

Deep-learned event variables for collider phenomenology

Doojin Kim^{1,*}, Kyoungchul Kong^{2,†}, Konstantin T. Matchev^{3,‡}, Myeonghun Park^{4,5,6,§} and Prasanth Shyamsundar^{7,||}

¹*Mitchell Institute for Fundamental Physics and Astronomy, Department of Physics and Astronomy, Texas A&M University, College Station, Texas 77843, USA*

²*Department of Physics and Astronomy, University of Kansas, Lawrence, Kansas 66045, USA*

³*Institute for Fundamental Theory, Physics Department, University of Florida, Gainesville, Florida 32611, USA*

⁴*Institute of Convergence Fundamental Studies, Seoultech, Seoul 01811, Korea*

⁵*School of Physics, KIAS, Seoul 02455, Korea*

⁶*Center for Theoretical Physics of the Universe, Institute for Basic Science, Daejeon 34126 Korea*

⁷*Fermilab Quantum Institute, Fermi National Accelerator Laboratory, Batavia, Illinois 60510, USA*



(Received 2 June 2021; accepted 12 February 2023; published 28 February 2023)

The choice of optimal event variables is crucial for achieving the maximal sensitivity of experimental analyses. Over time, physicists have derived suitable kinematic variables for many typical event topologies in collider physics. Here, we introduce a deep-learning technique to design good event variables, which are sensitive over a wide range of values for the unknown model parameters. We demonstrate that the neural networks trained with our technique on some simple event topologies are able to reproduce standard event variables like invariant mass, transverse mass, and stransverse mass. The method is automatable and completely general and can be used to derive sensitive, previously unknown, event variables for other, more complex event topologies.

DOI: [10.1103/PhysRevD.107.L031904](https://doi.org/10.1103/PhysRevD.107.L031904)

I. INTRODUCTION

Data in collider physics is very high dimensional, which brings a number of challenges for the analysis, encapsulated in “the curse of dimensionality” [1]. Mapping the raw data to reconstructed objects involves initial dimensionality reduction in several stages, including track reconstruction, calorimeter clustering, jet reconstruction, etc. Subsequently, the kinematics of the reconstructed objects is used to define suitable analysis variables, adapted to the specific channel and targeted event topology. Each such step is essentially a human-engineered feature-extraction process from complicated data to a handful of physically meaningful quantities. While some information loss is unavoidable, physics principles and symmetries help keep it to a minimum.

In this paper, we shall focus on the last stage of this dimensionality reduction chain, namely, the optimal

construction of kinematic variables, which is essential to expedite the discovery of new physics and/or to improve the precision of parameter measurements. By now, the experimentalist’s toolbox contains a large number of kinematic variables, which have been thoroughly tested in analyses with real data (see Refs. [2–5] for reviews). The latest important addition to this set is the so-called singularity variables [6–10], which are applicable to missing energy events—the harbingers of dark matter production at colliders. In the machine-learning era, a myriad of algorithms have been invented or adopted to tackle various tasks that arise in the analysis of collider data, e.g., signal-background discrimination (see Ref. [11] for a continuously updated complete review of the literature). *Under the hood*, the machines trained in these techniques could learn to construct useful features from the low-level event description because they are relevant to the task at hand. But it is difficult to interpret what exactly the machines have learned in the process [12,13]. Furthermore, it is rarely studied whether the human-engineered features are indeed the best event variables for certain purposes and whether machines can outperform theorists at constructing event variables.

These two issues, explainability and optimality, are precisely the two questions which we shall address in this paper. We shall introduce a new technique for training neural networks to directly *output* useful features or event variables (which offer sensitivity over a range of unknown

*doojin.kim@tamu.edu

†kckong@ku.edu

‡matchev@ufl.edu

§parc.seoultech@seoultech.ac.kr

||prasanth@fnal.gov

Published by the American Physical Society under the terms of the [Creative Commons Attribution 4.0 International](https://creativecommons.org/licenses/by/4.0/) license. Further distribution of this work must maintain attribution to the author(s) and the published article’s title, journal citation, and DOI. Funded by SCOAP³.

parameter values). This allows for explainability of the machine’s output by comparison against known features in the data. At the same time, it is important to verify that the variables obtained using our technique are indeed the optimal choice, and we will test this by directly comparing them against the human-engineered variables that are known to be optimal for their respective event topologies. Once we have validated our training procedure in this way, we could extend it to more complex event topologies and derive novel kinematic variables in interesting and difficult scenarios.

Understanding how and what a neural network (NN) learns is a difficult task. Here, we shall consider relatively simple physics examples that are nevertheless highly non-trivial from a machine learning point of view: (1) visible two-body decay (to two visible daughter particles), (2) semi-invisible two-body decay (to one visible and one invisible daughter particle), and (3) semi-invisible two-body decays of pair-produced particles. It is known that the relevant variables in those situations are the invariant mass m , the transverse mass m_T [14,15], and the stransverse mass m_{T2} [16], respectively. We will demonstrate that in each case the NN can be trained to learn the corresponding physics variable in the reduced latent space. The method can be readily generalized to more complex cases to derive deep-learned, next-generation event variables.

II. METHODOLOGY

Let X represent the high-dimensional input features from a collision event, e.g., the 4-momenta of the reconstructed physics objects. Let $V(X)$ be a low-dimensional event variable constructed from X . In this work, we shall model the function V using a neural network, where for notational convenience the dependence of V on the architecture and weights of the network will not be explicitly indicated. We imagine that V retains the relevant physics information and will be the centerpiece of an experimental study of a theory model with a set of unknown parameters Θ . The goal is to

train the NN encoding the function V to be “useful” over a wide range of values for Θ . For this purpose, we will need to train with events generated from a range of Θ values. Note that this is a departure from the traditional approach in particle physics, where training is done for specific study points with fixed values of Θ . In addition, we will have to quantify the usefulness of a given event variable $V(X)$, as explained in the remainder of this section.

Using intuition from information theory (see the Appendix), we propose the strategy schematically outlined in Fig. 1: train the event variable network so that the distributions $p_V \otimes p_\Theta$ and $p_{(V,\Theta)}$ are highly distinguishable, as quantified by an auxiliary classifier network. Here, p_V and p_Θ are the probability distribution functions of V and Θ , respectively, and $p_{(V,\Theta)}$ is their joint distribution.

A. Training data generation

To generate the training data, we start with the two distributions p_Θ and $p_{X|\Theta}$, where $p_{X|\Theta}$ is the distribution of the event X conditional on Θ . General-purpose event generators can be used to sample from $p_{X|\Theta}$. The specific choice of a prior distribution p_Θ is not crucial—as long as it allows us to sample θ over a sufficiently wide range (the one in which we want the event variable V to be sensitive), any distribution will do, and one is further free to impose theoretical prejudice like fine-tuning, etc. The overall distribution of X , namely, p_X , is given by

$$p_X(x) = \int_{\Omega} d\theta p_\Theta(\theta) p_{X|\Theta}(x|\theta). \quad (1)$$

Our training data consist of two classes, whose generation is illustrated in the left (green) block of Fig. 1. Each training data point is given by a 2-tuple (X, Θ) along with the class label $y_{\text{target}} \in \{0, 1\}$ of the data point. Under class 0, X and Θ are independent of each other, and their joint distribution is given by $p_X \otimes p_\Theta$. This is accomplished by simply replacing the true value of Θ used to generate X with

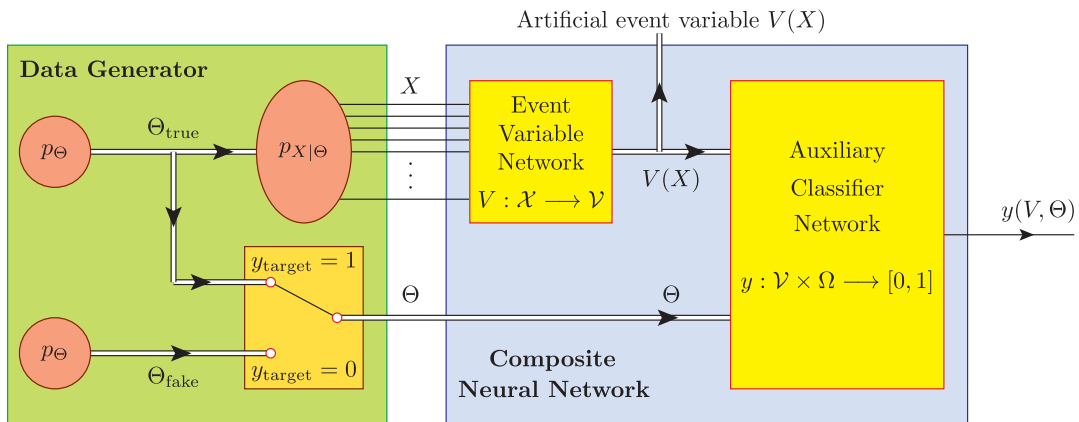


FIG. 1. A schematic diagram of the training strategy for the artificial event variable V . The left (green) and right (blue) blocks depict the generation of training data and the composite neural network layout, respectively.

a fake one for the data points in class 0. Under class 1, the joint distribution of (X, Θ) is given by

$$p_{(X, \Theta)}(x, \theta) = p_{X|\Theta}(x|\theta)p_{\Theta}(\theta). \quad (2)$$

B. Event variable training

As shown in the right (blue) block in Fig. 1, we then set up a composite network for classifying the data points (X, Θ) into the two classes. The composite network consists of two parts. First, an event variable network (EVN) takes the high-dimensional collider event information X of dimensionality $d_X \equiv \dim(X)$ as input and returns a low-dimensional $V(X)$ of dimensionality $d_V \equiv \dim(V)$ as output. As indicated, this network parametrizes the artificial event variable function $V(X)$, which is precisely what we are interested in training. The output layer of the EVN network does *not* use an activation function (or, equivalently, uses the identity activation). Since $d_V \ll d_X$, the main task of the EVN network is to perform the needed dimensionality reduction. However, to ensure that this retains the maximal amount of information, we introduce an auxiliary classifier network which takes the event variable $V(X)$ and the parameters Θ as input and returns a one-dimensional output, $y(V, \Theta) \in [0, 1]$. Note that the input received by the auxiliary network is distributed as $p_V \otimes p_{\Theta}$ under class 0 and as $p_{(V, \Theta)}$ under class 1.

The information bottleneck [17] $V(X)$ created by the EVN module is optimized by simply training the composite network as a classifier for the input data (X, Θ) , using the class labels y_{target} as the supervisory signal.

III. EXPERIMENTS

The EVN module in the network architecture from Fig. 1 reduces the original d_X -dimensional features to a d_V -dimensional subspace of event variables, which by construction are guaranteed to be highly sensitive to the theory model parameters Θ , but without any explicit dependence on them. Such variables have been greatly valued in collider phenomenology, and a significant number have been proposed and used in experimental analyses. As a proof of principle, we shall now demonstrate how our approach is able to reproduce the known kinematic variables in a few simple but nontrivial examples. Here, we shall only consider one variable at a time, i.e., $d_V = 1$, postponing the case of $d_V > 1$ to future work.

A. Example 1: Fully visible two-body decay

First, we consider the fully visible decay of a parent particle A into two massless visible daughter particles, $A \rightarrow bc$. The parameter Θ in this example is the mass m_A of the mother particle A . The event X is specified by the 4-momenta of the daughter particles p_b and p_c , leading to $d_X = 8$.

The prior p_{Θ} is chosen to sample m_A uniformly in the range $[100, 500]$ GeV. For each sampled value of m_A , we generate an event as follows. A generic boost for the parent particle A is obtained by isotropically picking the direction for its momentum and uniformly sampling its laboratory-frame energy in the range $[m_A, 1500]$ GeV. Subsequently, A is decayed on shell into two massless particles (isotropically in its own rest frame) so that the input X consists of the laboratory-frame final-state 4-momenta $\{p_b, p_c\} \equiv \{E_b, \vec{p}_b, E_c, \vec{p}_c\}$. The pair (X, m_A) forms a data point in class 1. For all data points in class 0, the true values of m_A are replaced with fake ones sampled from the same prior distribution p_{Θ} .

All the neural networks used in this work were implemented in TENSORFLOW [18]. For the event variable network, we used a sequential fully connected architecture with five hidden layers. The hidden layers, in order, have 128, 64, 64, 64, and 32 nodes, all of them using rectified linear unit (ReLU) as the activation function. The output layer has one node with no activation function. The classifier network is a fully connected network with three hidden layers (16 nodes each, with ReLU activation). The output layer of the classifier has one node with sigmoid activation. These two networks were combined as shown in the right (blue) block in Fig. 1 and trained with 2.5 million events total (50/50 split between classes 0 and 1), out of which 20% was set aside for validation. The network was trained for 20 epochs with a minibatch size of 50, using the Adam optimizer and the binary cross-entropy loss function.

For the event topology considered in this example, it is known that the event variable most sensitive to the value of m_A is the invariant mass of the daughter particles

$$m_{bc} = \sqrt{(E_b + E_c)^2 - (\vec{p}_b + \vec{p}_c)^2} \quad (3)$$

as well as any variable that is in one-to-one correspondence with it. To test whether our artificial event variable V learned by the NN correlates with m_{bc} , we show a heat map of the joint distribution of (V, m_{bc}) in the upper-left panel of Fig. 2. Here, and in what follows, the heat map is generated using a separate test dataset with 10^5 events. In the plot, we also show two nonparametric correlation coefficients, namely, Kendall's τ coefficient [19] and Spearman's rank correlation coefficients r_s [20]. A value of ± 1 for them would indicate one-to-one correspondence. Our results depict an almost perfect correspondence between V and m_{bc} . Here, and in what follows, we append an overall minus sign to V if needed, in order to make the correlations positive and the plots in Fig. 2 intuitive.

In practice, the artificial variable can be used to compare the data against templates simulated for different values of Θ . To illustrate this usage, in the lower-left panel of Fig. 2, we show unit-normalized distributions of the deep-learned variable V for several different values of $m_A = \{200, 280, 320, 400\}$ GeV. It is seen that the

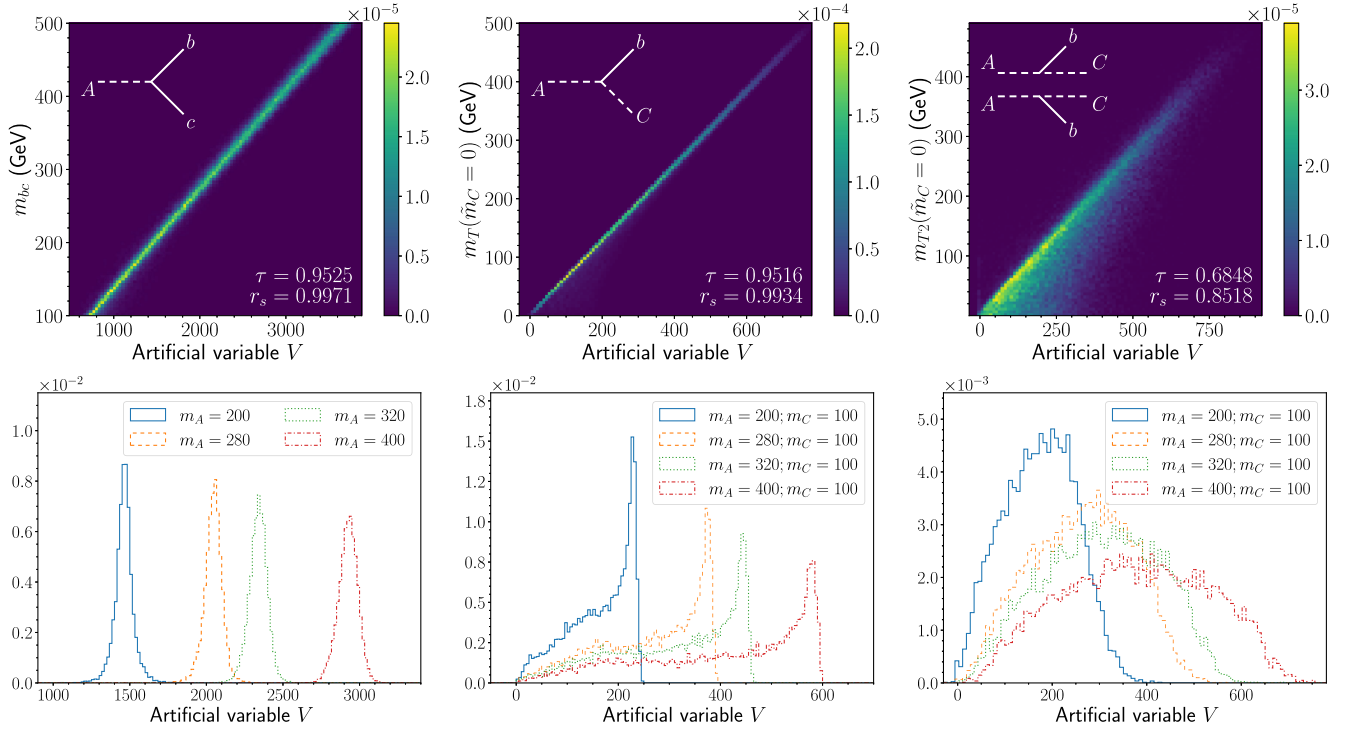


FIG. 2. Top row: correlation plots between the artificial variable V and a relevant human-engineered variable for each of the three examples considered in the text (m_{bc} , m_T , and m_{T2} from left to right). Bottom row: unit-normalized histograms of the corresponding artificial variable V for different mass inputs (10^4 events for each histogram).

distributions are highly sensitive to the parameter choice m_A and, if needed, V can be calibrated so that the peak location directly corresponds to m_A . The observed spread around the peak values in the histogram and the less-than-perfect correspondence between V and m_{bc} are due to limitations in the NN architecture and training.

B. Example 2: Semivisible two body decay

Next, we consider the semivisible two-body decay of a particle A into a massless visible particle b and a possibly massive invisible particle C , $A \rightarrow bC$, where A is singly produced (with zero transverse momentum). The parameter Θ is two dimensional: (m_A, m_C) . The event X is specified by the 4-momentum $p_b = \{E_b, \vec{p}_{bT}, p_{bz}\}$ of b and the missing transverse momentum, leading to $d_X = 6$.

We generate (m_A, m_C) by uniformly sampling (m_A, δ_m) in the region defined by $100 \text{ GeV} \leq m_A \leq 500 \text{ GeV}$ and $0 \leq \delta_m \leq m_A$, where $\delta_m \equiv (m_A - m_C)^2/m_A$. This choice of prior ensures that the relevant mass difference parameter in this event topology $\mu \equiv (m_A^2 - m_C^2)/m_A$ is adequately sampled in the range $[0, 500] \text{ GeV}$. For each sampled value of (m_A, m_C) , we generate an event as follows. The parent particle A is boosted along the beam axis $\pm z$ (with equal probability) to an energy chosen uniformly in the range $[m_A, 1500 \text{ GeV}]$. The particle A is decayed on shell into b and C , isotropically in its own rest frame. For data points in class 0, the values of (m_A, m_C) are replaced with

fake ones. The details of network architectures and training are the same as in Example 1.

The relevant variable for this event topology is the transverse mass m_T , which in our setup is given by

$$m_T(\tilde{m}_C) \equiv p_{bT} + \sqrt{p_{bT}^2 + \tilde{m}_C^2}, \quad (4)$$

where the choice of mass ansatz \tilde{m}_C for the mass of the invisible particle C does not affect the rank ordering of the events. For concreteness in what follows, we shall use $\tilde{m}_C = 0$. The corresponding heat map of the joint distribution (V, m_T) and unit-normalized distributions of the variable V for several choices of $m_A = \{200, 280, 320, 400\} \text{ GeV}$ and $m_C = 100 \text{ GeV}$ are shown in the middle panels of Fig. 2. Once again, we observe an almost perfect correlation between V and m_T , and a high sensitivity of the V distributions to the input masses.

C. Example 3: Symmetric semivisible two body decays

Finally, we consider the exclusive production at a hadron collider of two equal-mass parent particles A_1 and A_2 which decay semivisibly as $A_1 A_2 \rightarrow (b_1 C_1)(b_2 C_2)$. The parameter Θ is given by (m_A, m_C) , and the event X is described by the 4-momenta of b_1 and b_2 , and the missing transverse momentum, leading to $d_X = 10$.

The masses (m_A, m_C) are generated as in Example 2. To avoid fine tuning the network to the details of a particular

collider, we uniformly sampled the invariant mass $m_{A_1A_2}$ of the A_1A_2 system in the range $[2m_A, 1500 \text{ GeV}]$ and the laboratory-frame energy of the A_1A_2 system in the range $[m_{A_1A_2}, 2500 \text{ GeV}]$. The direction of the system was chosen to be along $\pm z$ with equal probability. The direction of A_1 is chosen isotropically in the rest frame of the A_1A_2 system. A_1 and A_2 are both decayed on shell, isotropically in their respective rest frames. For data points in class 0, the values of (m_A, m_C) are replaced with fake ones. The details of network architectures and training are the same as in Example 1.

The straightforward generalization of the idea of the transverse mass to the considered event topology leads to the stransverse mass variable $m_{T2}(\tilde{m}_C)$ [16]. In the upper-right panel of Fig. 2, we show a heat map of the joint distribution of $(V, m_{T2}(0))$, which reveals reasonably good, but not perfect, correlation, implying that the artificial event variable encapsulates information beyond m_{T2} . This could have been expected for the following two reasons: (1) unlike the previous two examples of singular variables with sharp features in their distributions, m_{T2} does not belong to the class of singular variables [10] and (2) m_{T2} only uses a subset of the available kinematic information, namely, the transverse momentum components. In contrast, the artificial kinematic variable can use all of the available information, and in a more optimal way. The lower-right panel of Fig. 2 displays unit-normalized distributions of the artificial variable for several choices of m_A and fixed $m_C = 100 \text{ GeV}$, again demonstrating the sensitivity of V to the mass spectrum.

IV. DISCUSSION AND OUTLOOK

We proposed a new deep-learning technique pictorially summarized in Fig. 1 which allows the construction of event variables from a set of training data produced from a given event topology. The novel component is the simultaneous training for varying parameters Θ , which allows the algorithm to capture the underlying phase space structure irrespective of the benchmark study point. This is the first such method for constructing event variables with neural networks and can be applied to other, more challenging event topologies in particle physics and beyond. In future applications of the method, one could enlarge the dimensionality of the latent space to $d_V > 1$ and supplement the training data with additional features, like tagging and timing information, etc. By manipulating the specifics of the generation of the training data, one can control what underlying physics effects are available for the machine to learn from and what physical parameters the machine-learned variable will be sensitive to. Our method opens the door to new investigations on interpretability and explainability by incorporating modern representation learning approaches like contrastive learning [21].

Code and data availability. The code and data that support the findings of this study are openly available in Ref. [22] under the directory named arXiv_2105.10126.

ACKNOWLEDGMENTS

We are indebted to the late Luc Pape for great insights and inspiration. We thank S. Gleyzer, K. Pedro, and J. Thaler for useful discussions. This work is supported in parts by U.S. DOE Grants No. DE-SC0021447 and No. DE-FG02-13ER41976. M. P. is supported by Basic Science Research Program through the National Research Foundation of Korea Research Grant No. NRF-2021R1A2C4002551. P. S. is partially supported by the U.S. Department of Energy, Office of Science, Office of High Energy Physics QuantISED program under the grants “HEP Machine Learning and Optimization Go Quantum”, Award No. 0000240323, and “DOE QuantISED Consortium QCCFP-QMLQCF”, Award No. DE-SC0019219. This manuscript has been authored by Fermi Research Alliance, LLC, under Contract No. DEAC02-07CH11359 with the U.S. Department of Energy, Office of Science, Office of High Energy Physics. The work of D. K. is supported by the DOE Grant No. DE-SC0010813.

APPENDIX

1. Intuition from information theory

Each event X carries some information about the underlying model parameter values from which it was produced. Some of this information could be lost when reducing the dimensionality of the data from d_X to d_V , as a consequence of the data processing inequality [23]. Good event variables minimize this information loss and efficiently retain the information about the underlying parameter values Θ [17,24]. This is precisely why the invariant mass m , the transverse mass m_T , and the stransverse mass m_{T2} have been widely used in particle physics for mass parameter measurements and for new physics searches.

The mutual information of V and Θ is given by

$$I(V; \Theta) \equiv \int_{\mathcal{V}} dv \int_{\Omega} d\theta p_{(V, \Theta)}(v, \theta) \ln \left[\frac{p_{(V, \Theta)}(v, \theta)}{p_V(v) p_{\Theta}(\theta)} \right], \quad (\text{A1})$$

where p_V and p_{Θ} are the probability distribution functions of V and Θ , respectively, and $p_{(V, \Theta)}$ is their joint distribution. \mathcal{V} and Ω are the domains of V and Θ , respectively. One can think of p_{Θ} as the prior distribution of Θ . The distributions $p_{(V, \Theta)}$ and p_V can then be derived from p_{Θ} and the conditional distribution $p_{V|\Theta}(v|\theta)$.

The mutual information $I(V; \Theta)$ quantifies the amount of information contained in V about Θ . Therefore, a good event variable V should have relatively high values of $I(V; \Theta)$. From Eq. (A1), one can see that $I(V; \Theta)$ is nothing but the Kullback-Leibler (KL) divergence from (a) the factorized distribution $p_V \otimes p_{\Theta}$ to (b) the joint distribution $p_{(V, \Theta)}$. The KL divergence, in turn, is a measure of how distinguishable the two distributions a and b are.

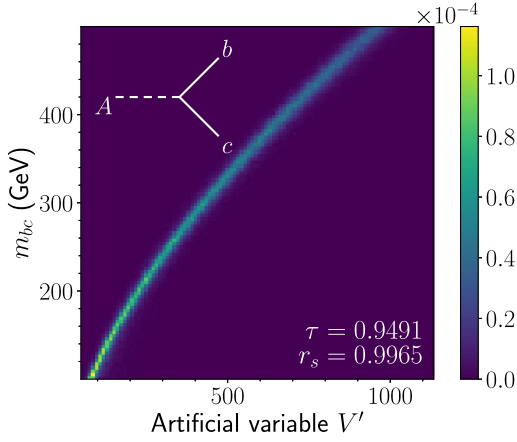


FIG. 3. The same as the upper left panel in Fig. 2, but using the modified activation function ϕ_{modified} in (A2).

2. Unexpected linear relationships

In the fully visible and semivisible two-body decay examples, the artificial variables learned by the neural networks appear to be linearly related to the corresponding theoretical variables, namely, m_{bc} and m_T (top-left and top-middle panels of Fig. 2). This feature is accidental, since one-to-one transformations of the artificial variable will leave its performance unaffected under our training principle. We found that the linearity persists under different runs of the pseudoexperiments with different random seeds. However, the relationship becomes nonlinear with a slight change in the activation function in the hidden layers of the event variable network from $\phi_{\text{ReLU}}(x) = \max[0, x]$ to

$$\phi_{\text{modified}}(x) = \max[0, x|x|^{0.1}]. \quad (\text{A2})$$

Training the modified network for the case of Example 1 now leads to the artificial variable V' shown in Fig. 3,

which clearly exhibits a nonlinear relationship with the theory variable m_{bc} .

3. Comparison to regression approaches

A popular ML technique for estimating per-data-point features is regression. A natural question that arises is whether sensitive kinematic variables can be learned by regressing collider events to the underlying model parameter values. Here, we will show that regression is not an effective technique for learning sensitive event observables, except in simplistic situations.

We begin by noting that, in order to use regression, the dimensionality of the NN output V should be the same as the dimensionality of the parameter Θ . However, in our Examples 2 and 3, the dimensionality of V is 1, and the dimensionality of Θ is 2. Such a mismatch in dimensionalities is incompatible with regression, which rules out the possibility of using regression in those cases.

Furthermore, even if the dimensionality of V is chosen to match that of Θ , regression will typically be ineffective for the following reasons. Let $V_{\text{reg}}(x)$ be the expected value of Θ under the training data distribution, conditional on $X = x$,

$$V_{\text{reg}}(x) = \frac{\int d\theta \theta p_{X|\Theta}(x|\theta) p_{\Theta}(\theta)}{\int d\theta p_{X|\Theta}(x|\theta) p_{\Theta}(\theta)}. \quad (\text{A3})$$

The event observable V_{reg} will be informative about Θ if the distribution of Θ conditional on X is localized around the true value of Θ . This is the case in Example 1, where the parameter m_A can exactly computed from a single event as the invariant mass m_{bc} . However, for event topologies featuring invisible particles, like Examples 2 and 3, this is not the case. In other words, performing regression for Examples 2 and 3 (with two-dimensional output) will not lead to the learning of a useful event variable.

[1] R. Bellman, *Dynamic Programming* (Princeton University Press, Princeton, NJ, 1957).
[2] T. Han, Collider phenomenology: Basic knowledge and techniques, [arXiv:hep-ph/0508097](https://arxiv.org/abs/hep-ph/0508097).
[3] A. J. Barr and C. G. Lester, A review of the mass measurement techniques proposed for the large hadron collider, *J. Phys. G* **37**, 123001 (2010).
[4] A. J. Barr, T. J. Khoo, P. Konar, K. Kong, C. G. Lester, K. T. Matchev, and M. Park, Guide to transverse projections and mass-constraining variables, *Phys. Rev. D* **84**, 095031 (2011).
[5] K. T. Matchev, F. Moortgat, and L. Pape, Dreaming awake: Disentangling the underlying physics in case of a SUSY-like discovery at the LHC, *J. Phys. G* **46**, 115002 (2019).

[6] I. W. Kim, Algebraic Singularity Method for Mass Measurement with Missing Energy, *Phys. Rev. Lett.* **104**, 081601 (2010).
[7] A. Rujula and A. Galindo, Measuring the W-boson mass at a hadron collider: A study of phase-space singularity methods, *J. High Energy Phys.* **08** (2011) 023.
[8] A. De Rujula and A. Galindo, Singular ways to search for the Higgs boson, *J. High Energy Phys.* **06** (2012) 091.
[9] D. Kim, K. T. Matchev, and P. Shyamsundar, Kinematic focus point method for particle mass measurements in missing energy events, *J. High Energy Phys.* **10** (2019) 154.
[10] K. T. Matchev and P. Shyamsundar, Singularity variables for missing energy event kinematics, *J. High Energy Phys.* **04** (2020) 027.

- [11] M. Feickert and B. Nachman, A living review of machine learning for particle physics, [arXiv:2102.02770](https://arxiv.org/abs/2102.02770).
- [12] S. Chang, T. Cohen, and B. Ostdiek, What is the machine learning?, *Phys. Rev. D* **97**, 056009 (2018).
- [13] T. Fauceett, J. Thaler, and D. Whiteson, Mapping machine-learned physics into a human-readable space, *Phys. Rev. D* **103**, 036020 (2021).
- [14] V. D. Barger, A. D. Martin, and R. J. N. Phillips, Perpendicular ν_e mass from W decay, *Z. Phys. C* **21**, 99 (1983).
- [15] J. Smith, W. L. van Neerven, and J. A. M. Vermaseren, The Transverse Mass and Width of the W Boson, *Phys. Rev. Lett.* **50**, 1738 (1983).
- [16] C. G. Lester and D. J. Summers, Measuring masses of semi-invisibly decaying particles pair produced at hadron colliders, *Phys. Lett. B* **463**, 99 (1999).
- [17] N. Tishby, F. C. Pereira, and W. Bialek, The information bottleneck method, *Proceeding of the 37th Annual Allerton Conference on Communication, Control and Computing* (1999), pp. 368–377, [arXiv:physics/0004057](https://arxiv.org/abs/physics/0004057).
- [18] Martín Abadi *et al.*, Large-scale machine learning on heterogeneous systems, <https://www.tensorflow.org/>.
- [19] Maurice G. Kendall, The treatment of ties in ranking problems, *Biometrika* **33**, 239 (1945).
- [20] D. Zwillinger and S. Kokoska, *CRC Standard Probability and Statistics Tables and Formulae* (Chapman & Hall, New York, 2000), Section 14.7.
- [21] P. Le-Khac, G. Healy, and A. Smeaton, Contrastive representation learning: A framework and review, *IEEE Access* **8**, 193907 (2020).
- [22] <https://gitlab.com/prasanthcakewalk/code-and-data-availability/>.
- [23] N. Beaudry, An intuitive proof of the data processing inequality, *Quantum Inf. Comput.* **12**, 432 (2012).
- [24] R. Iten, T. Metger, H. Wilming, L. del Rio, and R. Renner, Discovering Physical Concepts with Neural Networks, *Phys. Rev. Lett.* **124**, 010508 (2020).