = 8$ TeV with the ATLAS detector *J. High Energy Phys.* JHEP06(2016)093

[15] The CMS Collaboration 2009 A Cambridge-Aachen (C-A) based jet algorithm for boosted top-jet tagging *Technical Report* CMS-PAS-JME-09-001 (CERN) (available at: http://cds.cern.ch/record/1194489)

[16] The CMS Collaboration 2014 Boosted top jet tagging at CMS *Technical Report* CMS-PAS-JME-13-007 (CERN) (available at: http://cds.cern.ch/record/1647419)

[17] Aaboud M *et al* (The ATLAS Collaboration) 2019 Performance of top-quark and w-Boson tagging with ATLAS in Run 2 of the LHC *Eur. Phys. J.* C **79** 1

[18] Sirunyan A M *et al* (CMS Collaboration) 2020 Identification of heavy, energetic, hadronically decaying particles using machine-learning techniques *J. Instrum.* **15** 06005

[19] Pearkes J, Fedorko W, Lister A and Gay C 2017 Jet constituents for deep neural network based top quark tagging (arXiv:1704.02124)

[20] Moore L, Nordström K, Varma S and Fairbairn M 2019 Reports of my demise are greatly exaggerated: N-subjettiness taggers take on jet images *SciPost Phys.* **7** 036

[21] Datta K and Larkoski A 2017 How much information is in a jet? *J. High Energy Phys.* JHEP06(2017)073

[22] Louppe G, Cho K, Becot C and Cranmer K 2019 QCD-aware recursive neural networks for jet physics *J. High Energy Phys.* JHEP01(2019)057

[23] Butter A, Kasieczka G, Plehn T and Russell M 2018 Deep-learned top tagging with a Lorentz layer *SciPost Phys.* **5** 028

[24] Komiske P T, Metodiev E M and Thaler J 2019 Energy flow networks: deep sets for particle jets *J. High Energy Phys.* JHEP01(2019)121

[25] Qu H and Gouskos L 2020 Jet tagging via particle clouds *Phys. Rev.* D **101** 056019

[26] Macaluso S and Shih D 2018 Pulling out all the tops with computer vision and deep learning *J. High Energy Phys.* JHEP10(2018)121

[27] Erdmann M, Geiser E, Rath Y and Rieger M 2019 Lorentz boost networks: autonomous physics-inspired feature engineering *J. Instrum.* **14** P06006

[28] Egan S, Fedorko W, Lister A, Pearkes J and Gay C 2017 Long short-term memory (LSTM) networks with jet constituents for boosted top tagging at the LHC (arXiv:1711.09059)

[29] Bogatskiy A, Anderson B, Offermann J, Roussi M, Miller D and Kondor R 2020 Lorentz group equivariant neural network for particle physics *Int. Conf. on Machine Learning* (PMLR) pp 992–1002

[30] Moreno E A, Cerri O, Duarte J M, Newman H B, Nguyen T Q, Periwal A, Pierini M, Serikova A, Spiropulu M and Vlimant J-R 2020 JEDI-net: a jet identification algorithm based on interaction networks *Eur. Phys. J.* C **80** 1

[31] Gong S, Meng Q, Zhang J, Qu H, Li C, Qian S, Du W, Ma Z-M and Liu T-Y 2022 An efficient Lorentz equivariant graph neural network for jet tagging J. High Energy Phys. 2022 30 (arXiv:2201.08187)

[32] Bogatskiy A, Hoffman T, Miller D W and Offermann J T 2022 PELICAN: permutation equivariant and Lorentz invariant or covariant aggregator network for particle physics *Proc. 36th conf. on Neural Information Processing Systems* (arXiv:2211.00454)

[33] Qu H, Li C and Qian S 2022 Particle transformer for jet tagging *Proc. 39th Int. Conf. on Machine Learning* vol 162 pp 18281–92 (arXiv:2202.03772)

[34] Kasieczka G *et al* 2019 The machine learning landscape of top taggers *SciPost Phys.* **7** 14

[35] Hornik K, Stinchcombe M and White H 1989 Multilayer feedforward networks are universal approximators *Neural Netw.* **2** 359

[36] Chakraborty A, Lim S H and Nojiri M M 2019 Interpretable deep learning for two-prong jet classification with jet spectra *J. High Energy Phys.* JHEP07(2019)135

[37] Agarwal G, Hay L, Iashvili I, Mannix B, McLean C, Morris M, Rappoccio S and Schubert U 2021 Explainable AI for ML jet taggers using expert variables and layerwise relevance propagation *J. High Energy Phys.* JHEP05(2021)208

[38] Shanahan P, Terao K and Whiteson D 2022 Snowmass 2021 Computational frontier CompF03 topical group report: machine learning (arXiv:2209.07559)

[39] Seuß D 2021 Bridging the gap between explainable AI and uncertainty quantification to enhance trustability (arXiv:2105.11828)

[40] Grojean C, Paul A, Qian Z and Strümke I 2022 Lessons on interpretable machine learning from particle physics *Nat. Rev. Phys.* **4** 1

[41] Duarte J *et al* 2018 Fast inference of deep neural networks in FPGAs for particle physics *J. Instrum.* **13** 07027

[42] Iiyama Y *et al* 2021 Distance-weighted graph neural networks on FPGAs for real-time particle reconstruction in high energy physics *Front. Big Data* **3** 44

[43] Heintz A *et al* 2020 Accelerated charged particle tracking with graph neural networks on FPGAs Third workshop on machine learning and the physical sciences (NeurIPS 2020) (arXiv:2012.01563)

[44] Roy A and Neubauer M S 2022 Interpretability of an interaction network for identifying $H \to b\bar{b}$ jets *Proc. Sci.* **ICHEP2022** 223

[45] Neubauer M S and Roy A 2022 Explainable AI for high energy physics (arXiv:2206.06632) Contribution to Snowmass 2021

[46] Mokhtar F, Kansal R and Duarte J 2022 Do graph neural networks learn traditional jet substructure? *36th Conf. on Neural Information Processing Systems* p 11

[47] Moreno E A *et al* 2020 Interaction networks for the identification of boosted $h \to b\bar{b}$ decays *Phys. Rev.* D **102** 012010

[48] Kasieczka G , Plehn T, Thompson J and Russel M 2019 Top quark tagging reference dataset *Zenodo* (https://doi.org/10.5281/zenodo.2603256)

[49] Sjöstrand T, Ask S, Christiansen J R, Corke R, Desai N, Ilten P, Mrenna S, Prestel S, Rasmussen C O and Skands P Z 2015 An introduction to PYTHIA 8.2 *Comput. Phys. Commun.* **191** 159

[50] De Favereau J, Delaere C, Demin P, Giammanco A, Lemaitre V, Mertens A and Selvaggi M 2014 DELPHES 3: a modular framework for fast simulation of a generic collider experiment *J. High Energy Phys.* JHEP02(2014)057

[51] Cacciari M, Salam G P and Soyez G 2008 The anti-k_t jet clustering algorithm *J. High Energy Phys.* JHEP04(2008)063

[52] Cacciari M, Salam G P and Soyez G 2012 FastJet user manual *Eur. Phys. J.* C **72** 1

[53] Thaler J and Van Tilburg K 2011 Identifying boosted objects with N-subjettiness *J. High Energy Phys.* JHEP03(2011)015

[54] Ellis S D and Soper D E 1993 Successive combination jet algorithm for hadron collisions *Phys. Rev.* D **48** 3160

[55] Blazeya G C *et al* 2000 Run II jet physics *QCD and Weak Boson Physics in Run II* p 47

[56] Zaheer M, Kottur S, Ravanbhakhsh S, Póczos B, Salakhutdinov R and Smola A J 2017 Deep sets *Proc. 31st Int. Conf. on Neural Information Processing Systems* pp 3394–404

[57] Wang R and Tang K 2009 Feature selection for maximizing the area under the ROC curve *2009 IEEE Int. Conf. on Data Mining Workshops* (IEEE) pp 400–5

[58] van der Waa J, Nieuwburg E, Cremers A and Neerincx M 2021 Evaluating XAI: a comparison of rule-based and example-based explanations *Artif. Intell.* **291** 103404

[59] Jesus S *et al* 2021 How can I choose an explainer? An application-grounded evaluation of post-hoc explanations *Proc. 2021 ACM Conf. on Fairness, Accountability and Transparency* pp 805–15

[60] Tang J, Alelyani S and Liu H 2014 Feature selection for classification: a review *Data Classification: Algorithms and Applications* vol 37 (CRC Press)

[61] Ribeiro M T, Singh S and Guestrin C 2016 Why should I trust you? Explaining the predictions of any classifier *Proc. 22nd ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining* pp 1135–44

[62] Chen X-W and Wasikowski M 2008 Fast: a ROC-based feature selection metric for small samples and imbalanced data classification problems *Proc. 14th ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining* pp 124–32

[63] Serrano A J, Soria E, Martin J D, Magdalena R and Gomez J 2010 Feature selection using ROC curves on classification problems *The 2010 Int. Joint Conf. on Neural Networks (IJCNN)* (IEEE) pp 1–6

[64] Lundberg S M and Lee S-I 2017 A unified approach to interpreting model predictions *Advances in Neural Information Processing Systems (NeurIPS)* vol 30

[65] Ribeiro M T, Singh S and Guestrin C 2016 Model-agnostic interpretability of machine learning (arXiv:1606.05386)

[66] Binder A, Bach S, Montavon G, Müller K-R and Samek W 2016 Layer-wise relevance propagation for deep neural network architectures *Information Science and Applications (ICISA) 2016* (Springer) pp 913–22

[67] Montavon G, Binder A, Lapuschkin S, Samek W and Müller K-R 2019 Layer-wise relevance propagation: an overview *Explainable AI: Interpreting, Explaining and Visualizing Deep Learning* vol 193 (Springer)

[68] Bach S, Binder A, Montavon G, Klauschen F, Müller K-R and Samek W 2015 On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation *PLoS One* **10** e0130140

[69] Schnake T *et al* 2021 Higher-order explanations of graph neural networks via relevant walks *IEEE Trans. Pattern Anal. Mach. Intell.* **44** 1

[70] Toloşi L and Lengauer T 2011 Classification with correlated features: unreliability of feature ranking and solutions *Bioinformatics* **27** 1986

[71] Montavon G, Lapuschkin S, Binder A, Samek W and Müller K-R 2017 Explaining nonlinear classification decisions with deep Taylor decomposition *Pattern Recognit.* **65** 211

[72] Ayinde B O, Inanc T and Zurada J M 2019 Regularizing deep neural networks by enhancing diversity in feature extraction *IEEE Trans. Neural Netw. Learn. Syst.* **30** 2650

[73] Cogswell M, Ahmed F, Girshick R, Zitnick L and Batra D 2015 Reducing overfitting in deep networks by decorrelating representations (arXiv:1511.06068)

[74] Kaur H, Nori H, Jenkins S, Caruana R, Wallach H and Wortman Vaughan J 2020 Interpreting interpretability: understanding data scientists' use of interpretability tools for machine learning *Proc. 2020 CHI Conf. on Human Factors in Computing Systems* pp 1–14

[75] Srivastava N, Hinton G, Krizhevsky A, Sutskever I and Salakhutdinov R 2014 Dropout: a simple way to prevent neural networks from overfitting *J. Mach. Learn. Res.* **15** 1929

[76] Kingma D P and Welling M 2013 Auto-encoding variational Bayes (arXiv:1312.6114)

[77] Burgess C P, Higgins I, Pal A, Matthey L, Watters N, Desjardins G and Lerchner A 2018 Understanding disentangling in β-VAE *2017 NIPS workshop on learning disentangled representations* (arXiv:1804.03599)

[78] Hadjeres G, Nielsen F and Pachet F 2017 GLSR-VAE: geodesic latent space regularization for variational autoencoder architectures *2017 IEEE Symp. Series on Computational Intelligence (SSCI)* (IEEE) pp 1–7

[79] Bajaj C, Roy A and Zhang H 2021 Invariance-based multi-clustering of latent space embeddings for equivariant learning (arXiv:2107.11717)

[80] Zhao Q, Adeli E, Honnorat N, Leng T and Pohl K M 2019 Variational autoencoder for regression: application to brain aging analysis *Medical Image Computing and Computer Assisted Intervention–MICCAI 2019: 22nd Int. Conf. (Shenzhen, China, 13–17 October 2019), Proc., Part II 22* (Springer) pp 823–31

[81] Bortolato B, Smolkovič A, Dillon B M and Kamenik J F 2022 Bump hunting in latent space *Phys. Rev.* D **105** 115009

[82] Liu Z, Luo P, Wang X and Tang X 2015 Deep learning face attributes in the wild *Proc. 2015 IEEE Int. Conf. on Computer Vision (ICCV)* pp 3730–8 (arXiv:1411.7766)

[83] Jolliffe I T and Cadima J 2016 Principal component analysis: a review and recent developments *Phil. Trans. R. Soc.* A **374** 20150202