Automating the construction of jet observables with machine learning

Kaustuv Datta,[1,*] Andrew Larkoski,[2,†] and Benjamin Nachman[3,‡]
[1]*Institute for Particle Physics and Astrophysics, ETH Zürich, 8093 Zürich, Switzerland*
[2]*Physics Department, Reed College, Portland, Oregon 97202, USA*
[3]*Physics Division, Lawrence Berkeley National Laboratory, Berkeley, California 94720, USA*

(Received 6 March 2019; published 15 November 2019)

Machine-learning assisted jet substructure tagging techniques have the potential to significantly improve searches for new particles and Standard Model measurements in hadronic final states. Techniques with simple analytic forms are particularly useful for establishing robustness and gaining physical insight. We introduce procedures to automate the construction of a large class of observables that are chosen to completely specify M-body phase space. The procedures are validated on the task of distinguishing $H \to b\bar{b}$ from $g \to b\bar{b}$, where $M = 3$ and previous brute-force approaches to construct an optimal product observable for the M-body phase space have established the baseline performance. We then use the new methods to design tailored observables for the boosted Z' search, where $M = 4$ and brute-force methods are intractable. The new classifiers outperform standard two-prong tagging observables, illustrating the power of the new optimization method for improving searches and measurement at the LHC and beyond.

DOI: 10.1103/PhysRevD.100.095016

I. INTRODUCTION

Effective identification of hadronic decays of boosted heavy particles like the top quark or W, Z and Higgs (H) bosons is essential for analyses at the Large Hadron Collider (LHC). Jet substructure observables that identify specific discriminating information in the radiation pattern of jets originating from different particles are now necessary, both in the search for new physics and precision Standard Model (SM) measurements. As a result, there is an extensive literature developing observables and techniques for identifying boosted topologies to increase the efficacy of LHC analyses probing extreme regions of phase space [1,2].

Modern machine-learning (ML) methods have emerged as useful tools for automating the creation of optimal observables for classification. These methods are particularly powerful for high-dimensional, low-level inputs such as fixed-length sets of four-vectors [3], variable-length sets of four-vectors [4], physics-inspired bases [5–9], images [10–20], sequences [18,21–23], trees [24,25], and graphs [26]. Some deep-learning-based tagging schemes have already been demonstrated using collider data as well as with full detector simulations for top quark tagging [27,28], boson tagging [27,29], quark/gluon tagging [30,31], and b-jet tagging [32–35]. In addition to improving classification performance, ML techniques may also be able to make jet tagging more independent from simulation and robust to differences between simulation and data as well as between sideband and signal regions [36–43]. These and related techniques have also been proposed as more model-agnostic approaches to new particle searches [44–48].

One of the key challenges with ML taggers is to identify what information the machine is using for classification. Understanding the origin of discrimination can lead to robustness when taggers are applied outside of the region they were trained, can result in new theoretical insight for other applications, and may produce new simple observables that capture most of the information. While there are many proposals for ML metacognition [4,5,7,8,12,17,40], one particularly powerful approach is to identify simple product observables that capture most of the information from a ML algorithm trained on the full phase space [8]. This approach results in analytically tractable observables that can capture nearly all of the power of a more complicated algorithm, but are also very robust and insightful. One of the most challenging aspects of the approach presented in Ref. [8] is the fitting process for picking the optimal simple product observable.

In this paper, we describe two machine-learning procedures for automating the feature extraction originally

---

[*] kdatta@ethz.ch
[†] larkoski@reed.edu
[‡] bpnachman@lbl.gov

presented in Ref. [8]. One method attempts to learn a parametrized generative model for estimating the probability densities for a product observable. A second, simpler method uses linear regression on the logarithm of the product observable. These methods are applied to derive an optimal product observable for discriminating $H \to b\bar{b}$ vs $g \to b\bar{b}$ and the outcome is compared to the result of Ref. [8] which used a brute-force approach. Having validated the methods, a new classifier is developed to distinguish a Z' from generic quark and gluon jets. The phase space scan required in this later tagging task is too big for the brute-force approach and therefore the automated methods are required to find the optimal tagger. The resulting classifier has a simple form and is competitive with a tagger using high-dimensional, low-level inputs. In addition to Ref. [42], this is the only other study of the dependence on the mass of the new boson, which is timely given new searches for light boosted bosons [49–51].

This paper is organized as follows. The method for constructing product observables is described in Sec. II and the machine-learning approaches are detailed in Sec. III. Results for both the Higgs and Z' classification tasks are presented in Sec. IV. The paper ends with conclusions and a future outlook in Sec. V.

II. *N*-SUBJETTINESS PRODUCT OBSERVABLES

The information about the kinematic phase space of M-subjets in a jet is resolved with a set of $(3M-4)$ N-subjettiness [52–54] observables. By increasing M, one can identify the number of subjets required to saturate the classification performance based on the spanning set of N-subjettiness observables [7]:

$$\{\tau_1^{(0.5)}, \tau_1^{(1)}, \tau_1^{(2)}, \dots, \tau_{M-2}^{(0.5)}, \tau_{M-2}^{(1)}, \tau_{M-2}^{(2)}, \tau_{M-1}^{(1)}, \tau_{M-1}^{(2)}\},$$

where

$$\tau_N^{(\beta)} = \frac{1}{\sum_{i\in \mathrm{jet}} p_{\mathrm{T},i} R^\beta} \sum_{i\in \mathrm{jet}} p_{\mathrm{T},i} \min_{\mathrm{axes}\ j} (\Delta R_{j,i})^\beta, \qquad (1)$$

for some choice of N axes within the jet; R is the jet radius parameter, and $(\Delta R)^2 = (\Delta\phi)^2 + (\Delta\eta)^2$. Given the minimal M, one can posit an ansatz[1] for a simple product observable that captures most of the information contained in a neural network trained on the entire spanning set:

$$\beta_M^{\mathrm{ML}} = (\tau_1^{(0.5)})^a (\tau_1^{(1)})^b (\tau_1^{(2)})^c (\tau_2^{(1)})^d \cdots. \qquad (2)$$

For distinguishing $H \to b\bar{b}$ vs $g \to b\bar{b}$ jets, the authors of Ref. [8] showed that the useful information for classification is saturated by $M = 3$ and β_3^{ML} has nearly the same tagging performance as the full 3-body phase space. The parameters a, b, c, d, and e that specify β_3^{ML} were identified by randomly scanning the five-dimensional phase space and exploiting minimal correlations between some of the parameters. This becomes intractable when the optimal M is bigger than 3.

In this paper, we explore methods to overcome the difficulties of extending this procedure to higher dimensions. In one approach, we replace the random sampling segment of the procedure with a combination of neural networks carrying out regression from the parameter space to the distributions of the product observable for individual jets. Off-the-shelf minimization routines can then be used to optimize any metric of the classifier performance. A complementary and simpler approach is to directly use the form in Eq. (2) in the machine-learning optimization, where the learnable parameters are the exponents $\{a, b, c, \dots\}$. Further details are described in the next sections.

III. MACHINE-LEARNING IMPLEMENTATION

A. Dataset

Proton-proton collisions with $Z' \to$ hadrons, $H \to b\bar{b}$, and generic quark and gluon jets (QCD) at $\sqrt{s} = 13$ TeV are generated using PYTHIA8.226 [55,56]. For the $H \to b\bar{b}$ case, the background is enriched in $g \to b\bar{b}$ as in Ref. [57] by generating the gluon splitting matrix element in MadGraph 5 v2.5.4 [57]. All detector-stable particles excluding neutrinos and muons are clustered into jets using the anti-k_t algorithm [58] with $R = 0.8$ as implemented in Fastjet [59]. Events are required to have at least one jet with $p_T >$ 500 GeV and mass $>$25 GeV, and the leading such jet is considered for further analysis. There is no explicit requirement on the jet pseudorapidity, but due to the high p_T threshold, jets are mostly concentrated at central rapidities. Jets are groomed by reclustering the constituents using the Cambridge-Aachen algorithm [60,61] and applying the soft drop algorithm [62] with $\beta = 0$ and $z_{\mathrm{cut}} = 0.1$ [equivalent to modified mass drop tagging (mMDT) [63]]. The N-subjettiness observables are computed using the axes that minimize $\tau_N^{(\beta)}$, using the exclusive k_t algorithm [64,65] with standard E-scheme recombination [66]. For comparison with other state-of-the-art two-prong tagging techniques, the D_2 [67], N_2 [68] observables, and $\tau_{21}^{(\beta)}$ with winner-take-all (WTA) recombination [69–71], are also computed from the jet constituents.

B. Construction of optimized product observables

Using the approach followed in Ref. [8], the point of saturation of discrimination power is first identified using a

---

[1] The product form may not be flexible enough to capture the full discrimination power. We find that it can capture a significant portion of the classification performance, but Appendix E indicates that further information can be useful.

deep neural network (DNN) classifier. For Z' vs QCD and $H \to b\bar{b}$ vs $g \to b\bar{b}$ discrimination, we note that discrimination power saturates at 4-body (eight-dimensional) and 3-body phase space (five-dimensional), respectively. Then it is simple to form the product observable from the elements of the M-body basis corresponding to saturation.

We examine two approaches for finding the optimal product observable. A first method estimates the probability density functions (PDFs) of the product observables for a given set of exponents $\{a, b, c, \ldots\}$. These PDFs can then be used to construct likelihood ratios and scan for the optimal classifier. The second method makes use of the observation that any monotonic transformation of a classifier has the same performance as the original classifier. The logarithm of Eq. (2) is linear and so the optimal exponents can be simply optimized using linear regression. Linear regression has a unique solution and is easy to implement. Therefore, this second method has a clear advantage over the first method. However, linear regression would not apply if a more complex function was chosen instead of a simple product for Eq. (2). There may be additional use-cases for the generative model as well. Therefore, both methods are presented on equal footing below, though for the product case studied in this paper, we advocate for the linear regression in practice.

We begin by describing the generative model approach. For each task, the product observable is calculated for 25,000 signal and background jets for different values of the parameters $[a - e]$ ($H \to b\bar{b}$) or $[a - h]$ (Z'), in the range $[-5, 5]$. These distributions are then stored to generate training sets for the neural networks used to carry out regression from the parameter space to the calculation of β_M^{ML} with those exponents.

While there are multiple possibilities for learning the probability distribution of β_M given $\{a, b, c, \ldots\}$, such as generative adversarial networks [72] and variational autoencoders [73,74], the method that we found works well for the product observables is illustrated in Fig. 1. The network takes as input five (Higgs) or eight (Z') inputs and outputs 25,000 numbers, which represent a dataset that is the same size as the training data, but with the specified parameter values $\{a, b, c, \ldots\}$. From these 25,000 values, the probability distribution of β is formed for signal and background and the one-dimensional likelihood ratio is constructed for optimizing the classifier performance. Variations on this setup are possible, such as (significantly) reducing the number of points needed to specify the probability distributions, but this approach was found to be robust to perturbations in initialization and network architecture. For this paper, it was found that the network did not work well with fewer than 25,000 example jets per parameter point. For each network, 250,000 (450,000) parameter points were used for training in the Z' and ungroomed Higgs (groomed Higgs) case. In only the groomed Higgs case, a single network was trained for signal and background with a 1/0 switch added to the input. Separate networks were trained for signal and background in the Z' and ungroomed Higgs cases. To reduce the effects of numerical instability on the training of these networks, we train on samples after taking the natural logarithm of the 25,000 measured values of the product observables.

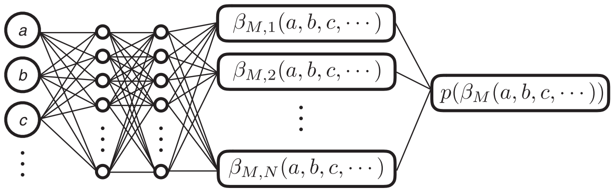

a
b
c
$\beta_{M,1}(a,b,c,\cdots)$
$\beta_{M,2}(a,b,c,\cdots)$
$\beta_{M,N}(a,b,c,\cdots)$
$p(\beta_M(a,b,c,\cdots))$

FIG. 1. A schematic diagram of the network architecture used to produce the probability distribution of β_M for a given set of input parameters $\{a, b, c, \ldots\}$. In this case, $N = 25,000$.

Aside from the use (or not) of the switch input, both the $H \to b\bar{b}$ and Z' tasks use simple fully connected neural networks with two hidden layers. The input layer is followed by a fully connected layer with either 250 or 500 nodes, then another fully connected layer with 100 or 250 nodes, followed by an output layer with 25,000 nodes using a linear activation. The number of nodes in the hidden layers were bigger for the Z' case with grooming compared with the Higgs case or the ungroomed Z' case.

We use leaky rectified linear units (leaky ReLU) as the activations for the hidden layers. The networks were compiled with a mean squared error loss function [on the penultimate layer shown in Fig. 1, not on $p(\beta_M)$ directly], using Adam optimization [75]. The regression networks were each trained for ~400 epochs. All deep learning tasks were carried out with the Keras [76] deep learning libraries, using the TensorFlow [77] backend.

Given the set of 25,000 values of the β_M observable for a given set of parameters, it is straightforward to use these networks in an optimality scan. For this purpose, we use SciPy's [78] basin-hopping [79] global minima finder using the nonlinear, derivative-free constrained optimization by linear approximation (COBYLA) [80] minimizer to scan over local minima. In the optimization, the networks are used to predict background and signal distributions for a given set of parameters. The one-dimensional binned likelihood distributions[2] of the observable, constructed from the network outputs, were then used to calculate the area under the receiver operating characteristics (ROC) curve, henceforth referred to as the AUC, to estimate the discrimination power, where (1-AUC) was explicitly chosen as the metric for the basin-hopping minimization.

---

[2] In principle, one can estimate the AUC without binning, but it was found that there was not a significant sensitivity to the choice of binning.

Appendix A illustrates that the regression networks can be used to accurately model the dependence of the AUC as a function of the parameters. The observable selected using this procedure will be denoted $\beta^{\mathrm{ML}}_{3,H\to b\bar{b}}$ in the next sections.

We also note that the space of possible inputs is degenerate since a monotonic function of an observable has the same discrimination power as the original observable. However, due to the finite binning required to calculate the AUC's from the likelihood distributions, and statistical fluctuations in a given data sample, the observables do not have precisely the same power as monotonic functions of themselves. The issue of degeneracies is not explicitly dealt with in the minimization procedure, but if the networks are adequately trained over the input space, it is sufficient to locate any one "global" minimum among local minima of similar depth, using basin-hopping or any other global minimizer.

The second approach to optimizing $\{a, b, c, \ldots\}$ directly uses Eq. (2). The product form can be used directly as a tunable function for predicting signal/background with tunable parameters $\{a, b, c, \ldots\}$. This is a more direct way of identifying the optimal solution without explicitly modeling the probability distributions. Optimizing a generic function is possible with methods like stochastic gradient decent, but the product observable is amenable to a significant simplification.[3] In particular, two classifiers that are monotonic transformations of each other result in the same classification performance. By taking the logarithm of Eq. (2), one can transform the problem into linear regression[4] where the inputs are $\log(\tau)$ and the coefficients are the exponents:

$$\log(\beta^{\mathrm{ML}}_{M}) = a\log(\tau_1^{(0.5)}) + b\log(\tau_1^{(1)}) + \cdots. \tag{3}$$

This approach uses the mean squared error loss to identify $\{a, b, c, \ldots\}$. The observable selected using this procedure will be denoted $\hat{\beta}^{\mathrm{ML}}_{3,H\to b\bar{b}}$ in the next sections.

IV. RESULTS

In this section, we present the new observables obtained for the different classification tasks for the ungroomed Z' samples (the groomed case is in Appendix C). For closure, we first demonstrate that this new procedure produces an observable for ungroomed $H\to b\bar{b}$ discrimination with the same performance as the β_3 observable proposed in Ref. [8] (the groomed case in Appendix B). Then we extend the procedure to higher M-body phase space by applying it to Z' discrimination for three values of $m_{Z'}$, and propose new observables for those classification tasks.

TABLE I. Summary of parameters for the product observables for ungroomed $H\to b\bar{b}$ discrimination as proposed in Ref. [8] and as constructed via the procedures presented in this work [Figs. 2(a) and 2(b)].

Observable	a	b	c	d	e	AUC
β_3	2.0	0.0	0.0	0.5	−1.0	0.823
$\beta^{\mathrm{ML}}_{3,H\to b\bar{b}}$	1.87	−0.02	−0.14	0.66	−0.98	0.823
$\hat{\beta}^{\mathrm{ML}}_{3,H\to b\bar{b}}$	−0.11	−0.58	0.09	−0.25	0.51	0.824

A. Ungroomed $H\to b\bar{b}$ vs $g\to b\bar{b}$ discrimination

Utilizing the result that discrimination power for ungroomed $H\to b\bar{b}$ vs $g\to b\bar{b}$ discrimination saturates at 3-body phase space, we use the procedures proposed in the previous section to find the optimal product observable. The final values for the parameters $\{a, \ldots, e\}$ obtained through the optimization are presented in Table I, along with those obtained in the previous study. Note that any set of exponents for which the observable is related by a monotonic transformation should have equivalent loss and thus are equally good from the point of view of minimization. This introduces some sensitivity to the stochastic nature of network training. Interestingly, the exponents with the ensemble method are nearly the same for a, b, d, and e, but slightly different for c. For the regression method, the exponents are nearly the same as the ensemble method up to a constant factor (approximately −2) for c, d, and e, but not for a and b. These results indicate the presence of multiple observables with comparable performance. While there is no obvious monotonic transformation between the two observables, a simple neural network trained on their combination does no better than each individually and thus they must be using the same information. This is true for the networks presented in later sections as well.

In Fig. 2(a), we plot the distributions of the new observable computed for signal and background, along with the prediction from the ensemble neural network. We note that the network provides a good match to the true distribution, where the latter is also calculated on ten times more jets. Further, in Fig. 2(b) we plot the distributions of the observable obtained via the ML regression method. We then compare the ROC curves for the new observables to $D_2^{(2)}$ [67], $N_2^{(2)}$ [68] observables, and $\tau_{21}^{(2)}$ in Fig. 2(c).

In addition, we also compare the new observables to β_3 in Fig. 2(d) to demonstrate that the three observables have essentially the same discrimination power as expected. Then, this allows us to proceed to applying the procedure on higher dimensional problems.

B. Ungroomed Z' vs QCD

We first train neural network classifiers on the M-body N-subjettiness bases, to identify the point of saturation of

---

[3] We thank Eric Metodiev for this insightful observation.

[4] Linear regression was proven to be sufficient for all IRC safe observables in Ref. [5]; however our results need not be IRC safe.

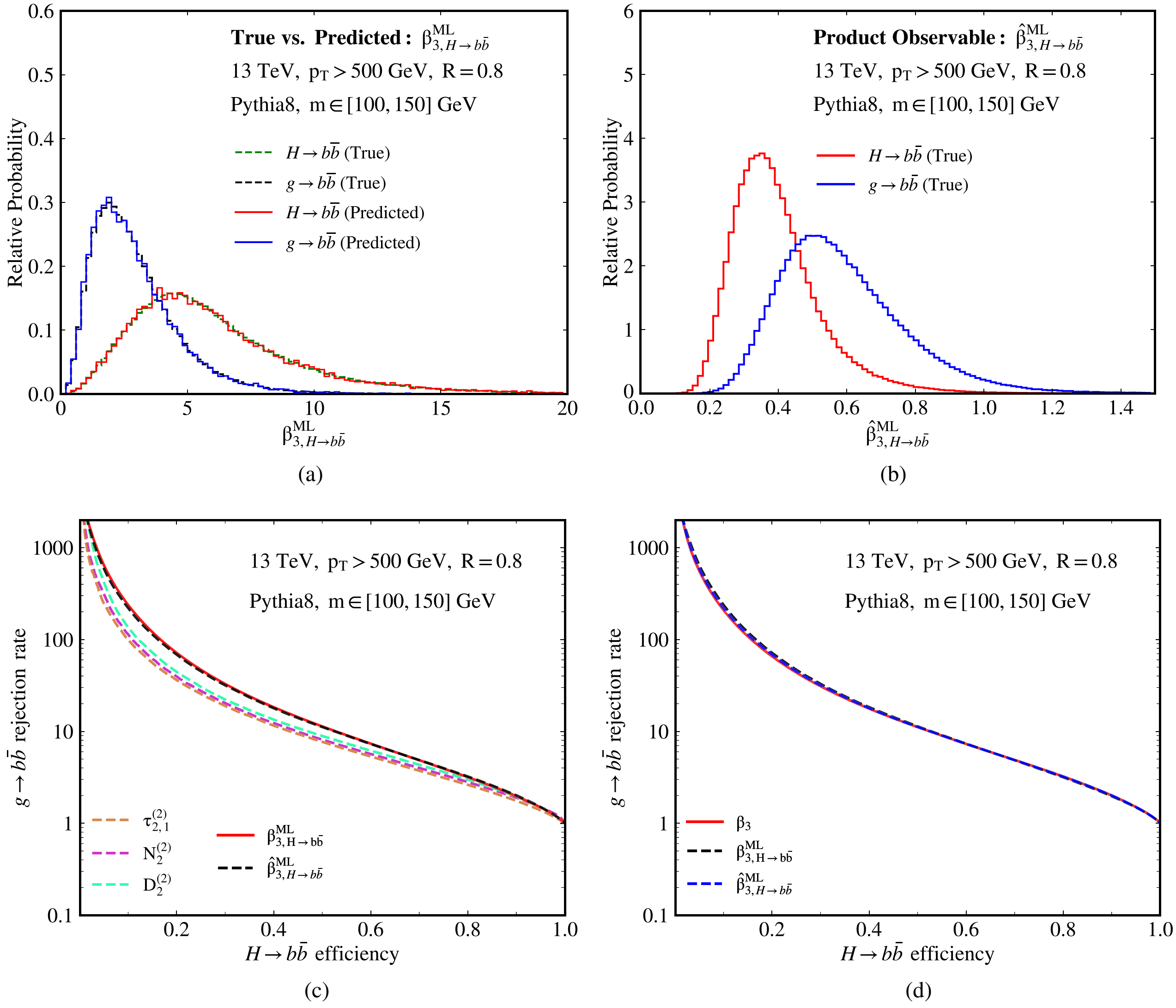
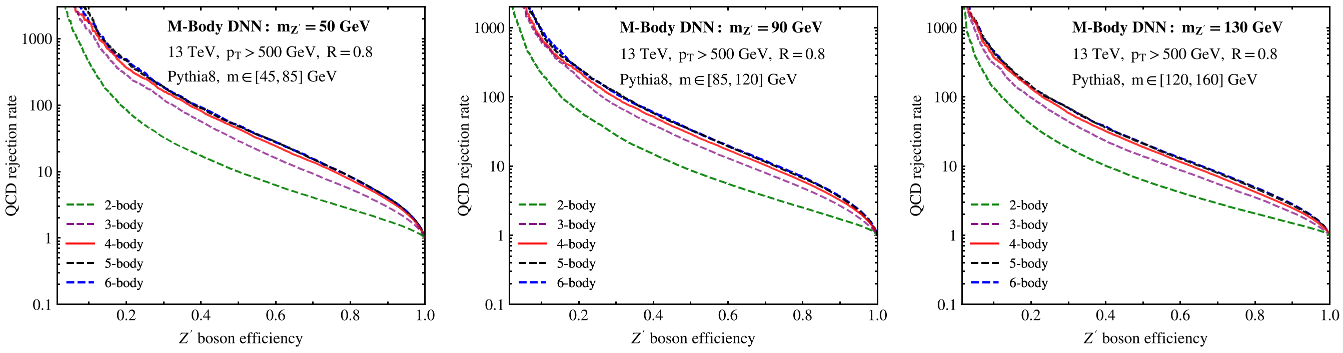
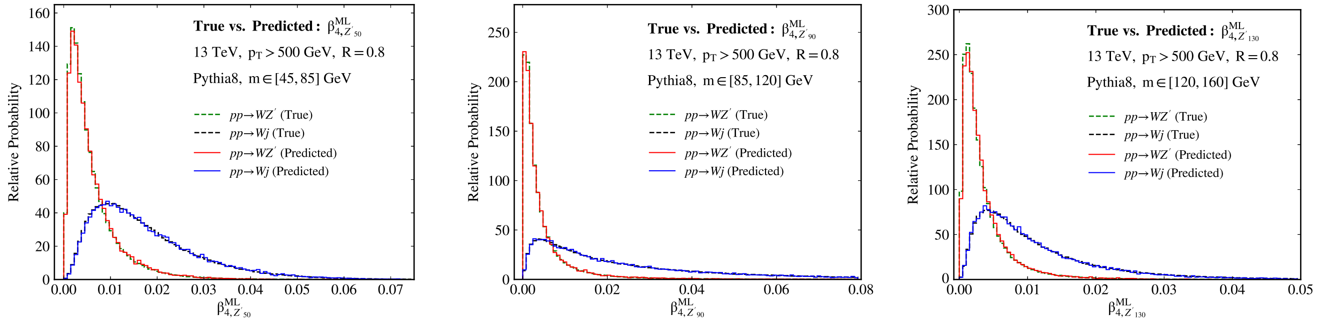


FIG. 2. (a) Comparison of the probability density function of the new $\beta^{\mathrm{ML}}_{3,H\to b\bar{b}}$ observables for ungroomed $H\to b\bar{b}$ discrimination, using ∼500, 000 signal and background samples, and the distributions of the regression DNN prediction. The distributions are rescaled by a constant for the sake of visual comparison. (b) Probability densities of $\hat{\beta}^{\mathrm{ML}}_{3,H\to b\bar{b}}$ obtained via linear regression. (c) Comparison of discrimination power of $\beta^{\mathrm{ML}}_{3,H\to b\bar{b}}$ and $\hat{\beta}^{\mathrm{ML}}_{3,H\to b\bar{b}}$ to standard observables. (d) Comparison of $\beta^{\mathrm{ML}}_{3,H\to b\bar{b}}$ and $\hat{\beta}^{\mathrm{ML}}_{3,H\to b\bar{b}}$ to β_3 proposed in Ref. [8]; we note that three observables provide essentially the same discrimination power.

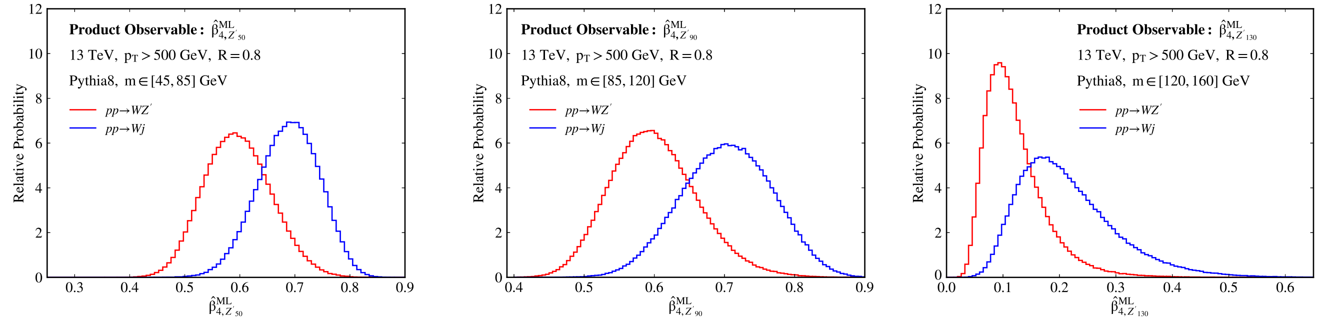


(a) $m_{Z'} = 50$ GeV (b) $m_{Z'} = 90$ GeV (c) $m_{Z'} = 130$ GeV

FIG. 3. *M*-body discrimination results for ungroomed Z' vs QCD jets. Discrimination power is effectively saturated at 4-body phase space for each case.

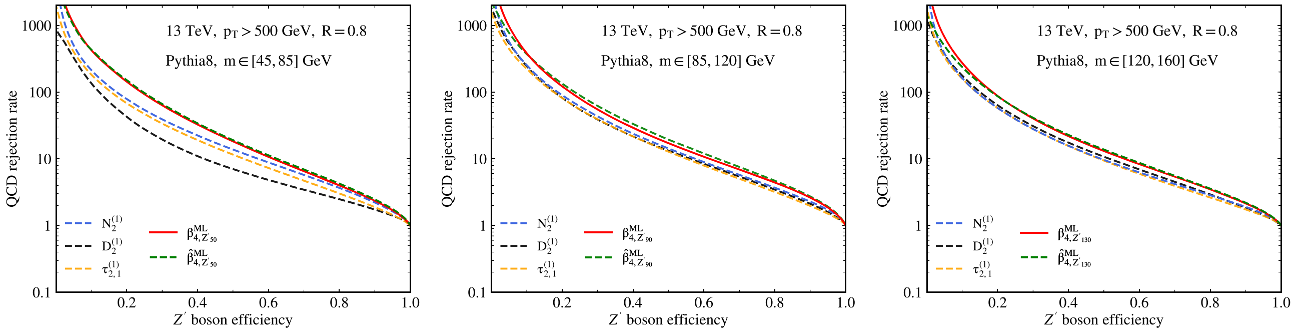


(a) $m_{Z'} = 50$ GeV (b) $m_{Z'} = 90$ GeV (c) $m_{Z'} = 130$ GeV

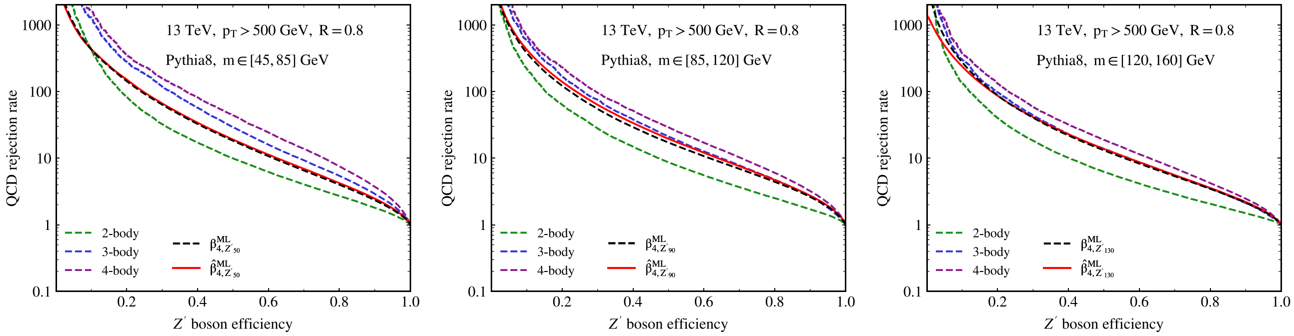


(d) $m_{Z'} = 50$ GeV (e) $m_{Z'} = 90$ GeV (f) $m_{Z'} = 130$ GeV

FIG. 4. Top panels [(a)–(c)]: Comparison of the probability density function of the new β_4^{ML} observables for ungroomed Z' discrimination, calculated for ~500, 000 signal and background samples, and the distributions of the regression DNN predictions of 25,000 observable values. The distributions are rescaled for the sake of visual comparison. Bottom panels [(d)–(f)]: Distributions of the $\hat{\beta}_4^{\mathrm{ML}}$ observables for ungroomed Z' discrimination that were obtained via linear regression.



(a) $m_{Z'} = 50$ GeV (b) $m_{Z'} = 90$ GeV (c) $m_{Z'} = 130$ GeV



(d) $m_{Z'} = 50$ GeV (e) $m_{Z'} = 90$ GeV (f) $m_{Z'} = 130$ GeV

FIG. 5. Top panels [(a)–(c)]: Comparison of discrimination power of β_4^{ML} observables to standard observables; the latter are calculated with an angular exponent of 1, for which they were observed to perform best. Bottom panels [(d)–(f)]: Comparison of β_4^{ML} to discrimination power of neural networks trained on the M-body observable bases; the observables seem to capture increasing amounts of the discrimination power of the 3- and 4-body neural networks with increasing $m_{Z'}$.

TABLE II. Summary of parameters for $\beta_4^{\rm ML}$ for ungroomed Z' vs QCD discrimination at three mass points.

$m_{Z'}$ (GeV)	a	b	c	d	e	f	g	h
50	2.72	−3.78	0.63	−2.77	1.54	0.20	2.36	−0.28
90	0.90	−2.87	0.18	−1.78	−0.72	1.79	2.48	−0.44
130	1.69	−2.98	0.75	−0.89	−0.38	0.77	1.37	0.30

TABLE III. Summary of parameters for $\hat{\beta}_4^{\rm ML}$ for ungroomed Z' vs QCD discrimination at three mass points.

$m_{Z'}$ (GeV)	a	b	c	d	e	f	g	h
50	1.06	−1.11	0.25	−0.56	0.43	−0.07	0.22	−0.01
90	1.02	−1.06	0.22	−0.27	0.15	0.00	0.18	0.02
130	−1.09	−0.43	0.25	−0.97	0.37	0.12	0.60	0.19

TABLE IV. AUC, from Fig. 5, of the standard observables and the $\beta_4^{\rm ML}$ observables, optimized for the corresponding signal, for ungroomed Z' vs QCD discrimination at three $m_{Z'}$ points. The ROC curves are calculated using the full datasets, with ~500,000 events passing the mass cut for each value of $m_{Z'}$. The peak AUC at 90 GeV is systematic and statistical; testing with additional samples at 88 and 92 GeV indicates that the AUCs are ordered as $88 > 90 > 92$.

$m_{Z'}$ (GeV)	$\hat{\beta}_4^{\rm ML}$	$\beta_4^{\rm ML}$	$\mathrm{N}_2^{(1)}$	$\mathrm{D}_2^{(1)}$	$\tau_{2,1}^{(1)}$
50	0.864	0.858	0.843	0.778	0.817
90	0.873	0.866	0.848	0.837	0.827
130	0.842	0.838	0.809	0.812	0.797

discrimination power for each value of $m_{Z'}$.[5] The results are presented in Fig. 3, showing that saturation occurs with the 4-body phase space for each case.

We then proceed to construct the $\beta_{4,Z'}^{\rm ML}$ and $\hat{\beta}_{4,Z'}^{\rm ML}$ product observables with the elements of the eight-dimensional 4-body basis, run the procedure described in Sec. III and construct the new observables optimized for Z' discrimination at three different values of $m_{Z'}$.

We present the distributions of the new observables for Z' discrimination in Fig. 4 and then compare their discrimination power to standard observables and DNNs trained on the spanning N-subjettiness bases in Fig. 5. The corresponding values of $\{a, b, c, \ldots, h\}$ and the AUCs are in Tables II, III, and IV, respectively. The comparison of the true and predicted distributions in Fig. 4 illustrates the excellent quality of the regression network. The ROC curves in Fig. 5 show that the learned $\beta^{\rm ML}$ and $\hat{\beta}^{\rm ML}$ outperform the state-of-the-art single physics-motivated observables (top row), though the product observables do not fully saturate the performance of the DNN trained on the full 4-body phase space (bottom row). This suggests that a more flexible form (other than a simple product) is required to build a simple observable to capture more of the classification information. The product values obtained from the ensemble and regression methods are not a simple scaling of each other, though the fact that both have a similar performance suggests that one is a monotonic transformation of the other.

The optimized $\beta^{\rm ML}$ and $\hat{\beta}^{\rm ML}$ observables are not identical for the different values of $m_{Z'}$ (Tables II and III), but it would be interesting to study to what extent the trends are physical or are due to the existence of multiple observables with similar performance. We leave this study to future work. However, a first indication that the observables contain similar physical information is studied in Appendix D, where the optimized product for one mass is applied to another mass. The ROC curves are similar for all three product observables when applied to the same $m_{Z'}$.

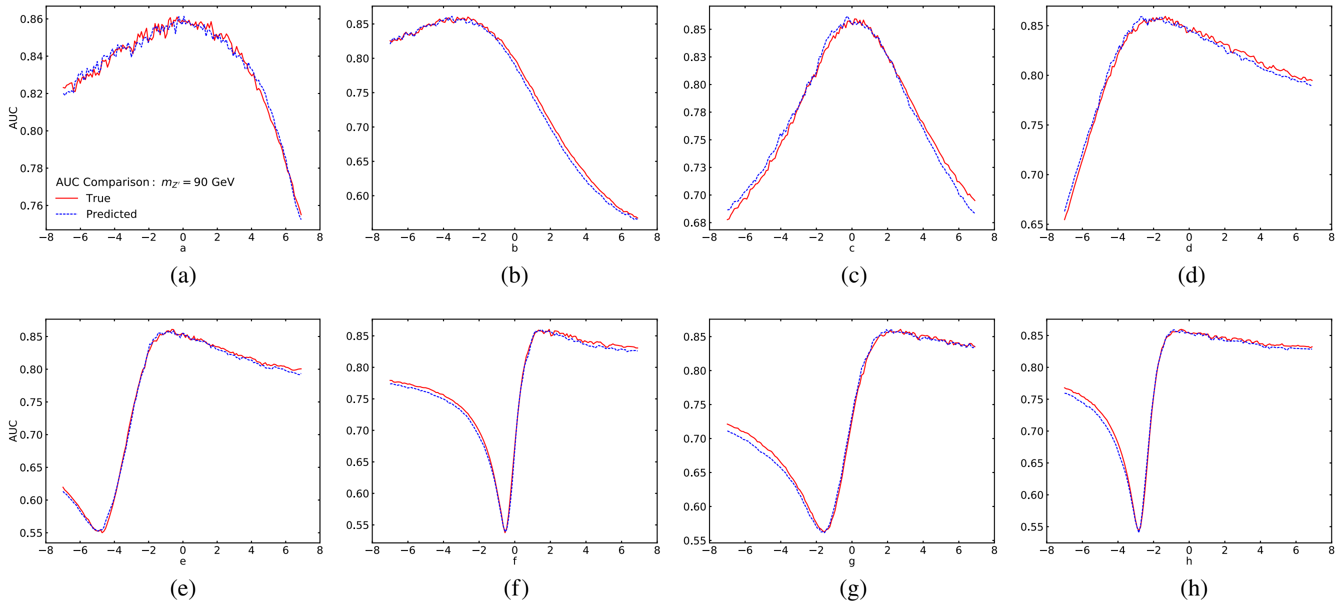
V. CONCLUSIONS

This paper has extended the growing literature of machine-learning assisted jet substructure-based tagging in two ways. First, we have developed procedures to automatically identify the optimal product observables, using the N-subjettiness features as an example. This is an important innovation because observables with relatively simple analytic forms are robust complements to complex neural network classifiers and prior to this work, there was no efficient way to identify the best coefficients in the product. Second, we have used this automated framework to identify the optimal product observables for searching for boosted resonances like the Z boson, but with beyond the Standard Model masses. Jet substructure has proven to be a powerful toolset for such searches, but until now, there have been few studies of the mass dependence of the optimal observables.

Future extensions of the methods introduced in this paper may be able to simplify the regression procedure, as well as study the connections between different classifiers with similar performance (including the ones connected by

---

[5] A single neural network architecture, consisting of seven fully connected (five hidden) layers, was utilized for all of the classification tasks. The first four dense layers consisted of 1000, 1000, 750 and 500 nodes respectively, and were assigned a dropout [81] regularization of 0.2, to prevent overfitting on training data. The next two dense layers consisted of 250 nodes with a dropout regularization 0.1, and 100 nodes without dropout. The input layer and all hidden layers utilized the ReLU activation function [82], while the output layer, consisting of a single node, used a sigmoid activation. The network was compiled with the binary cross-entropy loss minimization function, using the Adam optimization [75]. Models were trained with Keras's default EarlyStopping callback, with appropriate patience thresholds, to further negate possible overfitting.



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

(e) (f) (g) (h)

FIG. 6. Here we plot results for the calculation of the AUC from the distributions generated by the signal and background regression DNNs for the ungroomed $m_{Z'} = 90$ GeV case (blue dashed curve) along with the true AUC (red solid) computed on the same statistics. These are seen to be in good agreement with each other. The results demonstrate the usefulness of the networks to accurately reproduce the change in the signal and background PDFs, represented via the accurate reproduction of the AUCs calculated from them, as a function of the variation of each individual input parameter $\{a, \ldots, h\}$.

monotonic functions). The power of the method may also be extended by considering other parametric forms besides products. Classification problems demanding a higher M-body phase space are a natural extension of the examples presented here.

As machine-learning techniques are used more widely to guide the optimal selection of classifiers, there will be a growing need to simplify and interpret the guidance from the machines. We have prepared an automated approach to construct optimal observables with simple, analytic forms, which can be used for further theoretical and experimental studies. This technique will form the basis of multiple extensions in the future to improve classification performance and increase the robustness of searches and measurement at the LHC and beyond.

ACKNOWLEDGMENTS

This work was supported by the U.S. Department of Energy, Office of Science under Contract No. DE-AC02-05CH11231. We would like to thank Gregor Kasieczka, Patrick Komiske, Eric Metodiev, and Jesse Thaler for detailed comments on the analysis and manuscript.

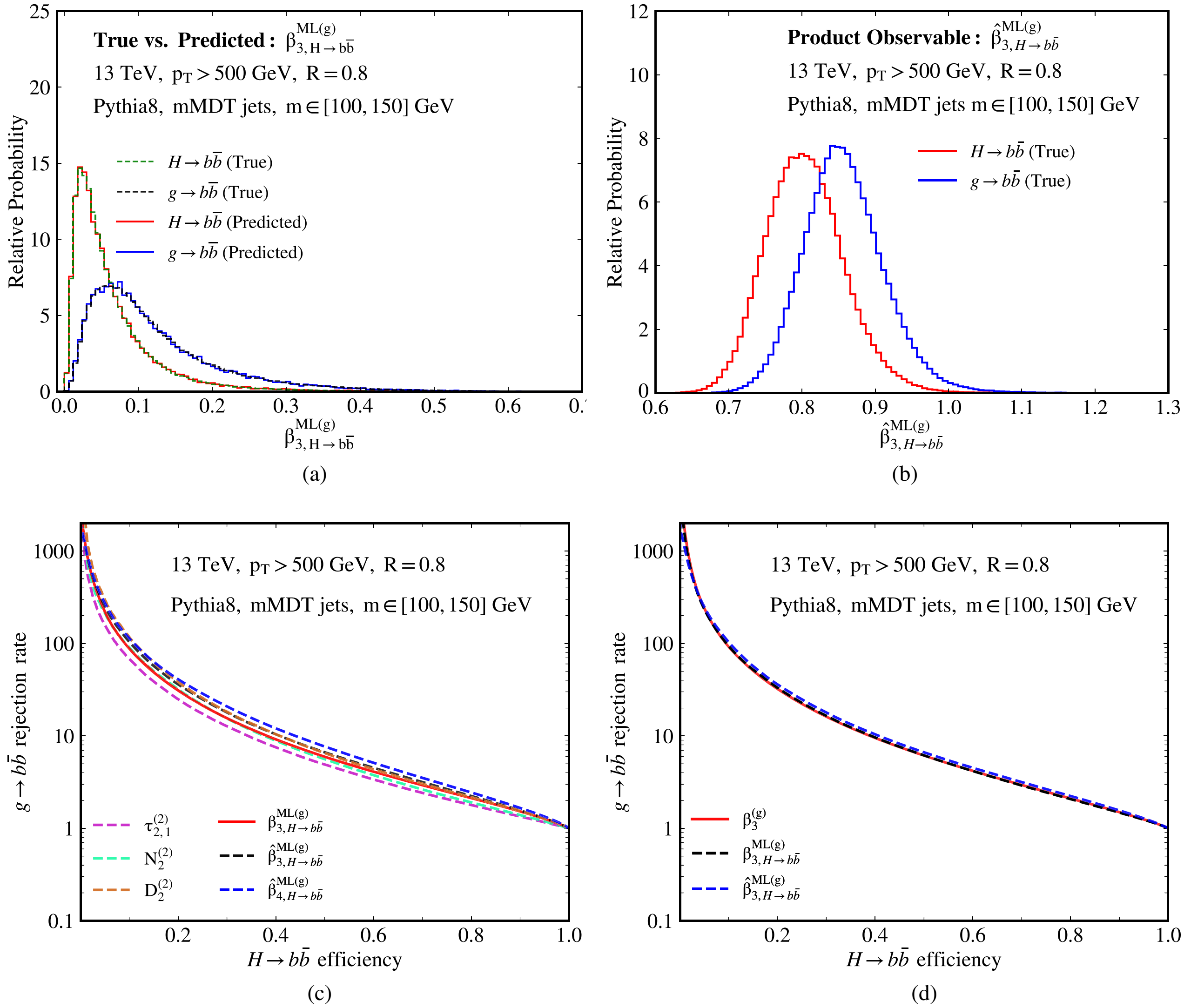
APPENDIX A: CROSS-CHECK FOR PERFORMANCE OF THE REGRESSION NETWORKS

Here we briefly demonstrate that the regression DNNs do actually learn to approximate the mapping from the input parameters of the product observables to their densities, i.e., a mapping from $\mathbb{R}^8 \to \mathbb{R}^{25,000}$. We specifically choose the ungroomed 90 GeV case, while choosing values of $\{a, \ldots, h\}$ for the optimal observable, as listed in Table II.

We then select one of the parameters and vary it between −7 and 7 with a step size of 0.1 while keeping the other parameters fixed. This allows us to study how the networks can be used to interpolate AUCs over a range of values around the optimum we locate and, in addition, by going beyond the training range of $[-5, 5]$ we also demonstrate that the networks can be used to extrapolate the aforementioned mapping to then still calculate the AUC with a good level of accuracy. The results for this study are shown in Fig. 6 and indicate that the regression networks allow us to accurately track the trajectories of the AUC in these one-dimensional slices of the parameter space.

TABLE V. Summary of parameters for the product observables for groomed $H \to b\bar{b}$ discrimination as proposed in Ref. [8] and as constructed via the procedure presented in this work [Fig. 7(a)].

Observable	a	b	c	d	e	AUC
$\beta_3^{(g)}$	−2.0	0.0	0.0	−2.0	2.0	0.745
$\beta_{3,H\to b\bar{b}}^{\mathrm{ML(g)}}$	0.67	−1.65	0.01	−1.90	2.07	0.744
$\hat{\beta}_{3,H\to b\bar{b}}^{\mathrm{ML(g)}}$	−1.54	1.01	−0.17	−0.15	0.16	0.758



(a)

(b)

(c)

(d)

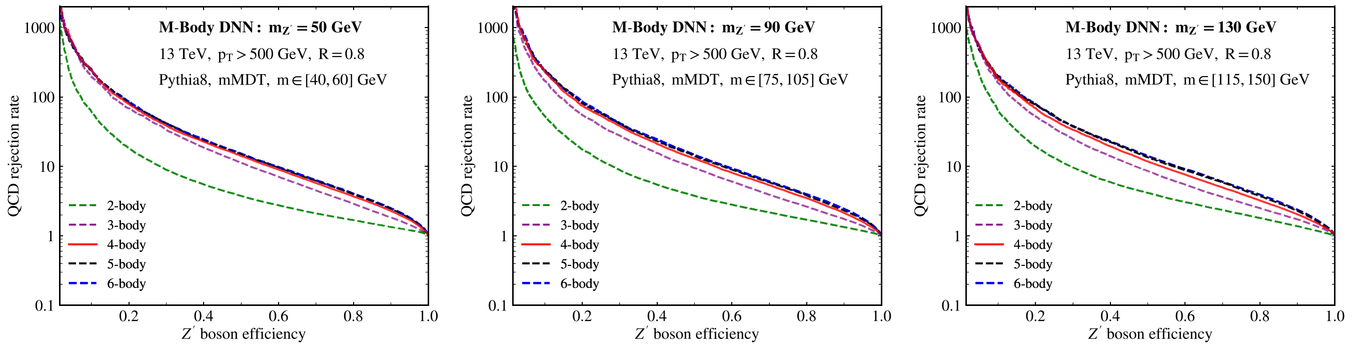
FIG. 7. (a) Comparison of PDFs of $\beta^{\mathrm{ML}(g)}_{3,H\to b\bar{b}}$ for mMDT groomed $H\to b\bar{b}$ discrimination, using ∼250,000 signal and background samples, and the distributions of the regression DNN prediction. The distributions are rescaled by a constant for the sake of visual comparison. (b) Probability density distributions of $\hat{\beta}^{\mathrm{ML}(g)}_{3,H\to b\bar{b}}$ obtained via linear regression. (c) Comparison of discrimination power of $\beta^{\mathrm{ML}(g)}_{3,H\to b\bar{b}}$, $\hat{\beta}^{\mathrm{ML}(g)}_{3,H\to b\bar{b}}$ and $\hat{\beta}^{\mathrm{ML}(g)}_{4,H\to b\bar{b}}$ to standard observables, where the 4-body product observable is seen to perform best for groomed $H\to b\bar{b}$ discrimination. (d) Comparison of $\hat{\beta}^{\mathrm{ML}(g)}_{3,H\to b\bar{b}}$ and $\beta^{\mathrm{ML}(g)}_{3,H\to b\bar{b}}$ to $\beta^{(g)}_3$ proposed in [8]; we note that the latter two 3-body product observables provide essentially the same discrimination power while the 3- and 4-body ones obtained with linear regression outperform them.

APPENDIX B: GROOMED $H\to b\bar{b}$ vs $g\to b\bar{b}$ DISCRIMINATION

Utilizing the result that discrimination power for mMDT groomed $H\to b\bar{b}$ vs $g\to b\bar{b}$ discrimination saturates at 3-body phase space [8], we use the procedure proposed in Sec. III to find the optimal product observable. The final values for the parameters $\{a,\ldots,e\}$ obtained through the optimization are presented in Table V, along with those obtained in the previous study. Interestingly, the exponents for $\beta^{\mathrm{ML(g)}}_{3,H\to b\bar{b}}$ are nearly the same for c, d, and e, but are quite different for a and b. The factors d and e are also similar for $\hat{\beta}^{\mathrm{ML(g)}}_{3,H\to b\bar{b}}$ up to a multiplicative factor.

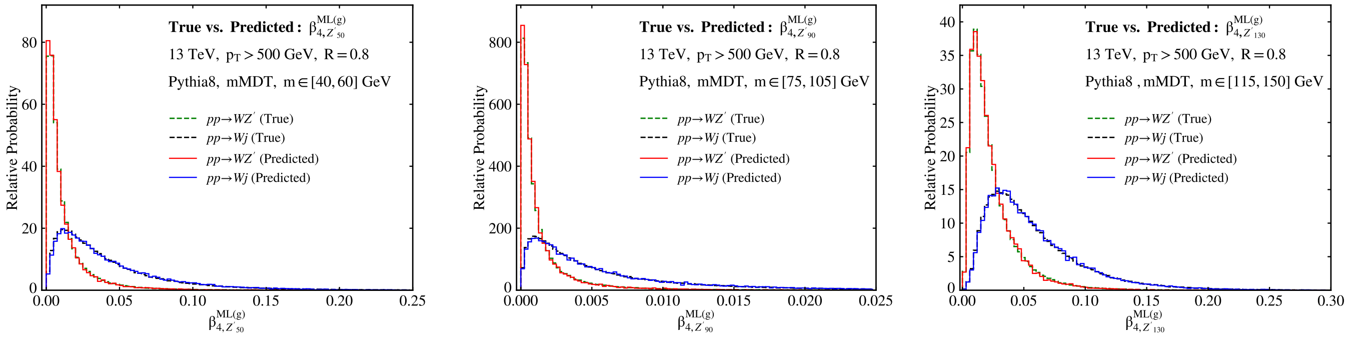
In Fig. 7(a), we plot the distributions of the new observable computed for signal and background, along with the prediction from the neural network. We note that the network provides a good match to the true distribution, where the latter is also calculated on ten times more jets. We then compare the ROC curves for the new observable to $D^{(2)}_2$ [67], $N^{(2)}_2$ [68] observables, and $\tau^{(2)}_{21}$ in Fig. 7(c).

In addition, we compare the new observable to β_3 in Fig. 7(d) to demonstrate that both observables have

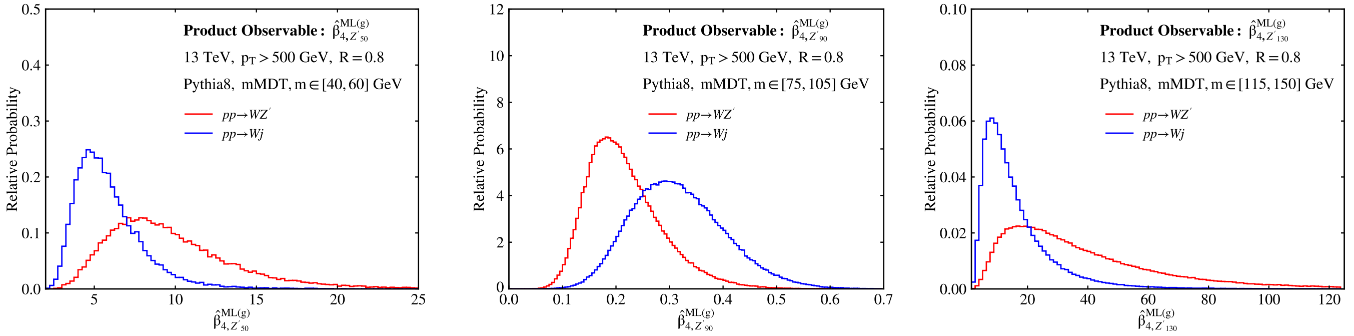


(a) $m_{Z'} = 50$ GeV (b) $m_{Z'} = 90$ GeV (c) $m_{Z'} = 130$ GeV

FIG. 8. M-body discrimination results of mMDT groomed Z' vs QCD jets. Here, discrimination power is again seen to effectively saturate at 4-body phase space for all considered values of $m_{Z'}$.



(a) $m_{Z'} = 50$ GeV (b) $m_{Z'} = 90$ GeV (c) $m_{Z'} = 130$ GeV



(d) $m_{Z'} = 50$ GeV (e) $m_{Z'} = 90$ GeV (f) $m_{Z'} = 130$ GeV

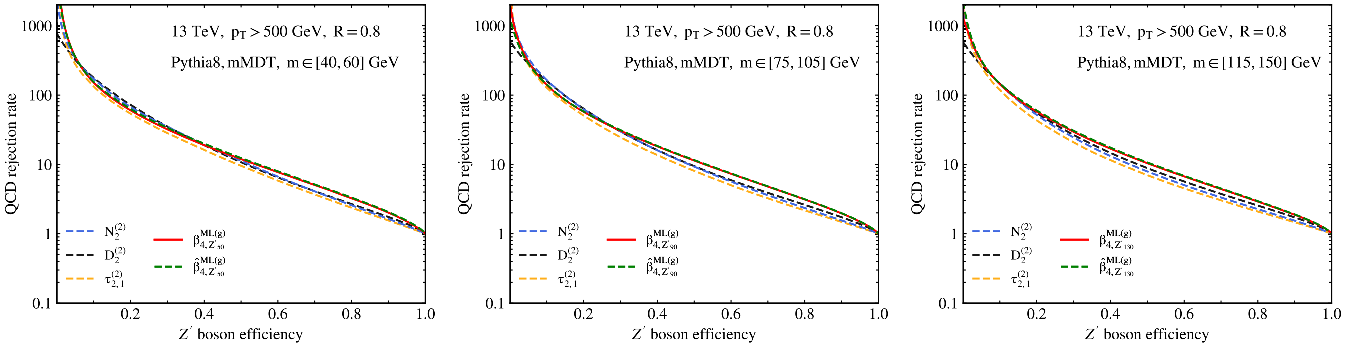
FIG. 9. Top panels [(a)–(c)]: Comparison of the probability density function of the new $\beta_4^{\mathrm{ML(g)}}$ observables for mMDT groomed Z' discrimination, calculated for ∼300,000 signal and background samples, and the distribution of the regression DNN predictions of 25,000 observable values. The distributions are rescaled for the sake of visual comparison. Bottom panels [(d)–(f)]: Distributions of the $\hat{\beta}_4^{\mathrm{ML(g)}}$ observables for ungroomed Z' discrimination that were obtained via linear regression.

TABLE VI. Summary of parameters for $\beta_4^{\mathrm{ML(g)}}$ for mMDT groomed Z' vs QCD discrimination at three mass points.

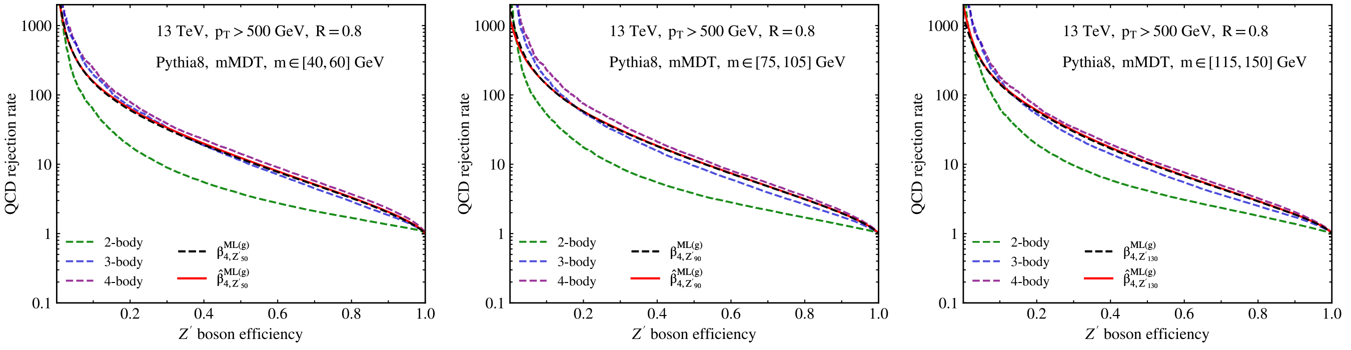
$m_{Z'}$ (GeV)	a	b	c	d	e	f	g	h
50	2.6	−0.41	−2.94	−2.79	0.20	0.93	−0.66	2.43
90	2.3	−1.35	−2.05	−1.64	−0.81	0.89	2.03	−0.44
130	0.80	−1.74	−0.28	−1.01	−0.38	0.56	0.82	0.69

TABLE VII. Summary of parameters for $\hat{\beta}_4^{\mathrm{ML(g)}}$ for mMDT groomed Z' vs QCD discrimination at three mass points.

$m_{Z'}$ (GeV)	a	b	c	d	e	f	g	h
50	−0.35	0.35	0.56	1.05	−0.17	−0.24	−0.34	0.51
90	0.26	−0.41	−0.39	−0.68	−0.15	0.11	0.25	0.42
130	1.28	0.54	0.35	1.09	0.09	−0.38	−1.06	−0.48



(a) $m_{Z'} = 50$ GeV (b) $m_{Z'} = 90$ GeV (c) $m_{Z'} = 130$ GeV



(d) $m_{Z'} = 50$ GeV (e) $m_{Z'} = 90$ GeV (f) $m_{Z'} = 130$ GeV

FIG. 10. Top panels [(a)–(c)]: Comparison of discrimination power of $\beta_4^{\mathrm{ML(g)}}$ observables to standard observables; the latter are computed with an angular exponent of 2, for which they were observed to perform best for mMDT groomed samples. Bottom panels [(d)–(f)]: Comparison of $\beta_4^{\mathrm{ML(g)}}$ to discrimination power of neural networks trained on the M-body observable bases; the observables capture almost all of the discrimination power of the 4-body neural networks.

essentially the same discrimination power as expected. Then, this allows us to proceed to applying the procedure on higher dimensional problems. Further, we plot the ROC curve for the 4-body product observable from the linear regression method, noting that it provides the best performance of the observables that have been explored for this problem.[6]

APPENDIX C: GROOMED Z' vs QCD

In this section we carry out the same set of studies for mMDT groomed Z' discrimination as for the ungroomed cases from Sec. IV B. As in the ungroomed case, Fig. 8 indicates that the saturation of discrimination power occurs at 4-body phase space.

The results for the final observables for the three $m_{Z'}$ points are presented in Tables VI and VII, and the observable distributions are plotted in Fig. 9. The performances of the new observables are compared to standard ones and M-body DNNs in Fig. 10 and the corresponding AUCs are shown in Table VIII for different mass points. The conclusions from this section are qualitatively the same as from Sec. IV B, with a slightly lower AUC from both the product observable and the physics-motivated observables. Importantly, the product observables for the groomed case appear to saturate the bounds from the M-body phase space better than in the ungroomed case.

APPENDIX D: MASS DEPENDENCE OF β_M^{ML}

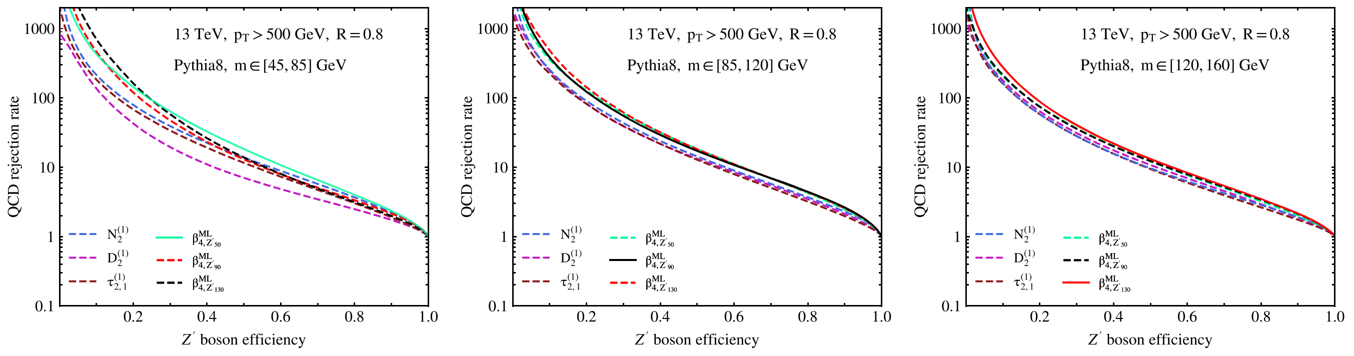
Here, we briefly study the performance of the new observables presented in Sec. IV B. They are tested on a different combination of signal and background samples from the ones they were optimized on; for example, we

TABLE VIII. AUC, from Fig. 10, of standard observables and the $\beta_4^{\mathrm{ML(g)}}$ observables, optimized for the corresponding signal, for mMDT groomed Z' vs QCD discrimination at three $m_{Z'}$ points. The ROC curves are calculated using the full datasets, with ~300,000 events passing the mass cut for each value of $m_{Z'}$.

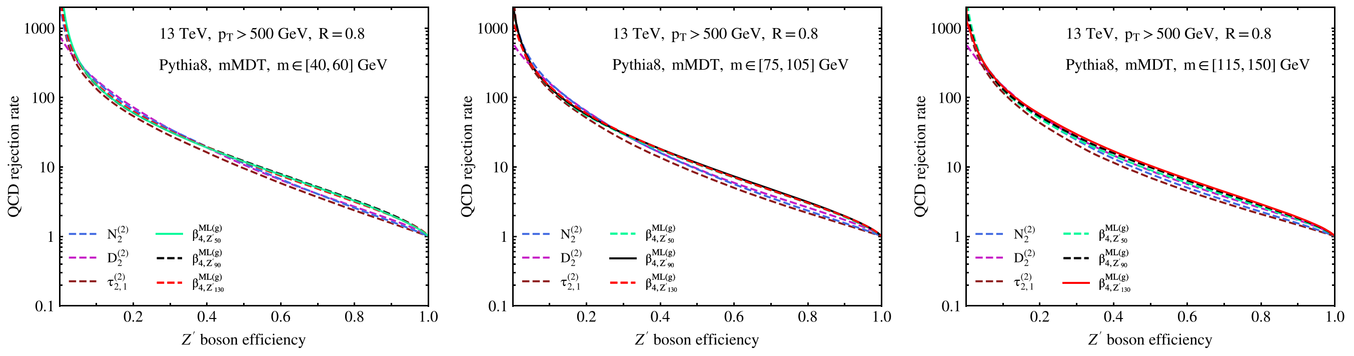
$m_{Z'}$ (GeV)	$\hat{\beta}_4^{\mathrm{ML(g)}}$	$\beta_4^{\mathrm{ML(g)}}$	$\mathrm{N}_2^{(2)}$	$\mathrm{D}_2^{(2)}$	$\tau_{2,1}^{(2)}$
50	0.830	0.826	0.796	0.803	0.780
90	0.822	0.821	0.780	0.796	0.763
130	0.814	0.811	0.769	0.791	0.751

---

[6] Explicitly, the optimal parameter values for $\hat{\beta}^{\mathrm{ML}(g)}_{4,H\to b\bar{b}}$ are as follows: $\{a, \ldots, h\} = \{-2.09, 1.46, -0.31, -0.49, 0.35, 0.03, -0.18, 0.23\}$, and it leads to an AUC of 0.778 in Fig. 7(c).



(a) $m_{Z'} = 50$ GeV (b) $m_{Z'} = 90$ GeV (c) $m_{Z'} = 130$ GeV



(d) $m_{Z'} = 50$ GeV (e) $m_{Z'} = 90$ GeV (f) $m_{Z'} = 130$ GeV

FIG. 11. Here we plot results for the new observables on Z' samples with a different mass point to that which they were optimized on, within the mass windows appropriate for the corresponding signal. We note that for all cases, all the new observables demonstrate very similar discrimination power, and outperform standard observables.

calculate the new observable for $m_{Z'} = 130$ GeV on signal samples for $m_{Z'} = 90$ GeV, and background, that pass the mass window on which the 90 GeV observable was optimized. The results for this study are presented in Fig. 11, and indicate that while these observables are optimized on samples from a specific mass point, they can be applied to other classification tasks and still provide better discrimination performance than standard observables. This also suggests that the different parameter sets in Tables II and III may represent observables with very similar physical information even though the N-subjettiness variables are not invariant under transverse boosts.

APPENDIX E: SATURATING THE DISCRIMINATION POWER OF $\hat{\beta}_M^{\mathrm{ML}}$

In this section we briefly study the flexibility of the product form ansatz using the $\hat{\beta}_M^{\mathrm{ML}}$ observables obtained via the linear regression procedure. For concreteness, we look at the $m_{Z'} = 90$ GeV case, and plot ROC curves for the product observables up to $M = 8$ in Fig. 12.

We observe that discrimination power gradually increases up to the inclusion of 7- or 8-body phase space variables. Compared to the ROC curve at the point of saturation, from the 4-body DNN classifier, these results suggest that while a DNN can adjust thresholds on the M-body inputs such that there is effectively only redundant discriminating information in higher M-body bases, as is also expected from the physics study in Ref. [7], the

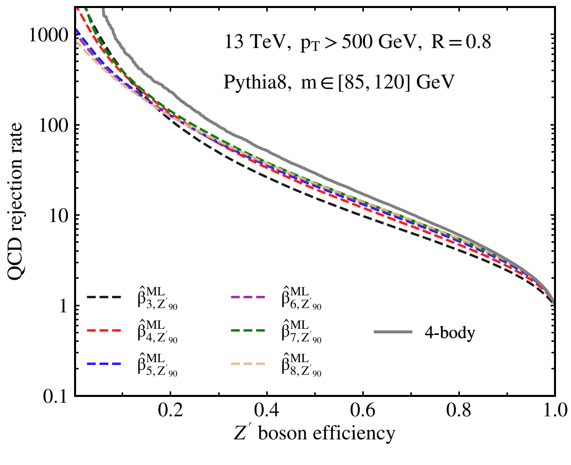


FIG. 12. Here we compare the discrimination power of $\hat{\beta}^{\mathrm{ML}}_{M,Z'_{90}}$ for ungroomed Z' discrimination for values of $M = 3, \ldots, 8$. We note that while discrimination power of the product form does increase with higher M (until the inclusion of 7- or 8-body phase space variables), it can only capture a limited amount of useful discriminating information from inclusion of variables from beyond the basis of the point of saturation of a DNN classifier (dark gray).

product observables do still benefit from including N-subjettiness variables from beyond the point of saturation.

Depending on the classification task, the product observables may even come very close to matching the performance of a saturated ML classifier (Fig. 10). However, ultimately it cannot capture all available information, due to the lack of further flexibility of the product form ansatz. These observations will of course vary based on the objects being studied. We leave further physics studies of the product form or other equivalent ansatz to future work.

[1] A. J. Larkoski, I. Moult, and B. Nachman, arXiv:1709.04464.
[2] L. Asquith *et al.*, arXiv:1803.06991.
[3] J. Pearkes, W. Fedorko, A. Lister, and C. Gay, arXiv:1704.02124.
[4] P. T. Komiske, E. M. Metodiev, and J. Thaler, J. High Energy Phys. 01 (2019) 121.
[5] P. T. Komiske, E. M. Metodiev, and J. Thaler, J. High Energy Phys. 04 (2018) 013.
[6] M. Erdmann, E. Geiser, Y. Rath, and M. Rieger, J. Instrum. **14**, P06006 (2019).
[7] K. Datta and A. Larkoski, J. High Energy Phys. 06 (2017) 073.
[8] K. Datta and A. J. Larkoski, J. High Energy Phys. 03 (2018) 086.
[9] A. Butter, G. Kasieczka, T. Plehn, and M. Russell, SciPost Phys. **5**, 028 (2018).
[10] J. Cogan, M. Kagan, E. Strauss, and A. Schwartzman, J. High Energy Phys. 02 (2015) 118.
[11] L. G. Almeida, M. Backovic, M. Cliche, S. J. Lee, and M. Perelstein, J. High Energy Phys. 07 (2015) 086.
[12] L. de Oliveira, M. Kagan, L. Mackey, B. Nachman, and A. Schwartzman, J. High Energy Phys. 07 (2016) 069.
[13] P. T. Komiske, E. M. Metodiev, and M. D. Schwartz, J. High Energy Phys. 01 (2017) 110.
[14] J. Barnard, E. N. Dawe, M. J. Dolan, and N. Rajcic, Phys. Rev. D **95**, 014018 (2017).
[15] G. Kasieczka, T. Plehn, M. Russell, and T. Schell, J. High Energy Phys. 05 (2017) 006.
[16] F. A. Dreyer, G. P. Salam, and G. Soyez, J. High Energy Phys. 12 (2018) 064.
[17] J. Lin, M. Freytsis, I. Moult, and B. Nachman, J. High Energy Phys. 10 (2018) 101.
[18] K. Fraser and M. D. Schwartz, J. High Energy Phys. 10 (2018) 093.
[19] Y.-T. Chien and R. Kunnawalkam Elayavalli, arXiv:1803.03589.
[20] S. Macaluso and D. Shih, J. High Energy Phys. 10 (2018) 121.
[21] D. Guest, J. Collado, P. Baldi, S.-C. Hsu, G. Urban, and D. Whiteson, Phys. Rev. D **94**, 112002 (2016).
[22] S. Egan, W. Fedorko, A. Lister, J. Pearkes, and C. Gay, arXiv:1711.09059.
[23] A. Andreassen, I. Feige, C. Frye, and M. D. Schwartz, Eur. Phys. J. C **79**, 102 (2019).
[24] T. Cheng, Comput. Software Big Sci. **2**, 3 (2018).
[25] G. Louppe, K. Cho, C. Becot, and K. Cranmer, J. High Energy Phys. 01 (2019) 057.
[26] I. Henrion *et al.*, DLPS at NIPS, 2017, https://dl4physicalsciences.github.io/files/nips_dlps_2017_29.pdf.
[27] M. Aaboud *et al.* (ATLAS Collaboration), Eur. Phys. J. C **79**, 375 (2019).
[28] CMS Collaboration, CERN Report No. CMS-DP-2017-049, 2017, https://cds.cern.ch/record/2295725.
[29] CMS Collaboration, CERN Report No. CMS-DP-2018-046, 2018, https://cds.cern.ch/record/2630438.
[30] ATLAS Collaboration, CERN Report No. ATL-PHYS-PUB-2017-017, 2017, http://cds.cern.ch/record/2275641.
[31] CMS Collaboration, CERN Report No. CMS-DP-2017-027, 2017, https://cds.cern.ch/record/2275226.
[32] ATLAS Collaboration, CERN Report No. ATL-PHYS-PUB-2017-003, 2017, https://cds.cern.ch/record/2255226.
[33] ATLAS Collaboration, CERN Report No. ATL-PHYS-PUB-2017-013, 2017, https://cds.cern.ch/record/2273281.
[34] CMS Collaboration, CERN Report No. CMS-DP-2018-058, 2018, https://cds.cern.ch/record/2646773.
[35] A. M. Sirunyan *et al.* (CMS Collaboration), J. Instrum. **13**, P05011 (2018).
[36] E. M. Metodiev and J. Thaler, Phys. Rev. Lett. **120**, 241602 (2018).
[37] P. T. Komiske, E. M. Metodiev, B. Nachman, and M. D. Schwartz, Phys. Rev. D **98**, 011502 (2018).
[38] E. M. Metodiev, B. Nachman, and J. Thaler, J. High Energy Phys. 10 (2017) 174.
[39] L. M. Dery, B. Nachman, F. Rubbo, and A. Schwartzman, J. High Energy Phys. 05 (2017) 145.
[40] T. Cohen, M. Freytsis, and B. Ostdiek, J. High Energy Phys. 02 (2018) 034.
[41] G. Louppe, M. Kagan, and K. Cranmer, arXiv:1611.01046.
[42] C. Shimmin, P. Sadowski, P. Baldi, E. Weik, D. Whiteson, E. Goul, and A. Sogaard, Phys. Rev. D **96**, 074034 (2017).
[43] ATLAS Collaboration, CERN Technical Report No. ATL-PHYS-PUB-2018-014, 2018, http://cds.cern.ch/record/2630973.
[44] J. H. Collins, K. Howe, and B. Nachman, Phys. Rev. Lett. **121**, 241803 (2018).
[45] J. H. Collins, K. Howe, and B. Nachman, Phys. Rev. D **99**, 014038 (2019).
[46] T. Heimel, G. Kasieczka, T. Plehn, and J. M. Thompson, SciPost Phys. **6**, 030 (2019).
[47] M. Farina, Y. Nakai, and D. Shih, arXiv:1808.08992.
[48] J. Hajer, Y.-Y. Li, T. Liu, and H. Wang, arXiv:1807.10261.
[49] A. M. Sirunyan *et al.* (CMS Collaboration), Phys. Rev. Lett. **119**, 111802 (2017).

[50] A. M. Sirunyan *et al.* (CMS Collaboration), J. High Energy Phys. 01 (2018) 097.

[51] M. Aaboud *et al.* (ATLAS Collaboration), Phys. Lett. B **788**, 316 (2019).

[52] I. W. Stewart, F. J. Tackmann, and W. J. Waalewijn, Phys. Rev. Lett. **105**, 092002 (2010).

[53] J. Thaler and K. Van Tilburg, J. High Energy Phys. 03 (2011) 015.

[54] J. Thaler and K. Van Tilburg, J. High Energy Phys. 02 (2012) 093.

[55] T. Sjostrand, S. Mrenna, and P. Z. Skands, J. High Energy Phys. 05 (2006) 026.

[56] T. Sjostrand, S. Ask, J. R. Christiansen, R. Corke, N. Desai, P. Ilten, S. Mrenna, S. Prestel, C. O. Rasmussen, and P. Z. Skands, Comput. Phys. Commun. **191**, 159 (2015).

[57] J. Alwall, R. Frederix, S. Frixione, V. Hirschi, F. Maltoni, O. Mattelaer, H. S. Shao, T. Stelzer, P. Torrielli, and M. Zaro, J. High Energy Phys. 07 (2014) 079.

[58] M. Cacciari, G. P. Salam, and G. Soyez, J. High Energy Phys. 04 (2008) 063.

[59] M. Cacciari, G. P. Salam, and G. Soyez, Eur. Phys. J. C **72**, 1896 (2012).

[60] Y. L. Dokshitzer, G. D. Leder, S. Moretti, and B. R. Webber, J. High Energy Phys. 08 (1997) 001.

[61] M. Wobisch and T. Wengler, in *Monte Carlo Generators for HERA Physics. Proceedings, Workshop, Hamburg, Germany, 1998-1999* (DESY and Fermilab, 1998), pp. 270–279.

[62] A. J. Larkoski, S. Marzani, G. Soyez, and J. Thaler, J. High Energy Phys. 05 (2014) 146.

[63] M. Dasgupta, A. Fregoso, S. Marzani, and G. P. Salam, J. High Energy Phys. 09 (2013) 029.

[64] S. Catani, Y. L. Dokshitzer, M. H. Seymour, and B. R. Webber, Nucl. Phys. **B406**, 187 (1993).

[65] S. D. Ellis and D. E. Soper, Phys. Rev. D **48**, 3160 (1993).

[66] G. C. Blazey *et al.*, in *QCD and Weak Boson Physics in Run II. Proceedings, Batavia, USA, 1999* (DESY and Fermilab, 2000), pp. 47–77, http://lss.fnal.gov/cgi-bin/find_paper.pl?conf-00-092.

[67] A. J. Larkoski, I. Moult, and D. Neill, J. High Energy Phys. 12 (2014) 009.

[68] I. Moult, L. Necib, and J. Thaler, J. High Energy Phys. 12 (2016) 153.

[69] D. Bertolini, T. Chan, and J. Thaler, J. High Energy Phys. 04 (2014) 013.

[70] A. J. Larkoski, D. Neill, and J. Thaler, J. High Energy Phys. 04 (2014) 017.

[71] A. J. Larkoski and J. Thaler, Phys. Rev. D **90**, 034010 (2014).

[72] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, in *Advances in Neural Information Processing Systems 27*, edited by Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger (Curran Associates, Inc., New York, 2014), pp. 2672–2680, http://papers.nips.cc/paper/5423-generative-adversarial-nets.pdf.

[73] D. P. Kingma and M. Welling, arXiv:1312.6114.

[74] D. J. Rezende, S. Mohamed, and D. Wierstra, in *Proceedings of the 31st International Conference on International Conference on Machine Learning–Volume 32* (JMLR.org, 2014), ICML'14, pp. II–1278–II–1286, http://dl.acm.org/citation.cfm?id=3044805.3045035.

[75] D. P. Kingma and J. Ba, arXiv:1412.6980.

[76] F. Chollet, Keras, https://github.com/fchollet/keras (2015).

[77] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin *et al.*, *TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems* (2015), software available from tensorflow.org, http://tensorflow.org/.

[78] P. Virtanen *et al.*, arXiv:1907.10121.

[79] D. J. Wales and J. P. K. Doye, J. Phys. Chem. A **101**, 5111 (1997).

[80] M. J. D. Powell, *A Direct Search Optimization Method That Models the Objective and Constraint Functions by Linear Interpolation* (Springer Netherlands, Dordrecht, 1994), pp. 51–67.

[81] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, J. Mach. Learn. Res. **15**, 1929 (2014).

[82] V. Nair and G. E. Hinton, in *ICML* (Omnipress, Wisconsin, 2010), pp. 807–814, http://dblp.uni-trier.de/db/conf/icml/icml2010.html#NairH10.