Automating the ABCD method with machine learning

Gregor Kasieczka,[1,*] Benjamin Nachman,[2,†] Matthew D. Schwartz,[3,§] and David Shih[2,4,5,‡]

[1]*Institut für Experimentalphysik, Universität Hamburg, Luruper Chaussee 149, D-22761 Hamburg, Germany*
[2]*Physics Division, Lawrence Berkeley National Laboratory, Berkeley, California 94720, USA*
[3]*Department of Physics, Harvard University, Cambridge, Massachusetts 02138, USA*
[4]*NHETC, Department of Physics and Astronomy, Rutgers University, Piscataway, New Jersey 08854, USA*
[5]*Berkeley Center for Theoretical Physics, University of California, Berkeley, California 94720, USA*

(Received 6 August 2020; accepted 3 February 2021; published 22 February 2021)

The ABCD method is one of the most widely used data-driven background estimation techniques in high energy physics. Cuts on two statistically independent classifiers separate signal and background into four regions, so that background in the signal region can be estimated simply using the other three control regions. Typically, the independent classifiers are chosen "by hand" to be intuitive and physically motivated variables. Here, we explore the possibility of automating the design of one or both of these classifiers using machine learning. We show how to use state-of-the-art decorrelation methods to construct powerful yet independent discriminators. Along the way, we uncover a previously unappreciated aspect of the ABCD method: its accuracy hinges on having low signal contamination in control regions not just overall, but *relative* to the signal fraction in the signal region. We demonstrate the method with three examples: a simple model consisting of three-dimensional Gaussians; boosted hadronic top jet tagging; and a recasted search for paired dijet resonances. In all cases, automating the ABCD method with machine learning significantly improves performance in terms of ABCD closure, background rejection, and signal contamination.

DOI: 10.1103/PhysRevD.103.035021

I. INTRODUCTION

A key component of high energy physics data analysis, whether for Standard Model (SM) measurements or searches beyond the SM, is background estimation. While powerful simulations and first-principles calculations exist and are constantly improving, they still remain inadequate for the task of precisely estimating backgrounds in many situations. For example, the cross section of the SM background for events with a large number of hadronic jets is difficult to estimate. Therefore methods for *data-driven* background estimation remain a crucial part of the experimental toolkit. The idea behind all data-driven background estimation strategies is to extrapolate or interpolate from some control regions which are background dominated into a signal region of interest.

One classic (see e.g., Ref. [1]) data-driven background method which is used in a multitude[1] of physics analyses at the Large Hadron Collider (LHC) and elsewhere is the *ABCD method*. The idea of the ABCD method is to pick two observables f and g (for example, the invariant mass of a dijet system and the rapidity of that system) which are approximately statistically independent for the background, and which are effective discriminators of signal versus background. Simple thresholds on these observables partition events into four regions. Three of these regions, called B, C, and D, are background dominated. The fourth, A, is the signal region. If the observables are independent, then the background in the signal region can be predicted from the other three regions via

$$N_A = \frac{N_B N_C}{N_D}, \tag{1.1}$$

where N_i is the number of events in region i. This setup is depicted schematically for signal and background distributions in Fig. 1.

*gregor.kasieczka@uni-hamburg.de
†bpnachman@lbl.gov
‡shih@physics.rutgers.edu
§schwartz@g.harvard.edu

[1]See Refs. [2,3] for recent examples and many more in the ATLAS and CMS search group webpages [4–8].

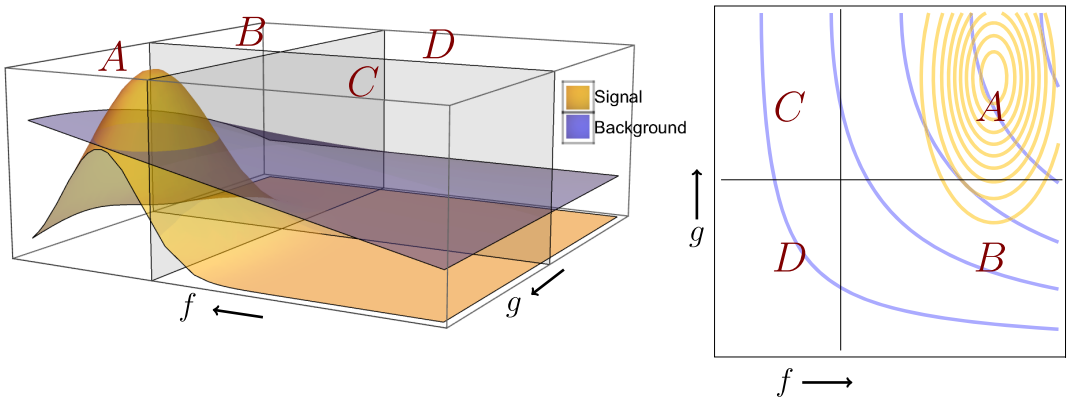

FIG. 1. The ABCD method is used to estimate the background in region A as $N_A = \frac{N_B N_C}{N_D}$. It requires the signal to be relatively localized in region A and the observables to be independent on background. The shaded planes (left) or lines (right) denote thresholds which isolate the signal in region A.

Typically, the observables f and g for the ABCD method are chosen to be simple, physically well-motivated features such as mass, H_T, and missing E_T. Their independence is always ensured manually, e.g., by choosing features that are known physically to have little correlation or by trial and error.[2] In some cases, independence can be guaranteed by using completely orthogonal sources of information, such as measurements from different subdetectors or properties of independently produced particles. However, more often than not, the features are not 100% independent and one has to apply a residual correction derived from simulations. Ideally, this simulation correction has small uncertainties—either because the effect itself is small, or because the correction is robust. But such corrections, together with the fact that simple kinematic features are typically not optimal discriminants of signal versus background, generally limit the effectiveness of the ABCD method and the sensitivity of the analysis in question. (See [10], however, for a proposal for extending the ABCD method using higher-order information when the features are not independent.)

In this paper, we will explore the systematic application of deep learning to the ABCD method. Deep learning has already demonstrated impressive success in finding observables that are effective at discrimination [11–65] and that are uncorrelated with other observables [66–81]. Building on previous success, we will aim to use deep learning to automate the selection of features used in the ABCD method, simultaneously optimizing their discrimination power while ensuring their independence.

The main tool we will use in automating the ABCD method will be a recently proposed method for training decorrelated deep neural networks [73]. This method uses a well-known statistical measure of nonlinear dependence known as *distance correlation* (DisCo) [82–85]. DisCo is a function of two random variables (or samples thereof) and is zero if and only if the variables are statistically independent; otherwise it is positive. Therefore it can be added as a regularization term in the loss function of a neural network to encourage the neural network output to be decorrelated against any other feature. In [73] it was shown that DisCo decorrelation achieves state-of-the-art decorrelation performance while being easier and more stable to train than approaches based on adversarial methods. Therefore it is ideally suited to automating the ABCD method.

We will propose two new ideas for automating the ABCD method, which we will call *single DisCo* and *double DisCo*, respectively. In single DisCo, we will train a single neural network classifier on signal and background and use DisCo regularization to force it to be independent in the background of a second, fixed feature (such as invariant mass). In double DisCo, we will train *two* neural network classifiers and use DisCo regularization to force them to be independent of one another.

We will study three examples to illustrate the effectiveness of these methods. The first example is a simple model where signal and background are drawn from three-dimensional Gaussian distributions. Here the aim is to understand many of the features of single and double DisCo in a fully controlled environment. The second example is boosted hadronic top tagging, where often sideband interpolation in mass is employed. For the ABCD method we treat a window selection on the mass as a classifier variable. Thus we use the invariant mass as the single DisCo fixed feature, and we then show how double DisCo can improve on this by combining mass with other information to produce more effective classification. Finally, we examine a search that currently uses the conventional ABCD method: the ATLAS paired dijet

[2]There are examples where f or g are chosen automatically, as is the case when one of them is a neural network (see e.g., Ref. [9]). However, such analyses do not have an automated procedure for ensuring that f and g are independent and the departure from Eq. (1.1) can be significant.

resonance search, motivated by R-parity violation (RPV) squark decays [86] (for a similar search by CMS, see [87]). We show that significant performance gains are possible using single and double DisCo.

In the course of our study of the ABCD method, we will uncover a hitherto unappreciated limitation of the method, which we call *normalized signal contamination*. Usually, practitioners are concerned with the overall signal-to-background ratio in the control regions; if this is small, then they are usually satisfied. We point out that in fact another relevant quantity for the significance calculation is the signal-to-background ratio in the control regions *relative* or *normalized* to the signal-to-background ratio in the signal region. In other words, the requirement of signal contamination is actually

$$\frac{N_{i,s}}{N_{i,b}} \ll \frac{N_{A,s}}{N_{A,b}} \tag{1.2}$$

in addition to $\frac{N_{i,s}}{N_{i,b}} \ll 1$ (where $N_{i,s}$ and $N_{i,b}$ are the numbers of signal and background events in region $i = A, B, C, D$). In many analyses (e.g., [86]), the signal fraction in the signal region is quite small, meaning that even a small amount of signal contamination in the control regions can bias the p values reported by the search. We will show that single and double DisCo not only improve the discrimination power and background closure of the ABCD method but can also significantly reduce the level of signal contamination at the same time.

This paper is organized as follows. Section II reviews the ABCD method, and Sec. III describes how the method can be automated using deep learning. Numerical results for examples described above are presented in Sec. IV. The paper ends with conclusions and outlook in Sec. VI.

II. THE ABCD METHOD

The ABCD method starts with two features f and g. Imposing thresholds f_c and g_c divides the feature space into four rectangular regions, A, B, C, and D with corresponding event counts:

$$\begin{aligned} N_{A,\ell} &= N_\ell \Pr(f \geq f_c \text{ and } g \geq g_c|\ell), \\ N_{B,\ell} &= N_\ell \Pr(f \geq f_c \text{ and } g < g_c|\ell), \\ N_{C,\ell} &= N_\ell \Pr(f < f_c \text{ and } g \geq g_c|\ell), \\ N_{D,\ell} &= N_\ell \Pr(f < f_c \text{ and } g < g_c|\ell), \end{aligned} \tag{2.1}$$

where $N_\ell = N_{A,\ell} + N_{B,\ell} + N_{C,\ell} + N_{D,\ell}$ is the total number of events of type ℓ and $\ell \in \{\text{signal}(s), \text{background}(b), \text{all}(a)\}$ and $\Pr(\cdot)$ is the probability. The regions B, C, and D can be used to predict N_A:

$$N_{A,b}^{\text{predicted}} \equiv \frac{N_{B,a}N_{C,a}}{N_{D,a}}. \tag{2.2}$$

For the ABCD method to be valid, we would need $N_{A,b} = N_{A,b}^{\text{predicted}}$.

There are two requirements for $N_{A,b}^{\text{predicted}}$ to be accurate. First, the Bernoulli random variables $f < f_c$ and $g < g_c$ must be independent for the background in order to guarantee that

$$N_{A,b} = \frac{N_{B,b}N_{C,b}}{N_{D,b}}. \tag{2.3}$$

To see this, note that (2.3) is equivalent to

$$N_b \times N_{A,b} = (N_{A,b} + N_{B,b}) \times (N_{A,b} + N_{C,b}). \tag{2.4}$$

Then, substituting in Eq. (2.1) to Eq. (2.4) yields

$$\Pr(f \geq f_c \quad \text{and} \quad g \geq g_c|b) = \Pr(f \geq f_c|b) \times \Pr(g \geq g_c|b), \tag{2.5}$$

which is a definition of independence. While it is sufficient to have one set of thresholds, having a range over which independence holds adds robustness to the estimation procedure. If the ABCD method holds for all values of f_c and g_c, then f and g themselves must be independent. Note that this condition is stronger than requiring zero *linear* correlation. Two random variables can have zero linear correlation yet be nonlinearly dependent. In general, such a case would invalidate Eq. (2.3).

The second requirement for the ABCD method involves the signal and the background:

$$\frac{N_{B,a}N_{C,a}}{N_{D,a}} = \frac{N_{B,b}N_{C,b}}{N_{D,b}}. \tag{2.6}$$

In particular, if the signal contamination in regions B, C, and D is large, then Eq. (2.2) will not hold. But what does large mean in this context? Typically, large signal contamination is taken to be an overall statement, i.e.,

$$\delta_i \equiv \frac{N_{i,s}}{N_{i,b}} \ll 1, \tag{2.7}$$

for regions $i = B, C, D$. However, we will now show that in addition to this criterion, another relevant quantity is *normalized signal contamination*

$$r \equiv \delta_A^{-1}(\delta_B + \delta_C - \delta_D) = \left(\frac{N_{A,s}}{N_{A,b}}\right)^{-1}\left(\frac{N_{B,s}}{N_{B,b}} + \frac{N_{C,s}}{N_{C,b}} - \frac{N_{D,s}}{N_{D,b}}\right), \tag{2.8}$$

and for the ABCD method to be valid, it must satisfy

$$|r| \ll 1. \tag{2.9}$$

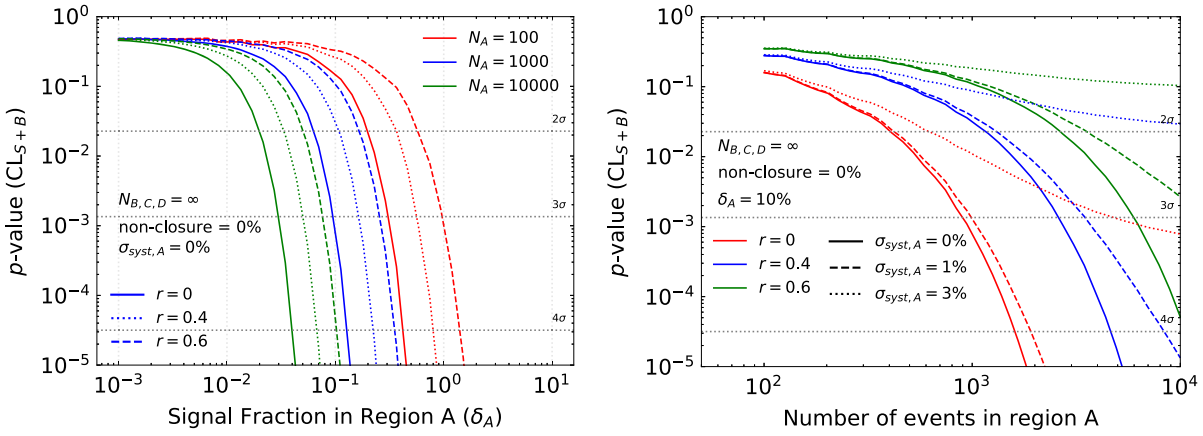

FIG. 2. The p-value ($\mathrm{CL}_{\mathrm{S+B}}$) for the ABCD method as a function of δ_A (N_A), and the signal fraction in region A (the number of background events in region A) for the left (right) plot. It is assumed that there is no uncertainty from regions C and D. The systematic uncertainty σ_{syst} is absolute (applied to the number of events in the signal region) and not relative to the signal fraction.

Note that this is often a much stronger requirement than (2.7). It is not enough that the signal fractions in each control region are small—they must be *small compared to the signal fraction in the signal region*. In many searches (e.g., the RPV stop search in Sec. IV C), signal to background can be quite small in the signal region, meaning that this can be a significant (and underappreciated) constraint on the ABCD method.

To see why (2.9) is required, suppose that the ABCD method closes exactly, so that Eq. (2.3) holds, but there is some signal contamination in all four regions. Then,

$$\begin{aligned} N_{A,b}^{\mathrm{predicted}} &= N_{B,b}(1+\delta_B)\times\frac{N_{C,b}(1+\delta_C)}{N_{D,b}(1+\delta_D)} \\ &= N_{B,b}\times\frac{N_{C,b}}{N_{D,b}}[1+\delta_B+\delta_C-\delta_D+\mathcal{O}(\delta^2)] \\ &= N_{A,b}[1+\delta_B+\delta_C-\delta_D+\mathcal{O}(\delta^2)], \end{aligned} \tag{2.10}$$

This will be compared with the number of events in region A, $N_{A,a}=N_{A,b}(1+\delta_A)$, to decide whether there is an excess or not. In order to detect the signal in A, one needs Eq. (2.9) to be satisfied. Note that we are still assuming that $\delta_{B,C,D}\ll 1$ in order for the subleading terms in Eq. (2.10) to be negligible.

Another point is that generally δ_D can be neglected compared to δ_B and δ_C (as it is diagonally opposite and should therefore be doubly suppressed). So we expect $r>0$ and an *overestimate* of the background in the signal region. This will make it much harder to discover new physics.

Finally, let us make the connection between the normalized signal contamination and classifier performance. For the fixed thresholds f_c and g_c, the signal (ϵ_s) and background (ϵ_b) efficiencies for each individual classifier can be computed as

$$\begin{aligned} \epsilon_{f,b} &= \frac{N_{A,b}+N_{B,b}}{N_b} \overset{\text{independence}}{=} \frac{N_{A,b}}{N_{A,b}+N_{C,b}}, \\ \epsilon_{g,b} &= \frac{N_{A,b}+N_{C,b}}{N_b} \overset{\text{independence}}{=} \frac{N_{A,b}}{N_{A,b}+N_{B,b}}, \\ \epsilon_{f,s} &= \frac{N_{A,s}+N_{B,s}}{N_s}, \\ \epsilon_{g,s} &= \frac{N_{A,s}+N_{C,s}}{N_s}. \end{aligned} \tag{2.11}$$

With these definitions and neglecting $N_{D,s}$, Eq. (2.8) can be re-written as

$$r=\frac{(1-\epsilon_{f,s})}{(1-\epsilon_{f,s}+\epsilon_{g,s})}\frac{\epsilon_{f,b}}{(1-\epsilon_{f,b})}+\frac{(1-\epsilon_{g,s})}{(1-\epsilon_{g,s}+\epsilon_{f,s})}\frac{\epsilon_{g,b}}{(1-\epsilon_{g,b})}. \tag{2.12}$$

The two terms in Eq. (2.12) are nearly the *diagnostic odds ratio* and importantly are minimized for a given signal efficiency when the background efficiency is as small as possible. This demonstrates that "classification performance" and "signal contamination" are synonymous in this context—the better a classifier is, the more likely it will be that there is a threshold which ensures a small relative signal contamination.

To illustrate these points, we show in Fig. 2 the effect of signal contamination on the p-value. The left plot shows the interplay between the relative signal contamination r and the number of events N_A in the signal region as a function of δ_A. For example, if the signal fraction in the signal region is $\delta_A=10\%$ and $N_A=1000$, the true p-value is 0.0015 while the reported value assuming negligible signal contamination would be 0.03 or 0.1 with an unaccounted for signal contamination (δ_B) of 4% and 6% in region B, respectively.

Correctly accounting for this signal contamination would require having a signal-model-dependent ABCD estimation. This could be done (see e.g., [88]), but would be much more complicated than most applications of the ABCD method. Adding an uncertainty to account for potential signal contamination is also not ideal—this is shown in the right plot of Fig. 2. Once again, for $N_A = 1000$ and $\delta_A = 10\%$, the true p-value is 0.0015 and a signal contamination of 4% in region B (δ_B) results in a p-value of 0.03. Adding an uncertainty ($\sigma_{\text{syst},A}$) of 5% increases this to 0.16 and an uncertainty of 10% further increases the p-value to 0.29. So while this would result in a conservative p-value, it means that potential discoveries would be masked.

III. AUTOMATING THE ABCD METHOD

Having described the requirements for the ABCD method (two strong classifiers that are independent for background), we now turn to the main idea of the paper: automating the ABCD method with machine learning.

Typically, when the ABCD method is used in experimental analyses, the two features are chosen by hand, based on physical intuition. Usually the features are simple quantities, such as mass, H_T, p_T, or missing E_T. In the remainder of the paper, we will investigate the benefits of allowing the ABCD features to be more complicated functions of the inputs. These functions will be obtained by training neural networks with suitable loss functions that ensure the ABCD objectives. We will see that machine learning has the potential to greatly improve the performance of the ABCD method.

The basic idea is that we want to train a classifier $f(X)$ where X are the input features (either low level inputs, such as four vectors or images, or high level inputs, such as p_T and mass) that is forced to be decorrelated against another classifier $g(X)$. This will achieve the first ABCD requirement of independent features. If the two classifiers are both good discriminants, this will satisfy the second ABCD requirement.

One can imagine two versions of this idea, both of them new:

(1) The second classifier is a simple, existing high-level variable (e.g., mass). In this case the problem is basically identical to the one that has been solved in the literature on decorrelation. We then just have to apply these approaches to the ABCD method.
(2) The second classifier is also a neural network. In this case we need to train two neural networks simultaneously while keeping them decorrelated from one another. This requires us to go beyond the usual literature on decorrelation against a fixed feature.

Regardless of whether $g(X)$ is fixed or learned, decorrelation can be achieved by any of the numerous methods that have been proposed [66–81]. In this paper we will use the DisCo method [73]. DisCo decorrelation proceeds through a positive-definite regularization term that penalizes statistical dependence. It achieves state-of-the-art performance while being significantly easier to train than adversarial decorrelation methods which rely on saddle-point extremization.

For the single DisCo ABCD method, we take the loss function to be the same as in [73]:

$$\mathcal{L}[f(X)] = \mathcal{L}_{\text{classifier}}[f(X), y] + \lambda \text{dCorr}^2_{y=0}[f(X), X_0], \quad (3.1)$$

where X are the features used for classification, $y \in \{0, 1\}$ are the labels, X_0 is the feature that one wants to be decorrelated from $f(X)$ (X_0 could be part of X), and $\mathcal{L}_{\text{classifier}}$ is the classifier loss such as the commonly used binary cross entropy. The subscript $y = 0$ in the second term of Eq. (3.1) ensures that the decorrelation is only applied to the background (class 0). Furthermore, $\lambda \geq 0$ is a hyperparameter that determines the decorrelation strength. The function $\text{dCorr}^2[f, g]$ is the squared distance correlation defined in [82–85] (see Appendix A). It has the property that $0 \leq \text{dCorr}[f, g] \leq 1$ and $\text{dCorr}[f, g] = 0$ if and only if f and g are independent. For Single DisCo, $g(X) = X_0$.

In practice, f is parametrized as a neural network and Eq. (3.1) is minimized using gradient-based methods. The distance correlation is computed for batches of data used to stochastically estimate the gradient. In the limit of small numbers of events, the naive distance covariance computed by replacing expectation values with sample averages is a biased estimator of the true distance correlation. Analogously to the case of sample variance (in which a factor of $\frac{1}{N-1}$ instead of $\frac{1}{N}$—where N denotes the minibatch size—is inserted to remove bias), there is an analytic low-N correction to the distance covariance that is unbiased [83,85]. Numerical results suggest that this correction is useful when N is low, but for sufficiently large training datasets with large enough batches, the correlation has little impact on the results.

For the double disco ABCD method, we use the loss function

$$\mathcal{L}[f, g] = \mathcal{L}_{\text{classifier}}[f(X), y] + \mathcal{L}_{\text{classifier}}[g(X), y] + \lambda \text{dCorr}^2_{y=0}[f(X), g(X)], \quad (3.2)$$

where now f and g are two neural networks that are trained simultaneously. When $\lambda = 0$, the loss will be minimized when $f = g$ is the optimal classifier (up to degeneracies). When $\lambda \to \infty$, f and g will be forced to be independent even if one or both of them does not classify well at all. In practice, if λ is taken too large, the DisCo term will tend to overwhelm the training and poor classification performance will result. Thus there should be an optimal λ at some finite value which can be determined by scanning over λ.

IV. APPLICATIONS

This section explores the efficacy of single and double DisCo in some applications of the ABCD method.

A. Simple example: Three-dimensional Gaussian random variables

We begin with a simple example to build some intuition and validate our methods. Consider a three-dimensional space (X_0, X_1, X_2), where the signal and background are both multivariate Gaussian distributions. We choose the means $\vec{\mu}$ and a covariance matrix $\mathbf{\Sigma}$ for background and signal as

$$\vec{\mu}_b = (0,0,0), \qquad \mathbf{\Sigma}_b = \sigma_b^2 \begin{pmatrix} 1 & \rho_b & 0 \\ \rho_b & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix},$$
$$\sigma_b = 1.5, \qquad \rho_b = -0.8, \tag{4.1}$$

and

$$\vec{\mu}_s = (2.5, 2.5, 2), \qquad \mathbf{\Sigma}_s = \sigma_s^2 \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}, \qquad \sigma_s = 1.5. \tag{4.2}$$

So for the background, all three features are centered at the origin and features X_0 and X_1 are correlated with each other but independent of X_2. For the signal, all three features are independent but are centered away from the origin. The first feature X_0 will play the role of the known feature for single DisCo in Sec. III.

All of the neural networks presented in this section use three hidden layers with 128 nodes per layer. The rectified linear unit (ReLU) activation function is used for the intermediate layers and the output is a sigmoid function. A hyperparameter of $\lambda = 1000$ is used for both single and double DisCo to ensure total decorrelation. The single DisCo training converged after 100 epochs while the double DisCo training required 200 epochs. Other networks only needed ten epochs. The double DisCo networks

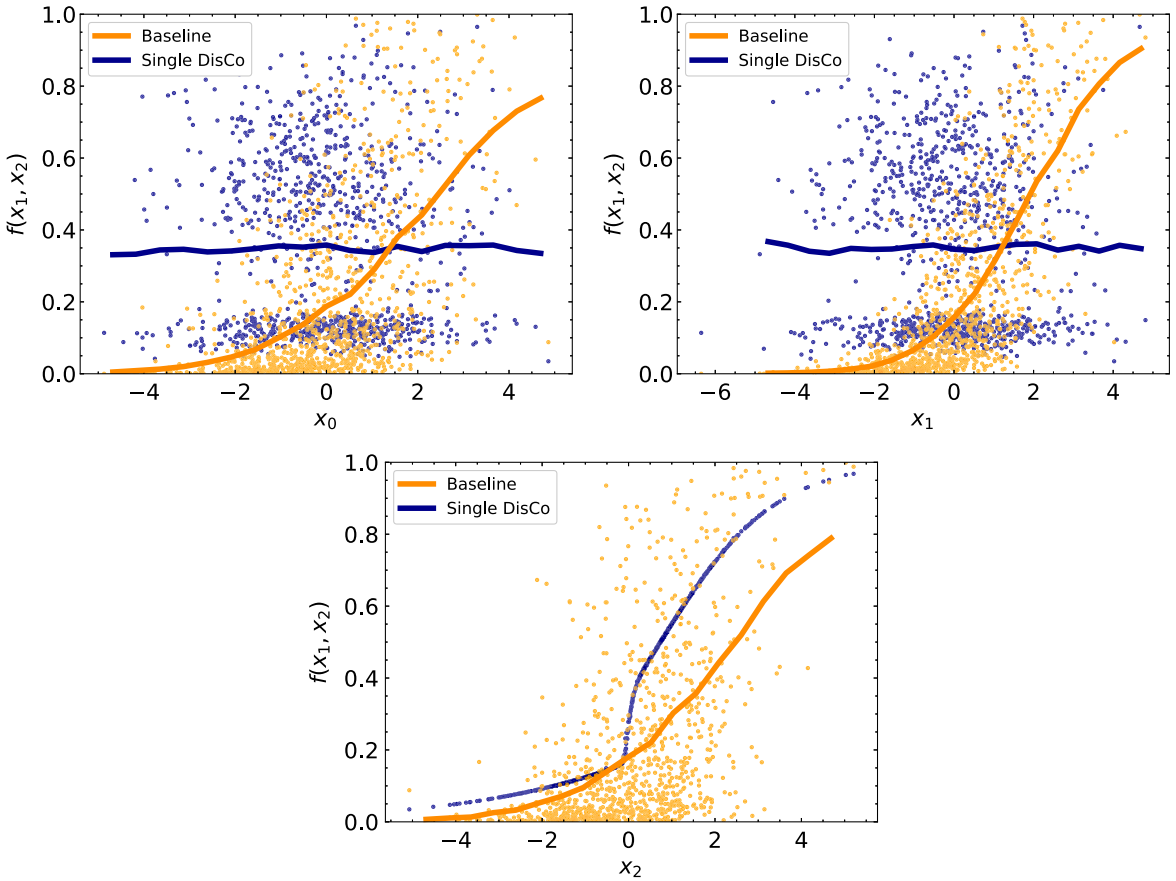

FIG. 3. Scatter plots showing the relationship (or lack thereof) between the three random variables X_0, X_1, and X_2 and (1) a baseline classifier $f_{\mathrm{BL}}(X_1, X_2)$ trained on X_1 and X_2 with no regularization, and (2) a classifier $f_{\mathrm{SD}}(X_1, X_2)$ trained with the single DisCo loss function that penalizes correlations with X_0. Only the background events are shown in these plots. The solid lines are the averages of the classifiers over events with the same value of X_0, X_1, or X_2. In the third panel, the scatter of the single DisCo classifier is already a line, so no average is needed.

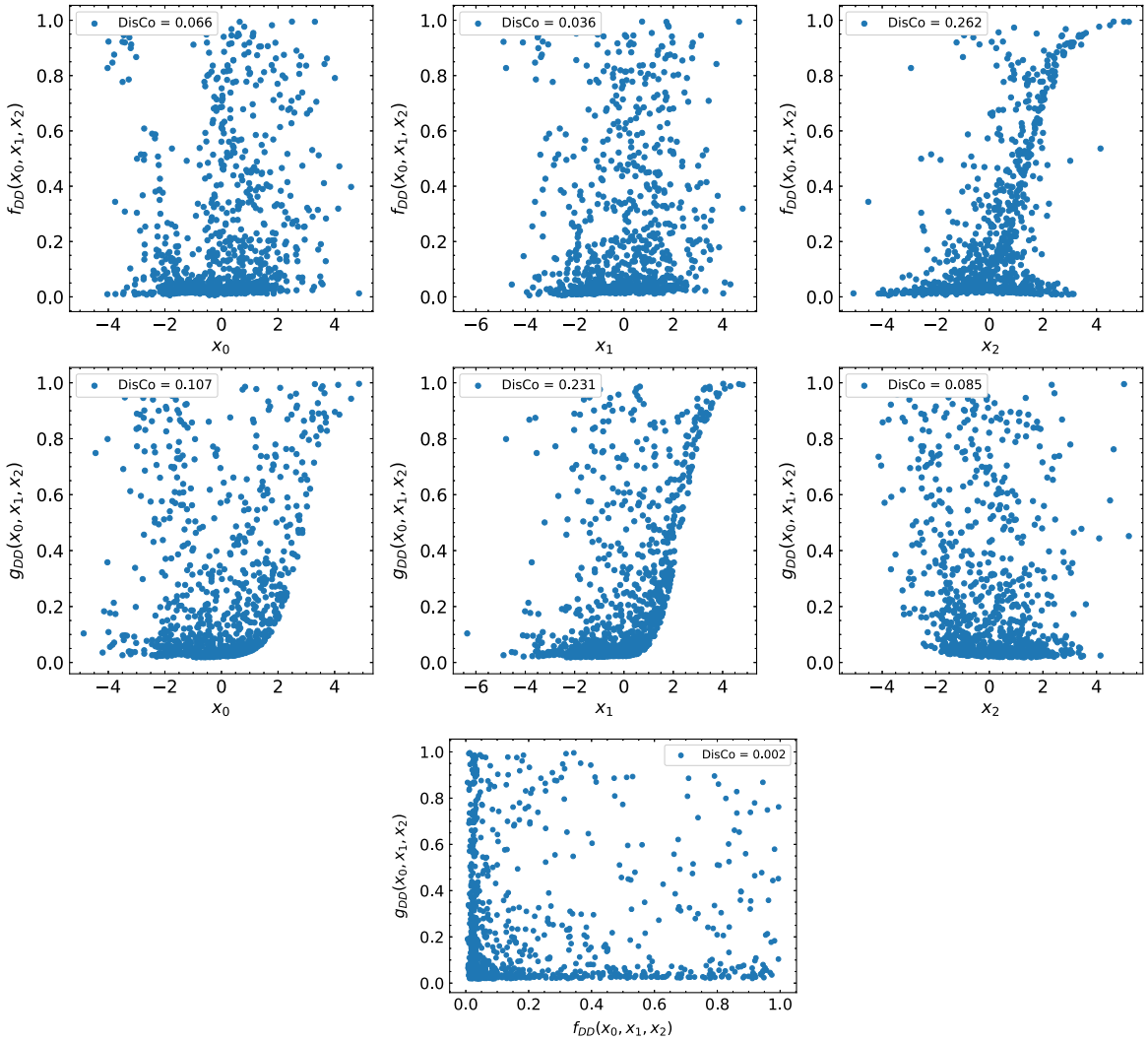

FIG. 4. Scatter plots showing the relationship between the three random variables X_0, X_1, X_2 and the two double DisCo neural networks f_{DD} and g_{DD} using only the background. The distance correlation between the two plotted observables is indicated in the legend.

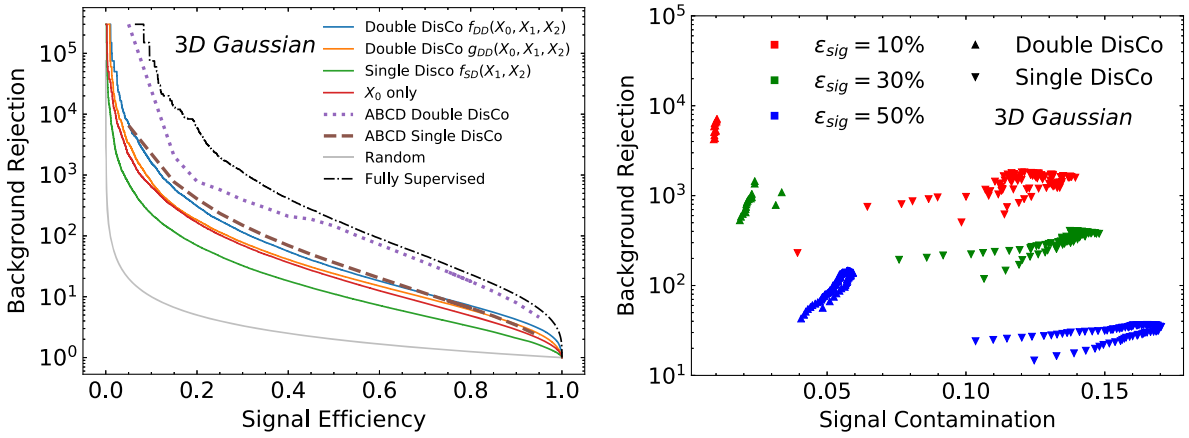

FIG. 5. Performance metrics for the Gaussian random variable model. Left: a receiver operating characteristic (ROC) curve. The lines marked ABCD DisCo are derived by scanning over rectangular thresholds on the two classifiers for points with ABCD closure within 10%. In the single DisCo case, one of the two classifiers is simply a NN trained with only X_0 (marked "X_0" only in the legend). Right: a scatter plot between background rejection and the normalized signal contamination for ABCD closure within 20%. For comparison, the left plot also shows the performance of the two double DisCo functions separately, the single DisCo function on its own, as well as a fully supervised classifier using all the available information all at once.

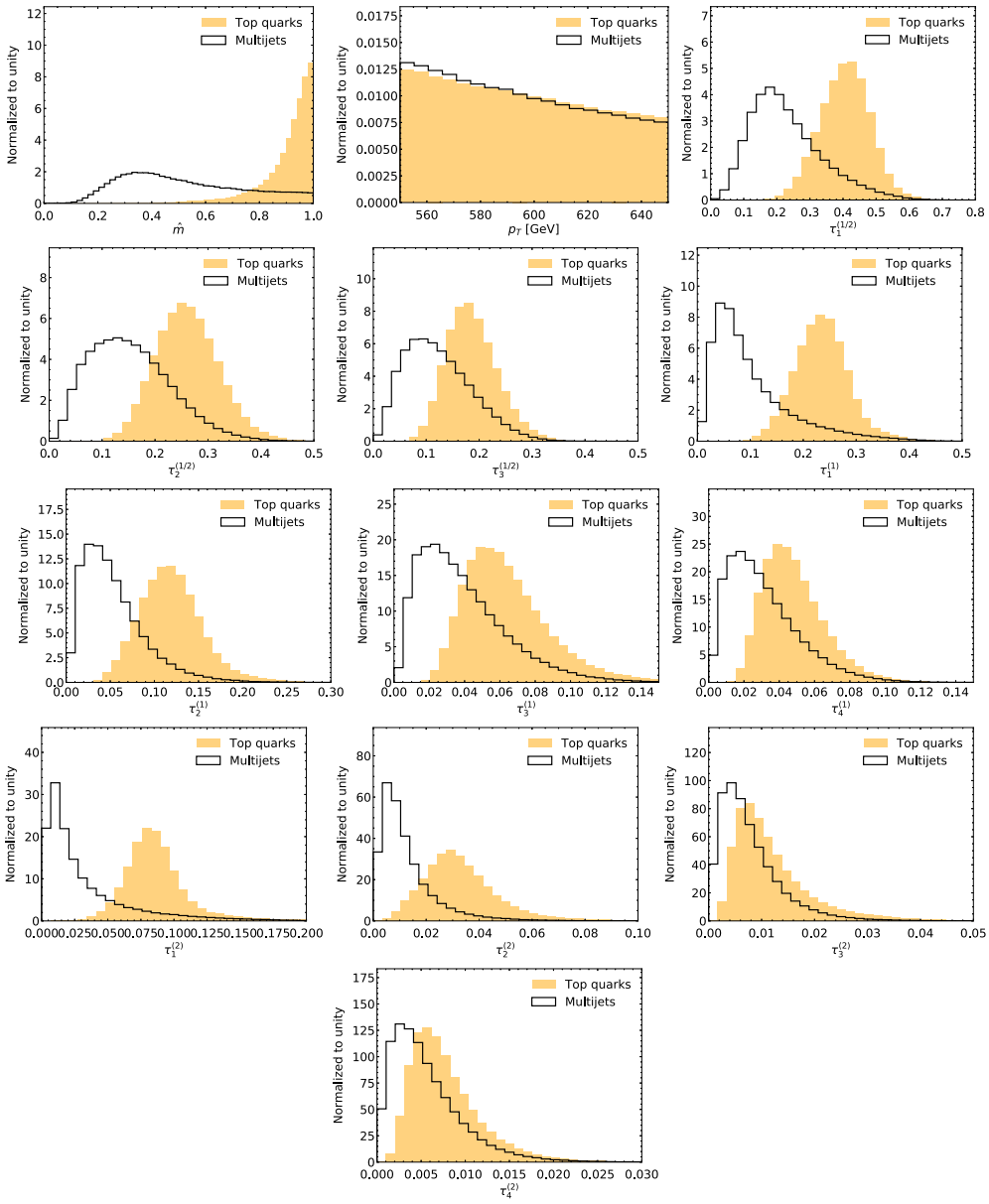

FIG. 6. The 13 features used for the boosted top analysis.

were trained using a single neural network with a two-dimensional output. All models were trained using Tensorflow [89] through Keras [90] with Adam [91] for optimization. Two million examples were generated with 15% used for testing. A batch size of 1% of the total was used for all networks to ensure an accurate calculation of the DisCo term in the relevant loss functions.

We first consider two classifiers: a baseline classifier $f_{\mathrm{BL}}(X_1, X_2)$ trained only on X_1 and X_2 and a single DisCo classifier $f_{\mathrm{SD}}(X_1, X_2)$ which includes a penalty for correlations between f_{SD} and X_0. The values of these classifiers for events drawn from the distributions are plotted in Fig. 3 against the X_0, X_1, or X_2 values of these events. We see that even though X_0 was not used in the training of the baseline, the classifier output is still correlated with X_0 because of the correlations between X_0 and X_1. In contrast to the baseline classifier, the single DisCo classifier is independent of both X_0 and X_1 and is simply a function of X_2. Intuitively, it makes sense that a classifier that must be independent of X_0 must also be independent of X_1. This is justified rigorously in Appendix B.

For double DisCo, we train two classifiers $f_{\mathrm{DD}}(X, Y, Z)$ and $g_{\mathrm{DD}}(X, Y, Z)$ according to the double DisCo loss function. The results are illustrated in Fig. 4. The first classifier depends mostly on Z and the second classifier depends mostly on X and Y. However, the residual dependence on all three observables is not a deficit of the training procedure: even though the three random variables are separable into two independent subsets (X, Y) and Z, the two classifiers learned by double DisCo

are nontrivial functions of all three variables. There is a large freedom in choosing the two functions f_{DD} and g_{DD} with a very small distance correlation and also excellent classification performance. Evidently, double DisCo prefers to partition the information differently than the naive partitioning in order to achieve better classification performance.

Figure 5 shows the performance of the single and double DisCo classifiers. The curve for the ABCD method is constructed by scanning 100 values of independent thresholds on the two features, evenly spaced in percentile of one classifier or the other to ensure a fixed signal efficiency. Above 50% signal efficiency, the ABCD double DisCo has nearly the same performance as the fully supervised classifier using all of the available information. The single DisCo performance is much lower than the double DisCo performance and is comparable to the best of the two double DisCo classifiers. The right plot of Fig. 5 demonstrates that double DisCo is not only more effective at a rejection background, but it also has a lower signal contamination.

B. Boosted tops

Next we turn to a physical example: boosted, hadronically decaying, tops. When top quarks are highly boosted, their hadronic decay products can be collimated into a single large jet and jet substructure methods are often necessary to distinguish them from QCD jet backgrounds [92]. One can estimate these backgrounds using sidebands in the jet mass around the top mass m_t. For the application of single and double DisCo, we will first reframe this estimation as an ABCD method and map mass to a variable where the signal peaks at 1 and the background peaks at a lower value:

$$\hat{m} \equiv 1 - \frac{|m_{\mathrm{jet}} - m_t|}{m_t}. \tag{4.3}$$

For our studies we will use the community top tagging comparison sample [13,46]. There are 2 million jets total, 1 million each of signal (top jets) and background (light quark and gluon QCD jets). Of these, half are used for training and the other half for validation.

We compute the following set of high level features suggested by [43]

$$\hat{m}, p_T, \tau_1^{1/2}, \tau_2^{1/2}, \tau_3^{1/2}, \tau_1^1, \tau_2^1, \tau_3^1, \tau_4^1, \tau_1^2, \tau_2^2, \tau_3^2, \tau_4^2. \tag{4.4}$$

Here, τ_N^a are the subjettiness variables introduced in [93,94] and are computed using FastJet [95]. This set of 13 variables is a complete basis for five-body phase space, and therefore it provides a complete description of the physics at the parton level at leading order [43,44,49,96]. It also offers a useful [96] feature space for modeling the top quark jets and inclusive jets after hadronization. Histograms of these features for signal and background are presented in Fig. 6. These features are not unique, but they offer a useful set for studying the performance of neural network-based taggers.

All the features are rescaled to be between 0 and 1. The neural network specification is three hidden layers of 64 nodes each, ReLU activations, and batch normalization after the first hidden layer. We train for 200 epochs with a fixed learning rate of 10^{-3} and the default Adam optimizer. We use a large batch size of 10k to ensure an accurate DisCo sampling estimate.

For single DisCo, we train a single neural network on just the subjettiness variables (we could have included $\hat{m}$ and p_T too with little change). For double DisCo, we train two neural networks on all the features ($\hat{m}$, p_T, and the subjettiness variables). The neural networks specifications, feature preprocessing, and training details are all the same for single and double DisCo. However, for double DisCo, in addition to the usual DisCo loss term described in Eq. (3.2), we include a *second* DisCo term which only takes the tail of the neural network outputs (again for background only) as inputs. This was found to help with the stability of the ABCD prediction for lower signal efficiencies, which can be sensitive to the extreme tails of the background. For the tail we required the simultaneous cuts of $y_1 > (y_1)_{\mathrm{bg},50}$ and $y_2 > (y_2)_{\mathrm{bg},50}$, where $y_{1,2}$ are the outputs of the two neural networks and "bg,50" refers to the 50th percentile cut on the background distributions.

For both single and double DisCo we have scanned over the following values of the DisCo parameter:

$$\lambda = 25, 50, 75, 100, 150, 200. \tag{4.5}$$

Values of λ larger than 200 tended to destabilize the training. We show in Fig. 7 the background rejection at

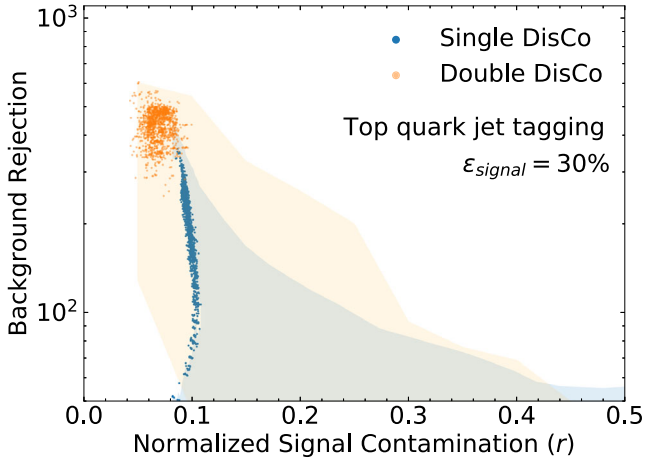

FIG. 7. A scatter plot of background rejection and normalized signal contamination (r) across DisCo parameters, epochs, and thresholds on the two features, for $\epsilon_{\mathrm{signal}} = 30\%$ and background ABCD closure better than 10%. High density regions are depicted with individual data points while low density regions are drawn as shaded regions.

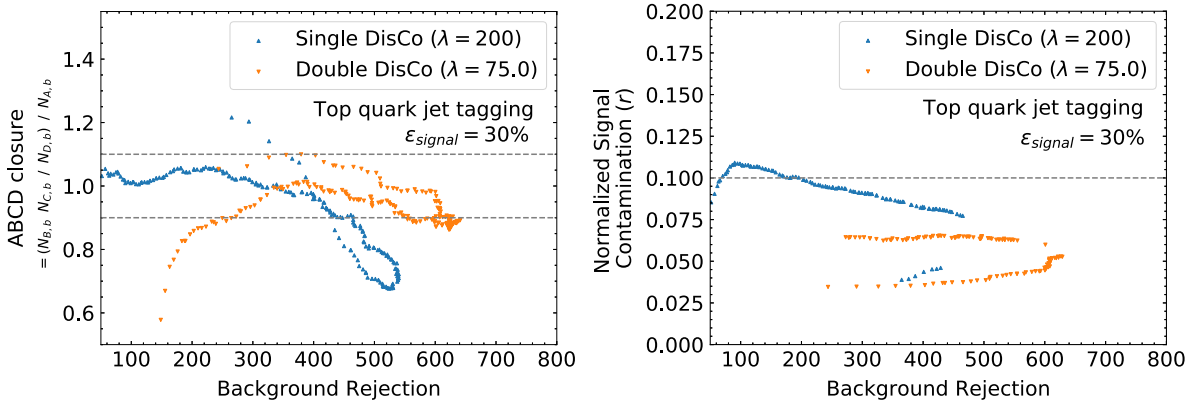

FIG. 8. Performance metrics for the boosted top analysis. Left: a scatter plot of the ABCD closure for the background versus the background rejection for $\epsilon_{\text{signal}} = 30\%$ in the boosted top analysis. Right: for the points in the left plot with ABCD closure within 10% of unity, this is a scatter plot of the normalized signal contamination (r) versus the background rejection. In both scatter plots, each point corresponds to a different rectangular cut on f and g, which we scan over keeping fixed the signal efficiency. The two branches in the plots correspond to tight cuts on either f or g, which is a feature of the rectangular nature of the cuts.

30% signal efficiency vs the normalized signal contamination r defined in Eq. (2.8), for every epoch, DisCo parameter, and value of rectangular cuts on the two classifiers that achieves the required signal efficiency (same method as Sec. IV A), subject only to the requirement that the ABCD closure for the background is accurate to within 10%: $|N_{A,b} - N_{A,b}^{\text{predicted}}| < 0.1$ [see Eq. (2.2)]. We see that double DisCo is able to achieve both higher background rejection and significantly lower signal contamination than single DisCo.

Figure 8 shows the "best" models for single DisCo and double DisCo, where best corresponds to an epoch and λ robustly reaches the upper left corner of Fig. 7. Here each point in the plot represents a choice of the rectangular cut that achieves 30% signal efficiency. We see that both single DisCo and double DisCo are able to achieve accurate ABCD closure and low signal contamination across a wide range of rectangular cuts.

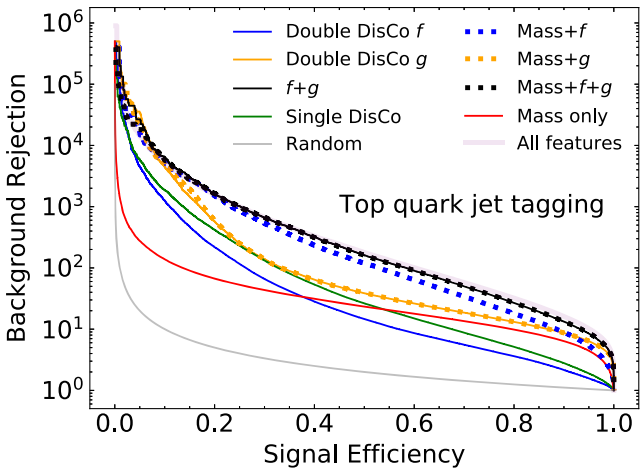

FIG. 9. ROC curve for the boosted top analysis. The background rejection is shown as a function of the signal efficiency for various combinations of single and double DisCo classifiers with or without mass.

Next we turn to the question of what did single and double DisCo learn—specifically how the available information was used by the individual NNs. Shown in Fig. 9 are a number of ROC curves. This includes ROC curves for mass, the individual classifiers in single and double DisCo, as well as additional NN classifiers obtained from training simple DNNs on various combinations of mass and NN1, NN2 from double DisCo.

A first observation is that one of the double DisCo classifiers (g) outperforms all the other individual classifiers without explicitly added mass information for all values of the signal efficiency. The next best performance is achieved by the single DisCo classifier, followed by the second one of the double DisCo classifiers (f).[3]

Jet mass by itself is very effective for loose selections (corresponding to a high signal efficiency). This can be understood from the good separation observed in Fig. 6 (top left). However, for tighter selections additional substructure information is needed.

Combining mass with one of the double DisCo classifiers (g) does not strongly alter its performance. This implies that the information contained in mass is learned by this NN. However, it clearly outperforms mass, meaning that g contains more features than just mass. On the other hand, combining mass with the weaker double DisCo classifier (f) dramatically improves it—it becomes almost, but not quite, optimal. This is to be expected as f is forced to be independent from g for background examples. If g contains mass completely, then f should be mostly

[3]Both f and g started with equivalent initial conditions, and their symmetry was spontaneously broken during network training.

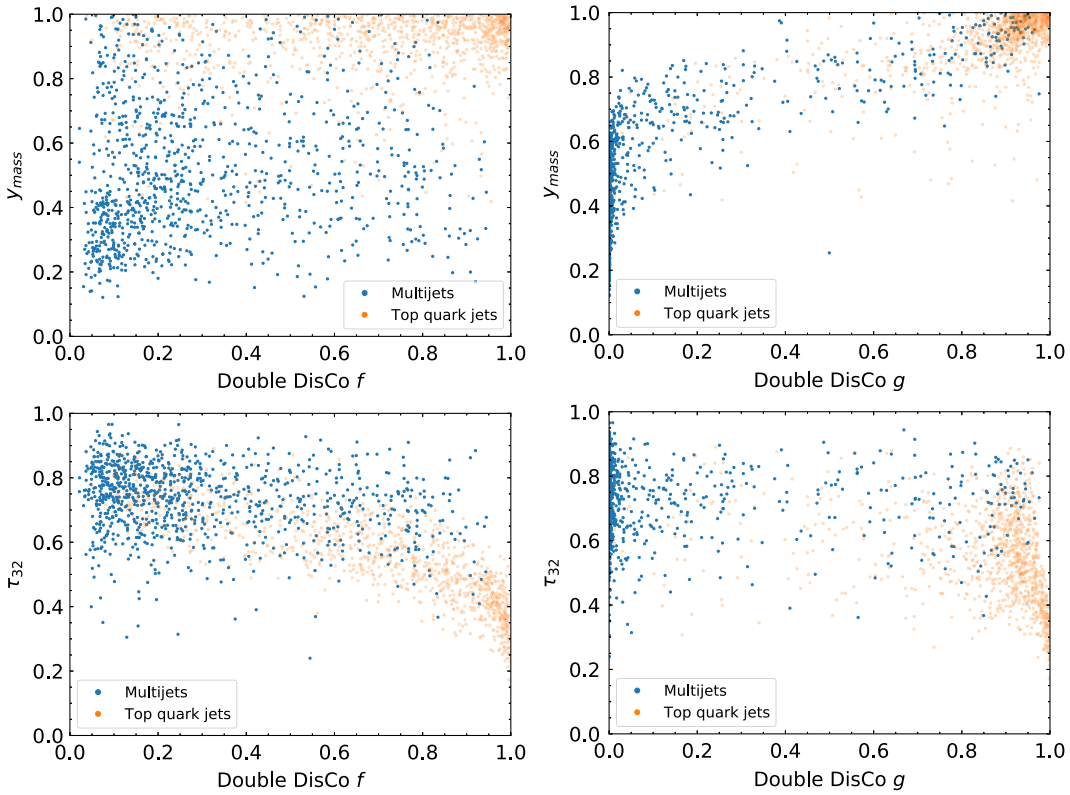

FIG. 10. Scatter plots of y_{mass} (top) and τ_{32} (bottom) with the double DisCo classifiers f (left) and g (right) in the boosted top analysis.

independent of mass, and adding it to f should result in a major performance boost.

Finally, there is no real difference between a combination of the two double DisCo classifiers $(f+g)$, a further combination also including the mass (mass $+ f + g$), and a direct training on all input features. This further confirms that the mass information has been fully absorbed by $f+g$—specifically g via the argument above. The maximally inclusive mass $+ f + g$ classifier of course should not be used as input to the ABCD method. However, we can compare its performance to results on the same dataset in Ref. [13]. A classifier based on multibody N-subjettiness trained following the procedure suggested in Ref. [96] achieved a background rejection of up to around $1/900$ for a signal efficiency of 30%. We observe a slightly weaker $1/700$ which is to be expected as a lower number of N-subjettiness observables is used as inputs here.

In the scatter plots of the double DisCo discriminators in Fig. 10, we again observe the larger discrimination power of g compared to f. Looking at the top left distribution, we indeed see no dependence of f on the mass while in the top right a clear correlation is there for g. On the other hand, in the bottom left, we see a trend between f and τ_{32} which encodes to which amount the jet is compatible with a three-prong substructure. This information is largely not learned by g.

We conclude that double DisCo can do better than single DisCo because it is partitioning the information differently than just mass versus everything else.

C. RPV SUSY

For our third example, we consider an actual "real-life" application of the ABCD method on LHC data: the $\sqrt{s} = 13$ TeV ATLAS search for paired dijet resonances [86]. Similar searches were conducted by CMS [87] and by both experiments at $\sqrt{s} = 8$ TeV [97,98]. These searches were motivated by pair production of identical squarks which each decay promptly to two jets via RPV couplings. For background estimation, these searches all used the standard ABCD method. In this section we will describe our recast of this search and the performance gains derived from training single and double DisCo on it.

The ATLAS search consisted of the following steps:

(i) *Preselection*: Events are required to have at least four jets with $p_T > 120$ GeV and $|\eta| < 2.4$. The leading four such jets are used to form two squark candidates based on nearest proximity in $\Delta R = \sqrt{(\Delta\phi)^2 + (\Delta\eta)^2}$. The minimum ΔR from the resulting pairings is defined as ΔR_{min} and the two dijet masses are used to form the average mass $m_{\text{avg}} = \frac{1}{2}(m_{\text{dijet 1}} + m_{\text{dijet 2}})$ and fractional mass

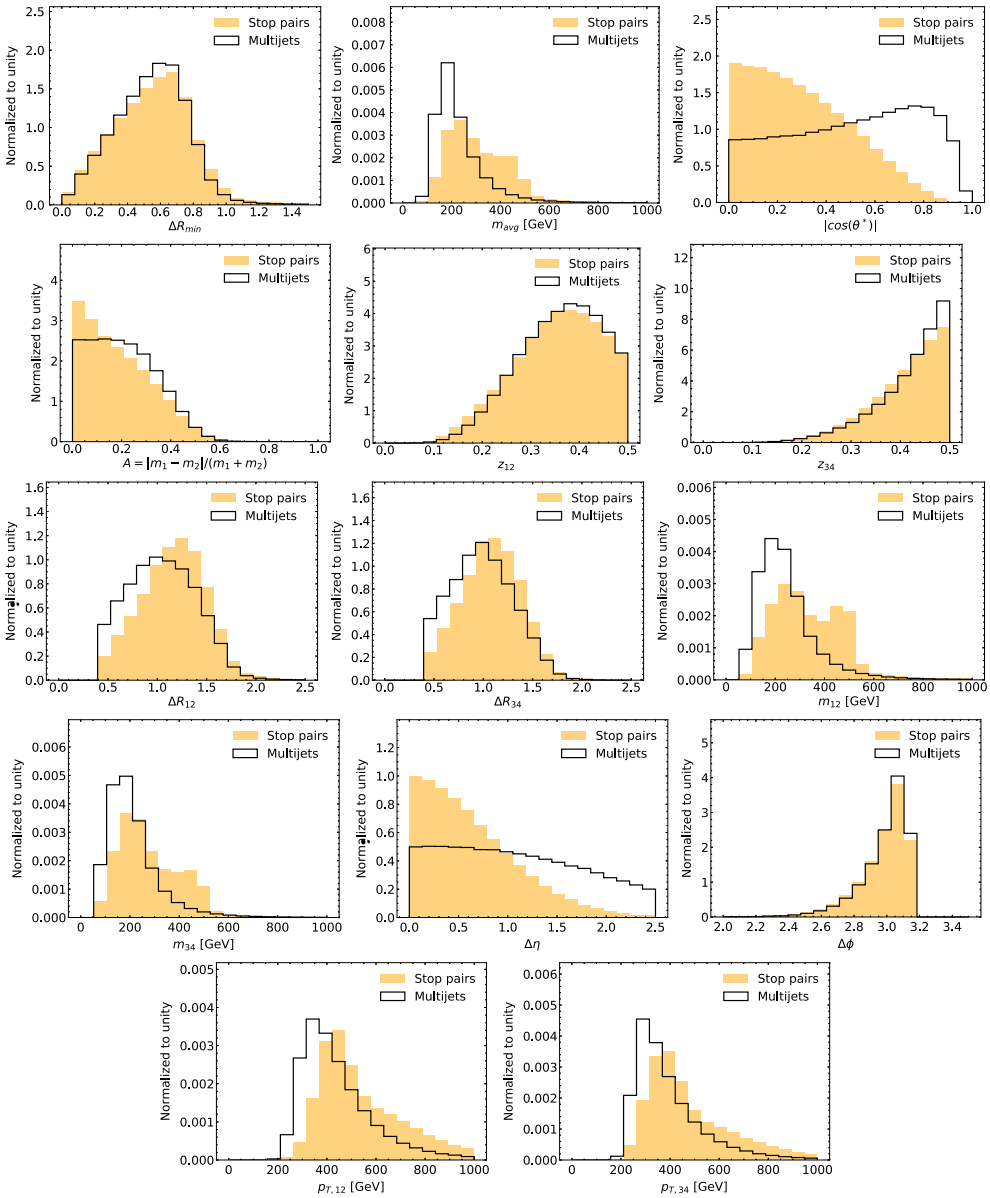

FIG. 11. The features used to train the RPV classification model.

asymmetry $A_{\text{mass}} = \frac{1}{m_{\text{avg}}}|m_{\text{dijet 1}} - m_{\text{dijet 2}}|$. Events with $m_{\text{avg}} < 255$ GeV must have $\Delta R_{\text{min}} < 0.72 - 0.002(m_{\text{avg}}/\text{GeV} - 255)$ and events with $m_{\text{avg}} \geq 255$ GeV must have $\Delta R_{\text{min}} < 0.72 - 0.0013(m_{\text{avg}}/\text{GeV} - 255)$.

(ii) *Final selection*: For the final selection, the ATLAS search performs counting experiments in successive windows of m_{avg}, and for background estimation uses the ABCD method in $|\cos\theta^*|$ and A_{mass}, where θ^* is the polar angle of one of the squarks in the squark-squark center-of-mass frame. The signal region is defined as $A_{\text{mass}} < 0.05$ and $|\cos\theta^*| < 0.3$.

ATLAS ended up setting a limit at approximately $m_{\text{squark}} = 500$ GeV, so we will also focus our analysis on this value of the squark mass. We repeat the preselection cuts but instead of the final selection on m_{avg}, A_{mass}, and $\cos\theta^*$, we instead feed a list of inputs to single and double DisCo to learn the optimal features. The inputs are

$$\Delta R_{\text{min}}, m_{\text{avg}}, \cos\theta^*, A_{\text{mass}}, z_{12}, z_{34}, \Delta R_{12}, \Delta R_{34}, m_{12}, m_{34}, \Delta\eta, \Delta\phi, p_{T,12}, p_{T,34}, \tag{4.6}$$

where z_{12} (z_{34}), ΔR_{12} (ΔR_{34}), m_{12} (m_{34}), $p_{T,12}$ ($p_{T,34}$) are the p_T of the subleading jet divided by the sum of the transverse momenta of both jets, the opening angle between the two jets, the invariant mass of the two jets, and the p_T of the two jets for the stop dijet pair with the leading small-radius jet (and the other stop dijet pair), respectively. Histograms of these features are shown in Fig. 11. All features are rescaled to the range [0, 1] before feeding to the

NNs. For single DisCo we use $\cos\theta^*$ rather than A_{mass} as the fixed variable X_0 ($\cos\theta^*$ is the stronger of these two features from the ATLAS RPV squark analysis) and feed everything else to the NN classifier. For double DisCo we feed everything to the two NN classifiers.

Squark pair events and multijet events are generated with PYTHIA 8.230 [99,100] at a center-of-mass-energy of $\sqrt{s} = 13$ TeV interfaced with DELPHES 3.4.1 [101] using the default CMS run card. Jets are clustered using the anti-k_t algorithm [102] with radius parameter $R = 0.4$ implemented in FastJet 3.2.1 [95,103]. The 1M signal events and 10M background events were generated, of which about 100k signal events and 60k background events pass the preselection. In order to ensure a high event selection efficiency for the background, events are generated using $2 \to 3$ matrix elements with a minimum separation of $R = 0.8$ and minimum $\hat{p}_T$ of 100 GeV for the softest parton and 200 GeV for the hardest parton. Signal events are produced using the SLHA [104,105] card from the recent ATLAS search [86,106] in which the squark mass is 500 GeV and all other super partners are decoupled.

The validation of the recasting of the ATLAS analysis is shown in Tables I and II. In the former we show the relative signal efficiencies after successive cuts. In the latter we show the relative fractions f_i (since we do not attempt to get the overall normalizations of our simulations correct) of data in ATLAS regions $i = D, A, F, C$ (for ATLAS D is the signal region (SR)); and the signal-to-background ratio δ_i in each region. Following ATLAS, for the data fractions, the counts are taken after the inclusive selection with no mass window cut, while for the signal-to-background ratios they are taken after the mass window cut. Overall, we see excellent agreement between the ATLAS numbers and our recasted numbers.

For training the NNs, we use 100k signal and 360k background events, while the validation sample consists of 25k signal and 250k background events. In the classifier loss, we rebalance the signal and background contributions as if they were 50/50.

We used the same hyperparameters as the top tagging example. (We also explored using 128 nodes per hidden layer but found that it did not help.) For DisCo parameters we chose

$$\lambda = 10, 20, 30, 40, 60, 100. \tag{4.7}$$

Unlike the top tagging example we do not add the additional DisCo term sensitive to the tails of the background distributions when training double DisCo; because the background rejections in this case were not as high as for top tagging, the additional term was found not to help.

The comparison of single and double DisCo is shown in Fig. 12. As in the top tagging section, we have plotted every epoch and every rectangular cut and every value of the disco parameter satisfying the 10% accuracy condition on the ABCD prediction. This shows the performance of the models in the plane of R_{10} (background rejection factor at 10% signal efficiency) vs total fractional signal contamination. We see that while double DisCo cannot surpass single DisCo in terms of raw performance (as measured by R_{10}), it can achieve dramatically lower signal contamination for roughly the same R_{10}.

We have also included scans over the features used in the ABCD method as used in the ATLAS RPV search, these are the green points in Fig. 12.[4] We note that ATLAS had significant normalized signal contamination with their

TABLE I. Relative efficiencies for each cut on the signal in the ATLAS RPV supersymmetry (SUSY) search and our recast.

Cut	ATLAS	Our recast
ΔR_{min}	13.0%	11.9%
Inclusive SR	10.2%	9.5%
Mass window	25%	23.3%

TABLE II. Relative fractions f_i of data in the regions $i = D, A, F$, and C used in the ATLAS RPV SUSY analysis and our recast, and signal-to-background ratios δ_i in each region.

	ATLAS		Our recast	
Region i	f_i	δ_i	f_i	δ_i
D (SR)	6.8%	6.3%	6.4%	6.3%
A	11.4%	3.1%	10.5%	3.4%
F	30.7%	0.2%	31.6%	0.3%
C	51.1%	0.07%	51.6%	0.2%

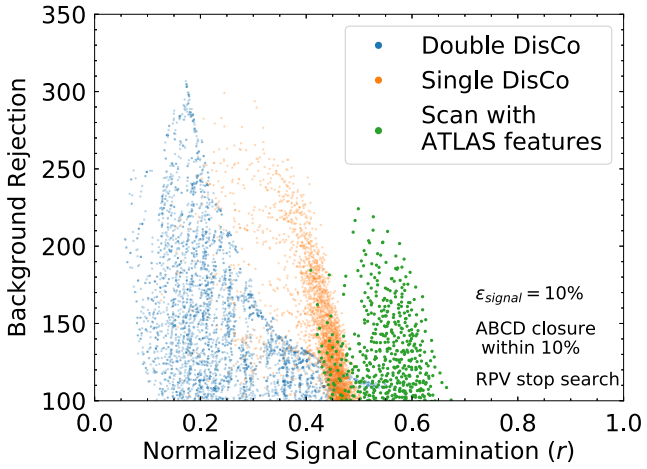

FIG. 12. A scatter plot of background rejection versus normalized signal contamination (r) in the RPV SUSY analysis for various epochs with single and double DisCo as well as a scan of three-dimensional thresholds on the features used by the ATLAS analysis.

[4]The actual ATLAS analysis used a working point that corresponds to about 2.5% signal efficiency. We found this to be suboptimal to a 10% value, which is why it is used in Fig. 12.

selection (40%–80%), which may result in a significant bias in the p-value (see Fig. 2).

Both single and double DisCo offer a marked improvement in both signal contamination and background rejection compared to the standard ABCD method with manually chosen high-level features. To gauge the impact of the improvements, we offer the following quantitative arguments:

(i) *Improved background rejection:* For the RPV stop example presented in Sec. IV C, the systematic uncertainty reported in Ref. [86] is about 3%. Our recasted p-value is 0.067, consistent with the fact that the 500 GeV stop is almost excluded in Ref. [86]. The enhanced signal rejection from Fig. 12 from double DisCo is about 20% at an r-value of 0.1. This would reduce the p-value to 0.041 when accounting for the same 3% uncertainty. Achieving this p-value without Double DisCo would require 2.5 times more data, and therefore the potential gain is significant.

(ii) *Reduced signal contamination:* If we use the numbers from the RPV stop search, we find that accounting for $r \sim 0.5$ would result in a p-value that is ≈ 0.2. We find that no amount of data would allow this 500 GeV point to be excluded with this value, accounting for the 3% systematic uncertainty (see Fig. 2). However, if $r \sim 0.1$, as with double DisCo, and the rejection is about 20% better, then the point would eventually be excluded with 25% more collisions.

V. DISCUSSION

The examples from the previous section have shown that single DisCo and double DisCo are able to effectively increase the discrimination power over traditional methods while also maintaining a low relative signal contamination. This section briefly discusses two interrelated features of these approaches connected to residual model dependence and systematic uncertainties.

One of the main goals of the ABCD method is to provide a data-driven background estimation strategy with minimal dependence on the background and signal model. By training classifiers with background and signal simulations, we are explicitly introducing model dependence. Mismodeled features will not result in a bias as long as the correlations are properly modeled.[5] Furthermore, this is exactly the same challenge that faces the classical ABCD method. An uncertainty is often determined by using an alternative background simulation and/or a nearby region in data. The same strategies could be applied for the automated ABCD methods using machine learning. Larger relative uncertainties may be tolerable for the automated methods if there is little absolute nonclosure.

For the signal model, there is a dependence in two ways. First, the automated decorrelation needs to be trained per signal model. It is likely that there is sensitivity for similar signals, but an analysis that scans over signal model parameters would likely need to train multiple models. Each of these models could correspond to a signal region. One may be able to extend this with parametrized networks [109,110]. The second aspect that relies on signal modeling is correcting for signal in the nonsignal regions. In many analyses, this is assumed small and is not modified. As we have shown, this is only valid when the relative signal contamination is negligible. Both of the automated approaches significantly improve the relative signal contamination, and so the size of this effect is significantly reduced for these methods compared with the traditional ABCD approach.

VI. CONCLUSIONS

Estimating backgrounds is essential for every experimental analysis in particle physics. One of the most well-established data-driven techniques for background estimation is the ABCD method. In this paper we have reexamined the criteria for the ABCD method to be effective and proposed a way to find the variables used to establish the ABCD regions using machine learning.

A general observation we make in this paper is that the signal contamination in the background region *normalized* to the signal fraction in the signal region drives the quality of the ABCD background estimate. This observation is independent of any machine-learning approaches to determining the features. We argue that controlling this normalized signal contamination should become a default procedure in applying the ABCD method, since neglecting it can lead to incorrect, and typically overly conservative, p-values.

Regardless of how one estimates contamination of the background samples, a necessary condition for the ABCD method to work is the availability of two independent classifiers. These classifiers are usually found by guessing observables that, on physical grounds, seem like they would be independent, and then verifying their independence with simulations or validation regions. Such a procedure is by no means guaranteed to yield optimal results. Indeed, observables either designed for classification by hand or learned by machine easily have better discrimination power than observables chosen to be independent. However, optimal observables aim to make maximum use of available information and will in general exhibit complex dependencies with all other observables.

[5]For example, the correlation between p_T and the number of particles inside jets can be described in perturbation theory and thus is known precisely [107,108] while the spectrum of the number of particles itself is non-perturbative and cannot be calculated from first principles.

In this paper, we proposed to use machine learning methodology to optimize the ABCD method. We considered two use cases: (1) single DisCo, where a first variable (such as mass) is fixed and another is learned to be decorrelated with it and optimize discrimination, and (2) double DisCo, where both variables are learned. For both methods, our machine learning approach builds upon the DisCo loss term, a recently developed method for automated decorrelation. This technique allows for the autonomous construction of a robust data-driven background estimation assuming a specific signal model.

We considered three examples: (1) a simple model of correlated random variables that demonstrates how single and double DisCo work, (2) boosted top tagging, and (3) an RPV squark search, based on an existing ATLAS analysis. We found that while single DisCo offers competitive performance in terms of pure background rejection, double DisCo achieves lower signal contamination levels in both of the physical examples considered. We note that while DisCo was used to demonstrate decorrelation in this paper, the general idea can be combined with any decorrelation method [66–81] and the best approach may be application specific.

On the surface, one advantage of the traditional ABCD method has over the proposed automated approaches is that it is largely signal model independent. However, even there, it is necessary to explicitly verify low signal contamination for all considered models using simulations. On the other hand, the training of single DisCo or double DisCo can be extended to a cocktail of signal models or parametrized as a function of the considered signal [109,110].

While the single and double DisCo approaches achieve excellent performance, even better sensitivity might be obtained by optimizing the necessary criteria of low signal contamination and good ABCD closure more directly. We argued in earlier sections that the single and double DisCo loss qualitatively capture these requirements, but direct optimization of the conditions is challenging as they cannot be readily cast in a differentiable form. One might, for example, try an iterated learning approach or one based on reinforcement learning, where the final p-value for ABCD searches is used as a score. Further studies in this direction are left to future work.

Finally, it is important to consider the task of background estimation in the broader context of analysis optimization. A variety of methods have been proposed to directly optimize analysis sensitivity including uncertainty [111–114]. Background estimation is a key part of analysis design and could be integrated into the ABCD method in order to further optimize the overall discovery potential. An orthogonal approach is to construct searches for new physics in a model independent way [14,80,115–138]. Such searches will also require robust and automated data-driven background predictions and—at least partially—can be trained with a single or double DisCo method.

In summary, we are able to increase the discovery potential of physics analyses by enabling robust background estimates for more powerful classifiers. This improvement is made possible by clearly defining the objectives and then using automated tools to optimize a parametric function to achieve them. The present work shows that even time-tested and widely deployed analysis methods can benefit from systematic optimization.

ACKNOWLEDGMENTS

We thank Alejandro Gomez Espinosa and Simone Pagan Griso for useful discussions and Simone for additionally providing feedback on the manuscript. We thank Olaf Behnke and Thomas Junk for helpful comments on the manuscript and especially for examples of early uses of the ABCD method. B. N., M. S., and D. S. were supported by the U.S. Department of Energy, Office of Science under Contracts No. DE-AC02-05CH11231, No. DE-SC0013607, and No. DOE-SC0010008, respectively. B. N. also thanks NVIDIA for providing Volta GPUs for neural network training. G. K. acknowledges support by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Germany's Excellence Strategy EXC 2121 "Quantum Universe" 390833306. D. S. is grateful to LBNL, BCTP, and BCCP for their generous support and hospitality during his sabbatical year.

APPENDIX A: DISTANCE CORRELATION

For two random variables f and g, the distance covariance is defined as

$$\begin{aligned}\mathrm{dCov}^2[f,g] = &\langle |f-f'| \times |g-g'| \rangle \\ &+ \langle |f-f'| \rangle \times \langle |g-g'| \rangle \\ &- 2\langle |f-f'| \times |g-g''| \rangle, \end{aligned} \tag{A1}$$

where (f,g), (f',g'), (f'',g'') are all independent and identically distributed from the same joint distribution. In practice, we evaluate $\mathrm{dCov}^2[f,g]$ by averaging $|f_i - f_j| \times |g_i - g_j|$, $|f_i - f_j|$, and $|g_i - g_j|$ over all pairs of events i, j, and $|f_i - f_j| \times |g_i - g_k|$ over all triplets of events i, j, k.

The distance correlation is then defined analogously to the usual correlation:

$$\mathrm{dCorr}^2[f,g] = \frac{\mathrm{dCov}^2[f,g]}{\mathrm{dCov}[f,f]\mathrm{dCov}[g,g]}. \tag{A2}$$

APPENDIX B: SINGLE DISCO IN THE GAUSSIAN CASE

In Sec. IV A, we observed that for the simple Gaussian model with three Gaussian random variables X_0, X_1, and X_2, the single DisCo classifier $f(X_1, X_2)$ trained to be independent of X_0 (which is correlated with X_1 but not X_2)

is only a function of X_2 and does not depend on X_1. The purpose of this Appendix is to prove this.

We start by rotating from (X_0, X_1, X_2) into another set of three Gaussian random variables that are mutually independent: X_0, W, and X_2 with $X_1 = \alpha X_0 + \beta W$, where α, β depend on ρ_b and W is independent from (X_0, X_2). Then, we can also write $h(X_0, W, X_2) = f(\alpha X_0 + \beta W, X_2)$. Let $Q = (W, X_2)$. Suppose that $h(X_0, Q)$ and X_0 are independent. Then for all sets A and B,

$$\Pr[h(X_0, Q) \in A \text{ and } X_0 \in B] = \Pr[h(X_0, Q) \in A] \times \Pr[X_0 \in B]. \tag{B1}$$

For any B, define $A_B = \{h(x_0, q) : x_0 \in B, \forall\, q\}$. Then, the probability that $h(X_0, Q) \in A_B$ given $X_0 \in B$ is unity,

$$\begin{aligned}&\Pr[h(X_0, Q) \in A_B \text{ and } X_0 \in B] \\ &= \Pr[h(X_0, Q) \in A_B | X_0 \in B] \times \Pr[X_0 \in B] \\ &= \Pr[X_0 \in B],\end{aligned} \tag{B2}$$

and so Eq. (B1) simply reduces to $\Pr[h(X_0, Q \in A_B] = 1$. This means that $h(x_0, q)$ cannot depend on x_0. Therefore, we conclude that if $h(X_0, Q)$ and X_0 are independent, then h does not depend on X_0. The only way for h to not depend on X_0 is for f to not depend on X_1.

[1] CDF Collaboration, A measurement of $\sigma B(W \to e\nu)$ and $\sigma B(Z^0 \to e^+e-)$ in $\bar{p}p$ collisions at $\sqrt{s} = 1800$ GeV, Phys. Rev. D **44**, 29 (1991).

[2] ATLAS Collaboration, Search for new phenomena in final states with large jet multiplicities and missing transverse momentum using $\sqrt{s} = 13$ TeV proton-proton collisions recorded by ATLAS in Run 2 of the LHC, J. High Energy Phys. 10 (2020) 062.

[3] CMS Collaboration, Search for supersymmetry in pp collisions at $\sqrt{s} = 13$ TeV with 137 fb^{-1} in final states with a single lepton using the sum of masses of large-radius jets, Phys. Rev. D **101**, 052010 (2020).

[4] ATLAS Collaboration, Exotic physics searches, 2018, https://twiki.cern.ch/twiki/bin/view/AtlasPublic/ExoticsPublicResults.

[5] ATLAS Collaboration, Supersymmetry searches, 2018, https://twiki.cern.ch/twiki/bin/view/AtlasPublic/SupersymmetryPublicResults.

[6] CMS Collaboration, CMS exotica public physics results, 2018, https://twiki.cern.ch/twiki/bin/view/CMSPublic/PhysicsResultsEXO.

[7] CMS Collaboration, CMS supersymmetry physics results, 2018, https://twiki.cern.ch/twiki/bin/view/CMSPublic/PhysicsResultsSUS.

[8] CMS Collaboration, CMS beyond-two-generations (B2G) public physics results, 2018, https://twiki.cern.ch/twiki/bin/view/CMSPublic/PhysicsResultsB2G.

[9] ATLAS Collaboration, Search for Higgs Boson Decays into a Z Boson and a Light Hadronically Decaying Resonance Using 13 TeV pp Collision Data from the ATLAS Detector, Phys. Rev. Lett. **125**, 221802 (2020).

[10] S. Choi and H. Oh, Improved extrapolation methods of data-driven background estimation in high-energy physics, arXiv:1906.10831.

[11] A. J. Larkoski, I. Moult, and B. Nachman, Jet substructure at the large hadron collider: A review of recent advances in theory and machine learning, Phys. Rep. **841**, 1 (2020).

[12] D. Guest, K. Cranmer, and D. Whiteson, Deep learning and its application to LHC physics, Annu. Rev. Nucl. Part. Sci. **68**, 161 (2018).

[13] G. Kasieczka, T. Plehn *et al.*, The machine learning landscape of top taggers, SciPost Phys. **7**, 014 (2019).

[14] HEP ML Community, A living review of machine learning for particle physics, https://iml-wg.github.io/HEPML-LivingReview/.

[15] J. Cogan, M. Kagan, E. Strauss, and A. Schwartzman, Jet-images: Computer vision inspired techniques for jet tagging, J. High Energy Phys. 02 (2015) 118.

[16] L. G. Almeida, M. Backović, M. Cliche, S. J. Lee, and M. Perelstein, Playing tag with ANN: Boosted top identification with pattern recognition, J. High Energy Phys. 07 (2015) 086.

[17] L. de Oliveira, M. Kagan, L. Mackey, B. Nachman, and A. Schwartzman, Jet-images deep learning edition, J. High Energy Phys. 07 (2016) 069.

[18] ATLAS Collaboration, Quark versus gluon jet tagging using jet images with the ATLAS detector, CERN Tech. Report No. ATL-PHYS-PUB-2017-017, 2017, http://cds.cern.ch/record/2275641.

[19] J. Lin, M. Freytsis, I. Moult, and B. Nachman, Boosting $H \to b\bar{b}$ with machine learning, J. High Energy Phys. 10 (2018) 101.

[20] P. T. Komiske, E. M. Metodiev, B. Nachman, and M. D. Schwartz, Learning to classify from impure samples with high-dimensional data, Phys. Rev. D **98**, 011502 (2018).

[21] J. Barnard, E. N. Dawe, M. J. Dolan, and N. Rajcic, Parton shower uncertainties in jet substructure analyses with deep neural networks, Phys. Rev. D **95**, 014018 (2017).

[22] P. T. Komiske, E. M. Metodiev, and M. D. Schwartz, Deep learning in color: Towards automated quark/gluon jet discrimination, J. High Energy Phys. 01 (2017) 110.

[23] G. Kasieczka, T. Plehn, M. Russell, and T. Schell, Deep-learning top taggers or the end of QCD?, J. High Energy Phys. 05 (2017) 006.

[24] S. Macaluso and D. Shih, Pulling out all the tops with computer vision and deep learning, J. High Energy Phys. 10 (2018) 121.

[25] T. Q. Nguyen, D. Weitekamp, D. Anderson, R. Castello, O. Cerri, M. Pierini, M. Spiropulu, and J.-R. Vlimant,

Topology classification with deep learning to improve real-time event selection at the LHC, Comput. Softw. Big Sci. **3**, 12 (2019).
[26] ATLAS Collaboration, Convolutional neural networks with event images for pileup mitigation with the ATLAS detector, CERN Tech. Report No. ATL-PHYS-PUB-2019-028, 2019, http://cds.cern.ch/record/2684070.
[27] M. Andrews, M. Paulini, S. Gleyzer, and B. Poczos, End-to-end physics event classification with CMS open data: Applying image-based deep learning to detector data for the direct classification of collision events at the LHC, Comput. Softw. Big Sci. **4**, 6 (2020).
[28] D. Guest, J. Collado, P. Baldi, S.-C. Hsu, G. Urban, and D. Whiteson, Jet flavor classification in high-energy physics with deep neural networks, Phys. Rev. D **94**, 112002 (2016).
[29] G. Louppe, K. Cho, C. Becot, and K. Cranmer, QCD-aware recursive neural networks for jet physics, J. High Energy Phys. 01 (2019) 057.
[30] T. Cheng, Recursive neural networks in quark/gluon tagging, Comput. Softw. Big Sci. **2**, 3 (2018).
[31] I. Henrion, K. Cranmer, J. Bruna, K. Cho, J. Brehmer, G. Louppe, and G. Rochette, Neural Message Passing for Jet Physics (2017), https://dl4physicalsciences.github.io/files/nips_dlps_2017_29.pdf.
[32] X. Ju *et al.*, Graph neural networks for particle reconstruction in high energy physics detectors, arXiv:2003.11603.
[33] J. Arjona Martnez, O. Cerri, M. Pierini, M. Spiropulu, and J.-R. Vlimant, Pileup mitigation at the large hadron collider with graph neural networks, Eur. Phys. J. Plus **134**, 333 (2019).
[34] E. A. Moreno, O. Cerri, J. M. Duarte, H. B. Newman, T. Q. Nguyen, A. Periwal, M. Pierini, A. Serikova, M. Spiropulu, and J.-R. Vlimant, JEDI-net: A jet identification algorithm based on interaction networks, Eur. Phys. J. C **80**, 58 (2020).
[35] S. R. Qasim, J. Kieseler, Y. Iiyama, and M. Pierini, Learning representations of irregular particle-detector geometry with distance-weighted graph networks, Eur. Phys. J. C **79**, 608 (2019).
[36] A. Chakraborty, S. H. Lim, and M. M. Nojiri, Interpretable deep learning for two-prong jet classification with jet spectra, J. High Energy Phys. 07 (2019) 135.
[37] A. Chakraborty, S. H. Lim, M. M. Nojiri, and M. Takeuchi, Neural network-based top tagger with two-point energy correlations and geometry of soft emissions, J. High Energy Phys. 07 (2020) 111.
[38] M. Abdughani, D. Wang, L. Wu, J. M. Yang, and J. Zhao, Probing triple Higgs coupling with machine learning at the LHC, arXiv:2005.11086.
[39] E. Bernreuther, T. Finke, F. Kahlhoefer, M. Krämer, and A. Mück, Casting a graph net to catch dark showers, arXiv:2006.08639.
[40] P. T. Komiske, E. M. Metodiev, and J. Thaler, Energy flow networks: Deep sets for particle jets, J. High Energy Phys. 01 (2019) 121.
[41] H. Qu and L. Gouskos, ParticleNet: Jet tagging via particle clouds, Phys. Rev. D **101**, 056019 (2020).
[42] K. Datta, A. Larkoski, and B. Nachman, Automating the construction of jet observables with machine learning, Phys. Rev. D **100**, 095016 (2019)..
[43] K. Datta and A. Larkoski, How much information is in a jet?, J. High Energy Phys. 06 (2017) 073.
[44] K. Datta and A. J. Larkoski, Novel jet observables from machine learning, J. High Energy Phys. 03 (2018) 086.
[45] P. T. Komiske, E. M. Metodiev, and J. Thaler, Energy flow polynomials: A complete linear basis for jet substructure, J. High Energy Phys. 04 (**2018**) 013.
[46] A. Butter, G. Kasieczka, T. Plehn, and M. Russell, Deep-learned top tagging with a Lorentz layer, SciPost Phys. **5**, 028 (2018).
[47] Y.-C. J. Chen, C.-W. Chiang, G. Cottin, and D. Shih, Boosted W and Z tagging with jet charge and deep learning, Phys. Rev. D **101**, 053001 (2020).
[48] K. Fraser and M. D. Schwartz, Jet charge and machine learning, J. High Energy Phys. 10 (2018) 093.
[49] K. Datta, A. Larkoski, and B. Nachman, Automating the construction of jet observables with machine learning, Phys. Rev. D **100**, 095016 (2019).
[50] E. A. Moreno, T. Q. Nguyen, J.-R. Vlimant, O. Cerri, H. B. Newman, A. Periwal, M. Spiropulu, J. M. Duarte, and M. Pierini, Interaction networks for the identification of boosted $H \to b\bar{b}$ decays, Phys. Rev. D **102**, 012010 (2020).
[51] M. Stoye, J. Kieseler, M. Verzetti, H. Qu, L. Gouskos, and A. Stakia (CMS Collaboration), DeepJet: Generic physics object based jet multiclass classification for LHC experiments (2017), https://dl4physicalsciences.github.io/files/nips_dlps_2017_10.pdf.
[52] Y.-T. Chien and R. Kunnawalkam Elayavalli, Probing heavy ion collisions using quark and gluon jet substructure, arXiv:1803.03589.
[53] G. Kasieczka, N. Kiefer, T. Plehn, and J. M. Thompson, Quark-gluon tagging: Machine learning vs detector, SciPost Phys. **6**, 069 (2019).
[54] G. Kasieczka, S. Marzani, G. Soyez, and G. Stagnitto, Towards machine learning analytics for jet substructure. J. High Energy Phys. 09 (**2020**) 195.
[55] S. Diefenbacher, H. Frost, G. Kasieczka, T. Plehn, and J. M. Thompson, CapsNets continuing the convolutional quest, SciPost Phys. **8**, 023 (2020).
[56] Y. Nakai, D. Shih, and S. Thomas, Strange jet tagging, arXiv:2003.09517.
[57] CMS Collaboration, Identification of heavy-flavour jets with the CMS detector in pp collisions at 13 TeV, J. Instrum. **13**, P05011 (2018).
[58] J. Bielcikoa, R. K. Elayavalli, G. Ponimatkin, J. H. Putschke, and J. Sivic, Identifying heavy-flavor jets using vectors of locally aggregated descriptors, arXiv:2005.01842.
[59] P. Baldi, P. Sadowski, and D. Whiteson, Searching for exotic particles in high-energy physics with deep learning, Nat. Commun. **5**, 4308 (2014).
[60] CMS Collaboration, A deep neural network to search for new long-lived particles decaying to jets, Mach. Learn. Sci. Technol. **1**, 035012 (2020).
[61] J. Alimena, Y. Iiyama, and J. Kieseler, Fast convolutional neural networks for identifying long-lived particles in

a high-granularity calorimeter, J. Instrum. **15**, P12006 (2020).
[62] L. De Oliveira, B. Nachman, and M. Paganini, Electromagnetic showers beyond shower shapes, Nucl. Instrum. Methods Phys. Res., Sect. A **951**, 162879 (2020).
[63] M. Paganini, L. de Oliveira, and B. Nachman, Survey of Machine Learning Techniques for High Energy Electromagnetic Shower Classification (2017), https://dl4physicalsciences.github.io/files/nips_dlps_2017_24.pdf.
[64] B. Hooberman, A. Farbin, G. Khattak, V. Pacela, M. Pierini, J.-R. Vlimant, M. Spiropulu, W. Wei, M. Zhang, and S. Vallecorsa, Calorimetry with Deep Learning: Particle Classification, Energy Regression, and Simulation for High-Energy Physics (2017), https://dl4physicalsciences.github.io/files/nips_dlps_2017_15.pdf.
[65] D. Belayneh *et al.*, Calorimetry with deep learning: Particle simulation and reconstruction for collider physics, Eur. Phys. J. C **80**, 688 (2020).
[66] G. Louppe, M. Kagan, and K. Cranmer, Learning to pivot with adversarial networks, arXiv:1611.01046.
[67] J. Dolen, P. Harris, S. Marzani, S. Rappoccio, and N. Tran, Thinking outside the ROCs: Designing decorrelated taggers (DDT) for jet substructure, J. High Energy Phys. 05 (2016) 156.
[68] I. Moult, B. Nachman, and D. Neill, Convolved substructure: Analytically decorrelating jet substructure observables, J. High Energy Phys. 05 (2018) 002.
[69] J. Stevens and M. Williams, uBoost: A boosting method for producing uniform selection efficiencies from multivariate classifiers, J. Instrum. **8**, P12013 (2013).
[70] C. Shimmin, P. Sadowski, P. Baldi, E. Weik, D. Whiteson, E. Goul, and A. Sgaard, Decorrelated jet substructure tagging using adversarial neural networks, Phys. Rev. D **96**, 074034 (2017).
[71] L. Bradshaw, R. K. Mishra, A. Mitridate, and B. Ostdiek, Mass agnostic jet taggers, SciPost Phys. **8**, 011 (2020).
[72] ATLAS Collaboration, Performance of mass-decorrelated jet substructure observables for hadronic two-body decay tagging in ATLAS, CERN Report No. ATL-PHYS-PUB-2018-014, 2018, http://cds.cern.ch/record/2630973.
[73] G. Kasieczka and D. Shih, DisCo Fever: Robust Networks Through Distance Correlation, Phys. Rev. Lett. **125**, 122001 (2020).
[74] L.-G. Xia, QBDT, a new boosting decision tree method with systematical uncertainties into training for high energy physics, Nucl. Instrum. Methods Phys. Res., Sect. A **930**, 15 (2019).
[75] C. Englert, P. Galler, P. Harris, and M. Spannowsky, Machine learning uncertainties with adversarial neural networks, Eur. Phys. J. C **79**, 4 (2019).
[76] S. Wunsch, S. Jrger, R. Wolf, and G. Quast, Reducing the dependence of the neural network function to systematic uncertainties in the input space, Comput. Softw. Big Sci. **4**, 5 (2020).
[77] A. Rogozhnikov, A. Bukva, V. V. Gligorov, A. Ustyuzhanin, and M. Williams, New approaches for boosting to uniformity, J. Instrum. **10**, T03002 (2015).
[78] CMS Collaboration, A deep neural network to search for new long-lived particles decaying to jets, Mach. Learn. Sci. Technol. **1**, 035012 (2020).
[79] J. M. Clavijo, P. Glaysher, and J. M. Katzy, Adversarial domain adaptation to reduce sample bias of a high energy physics classifier, arXiv:2005.00568.
[80] J. A. Aguilar-Saavedra, J. H. Collins, and R. K. Mishra, A generic anti-QCD jet tagger, J. High Energy Phys. 11 (2017) 163.
[81] CMS Collaboration, Identification of heavy, energetic, hadronically decaying particles using machine-learning techniques, J. Instrum. **15**, P06005 (2020).
[82] G. J. Szkely, M. L. Rizzo, and N. K. Bakirov, Measuring and testing dependence by correlation of distances, Ann. Stat. **35**, 2769 (2007).
[83] G. J. Szkely and M. L. Rizzo, Brownian distance covariance, Ann. Appl. Stat. **3**, 1236 (2009).
[84] G. J. Székely and M. L. Rizzo, The distance correlation t-test of independence in high dimension, J. Multivariate Anal. **117**, 193 (2013).
[85] G. J. Szkely and M. L. Rizzo, Partial distance correlation with methods for dissimilarities, Ann. Stat. **42**, 2382 (2014).
[86] ATLAS Collaboration, A search for pair-produced resonances in four-jet final states at $\sqrt{s} = 13$ TeV with the ATLAS detector, Eur. Phys. J. C **78**, 250 (2018).
[87] CMS Collaboration, Search for pair-produced resonances decaying to quark pairs in proton-proton collisions at $\sqrt{s} = 13$ TeV, Phys. Rev. D **98**, 112014 (2018).
[88] ATLAS Collaboration, Search for Higgs boson decays into pairs of light (pseudo)scalar particles in the $\gamma\gamma jj$ final state in pp collisions at $\sqrt{s} = 13$ TeV with the ATLAS detector, Phys. Lett. B **782**, 750 (2018).
[89] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard *et al.*, Tensorflow: A system for large-scale machine learning, in *OSDI* (2016), Vol. 16, pp. 265–283.
[90] F. Chollet, Keras, https://github.com/fchollet/keras (2017).
[91] D. Kingma and J. Ba, Adam: A method for stochastic optimization, arXiv:1412.6980.
[92] D. E. Kaplan, K. Rehermann, M. D. Schwartz, and B. Tweedie, Top Tagging: A Method for Identifying Boosted Hadronically Decaying Top Quarks, Phys. Rev. Lett. **101**, 142001 (2008).
[93] J. Thaler and K. Van Tilburg, Identifying boosted objects with N-subjettiness, J. High Energy Phys. 03 (2011) 015.
[94] J. Thaler and K. Van Tilburg, Maximizing boosted top identification by minimizing N-subjettiness, J. High Energy Phys. 02 (2012) 093.
[95] M. Cacciari, G. P. Salam, and G. Soyez, FastJet user manual, Eur. Phys. J. C **72**, 1896 (2012).
[96] L. Moore, K. Nordstrom, S. Varma, and M. Fairbairn, Reports of my demise are greatly exaggerated: N-subjettiness taggers take on jet images, SciPost Phys. **7**, 036 (2019).
[97] ATLAS Collaboration, A search for top squarks with R-parity-violating decays to all-hadronic final states with

the ATLAS detector in $\sqrt{s} = 8$ TeV proton-proton collisions, J. High Energy Phys. 06 (2016) 067.
[98] CMS Collaboration, Search for pair-produced resonances decaying to jet pairs in proton–proton collisions at $\sqrt{s} = 8$ TeV, Phys. Lett. B **747**, 98 (2015).
[99] T. Sjöstrand, S. Ask, J. R. Christiansen, R. Corke, N. Desai, P. Ilten, S. Mrenna, S. Prestel, C. O. Rasmussen, and P. Z. Skands, An introduction to PYTHIA 8.2, Comput. Phys. Commun. **191**, 159 (2015).
[100] T. Sjöstrand, S. Mrenna, and P. Z. Skands, PYTHIA 6.4 physics and manual, J. High Energy Phys. 05 (2006) 026.
[101] J. de Favereau, C. Delaere, P. Demin, A. Giammanco, V. Lematre, A. Mertens, and M. Selvaggi (DELPHES 3 Collaboration), DELPHES 3, A modular framework for fast simulation of a generic collider experiment, J. High Energy Phys. 02 (2014) 057.
[102] M. Cacciari, G. P. Salam, and G. Soyez, The Anti-k(t) jet clustering algorithm, J. High Energy Phys. 04 (2008) 063.
[103] M. Cacciari and G. P. Salam, Dispelling the N^3 myth for the k_t jet-finder, Phys. Lett. B **641**, 57 (2006).
[104] P. Z. Skands *et al.*, SUSY Les Houches accord: Interfacing SUSY spectrum calculators, decay packages, and event generators, J. High Energy Phys. 07 (2004) 036.
[105] B. Allanach *et al.*, SUSY Les Houches Accord 2, Comput. Phys. Commun. **180**, 8 (2009).
[106] ATLAS Collboration, A search for pair-produced resonances in four-jet final states at $\sqrt{s} = 13$ TeV with the ATLAS detector (2017), https://doi.org/10.17182/hepdata.79059.
[107] I. Dremin and J. Gary, Energy dependence of mean multiplicities in gluon and quark jets at the next-to-next-to-next-to leading order, Phys. Lett. B **459**, 341 (1999).
[108] A. Capella, I. Dremin, J. Gary, V. Nechitailo, and J. Tran Thanh Van, Evolution of average multiplicities of quark and gluon jets, Phys. Rev. D **61**, 074009 (2000).
[109] P. Baldi, K. Cranmer, T. Faucett, P. Sadowski, and D. Whiteson, Parameterized neural networks for high-energy physics, Eur. Phys. J. C **76**, 235 (2016).
[110] K. Cranmer, J. Pavez, and G. Louppe, Approximating Likelihood ratios with calibrated discriminative classifiers, arXiv:1506.02169.
[111] S. Wunsch, S. Jorger, R. Wolf, and G. Quast, Optimal statistical inference in the presence of systematic uncertainties using neural network optimization based on binned Poisson likelihoods with nuisance parameters, Comput. Softw. Big Sci. **5**, 4 (2021).
[112] P. De Castro and T. Dorigo, INFERNO: Inference-aware neural optimisation, Comput. Phys. Commun. **244**, 170 (2019).
[113] A. Elwood and D. Krcker, Direct optimisation of the discovery significance when training neural networks to search for new physics in particle colliders, arXiv:1806.00322.
[114] T. Dorigo and P. de Castro, Dealing with nuisance parameters using machine learning in high energy physics: A review, arXiv:2007.09121.
[115] R. T. D'Agnolo and A. Wulzer, Learning new physics from a machine, Phys. Rev. D **99**, 015014 (2019).
[116] J. H. Collins, K. Howe, and B. Nachman, Anomaly Detection for Resonant New Physics with Machine Learning, Phys. Rev. Lett. **121**, 241803 (2018).
[117] J. H. Collins, K. Howe, and B. Nachman, Extending the search for new resonances with machine learning, Phys. Rev. D **99**, 014038 (2019).
[118] R. T. D'Agnolo, G. Grosso, M. Pierini, A. Wulzer, and M. Zanetti, Learning multivariate new physics, Eur. Phys. J. C **81**, 89 (2021).
[119] M. Farina, Y. Nakai, and D. Shih, Searching for new physics with deep autoencoders, Phys. Rev. D **101**, 075021 (2020).
[120] T. Heimel, G. Kasieczka, T. Plehn, and J. M. Thompson, QCD or what?, SciPost Phys. **6**, 030 (2019).
[121] T. S. Roy and A. H. Vijay, A robust anomaly finder based on autoencoder, arXiv:1903.02032.
[122] O. Cerri, T. Q. Nguyen, M. Pierini, M. Spiropulu, and J.-R. Vlimant, Variational autoencoders for new physics mining at the large hadron collider, J. High Energy Phys. 05 (2019) 036.
[123] A. Blance, M. Spannowsky, and P. Waite, Adversarially-trained autoencoders for robust unsupervised new physics searches, J. High Energy Phys. 10 (2019) 047.
[124] J. Hajer, Y.-Y. Li, T. Liu, and H. Wang, Novelty detection meets collider physics, Phys. Rev. D **101**, 076015 (2020).
[125] A. De Simone and T. Jacques, Guiding new physics searches with unsupervised learning, Eur. Phys. J. C **79**, 289 (2019).
[126] A. Mullin, H. Pacey, M. Parker, M. White, and S. Williams, Does SUSY have friends? A new approach for LHC event analysis, arXiv:1912.10625.
[127] G. M. Alessandro Casa, Nonparametric semisupervised classification for signal detection in high energy physics, arXiv:1809.02977.
[128] B. M. Dillon, D. A. Faroughy, and J. F. Kamenik, Uncovering latent jet substructure, Phys. Rev. D **100**, 056002 (2019).
[129] A. Andreassen, B. Nachman, and D. Shih, Simulation assisted likelihood-free anomaly detection, Phys. Rev. D **101**, 095004 (2020).
[130] B. Nachman and D. Shih, Anomaly detection with density estimation, Phys. Rev. D **101**, 075042 (2020).
[131] M. Romo Crispim, N. Castro, R. Pedro, and T. Vale, Transferability of deep learning models in searches for new physics at colliders, Phys. Rev. D **101**, 035042 (2020).
[132] M. C. Romao, N. Castro, J. Milhano, R. Pedro, and T. Vale, Use of a generalized energy mover's distance in the search for rare phenomena at colliders, arXiv:2004.09360.
[133] O. Knapp, G. Dissertori, O. Cerri, T. Q. Nguyen, J.-R. Vlimant, and M. Pierini, Adversarially learned anomaly detection on CMS open data: Re-discovering the top quark, arXiv:2005.01598.
[134] ATLAS Collaboration, Dijet resonance search with weak supervision using 13 TeV pp collisions in the ATLAS detector, Phys. Rev. Lett. **125**, 131801 (2020).

[135] B. M. Dillon, D. A. Faroughy, J. F. Kamenik, and M. Szewc, Learning the latent structure of collider events, J. High Energy Phys. 10 (2020) 206.

[136] M. C. Romao, N. Castro, and R. Pedro, Finding new physics without learning about it: Anomaly detection as a tool for searches at colliders, Eur. Phys. J. C **81**, 27 (2021).

[137] O. Amram and C. M. Suarez, Tag N' train: A technique to train improved classifiers on unlabeled data, J. High Energy Phys. 01 (2021) 153.

[138] T. Cheng, J.-F. Arguin, J. Leissner-Martin, J. Pilette, and T. Golling, Variational autoencoders for anomalous jet tagging, arXiv:2007.01850.