What is the machine learning?

Spencer Chang, Timothy Cohen, and Bryan Ostdiek
Institute of Theoretical Science, University of Oregon, Eugene, Oregon 97403, USA

(Received 19 October 2017; published 13 March 2018)

Applications of machine learning tools to problems of physical interest are often criticized for producing sensitivity at the expense of transparency. To address this concern, we explore a data planing procedure for identifying combinations of variables—aided by physical intuition—that can discriminate signal from background. Weights are introduced to smooth away the features in a given variable(s). New networks are then trained on this modified data. Observed decreases in sensitivity diagnose the variable's discriminating power. Planing also allows the investigation of the linear versus nonlinear nature of the boundaries between signal and background. We demonstrate the efficacy of this approach using a toy example, followed by an application to an idealized heavy resonance scenario at the Large Hadron Collider. By unpacking the information being utilized by these algorithms, this method puts in context what it means for a machine to learn.

DOI: 10.1103/PhysRevD.97.056009

I. INTRODUCTION

A common argument against using machine learning for physical applications is that they function as a black box: send in some data and out comes a number. While this kind of nonparametric estimation can be extremely useful, a physicist often wants to understand what aspect of the input data yields the discriminating power, in order to learn/confirm the underlying physics or to account for their systematics. A physical example studied below is the Lorentz invariant combination of final state four-vectors, which exhibit a Breit-Wigner peak in the presence of a new heavy resonance. The simple example illustrated in Fig. 1 exposes the subtlety inherent in extracting what the machine has "learned." The left panel shows red and blue data, designed to be separated by a circular border. The right panel shows the boundary between signal and background regions that the machine (a neural network with one hidden layer composed of 10 nodes) has inferred. Under certain assumptions, a deep neural network can approximate any function of the inputs, e.g., [1], and thus produces a fit to the training data. While any good classifier would find a "circular" boundary, simply due to the distribution of the training data, one (without additional architecture) has no mechanism of discovering it is a circle. In light of this, our goal is to unpack the numerical discriminator into a set of human-friendly variables that best characterize the data. While we are not inverting the network to find its functional form, we are providing a scheme for understanding classifiers.

For context, we acknowledge related studies within the growing machine learning for particle physics literature. The authors of [2–5] emphasized the ability of deep learning to outperform physics inspired high-level variables. We use the "uniform phase space" scheme to flatten discriminating variables, which was introduced in [6] to quantify the information learned by deep neural networks. For other suggestions on testing what the machines are learning, see [7–12]. A nice summary of these ideas can be found in [13]. Additionally, progress has recently been made in the related question of *how* the machine learns [14,15].

Section II introduces a simple weighting scheme, which we call "data planing."[1] Applications to a toy model will be presented to illustrate the features of this approach. As we demonstrate, it is possible to plane away all the underlying discriminating characteristics of this toy by utilizing combinations of linear and nonlinear variables. This highlights another salient attribute of data planing: by comparing the performance of linear and deep neural networks, one can infer to what extent the encoded information is a linear versus nonlinear function of the inputs. Then in Sec. III we show that these features can be realized in a more realistic particle physics setting. Finally, Sec. IV concludes this paper with a discussion of future investigations.

[1]Planing is a woodworking technique for smoothing a surface.

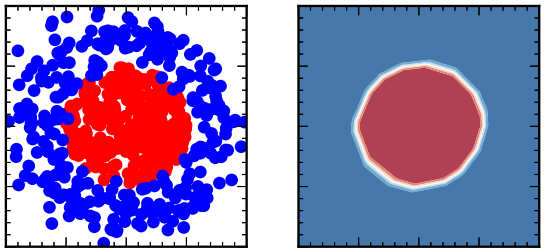

FIG. 1. (Left panel) The machine is trained using rectilinear coordinates to distinguish blue and red as defined by the displayed training data. (Right panel) The classifier output ranges from blue to red.

II. DATA PLANING

Our starting assumption is that a sufficiently deep network with ample training can take advantage of all inherent information to discriminate signal from background; i.e., the network approximately attains Bayes error [16,17], the lowest possible error rate. The approach advocated here is to then remove information, where the performance degradation of the new networks provides diagnostic value (this procedure was first introduced in the "uniform phase space" section of [6]). To plane the data, we weight the events, which are labeled by i and characterized by input variables $\vec{x}_i$. After choosing a variable m, the planing weights are computed using

$$[w(\vec{x}_i)]^{-1} = C \frac{\mathrm{d}\sigma(\vec{x}_i)}{\mathrm{d}m}\bigg|_{m=m_i}, \tag{1}$$

where $\mathrm{d}\sigma/\mathrm{d}m$ is the differential cross section (or more generally the underlying distribution for the training events), and a constant C is required by dimensional analysis and should be common to signal and background samples. In practice, we execute Eq. (1) by uniformly binning the input events and inverting the histogram, which introduces some finite bin effects as will be apparent below. Note for a different purpose, the experimental collaborations frequently weight events to match the transverse momentum spectrum of different samples (e.g., [18–20]).

Next, we train a new network on the planed input data. The performance drop yields a measurement of the discriminating information contained in the variable m. This procedure can be iterated, by choosing the next variable to plane with, until the network is unable to discriminate between the fully planed signal and background. This end point demonstrates that all of the information available to distinguish signal from background is encoded in the planing variables, thereby providing a procedure to concretely frame the question posed by the title of this paper.

Planing is one of many different approaches to understanding a network's discrimination power as mentioned in the introduction and reviewed in [13]. In what follows, as we study planing we will also utilize a technique (see [2–5,11,12]) which we refer to as "saturation," that compares a network trained on only low-level inputs with networks trained after adding higher-level variables. Saturation provides a tool to ensure that our networks are sufficiently deep, by checking that the new network's performance does not improve by much.[2]

Saturation additionally suggests another method to uncover what information a machine is utilizing. One could consider training networks using only the high-level variable(s) of interest as inputs, where in contrast to the saturation technique, no low-level information is being provided to the network. The diagnostic test would be to compute if the resulting network can achieve performance similar to that of a deep network that had been trained on only the low-level inputs. If the metrics were comparable, it would suggest that a machine can use the high-level variables alone to classify the data. However, the planing method has two advantages. First, the number of input parameters would typically change when going from only low-level to only high-level variables; unlike planing, this requires altering the network architecture. This in turn can impact the optimization of hyper-parameters, thereby complicating the comparison. Furthermore, this method suffers the same issue as saturation in that as the limit towards ideal performance is achieved, one is forced to take seriously small variations in the metrics. If there are not enough training trials to adequately determine the errors, these small variations could be incorrectly interpreted as consistent with zero. This can again be contrasted with planing in that our approach yields a qualitative drop in performance and is more straightforward to interpret.

For all results presented below, we will distinguish the performance of a *linear* versus *deep* network. This provides a diagnostic tool as to what extent the remaining information is a (non)linear function of the inputs. The machines used here are neural networks implemented within the Keras [21] package with the TensorFlow [22] backend. We choose either zero or three hidden layers to define the linear and deep networks, respectively; each hidden layer has 50 nodes. Note that a network with no hidden layer is equivalent to standard logistic regression. The inputs to each node are passed through the ReLu activation function, except that the Sigmoid is applied to the output layer. Training is done using the Adam optimizer [23]. For each classification, 10% of the events are used as a test set, and 4.5% are used for validation. Our metrics are computed on the test set using scikit-learn [24]. We provide the standard metric for performance: the area under the receiver operating characteristic curve (AUC). We compute standard

[2]Saturation can also be used to determine what high-level variables provide information (e.g., [11,12]). Planing tests can be easier to interpret, due to their larger dynamic range in performance metrics.

TABLE I. The AUC output for a variety of input configurations applied to toy signal data pulled from Eq. (2) and a flat background. The variable r is the cylindrical radius.

(x, y, z)	r	Planed	Linear AUC	Deep AUC
✓	✗	✗	0.61275(01)	0.81243(45)
✓	✓	✗	0.79672(01)	0.81388(23)
✓	✗	r	0.61030(01)	0.61026(02)
✓	✗	(r, z)	0.5081(16)	0.49998(03)

deviation of the AUC by using the output of ten networks trained with randomly chosen initial conditions. This is provided in the tables and gives a sense for the stability of the minimization.

We will first demonstrate how to plane in a concrete toy example. Assume the input data are given by three coordinates $\vec{x} = (x, y, z)$, and the signal is drawn from the distribution

$$f(\vec{x}) = [\Theta(r_0 - r) + C_r] \cdot [z \cdot B_z + C_z], \tag{2}$$

where $\Theta(x)$ is the step function, $r = \sqrt{x^2 + y^2}$, the C_i are constants, r_0 is the radius of a circular feature in the x-y plane, and B_z is the slope of the z-component of the signal. The background distribution is uniform in x, y, and z. This toy model contains both linear (z) and nonlinear (x-y) differences between signal and background, and it is also factorized such that there are no correlations between r and z.

The results of the study are presented in Table I. First, note that when training the networks on only the low-level inputs, the deep network is more powerful. This points to the presence of a nonlinearity, a consequence of the cylindrical shape of the underlying distribution. Next, in the spirit of the saturation approach, we add the 2-D radius r to the list of inputs and train another network. We see that the linear and deep networks perform nearly identically to the deep network trained only on the low-level inputs, which implies the remaining discriminating power is a linear function of the inputs, as it had to be from Eq. (2). The third row shows the results when training on data whose r-dependence has been planed away. All that remains is the z-dependence, which is linear as demonstrated by comparing the linear and deep outputs (see also Fig. 3, left). Finally, we plane in r and z simultaneously. The bottom row of the table shows the AUC approaching $1/2$, signaling that all discriminating power is captured by r and z.

III. APPLICATION TO PARTICLE PHYSICS

This section provides a planing application to a physical scenario. We extend the Standard Model with a single particle, a massive vector boson Z' that decays to an electron (e^-) positron (e^+) pair. This example was chosen because the best discriminator against the smoothly falling photon background is the invariant mass $m^2 = (p_{e^+} + p_{e^-})^2$, a nonlinear combination of the input four-vectors p. Furthermore, depending on how we choose the helicity structure of the coupling between the Z' and the Standard Model particles, additional discriminating power beyond invariant mass may be present.

We use a phenomenological parametrization:

$$\mathcal{L} \supset Z'_\mu \sum_f Q_f (g_{Z',L} \bar{f} \gamma^\mu P_L f + g_{Z',R} \bar{f} \gamma^\mu P_R f), \tag{3}$$

where f are the Standard Model fermions, Q_f is the electric charge, $P_{L(R)}$ are the left (right) projection operators, and $g_{Z',L(R)}$ is the strength of the coupling between the left (right) handed fermions and the Z'. We take $M_{Z'} = 1$ TeV and the width $\Gamma_{Z'} = 10$ GeV. This model is excluded by Large Hadron Collider (LHC) data over a wide parameter space; we present it here solely as an instructive tool.

We will focus our attention on two cases: Z'_V with vector coupling, where $g_{Z'L} = g_{Z'R}$ (the same as the helicity structure of the photon), and Z'_L with left couplings active and $g_{Z'R} = 0$. The models are implemented using FeynRules [25]. The Monte Carlo event generator MadGraph [26] is used to simulate 10^6 proton-proton collisions with an invariant mass between 500 and 1500 GeV for γ^*, Z'_V, and Z'_L intermediate states. Using information contained in $pp \to e^+e^-$ events, the goal is to distinguish the Z' signal models from the photon background.

We take the low-level training inputs to be the four-vectors $(E, \vec{p})$ of the $e^\pm$. We know that the best discriminator between signal and background is the invariant mass. This is the only distinguishing feature between the Z'_V and the photon. However, due to the nontrivial helicity structure of the Z'_L model, there are additional features in the high-level variable rapidity, $y \equiv \frac{1}{2}\log[(E + p_z)/(E - p_z)]$, that distinguish it from the photon. The distributions of the high-level variables are shown in the upper panels of Fig. 2.

The results of classifying the Z'_V against the photon are shown in Table II. We train the linear and deep networks on the low-level variables and again on the low-level variables plus invariant mass. The deep network performance is very similar with or without the invariant mass; following the logic of the saturation approach, this shows that the low-level deep network is a nearly ideal discriminator. For comparison, the low-level linear network performance is far below that of the deep network. We infer that nonlinear combinations of the input variables are needed to optimally classify the data. When invariant mass is added to the linear network, the resulting performance significantly improves, but it still does not match the power of the deep networks. One is tempted to (falsely) conjecture that there is extra discriminating power to uncover, and the top row of Fig. 2 seems to add support. It is also possible that the linear network aided by m does not perform as well as the deep

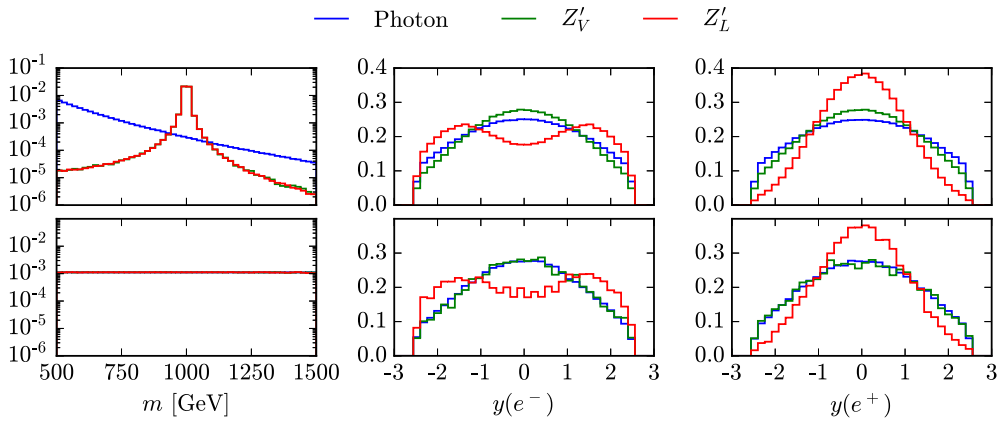

FIG. 2. Histograms of the constructed variables normalized to unity. The top (bottom) panels are before (after) planing the input events using the invariant mass m. The rapidity of the electron (positron) is specified by $y(e^-)$ ($y(e^+)$).

network, even though it contains all of the relevant information, because it can only make a one-sided cut.

However, due to the vector nature of the photon couplings (and the masslessness of the final state particles), we know that the only difference between signal and background should be captured by the invariant mass of the electron positron pair. To determine the correct interpretation, we plane signal and background in invariant mass as shown in the lower row of Fig. 2. As expected, the photon and the vector Z' have nearly identical distributions up to the noise induced by the histogramming procedure for computing the weights.

In order to quantify if there is information hidden in any of the other distributions, linear and deep networks are trained on the planed inputs. The results are shown in the lower section of Table II as measured on the planed test set. Both networks have an AUC approaching 0.5, so no noticeable discriminating power remains. Since the planing process removed the invariant mass information, the networks cannot tell the difference between the massless and massive vector boson propagators, showing that mass is in fact the only discriminator.

Next, we explore the Z'_L signal model where we expect additional discriminants to be present. Networks are trained to distinguish the Z'_L from the photon, with results shown in Table III. Initially, we see a pattern similar to that in the previous examples. Note that now the AUCs are slightly closer to unity as compared to the Z'_V model, again indicating the presence of information beyond the invariant mass. An inspection of the distributions that have been planed using m, which are plotted in the lower panels of Fig. 2, reveals the source of this additional discriminating power. The Z'_L clearly manifests differences in the rapidities for the electron and positron, where the magnitude of the electron rapidity is usually larger than the magnitude of the positron rapidity for the Z'_L. This results from the choice of chiral couplings and the shape of the parton distribution functions. This suggests that a variable $\Delta|y| \equiv |y(e^-)| - |y(e^+)|$ should be a useful discriminator (the more traditional approach is to utilize asymmetry observables, e.g., the reviews [27,28]). This can be further quantified by computing the correlation between the linear network response (before the Sigmoid activation) and $\Delta|y|$, as shown in the right panel of Fig. 3. A correlation of 0.90 is observed, implying that much of the remaining information is contained in $\Delta|y|$. As a comparison, we also show the equivalent result derived for the toy model of Sec. II in the left panel of Fig. 3. Since the signal was linear in z by construction, a perfect correlation is expected and demonstrated. Performing this test on any new variables is a powerful and quick method to assess their performance and test their linearity.

Next, we plane the inputs using the full m-$\Delta|y|$ dependence and train new networks. The results are provided in the last row of Table III. We see that an AUC approaching $1/2$ is achieved for both the linear and deep networks. The remaining bits of discriminating power could be resolved

TABLE II. The AUC output for a variety of input configurations applied to the Z'_V model and the photon background.

$(E, \vec{p})$	m	Planed	Linear AUC	Deep AUC
✓	✗	✗	0.746221(01)	0.988510(98)
✓	✓	✗	0.938967(01)	0.989007(03)
✓	✗	$\boldsymbol{m}$	0.50550(29)	0.4942(48)

TABLE III. The AUC output for a variety of input configurations applied to the Z'_L model and the photon background. The variable $\Delta|y| \equiv |y(e^-)| - |y(e^+)|$.

$(E, \vec{p})$	m	Planed	Linear AUC	Deep AUC
✓	✗	✗	0.763280(05)	0.989353(59)
✓	✓	✗	0.942004(02)	0.989826(10)
✓	✗	$\boldsymbol{m}$	0.626648(28)	0.6258(24)
✓	✗	$(\boldsymbol{m}, \boldsymbol{\Delta|y|})$	0.52421(15)	0.5320(25)

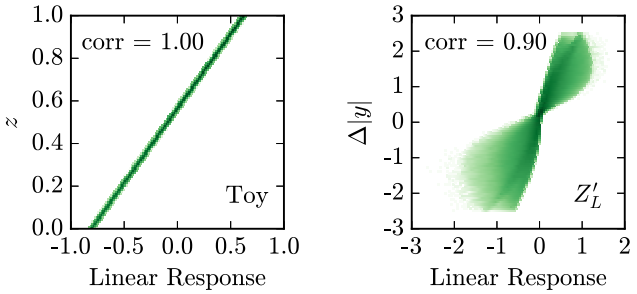

FIG. 3. (Left panel) Density of events for the planed linear network output versus z for the toy model presented in Sec. II. (Right panel) Density of events for the planed linear network output and $\Delta|y|$ for the Z'_L model. Both signal and background events are plotted. The correlation measure is provided in the top of each panel. Perfect correlation would imply that the variable and linear network represent the same information.

by planing in 3D: $(m, y(e^+), y(e^-))$. This would determine to what extent it is due to physics as opposed to noise from the histogramming procedure.

IV. OUTLOOK

We explored data planing, a probe of machine learning algorithms designed to remove features in a given variable; see also [6]. By iteratively planing training data, it is possible to remove the machine's ability to classify. As a by-product, the planed variables determine combinations of input variables that explain the machine's discriminating power. This procedure can be explored systematically but is most efficient in tandem with physics intuition.

In the future, it would be interesting to examine this procedure with more realistic training data that include initial/final state radiation and detector effects. The application to more complicated signals should also be tested. With exotic signals, planing may need to be done in many dimensions; perhaps a kernel smoothing procedure should be applied, or a network can be trained to compute the weights directly, which can then be utilized when training the planed network. Choosing which variables to plane will be increasingly challenging in higher dimensional phase space, as highlighted in the example of jet images [6]. Careful treatment of correlations will also be relevant; see [29,30] for related ideas.

One interesting extension would be to systematically test a large set of Lorentz invariants in order to find the combination that yields the largest performance drop. This could reveal new variables for traditional searches. Finally, what information is contained in jets could be explored with planing to complement the existing saturation analyses [11,12]. We intend to investigate many of these applications in future studies.

ACKNOWLEDGMENTS

We would like to thank Marat Freytsis and Benjamin Nachman for useful comments on the draft. T. C. is especially grateful to Ronnie Cohen for teaching him to use a plane in the real world. This work is supported by the U.S. Department of Energy under Awards No. DE-SC0011640 (to S. C. and B. O.) and No. DE-SC0018191 (to T. C.).

[1] G. Cybenko, Approximation by superpositions of a sigmoidal function, Mathematics of Control, Math. Control Signals Syst. **2**, 303 (1989).
[2] P. Baldi, P. Sadowski, and D. Whiteson, Searching for exotic particles in high-energy physics with deep learning, Nat. Commun. **5**, 4308 (2014).
[3] P. Baldi, P. Sadowski, and D. Whiteson, Enhanced Higgs Boson to $\tau^+\tau^-$ Search with Deep Learning, Phys. Rev. Lett. **114**, 111801 (2015).
[4] P. Baldi, K. Bauer, C. Eng, P. Sadowski, and D. Whiteson, Jet substructure classification in high-energy physics with deep neural networks, Phys. Rev. D **93**, 094034 (2016).
[5] 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).
[6] L. de Oliveira, M. Kagan, L. Mackey, B. Nachman, and A. Schwartzman, Jet-images deep learning edition, J. High Energy Phys. 07 (2016) 069.
[7] J. Stevens and M. Williams, uBoost: A boosting method for producing uniform selection efficiencies from multivariate classifiers, J. Instrum. **8**, P12013 (2013).
[8] A. Rogozhnikov, A. Bukva, V. V. Gligorov, A. Ustyuzhanin, and M. Williams, New approaches for boosting to uniformity, J. Instrum. **10**, T03002 (2015).
[9] L. G. Almeida, M. Backovic, 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.
[10] 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.
[11] K. Datta and A. Larkoski, How much information is in a jet?, J. High Energy Phys. 06 (2017) 073.
[12] J. A. Aguilar-Saavedra, J. H. Collins, and R. K. Mishra, A generic anti-QCD jet tagger, J. High Energy Phys. 11 (2017) 163.

[13] 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, arXiv:1709.04464.
[14] P. Mehta and D. J. Schwab, An exact mapping between the Variational Renormalization Group and Deep Learning, arXiv:1410.3831.
[15] R. Shwartz-Ziv and N. Tishby, Opening the black box of deep neural networks via information, arXiv:1703.00810).
[16] K. Fukunaga, *Introduction to Statistical Pattern Recognition*, 2nd ed. (Academic Press, Boston, 1990).
[17] K. Tumer and J. Ghosh, *Estimating the Bayes error rate through classifier combining* (Institute of Electrical and Electronics Engineers Inc., Piscataway, 1996), Vol. 2, pp. 695–699.
[18] G. Aad *et al.* (ATLAS Collaboration), Measurements of Higgs boson production and couplings in diboson final states with the ATLAS detector at the LHC, Phys. Lett. B **726**, 88 (2013); Corrigendum, Phys. Lett. B **734**, 406(E) (2014).
[19] S. Chatrchyan *et al.*(CMS Collaboration), Evidence for the 125 GeV Higgs boson decaying to a pair of τ leptons, J. High Energy Phys. 05 (**2014**) 104.
[20] G. Aad *et al.* (ATLAS Collaboration), Identification of boosted, hadronically decaying W bosons and comparisons with ATLAS data taken at $\sqrt{s} = 8$ TeV, Eur. Phys. J. C **76**, 154 (2016).
[21] F. Chollet *et al.*, Keras, https://github.com/fchollet/keras.
[22] M. Abadi *et al.*, TensorFlow: Large-scale machine learning on heterogeneous systems., http://tensorflow.org/.
[23] D. P. Kingma and J. Ba, Adam: A method for stochastic optimization, arXiv:1412.6980.
[24] F. Pedregosa *et al.*, Scikit-learn: Machine learning in Python, J. Machine Learning Res. **12**, 2825 (2011).
[25] A. Alloul, N. D. Christensen, C. Degrande, C. Duhr, and B. Fuks, FeynRules 2.0: A complete toolbox for tree-level phenomenology, Comput. Phys. Commun. **185**, 2250 (2014).
[26] J. Alwall, R. Frederix, S. Frixione, V. Hirschi, F. Maltoni, O. Mattelaer, H. S. Shao, T. Stelzer, P. Torrielli, and M. Zaro, The automated computation of tree-level and next-to-leading order differential cross sections, and their matching to parton shower simulations, J. High Energy Phys. 07 (2014) 079.
[27] A. Leike, The phenomenology of extra neutral gauge bosons, Phys. Rep. **317**, 143 (1999).
[28] T. G. Rizzo, Z' phenomenology and the LHC, arXiv:hep-ph/0610104.
[29] 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.
[30] 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).