Is the machine smarter than the theorist: Deriving formulas for particle kinematics with symbolic regression

Zhongtian Dong,[1,*] Kyoungchul Kong,[1,†] Konstantin T. Matchev,[2,‡] and Katia Matcheva[2,§]
[1]*Department of Physics and Astronomy, University of Kansas, Lawrence, Kansas 66045, USA*
[2]*Institute for Fundamental Theory, Physics Department, University of Florida, Gainesville, Florida 32611, USA*

(Received 18 November 2022; accepted 27 February 2023; published 13 March 2023)

We demonstrate the use of symbolic regression in deriving analytical formulas, which are needed at various stages of a typical experimental analysis in collider phenomenology. As a first application, we consider kinematic variables like the stransverse mass, M_{T2}, which are defined algorithmically through an optimization procedure and not in terms of an analytical formula. We then train a symbolic regression and obtain the correct analytical expressions for all known special cases of M_{T2} in the literature. As a second application, we reproduce the correct analytical expression for a next-to-leading order (NLO) kinematic distribution from data, which is simulated with a NLO event generator. Finally, we derive analytical approximations for the NLO kinematic distributions after detector simulation, for which no known analytical formulas currently exist.

DOI: 10.1103/PhysRevD.107.055018

I. INTRODUCTION

Being able to describe the data collected from the observations of various physical phenomena with simple analytical equations and formulas is the holy grail in theoretical physics—the physicists who are lucky enough to find such relationships typically get those laws named after them. In the era of big data, this task is becoming increasingly difficult for a human—the data is just too complex and/or very high dimensional. Recent advances in computer science and theoretical modeling have allowed us to entertain the idea that the discovery process could perhaps be automated (at least as a matter of principle) and novel laws of phenomenological behavior can be constructed entirely with a machine and without any human intervention [1–11]. A less ambitious, but still worthy, task is to simply let the machine rederive the known classical physics laws from data [12–15].

Spurred by the extensive recent research on symbolic learning in the machine learning (ML) community, the above program was recently successfully applied to examples in a wide range of physics areas, e.g., in astrophysics [13,16,17], in astronomy for the study of orbital dynamics [18,19] and exoplanet transmission spectroscopy [20], in collider physics [21–24], in materials science [25], and in behavioral science [26]. A common ML tool used in such studies is symbolic regression—an interpretable machine learning algorithm which searches the space of functions until it finds an algebraic expression that approximates the dataset well. While most current applications of symbolic regression are limited to low-dimensional data, the approach can be easily extended to higher-dimensional spaces by using a neural network as a proxy, as illustrated in Ref. [13] with the example of N-body problems.

The basic task in symbolic regression is to learn an analytical expression $f(\mathbf{x})$ given some labeled data $(\mathbf{x}, y)$, where $\mathbf{x}$ are input features, typically high-dimensional, and y is the output target label.[1] The learned function $f(\mathbf{x})$ can be scrutinized further in three aspects corresponding to fundamental principles of explainable AI [27]:

(i) *Explanation accuracy.* The first question is, how good is the result, i.e., how well does $f(\mathbf{x})$ fit the training data. Typical datasets are imperfect, due to noise, experimental errors, etc., in which case the fitted function will provide only an approximate description of the data. The fit is only expected to get worse as the errors in the data increase [28]. On the other hand, even if the data is perfect, the fit may

*cdong@ku.edu
†kckong@ku.edu
‡matchev@ufl.edu
§matcheva@ufl.edu

[1]In principle, y can also be high dimensional, however, for simplicity in this paper we shall focus on a single y.

be suboptimal due to factors related to the training of the symbolic regression itself. For example, one may have started with the wrong choice of basis functions, one may have unnecessarily restricted the functional complexity, or the training may simply not converge to the right answer. Our numerical examples considered in this paper shall illustrate many of those situations.

(ii) *Generalizability (knowledge limits).* A system should only operate under conditions for which it was designed. In the case of symbolic regression, extrapolating into the regions away from the training data in principle could be dangerous and should be handled with care. At the same time, physics laws are universal—if we find the correct relationship, it should be valid over the full allowed domain of the input variables. As shown below, this principle could be used to narrow down the list of candidate analytical expressions.

(iii) *Explainability (meaningful).* A system must provide explanations that are understandable to the intended consumers, and furthermore, these explanations must correctly reflect the reason for generating the output and/or the system's process. A common criticism of deep learning models is that they are black boxes which provide little insight into the fundamental processes that are at work. A symbolic regression is arguably the most intuitive and meaningful approach from the point of view of a theorist —theorists are used to working with analytical formulas and from experience can often find the physical interpretations of the various terms in an analytical expression.

In this paper we consider several applications of symbolic regression to problems in collider physics and specifically particle kinematics. These examples will be presented in order of increasing difficulty, starting from simple cases in which the exact theoretical formula is known. Nevertheless, rederiving those answers with a symbolic regression will serve as an important illustration and validation of our procedure.

Symbolic regression is a promising machine learning method that searches over a large space of functions until it finds an expression which is both (a) relatively simple, and (b) a good fit to the training data. Because the evolutionary algorithm requires diversity in order to effectively explore the search space, the result of the symbolic regression is a collection of several high-scoring models, which need to be scrutinized by the user to identify an approximation that offers a good trade-off between accuracy and simplicity. At the same time, training a symbolic regression is a computationally expensive process, since the function space to be scanned is in principle infinite. This is why, as a proof-of-concept, in this paper we shall limit ourselves to a few simple examples, which do not require a high-performance cluster, and can be done on a personal laptop.

To train a symbolic regression, we shall make use of the PySR software package [29], which models the data set with a graph neural network before applying symbolic regression to fit different internal parts of the learned model that operate on reduced dimension representations [13]. We shall not attempt any hyperparameter optimization and for the most part will use the default configuration in the PySR version 0.10.1 distribution.

The paper is organized as follows. In Sec. II we shall use parton-level data (in the narrow-width approximation) to rederive some known analytical results for the Cambridge M_{T2} variable. In Sec. III we repeat the same exercise and try to derive the splitting function $\mathcal{F}(E,\theta)$ for the ISR photon at an e^+e^- collider, which gives us the probability to radiate a photon with a given energy E and a given polar angle θ. We perform two versions of the exercise. First, in Sec. III A we sample the $\mathcal{F}$ function directly to create a perfect data sample with no statistical fluctuations. Then, in Sec. III B we use a sample of Monte Carlo (MC) generated events to first obtain a binned estimate of $\mathcal{F}$ (which is subject to statistical errors) before applying the symbolic regression. In Sec. III C we perform a more realistic analysis by adding detector resolution effects. Section IV is reserved for a summary and outlook.

II. DERIVING ANALYTIC EXPRESSIONS FOR ALGORITHMICALLY DEFINED KINEMATIC VARIABLES: M_{T2}

A standard analysis of particle physics data (such as events from collisions at the Large Hadron Collider (LHC) at CERN) involves the study of distributions of kinematic variables, which are typically defined in terms of the energies and momenta of the particles observed in the detector (for recent reviews of the kinematic variables commonly used in collider phenomenology, see [30–33]). Many of these variables, e.g., invariant mass, missing transverse momentum, etc., are defined in terms of simple analytical expressions and can be readily computed from the collections of particle 4-momenta in the event. However, there also exist another class of kinematic variables, which are defined algorithmically, i.e., through a well-defined optimization procedure which involves the minimization (or maximization) of a relevant kinematic function. In that case, the kinematic variable is a quantity which can be computed only once the algorithm has converged, and typically there is no *a priori* known analytical expression for it in the general case. Examples of such variables include many traditional event shape variables (thrust, sphericity, etc.) [33,34], some modern substructure variables like N-jettiness [35] and N-subjettiness [36], and many others. Another large class of algorithmic variables which have received a lot of attention in the last 15 years, are the so-called constrained

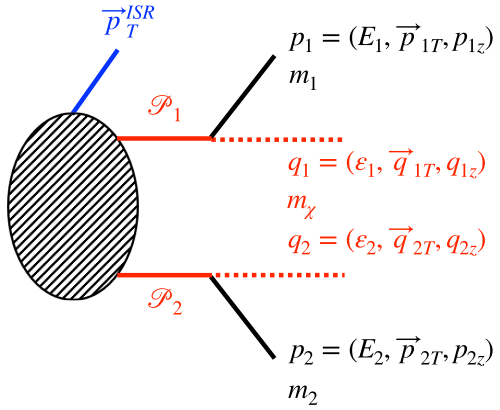

FIG. 1. The generic $\not{E}_T$ event topology applicable to the M_{T2} variable. The parent particles $\mathcal{P}_1$ and $\mathcal{P}_2$ are produced in association with some visible upstream transverse momentum $\vec{p}_T^{\,\mathrm{ISR}}$. The remaining visible final state particles are divided into two groups (solid black lines), with 4-momenta p_1 and p_2 and masses m_1 and m_2, respectively. The two invisible final state particles (red dashed lines) have 4-momenta q_1 and q_2 and are assumed to have a common mass m_χ.

mass variables which are computed via constrained minimization of a kinematic function of the particle 4-momenta [30–33]. The minimization is typically performed over the energy and momentum components of invisible particles in the event (neutrinos or dark matter candidates). Examples of constrained mass variables include the Oxbridge variable M_{T2} [37,38] and its 4-dimensional generalization M_2 [39–41], the variable M_{2C} [42], etc. In this paper, for concreteness we shall focus on the well-known M_{T2} variable [37,38], which is algorithmically defined and does not have a known analytical formula in the general case. The advantage of M_{T2} is that there exist formulas for special cases of certain momentum configurations for the visible final state particles. As a warm-up, in this section we shall use these special M_{T2} cases to validate and illustrate the use of symbolic regression for the purpose of deriving new formulas for computing kinematic variables.

A well-motivated class of new physics models which generically predict a $\not{E}_T$ signature, are models with dark-matter candidates. In such models, the lifetime of the dark-matter particle is typically protected by an exact discrete symmetry, which implies that the collider signals will involve not one, but *two* decay chains, each terminating in a dark-matter particle invisible in the detector. The simplest $\not{E}_T$ event topology of this type is illustrated in Fig. 1, where two identical parent particles $\mathcal{P}_1$ and $\mathcal{P}_2$ are produced with additional objects, typically from initial state radiation (ISR). Each parent particle $\mathcal{P}_i$, $(i = 1, 2)$, decays to a visible particle system with invariant mass m_i and 4-momentum $p_i = (E_i, \vec{p}_{iT}, p_{iz})$ and an invisible particle χ_i with 4-momentum $q_i = (\varepsilon_i, \vec{q}_{iT}, q_{iz})$. The masses of the invisible particles are *a priori* unknown. Here, we shall assume that the invisible particles χ_1 and χ_2 are identical and have a common mass m_χ. Momentum conservation in the transverse plane implies

$$\vec{q}_{1T} + \vec{q}_{2T} = \not{\vec{p}}_T, \tag{1}$$

where the missing transverse momentum vector is given by

$$\not{\vec{p}}_T = -(\vec{p}_{1T} + \vec{p}_{2T}) - \vec{p}_T^{\,\mathrm{ISR}}. \tag{2}$$

The transverse momentum vectors $\vec{p}_{iT}$, $\vec{q}_{iT}$, $\not{\vec{p}}_T$, and $\vec{p}_T^{\,\mathrm{ISR}}$ are illustrated in Fig. 2.

The two main ingredients in the M_{T2} calculation are the transverse masses $M_{T\mathcal{P}_i}$ of the two parent particles $\mathcal{P}_i$,

$$M_{T\mathcal{P}_i}(\vec{q}_{iT}, m_\chi) = \sqrt{m_i^2 + m_\chi^2 + 2(E_{iT}\varepsilon_{iT} - \vec{p}_{iT}\cdot\vec{q}_{iT})}, \tag{3}$$

where the transverse energies are defined as

$$E_{iT} = \sqrt{\vec{p}_{iT}^{\,2} + m_i^2}, \qquad \varepsilon_{iT} = \sqrt{\vec{q}_{iT}^{\,2} + m_\chi^2}. \tag{4}$$

The M_{T2} is defined as [37,38]

$$M_{T2}(\tilde{m}) \equiv \min_{\vec{q}_{1T},\vec{q}_{2T}} \{\max\left[M_{T\mathcal{P}_1}(\vec{q}_{1T}, \tilde{m}), M_{T\mathcal{P}_2}(\vec{q}_{2T}, \tilde{m})\right]\},$$
$$\not{\vec{p}}_T = \vec{q}_{1T} + \vec{q}_{2T}, \tag{5}$$

where the *a priori* unknown invisible daughter mass m_χ has been replaced with a test mass parameter $\tilde{m}$. This construction guarantees that on an event-by-event basis the computed value of M_{T2} does not exceed the mass of the parent $\mathcal{P}_i$.

In general, the minimization in (5) has to be done numerically. However, for certain special cases, analytical solutions have been derived [38,43–47]. In this section, we shall apply symbolic regression to rederive several of those analytical solutions. Having such analytical solutions is motivated by two reasons—first, a purely mathematical interest in the behavior and properties of the M_{T2} function,

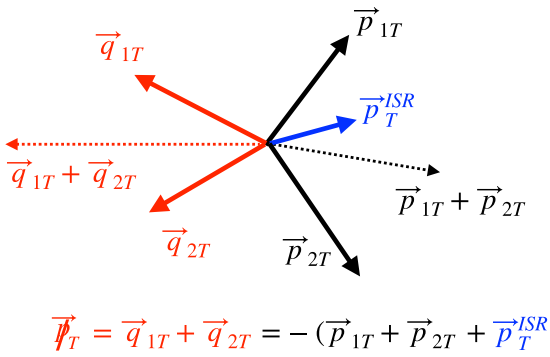

FIG. 2. A generic configuration of the transverse momentum vectors $\vec{p}_{iT}$, $\vec{q}_{iT}$, $\not{\vec{p}}_T$, and $\vec{p}_T^{\,\mathrm{ISR}}$ entering the definition (5) of M_{T2}.

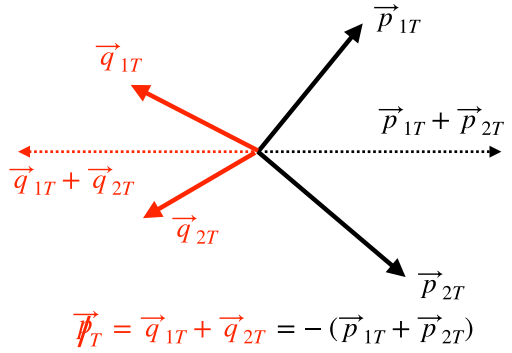

FIG. 3. The special momentum configuration with $\vec{p}_T^{\,\mathrm{ISR}} = 0$ considered in Sec. II A. The missing transverse momentum $\not{\vec{p}}_T$ exactly balances the total visible transverse momentum $\vec{p}_{1T}+\vec{p}_{2T}$.

and second, the potential use of M_{T2} as a trigger variable, which is extensively discussed in Appendix A of Ref. [43]. For this purpose, the computation of M_{T2} should be fast enough in order to fit within the trigger bandwidth; the fastest currently known iterative algorithm for M_{T2} is capable of ∼200 kHz evaluation rates at machine precision [48], which is still much slower than computing an analytical formula.

A. The case of no upstream momentum

The minimization in Eq. (5) may result in one of two distinct possibilities: the transverse masses of the parents are equal, $M_{T\mathcal{P}_1} = M_{T\mathcal{P}_2}$, which is known as the balanced solution, or the transverse masses of the parents are unequal, $M_{T\mathcal{P}_1} \neq M_{T\mathcal{P}_2}$, known as the unbalanced case. The analytical expression for M_{T2} in the unbalanced case is simply given by Eq. (3) [38], so the balanced case is the only one we need to worry about. Unfortunately, there is no known analytical formula for the balanced M_{T2} solution for generic momentum configurations like the one in Fig. 2. However, for the special momentum configuration shown in Fig. 3, where $\vec{p}_T^{\,\mathrm{ISR}} = 0$, the analytical formula for the balanced M_{T2} solution is known to be [45,46]

$$M_{T2}^2(\tilde{m}) = \tilde{m}^2 + A_T + \sqrt{\left(1+\frac{4\tilde{m}^2}{2A_T - m_1^2 - m_2^2}\right)(A_T^2 - m_1^2 m_2^2)}, \quad (6)$$

where A_T is a convenient shorthand notation introduced in [47]

$$A_T = E_{1T}E_{2T} + \vec{p}_{1T}\cdot\vec{p}_{2T}. \quad (7)$$

In order to avoid always taking an extra square root, from now on for convenience we shall focus on the M_{T2} variable *squared*.

In what follows an important attribute of an analytical expression will be the so-called complexity C (defined as the number of leaf nodes in the binary tree representing the analytical expression) [13,29]. Clearly, functions of higher complexity in turn will demand more extensive computational resources, including longer computational times. The function (6) is of complexity 24, which is already a formidable challenge. Given (a) our rather modest computational budget, and (b) our goal of to demonstrate the method as a proof of principle, here we shall limit ourselves to four simple, yet nontrivial, special cases of (6) which have lower complexity, namely

(i) *Massless visible and massless invisible final state particles.* Setting $m_1 = m_2 = 0$ and $\tilde{m} = 0$ in (6), we obtain

$$M_{T2}^2(\tilde{m}) = 2A_T = 2(E_{1T}E_{2T} + \vec{p}_{1T}\cdot\vec{p}_{2T}). \quad (8)$$

(ii) *Massless visible and massive invisible final state particles.* Substituting $m_1 = m_2 = 0$ and $\tilde{m} \neq 0$ into (6), we get

$$M_{T2}^2(\tilde{m}) = \tilde{m}^2 + A_T + \sqrt{A_T(A_T + 2\tilde{m}^2)}. \quad (9)$$

(iii) *Equally massive visible and massless invisible final state particles.* Alternatively, choosing $m_1 = m_2 = m \neq 0$ and $\tilde{m} = 0$ in (6), we find

$$M_{T2}^2(\tilde{m} = 0) = A_T + \sqrt{A_T^2 - m^4}. \quad (10)$$

(iv) *Equally massive visible and massive invisible final state particles.* Finally, choosing $m_1 = m_2 = m \neq 0$ and $\tilde{m} \neq 0$ in (6), we find

$$M_{T2}^2(\tilde{m}) = \tilde{m}^2 + A_T + \sqrt{(A_T - m^2 + 2\tilde{m}^2)(A_T + m^2)}. \quad (11)$$

We shall now try to reproduce[2] each of those expressions (8)–(11) with the symbolic regression algorithm implemented in PySR [13,29]. For this purpose, we shall generate a large sample of events, compute the target variable M_{T2}^2 numerically from the defining formula (5), using the Python code MT2 1.2.0 [48], and then ask the symbolic regression to "discover" the analytical results (8)–(11).

In the case of no upstream momentum ($\vec{p}_T^{\,\mathrm{ISR}} = 0$) considered in this subsection, there are seven input degrees of freedom, which naively can be taken to be the two transverse momentum components of each visible particle, $\vec{p}_{1T}$ and $\vec{p}_{2T}$, their masses m_1 and m_2, and the invisible test

[2]Note that all of these results would have been completely novel prior to 2007, i.e., only 15 years ago.

mass $\tilde{m}$. In principle, one can use this set of primitive variables as inputs to the symbolic regression, but the disadvantage is that the machine will need to learn the physics principles from scratch. In order to improve and speed up the performance of the symbolic regression, it is crucial to use an optimized set of input variables which reflects the underlying physics principles of the problem. One possibility is to use dimensional analysis and feed only groups of variables which have the proper physics dimensions [20]. In our case here, since we are looking for a formula for a mass-squared quantity, M_{T2}^2, it makes sense if all of our input variables have mass-dimension 2, otherwise the complexity of the function will increase, making it more difficult for the symbolic regression to find it. Furthermore, we know that the answer must be rotationally invariant, back-to-back boost invariant [46,47], and symmetric with respect to permutations among the visible particles $(1 \leftrightarrow 2)$. These considerations restrict the relevant set of variables to fewer degrees of freedom, which further improves the performance of the symbolic regression. For example, in the case of the function (8) we shall consider as inputs the set $\{E_{1T}E_{2T}, \vec{p}_{1T}\cdot\vec{p}_{2T}, |\vec{p}_{1T}+\vec{p}_{2T}|\}$, in terms of which the answer (8) is only of complexity 5. Similarly, in the case of the function (9), we shall input the values of A_T and $\tilde{\mu}\equiv\tilde{m}^2$, which results in complexity 12. Then for the function (10) we shall use the values of A_T and $\mu\equiv m^2$ as inputs, and the corresponding complexity is 8. Finally, for the function (11) we shall feed in $\{A_T,\mu,\tilde{\mu}\}$ and the complexity is 15.

In order to train the symbolic regression, we need to create suitable training data. For the exercise in this subsection, we sample the transverse momenta $\vec{p}_{1T}$ and $\vec{p}_{2T}$ of the two visible particles, which also fixes the missing transverse momentum vector as $\not{\vec{p}}_T=-(\vec{p}_{1T}+\vec{p}_{2T})$ (see Fig. 3). From those momenta we compute the input features (of mass dimension 2) to the symbolic regression as explained above. The target variable M_{T2}^2 is then calculated numerically with the MT2 code [48]. This exercise is performed four different times, depending on the choice for the mass parameters μ and $\tilde{\mu}$ being zero or nonzero, leading to the four different cases in Eqs. (8)–(11).

In each of these four cases, we train the PySR symbolic regression algorithm on 10,000 events. We mostly use the default hyperparameter configuration in the PySR distribution. Due to the relatively high complexity of our functions, we increased the number of iterations to 10. We allow for the simple arithmetic operators addition (+), subtraction (−), multiplication (∗), division (/), and square root ($\sqrt{\cdot}$). The loss function is the mean squared error (MSE). The typical training time on a single CPU with the default PySR settings was on the order of a few minutes.

The output from a typical PySR run is a set of functions of increasing complexity C, together with their MSE and score. The score is calculated by the fractional drop in the MSE over the increase in the complexity from the next best model [13]

$$\text{Score}=-\frac{\Delta\log(\text{MSE})}{\Delta c}. \tag{12}$$

The results from the four exercises in this subsection are displayed in Table I. In each case, the symbolic regression was able to eventually reproduce the correct functional dependence, once the required complexity was reached.

TABLE I. Results from the M_{T2} exercise with no ISR considered in Sec. II A. In each case, we show the best fitted functions at several representative values of the complexity. The correct answers are given by Eqs. (8)–(11), with the substitutions $\tilde{m}^2\to\tilde{\mu}$ and $m^2\to\mu$.

Case	Complexity	Fitted function	MSE	Score
$\mu=0,\ \tilde{\mu}=0$	1	$\vec{p}_{T1}\cdot\vec{p}_{T2}$	7×10^{7}	0
	3	$\lvert\vec{p}_{T1}+\vec{p}_{T2}\rvert^2$	2.2×10^{6}	1.74
	5	$2(\vec{p}_{T1}\cdot\vec{p}_{T2}+E_{T1}E_{T2})$	0	∞
$\mu=0,\ \tilde{\mu}\neq0$	9	$2A_T+1.8(\tilde{\mu}-3.91)$	7.345×10^{3}	6.73×10^{-3}
	11	$2A_T+\tilde{\mu}/0.556-9.51$	7.316×10^{3}	1.985×10^{-3}
	13	$2A_T+\tilde{\mu}+0.10\tilde{\mu}A_T^{1/4}$	6.377×10^{3}	6.870×10^{-2}
	14	$\tilde{\mu}+A_T+\sqrt{A_T^2+2\tilde{\mu}A_T}$	0	∞
$\mu\neq0,\ \tilde{\mu}=0$	5	$2.02(A_T-133.35)$	6.67×10^{4}	0.23
	7	$2.03A_T-0.44\mu$	3.39×10^{4}	0.34
	9	$2A_T-\mu^2/A_T$	2.06×10^{4}	0.25
	10	$A_T+\sqrt{(A_T-\mu)(A_T+\mu)}$	0	∞
$\mu\neq0,\ \tilde{\mu}\neq0$	12	$A_T+\sqrt{2}\sqrt{(A_T-\mu+\tilde{\mu})A_T}$	3.90×10^{4}	0.28
	13	$A_T+\sqrt{2}\sqrt{(A_T-0.99\mu)A_T}$	3.34×10^{4}	0.16
	14	$A_T+\sqrt{(A_T-\mu+2.74\tilde{\mu})(A_T+\mu)}$	3.488×10^{3}	2.26
	16	$\tilde{\mu}+A_T+\sqrt{(A_T-\mu+2\tilde{\mu})(A_T+\mu)}$	1.12×10^{-6}	10.93

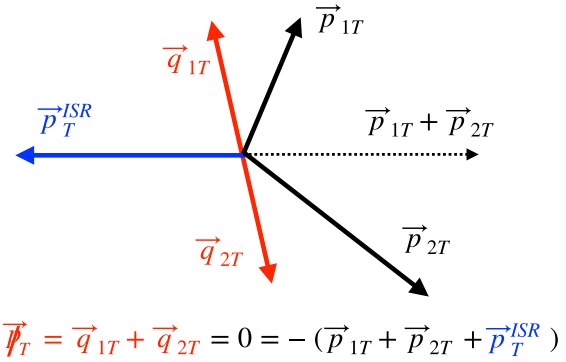

FIG. 4. The special momentum configuration with $\vec{\not{p}}_T = 0$ considered in Sec. II B. The invisible particles have equal and opposite momenta in the transverse plane. As a result, $\vec{p}_T^{\rm ISR}$ exactly balances the total visible transverse momentum $\vec{p}_{1T}+\vec{p}_{2T}$.

We note that Eq. (11) turned out to be more challenging than the others, so for that case we increased the population-size parameter to 50 and used 40,000 training samples with batchsize 5,000.

Note that sometimes we obtain an equivalent expression of slightly higher complexity. For example, in the case of Eq. (9), the answer has expanded the parentheses under the square root, which leads to an equivalent expression, but formally increases the complexity of the function to 14. Also note that since in this exercise the data is sampled from the exact function (no noise or errors), the MSE for the right answer is zero (or very close to it) and the score is infinite (or very large). The successful replication of the known special cases (8)–(11) validates our use of symbolic regression as implemented in PySR and motivates us to consider more realistic examples in the following sections.

B. The case of no missing transverse momentum

Recently, Ref. [43] pointed out a new special case which also allows an analytical formula for M_{T2}. Its momentum configuration is shown in Fig. 4, where the two invisible momenta are equal and opposite, and as a result $\vec{\not{p}}_T = 0$. This cancellation of the invisible momenta is purely accidental, which is why this case is mostly of academic interest—there will be very few events (if any) of this type in the data. Nevertheless, for completeness we shall explore this situation as well.

For simplicity, we shall focus on the case when the masses of the visible final state particles are the same, i.e., $m_1 = m_2 \equiv m$. The formula for M_{T2} is given by [43]

$$M_{T2}^2(\tilde{\mu}) = \tilde{\mu} + \mu + \sqrt{2\tilde{\mu}(\mu + E_{1T}E_{2T} + \vec{p}_{1T}\cdot\vec{p}_{2T})} \quad (13a)$$

$$= \tilde{\mu} + \mu + \sqrt{2\tilde{\mu}(\mu + A_T)}, \quad (13b)$$

where to facilitate later comparisons to the PySR output, we have used the mass squared parameters $\tilde{\mu} = \tilde{m}^2$ and $\mu = m^2$.

Once again, we may consider several special cases, depending on the masses of the visible and invisible particles. The case of massless invisible particles ($\tilde{\mu} = 0$) leads to a trivial function $M_{T2}^2 = \mu$ and will not be considered further. On the other hand, the case of massless visible particles [$\mu = 0$ in (13)] gives a nontrivial function

$$M_{T2}^2(\tilde{\mu}) = \tilde{\mu} + \sqrt{2\tilde{\mu}(E_{1T}E_{2T} + \vec{p}_{1T}\cdot\vec{p}_{2T})}. \quad (14)$$

Keeping in mind that the answer must be symmetric with respect to interchanging $1 \leftrightarrow 2$, we can use the set of mass-dimension 2 variables $\{E_{1T}E_{2T}, \vec{p}_{1T}\cdot\vec{p}_{2T}, \tilde{\mu}\}$, in terms of which the function (14) is of complexity 10.

Proceeding as in Sec. II A, we train a symbolic regression with the default parameter configuration in PySR on 10,000 events in the $\vec{\not{p}}_T = 0$ configuration of Fig. 4. We repeat the exercise twice—once for massless visible particles ($\mu = 0$) and then again for massive visible particles ($\mu \neq 0$). The value of M_{T2} is always computed with massive invisible particles ($\tilde{\mu} \neq 0$). The results are displayed in Table II. In the case $\mu = 0$, the exact formula (14) is reproduced, albeit in a mathematically equivalent form of slightly higher complexity. In the massive case ($\tilde{\mu} \neq 0$), our set of input variables was taken to be $\{\mu, \tilde{\mu}, A_T\}$, and the

TABLE II. Results from the M_{T2} exercise considered in Sec. II B for the momentum configuration with $\vec{\not{p}}_T = 0$ displayed in Fig. 4.

Case	Complexity	Fitted function	MSE	Score
$\mu = 0, \tilde{\mu} \neq 0$	8	$\sqrt{\tilde{\mu}(\vec{p}_{T1}\cdot\vec{p}_{T2} + E_{T1}E_{T2})/0.468}$	26.9	2.946
	10	$\tilde{\mu} + \sqrt{\tilde{\mu}(\vec{p}_{T1}\cdot\vec{p}_{T2} + E_{T1}E_{T2})/0.5}$	2.91×10^{-5}	6.87
	12	$\tilde{\mu} + \sqrt{2\tilde{\mu}(\vec{p}_{T1}\cdot\vec{p}_{T2} + E_{T1}E_{T2})} + 0.005$	1.25×10^{-5}	4.24×10^{-1}
	13	$\tilde{\mu} + (\sqrt{\tilde{\mu}})\sqrt{\vec{p}_{T1}\cdot\vec{p}_{T2} + E_{T1}E_{T2}}\sqrt{2}$	0	∞
$\mu \neq 0, \tilde{\mu} \neq 0$	6	$\sqrt{\tilde{\mu}A_T/0.22}$	5.33×10^5	9.168×10^{-1}
	8	$(\mu + \sqrt{\tilde{\mu}A_T/0.296})$	1.64×10^5	5.89×10^{-1}
	10	$1.29(\tilde{\mu} + \mu + \sqrt{\tilde{\mu}A_T})$	7.08×10^3	1.57
	12	$\tilde{\mu} + \mu + \sqrt{2\tilde{\mu}(\mu + A_T)}$	0	∞

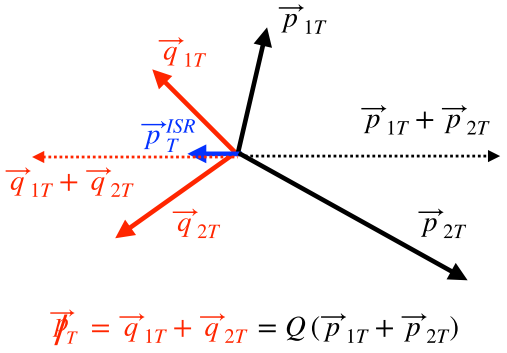

FIG. 5. The special balanced momentum configuration $\vec{\not{p}}_T = Q(\vec{p}_{1T}+\vec{p}_{2T})$ considered in Sec. II C. In general, the proportionality factor Q can be positive or negative, while $Q=0$ reduces to the case considered in Sec. II A. Note that $\vec{p}_T^{\,\rm ISR}$ is also necessarily collinear with $\vec{\not{p}}_T$ and the total visible transverse momentum $\vec{p}_{1T}+\vec{p}_{2T}$.

result was again successful, reproducing the correct function (13b) at complexity level 12.

C. The collinear momentum configuration

A second special case discussed in Ref. [43] is that of the collinear momentum configuration in Fig. 5, where the three transverse vectors $\vec{\not{p}}_T$, $\vec{p}_T^{\,\rm ISR}$ and $\vec{p}_{1T}+\vec{p}_{2T}$ all lie along the same line in the transverse plane. Reference [43] parametrized this case through a proportionality factor Q defined by

$$\vec{\not{p}}_T = Q(\vec{p}_{1T}+\vec{p}_{2T}) \equiv Q\vec{p}_{T12}, \tag{15}$$

where we have introduced a shorthand notation $\vec{p}_{T12}$ for the total visible transverse momentum $\vec{p}_{1T}+\vec{p}_{2T}$. Note that Q is unbounded and can take both positive and negative values, i.e., $-\infty < Q < \infty$. For definiteness, in Fig. 5 we show the case of $Q<0$.

Like before, we only consider the case of $\mu=0$ in which case the formula is

$$M_{T2}^2(\tilde{\mu}) = \tilde{\mu} - QA_T + \sqrt{A_T(2\tilde{\mu}+Q^2A_T)}. \tag{16}$$

For completeness we also consider separately the special case $\tilde{\mu}=0$ when the formula simplifies to

$$M_{T2}^2(\tilde{\mu}) = -QA_T + |QA_T| = \begin{cases} 0, & \text{for } Q>0, \\ 2A_T|Q| = 2A_T\frac{|\not{p}_T|}{|\vec{p}_{T12}|}, & \text{for } Q<0. \end{cases} \tag{17}$$

For simplicity we shall only test the nontrivial case given by the second line in (17).

Training PySR as before, we find the results shown in Table III. In the case of $\tilde{\mu}=0$, we choose the variables A_T, $|\vec{\not{p}}_T|$, and $|\vec{p}_{T12}|$ as our input features, while in the case of $\tilde{\mu}\neq 0$, our input features were A_T, Q, and $\tilde{\mu}$. We see that the correct answers are reproduced at complexities 7 and 18, respectively.

III. DERIVING ANALYTIC EXPRESSIONS FOR NLO KINEMATIC DISTRIBUTIONS

As our second example, we shall apply symbolic regression to learn the shapes of kinematic distributions at next-to-leading order (NLO). For simplicity, we shall consider the simplest possible process at leading order (LO), namely, the pair-production $e^+e^-\to\chi\chi$ of two invisible particles at a lepton collider with CM energy $\sqrt{s}$. Here the χ particles can be neutrinos or stable BSM dark matter candidates that escape undetected. In order to observe such events, we have to tag with a photon from initial state radiation (ISR), i.e., consider the NLO process $e^+e^-\to\chi\chi+\gamma$ [49].[3]

In general, there is no model-independent exact theoretical prediction for the resulting kinematic distribution of the ISR photon (for model-dependent studies, see [51–54]). However, if the emitted photon is either *soft* or *collinear* with the incoming electron or positron, soft/collinear factorization theorems provide an approximate model-independent relation between the LO and NLO differential cross sections,

$$\frac{d\sigma(e^+e^-\to\chi\chi+\gamma)}{dx\,d\cos\theta} \approx \mathcal{F}(x,\sin\theta)\hat{\sigma}(e^+e^-\to\chi\chi), \tag{18}$$

where θ is the angle between the photon direction and the direction of the incoming electron beam, and the dimensionless quantity

$$x = \frac{2E_\gamma}{\sqrt{s}} \tag{19}$$

is a measure of the photon energy E_γ, normalized by the beam energy $\sqrt{s}/2$. Further, $\hat{\sigma}$ is the LO χ pair-production cross section evaluated at the reduced center of mass energy, $\hat{s}=(1-x)s$. Finally, $\mathcal{F}$ denotes the splitting function

$$\mathcal{F}(x,\sin\theta) = \frac{\alpha}{\pi}\frac{1+(1-x)^2}{x}\frac{1}{\sin^2\theta}, \tag{20}$$

which upon integration over θ, reproduces the familiar Weizsacker-Williams distribution function. The factor $\mathcal{F}$ is universal; it does not depend on the nature of the (electrically neutral) particles produced in association with the photon.

[3]The analysis of this section is also applicable to hadron colliders like the LHC, where the LO process $pp\to\chi\chi$ can be tagged with an ISR jet as $pp\to\chi\chi+$ jet [50].

TABLE III. Results from the M_{T2} exercise with the collinear momentum configuration in Sec. II C.

Case	Complexity	Fitted function	MSE	Score
$\mu = 0, \tilde{\mu} = 0$	3	$\lvert\not{\vec{p}}_T\rvert\lvert\vec{p}_{T12}\rvert$	3.13×10^5	1.59
	5	$\lvert\not{\vec{p}}_T\rvert(\lvert\vec{p}_{T12}\rvert - 4.04)$	2.11×10^5	0.20
	7	$2A_T\lvert\not{\vec{p}}_T\rvert/\lvert\vec{p}_{T12}\rvert$	8.65×10^{-8}	14.26
$\mu = 0, \tilde{\mu} \neq 0$	14	$\tilde{\mu} - QA_T + \sqrt{A_T(\tilde{\mu} + A_T)/0.50}$	52.83	1.08
	16	$\tilde{\mu} - QA_T/0.897 + \sqrt{A_T(2\tilde{\mu} + A_T)}$	44.51	8.57×10^{-2}
	18	$\tilde{\mu} - QA_T + \sqrt{A_T(2\tilde{\mu} + Q^2A_T)}$	3.86×10^{-5}	6.98

Note that the normalization of (18) depends on the fine structure coupling constant α appearing in (20). Our main goal in this section will be to apply symbolic regression and learn *the shape* of the splitting function (20) from a sample of MC events generated either according to the soft/collinear approximation (18) (see Secs. III B 1 and III C 1) or using the full matrix element in a specific model (see Secs. III B 2 and III C 2). In Sec. III B (Sec. III C) the exercise will be performed without (with) detector effects, i.e., smearing the photon energy according to the calorimeter resolution.

A. Warm-up toy exercise: Learning the splitting function directly

First we begin with a toy exercise where we create the training data by sampling the function $\mathcal{F}$ directly, i.e., for a given choice of x and θ, we compute the target variable y directly from Eq. (20). In other words, our training dataset will be the set

$$\left(x, \sin\theta, \frac{\mathcal{F}(x, \sin\theta)}{\alpha/\pi}\right), \tag{21}$$

where for simplicity we have factored out the constant α/π. One can view this exercise as corresponding to the case of infinite MC statistics in the absence of any detector effects.

We generate training data (21) by sampling $x \in [0.1, 1]$ and $\sin\theta \in [0.1, 1]$ on a 100×100 grid. Using the default parameter options in PySR, we obtain the results shown in Table IV for the target function in this case, $\frac{\pi}{\alpha}\mathcal{F}(x, \sin\theta)$. We see that the correct analytical expression, $(1 + (1 - x)^2)/(x \sin^2\theta)$, is recovered at complexity 11, as indicated by the sharp drop of the MSE loss (note also the drastic improvement in the score at complexity 11). This is pictorially illustrated in Fig. 6, which shows the evolution of the MSE loss as a function of complexity.

TABLE IV. Results from the warm-up symbolic regression exercise considered in Sec. III A.

Complexity	Fitted function	MSE	Score
5	$(3.73)/\sin^2\theta$	5.56×10^3	0.14
7	$1.60/(x \sin^2\theta)$	2.08×10^2	1.64
9	$(-1.15 + \frac{1.89}{x})/\sin^2\theta$	8.63	1.59
11	$(x - 2 + \frac{2}{x})/\sin^2\theta$	5.71×10^{-11}	12.87

B. Learning from gen-level MC data

Having validated our symbolic regression procedure with the toy example of the previous subsection, we shall now modify this exercise, making it more realistic in several ways:

(i) Instead of considering infinite statistics, we shall now limit ourselves to a finite event sample, thereby introducing statistical errors in the target values of the function which are used for the training of the symbolic regression.

(ii) In the toy example of Sec. III A, we generated the training data by simply looking up the value of the target function at a given x and $\sin\theta$ from the correct formula. In reality this will be impossible, and the target values in the training data will have to be determined from experimental or MC simulated data via some sort of density estimation, e.g., through bin counts. Therefore, from now on we shall always rely on MC simulated data to obtain the values for the (unit-normalized) target function from event counts in suitably chosen bins. This approach is a better representation of what would be done in an actual experiment.

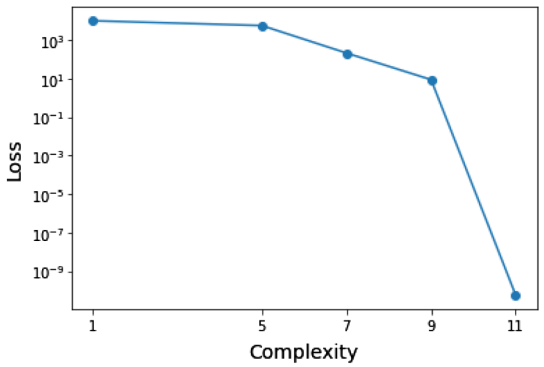

FIG. 6. The MSE loss as a function of complexity for the warm-up symbolic regression exercise considered in Sec. III A.

TABLE V. Results for a few representative complexities from the symbolic regression exercise performed in Sec. III B 1.

Complexity	Function fitted	MSE	Score
9	$(-0.039 + 0.063/x)/\sin^2\theta$	8.32×10^{-3}	1.59
11	$[-0.030 + 0.063/(x - 0.012)]/\sin^2\theta$	2.72×10^{-3}	0.558
13	$(-0.068 + 0.068/x + 0.034x)/\sin^2\theta$	1.53×10^{-4}	1.44
15	$[(-0.067 + 0.067/x + 0.034x)/\sin\theta - 0.001]/\sin\theta$	1.51×10^{-4}	6.80×10^{-3}

(iii) While in this subsection we shall restrict ourselves to gen-level data, in Sec. III C we shall account for the finite detector resolution by smearing the photon energy.

1. Data generated using a splitting function

For this version of the symbolic regression exercise, we first generate MC data according to the approximate model-independent differential cross section (18). We avoid the soft/collinear singularity at $x = 0$ and $\sin\theta = 0$ by focusing on the previously considered region $x \in [0.1, 1]$ and $\sin\theta \in [0.1, 1]$ binned on a 100×100 grid. We then sample 100 million events and populate the bins, whose final event counts then serve as the values of the target function (after unit-normalization) to be used in the training. The input features are again x and $\sin\theta$ and we use the default parameter setup in PySR.

Our results are shown in Table V and Fig. 7 in complete analogy to the earlier Table IV and Fig. 6. Once again, PySR finds the correct expression which is now of complexity 13 (the increase by 2 relative to the result in Table IV is due to the numerical prefactor in front of the linear x term). However, the MSE error this time does not go down to machine precision, and instead saturates at around 10^{-4}, which is due to the statistical uncertainties on the target function values in our training data. Note that "the knee" in Fig. 7 is a marker for the true complexity of our target function.

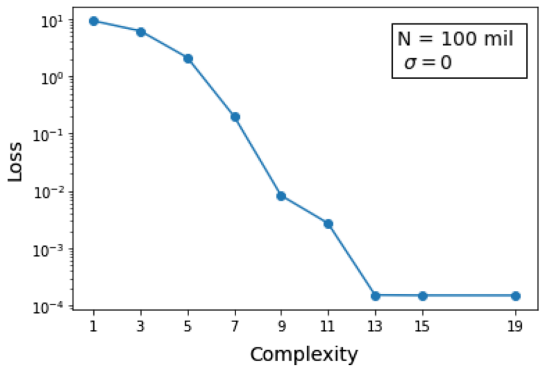

FIG. 7. The same as Fig. 6, but for the symbolic regression exercise performed in Sec. III B 1.

As mentioned in the introduction, an important principle of explainable AI is "generalizability," i.e., extrapolating into the region away from the training data. In order to demonstrate this, we repeat the exercise, but this time we train only on the data within the restricted domain shown with the dashed rectangle in Fig. 8. We then compare the predictions from the fitted functions found by PySR to the true target function, by plotting the difference as a heatmap in the $(x, \sin\theta)$ plane (see Fig. 6 in [20]). Note that in all four cases, the fit within the training domain is reasonably good, but the extrapolation away from it is successful only for the correct answers at complexities 13 and 15. Furthermore, a careful inspection of the plots in the lower row reveals that the extrapolation is better for complexity 13 compared to complexity 15, even though within the training domain the performance is similar. This fact favors the complexity 13 answer over its competitor.

2. Data generated with MadGraph

The training data used in the previous Sec. III B 1 was generated with the approximate factorized formula (18) which is valid in the soft/collinear limit. The advantage of doing so is that we knew the answer that we were supposed to get, which allowed us to judge and validate the performance of PySR. In this subsection, we shall instead generate our training data with a full blown event generator, MadGraph5_aMC@NLO [55], which avoids the soft/collinear approximation. For concreteness, we shall use one of the low energy supersymmetry study points from Ref. [49], namely, the one with neutralino mass of $M_\chi = 225$ GeV. We choose $\sqrt{s} = 500$ GeV at the International Linear Collider. We assumed electromagnetic calorimeter acceptance of $\sin\theta > 0.1$, and required $p_{T\gamma} = E_\gamma \sin\theta > 7.5$ GeV corresponding to the mask calorimeter acceptance of 1 degree. With that setup, we generated 10 million events as our training data, and repeated the symbolic regression exercise with default PySR parameters.

In analogy to the earlier Tables IV and V and Figs. 6 and 7, we present the results in Table VI and Fig. 9, where for simplicity we focus on the x-dependence only. The knee in Fig. 9 is observed at complexity 9, which also has the highest score in Table VI. The form of the function resembles that of (20), but the coefficients are modified. The expressions at higher complexities (11, 13, and 15),

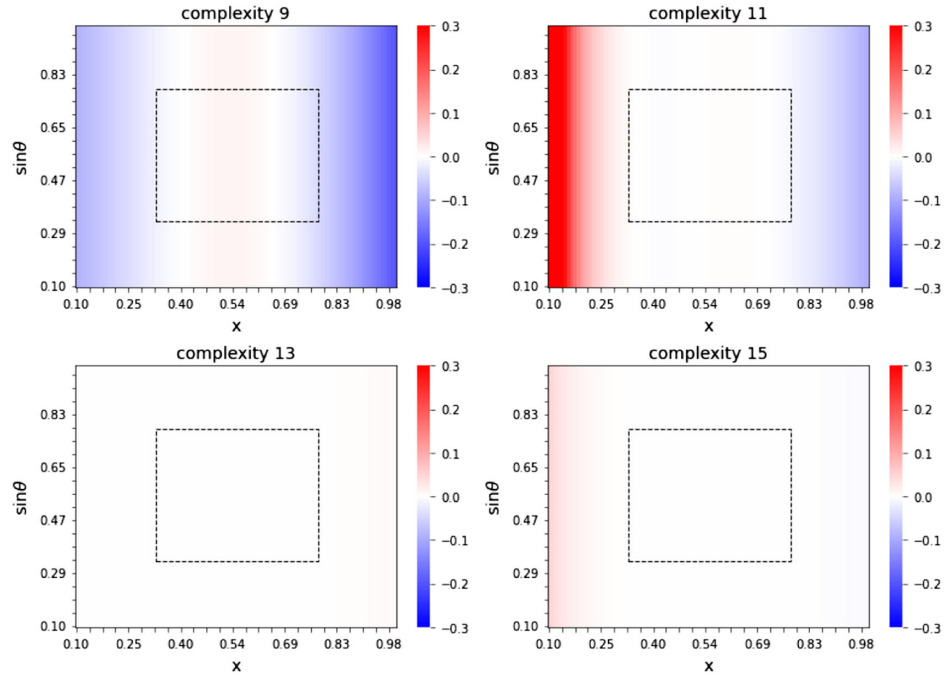

FIG. 8. Heatmaps in the $(x, \sin\theta)$ plane of the differences between the fit functions found by the symbolic regression and the true target function. The rectangular box marked with a dashed line delineates the domain of values on which the symbolic regression was trained.

while having comparable MSE, might be disfavored using the method discussed at the end of the previous subsection, see Fig. 8.

Since in this example we do not have a simple analytical answer as a point of reference, the only way to judge the quality of the answer is to numerically compare to the distribution in the training data. In Fig. 10, we show the unit-normalized distribution of the events in the training data (red) and PySR output (blue). The results from the current subsection are shown in the left panel, where the blue line corresponds to the fitted function at complexity 9. We see that the symbolic regression was capable of producing a simple analytical expression which describes the data quite well, the main visible discrepancy is in the low statistics tail which is not represented well in the training data, and furthermore, is not relevant for the experimental analysis.

TABLE VI. The same as Table V, but for the symbolic regression exercise with MadGraph5_aMC@NLO training data performed in Sec. III B 2.

Complexity	Fitted function	MSE	Score
5	$17.07 - 98.85x$	1.968	0.671
7	$17.08 - 98.85x + x^2$	1.968	1.6×10^{-5}
9	$-11.72 + \frac{2.42 - 0.057/x}{x}$	0.115	1.419
11	$x - 11.97 + \frac{2.44 - 0.057/x}{x}$	0.113	0.007
13	$2x - 12.23 + \frac{2.46 - 0.058/x}{x}$	0.112	0.007
15	$3x - 12.48 + \frac{2.48 + 0.058/x}{x}$	0.111	0.006

C. Learning from detector-level MC data

So far in this section we have been ignoring any instrumental effects, so that the observed distribution followed the theoretical formula (up to statistical errors). In this section we shall add the effects of the detector resolution which would in principle cause the result from the symbolic regression to differ slightly from the theoretical prediction at gen-level.

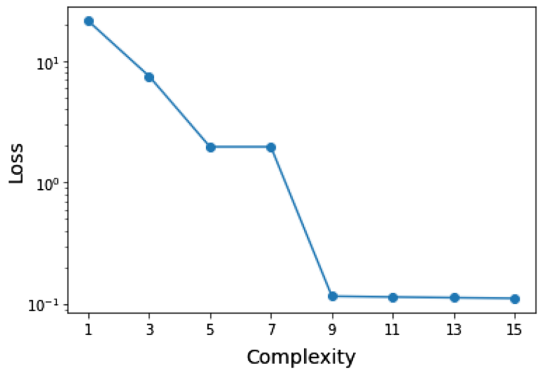

FIG. 9. The same as Fig. 7, but for the symbolic regression exercise with MadGraph5_aMC@NLO training data performed in Sec. III B 2.

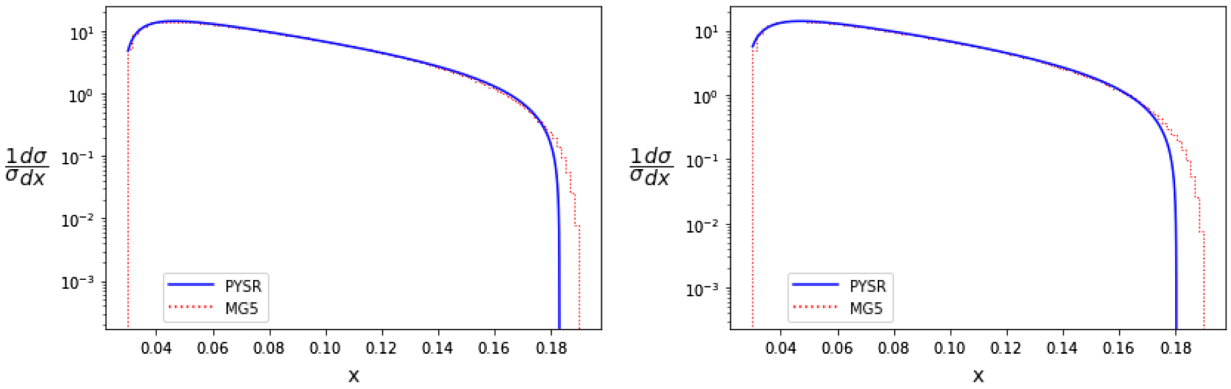

FIG. 10. Unit-normalized distribution of the events in the training data (red) and PySR output (blue). The results in the left panel are from Sec. III B 2 and do not include detector effects, while the results in the right panel are from Sec. III C 2 and account for the detector resolution.

1. Data generated using a splitting function

Here we repeat the exercise from Sec. III B 1, but account for the detector resolution via Gaussian smearing of the energy (but not direction) of the photon with some resolution parameter σ. By varying the value of σ, we shall investigate the impact of the detector on our results, which are collected in Table VII for four different values of σ: 0.01, 0.03, 0.05, and 0.10. In all cases, we observe the expected $\sin^{-2}\theta$ dependence. We note that when the detector effects are relatively mild, $\sigma \lesssim 5\%$, the x dependence is well recovered as well. This fact—that symbolic regression appears to be robust against noise—has been observed in other independent studies as well [56].

In analogy to Fig. 7, in Fig. 11 we show the evolution of the MSE loss with the complexity of the fitted function, for several different values of σ: 0.01, 0.05, 0.10, and 0.20. The "knee" structure is again evident, and the location of the knee depends slightly on the amount of applied smearing.

Since the exercises in this subsection include both errors due to the finite statistics and due to the detector resolution, it is instructive to look at the interplay of the two types of errors as a function of the number of events N_{events} in the training data, see Fig. 12. When the detector effects are absent ($\sigma = 0$, black line), the average loss improves as N_{events} increases, since statistical errors scale as $1/\sqrt{N_{\text{events}}}$. On the other hand, the detector effects are not influenced by

TABLE VII. Results from the symbolic regression exercise performed in Sec. III C 1 for several values of the detector resolution parameter σ: 0.01, 0.03, 0.05, and 0.10.

Complexity	Fitted function	MSE	Score
	$\sigma = 0.01$		
9	$(-0.039 + 0.064/x)/\sin^2\theta$	8.41×10^{-3}	1.58
11	$0.056/(x \sin^2\theta(x + 0.82))$	5.91×10^{-4}	1.33
13	$(-0.068 + 0.068/x + 0.034x)/\sin^2\theta$	2.46×10^{-4}	0.438
17	$[(-0.067 + 0.067/x + 0.034x + \sin\theta)/\sin\theta - 1.00]/\sin\theta$	2.44×10^{-4}	2.11×10^{-3}
	$\sigma = 0.03$		
9	$(-0.039 + 0.064/x)/\sin^2\theta$	8.71×10^{-3}	1.57
11	$0.056/(x \sin^2\theta(x + 0.81))$	7.95×10^{-4}	1.20
13	$(-0.068 + 0.068/x + 0.034x)/\sin^2\theta$	5.33×10^{-4}	0.20
15	$(-0.068 + 0.068/x + 0.034x)/\sin^2\theta - 1.24 \times 10^{-3}$	5.32×10^{-4}	1.06×10^{-3}
	$\sigma = 0.05$		
9	$(-0.039 + 0.064/x)/\sin^2\theta$	1.16×10^{-2}	1.44
11	$0.056/(x \sin^2\theta(x + 0.81))$	3.89×10^{-3}	5.46×10^{-1}
13	$(-0.067 + 0.068/x + 0.033x)/\sin^2\theta$	3.70×10^{-3}	2.51×10^{-2}
15	$[(-0.067 + 0.068/x + 0.033x)/\sin\theta - 1.10 \times 10^{-3}]/\sin\theta$	3.70×10^{-3}	2.78×10^{-4}
	$\sigma = 0.1$		
9	$(-0.039 + 0.064/x)/\sin^2\theta$	3.90×10^{-2}	9.00×10^{-1}
11	$0.056/(x \sin^2\theta(x + 0.82))$	3.30×10^{-2}	8.28×10^{-2}
13	$(-0.064 + 0.067/x + 0.029x)/\sin^2\theta$	3.27×10^{-2}	3.60×10^{-3}
15	$0.078/[\sin^2\theta(1.62x^2 + x + 0.014)]$	3.17×10^{-2}	1.75×10^{-2}

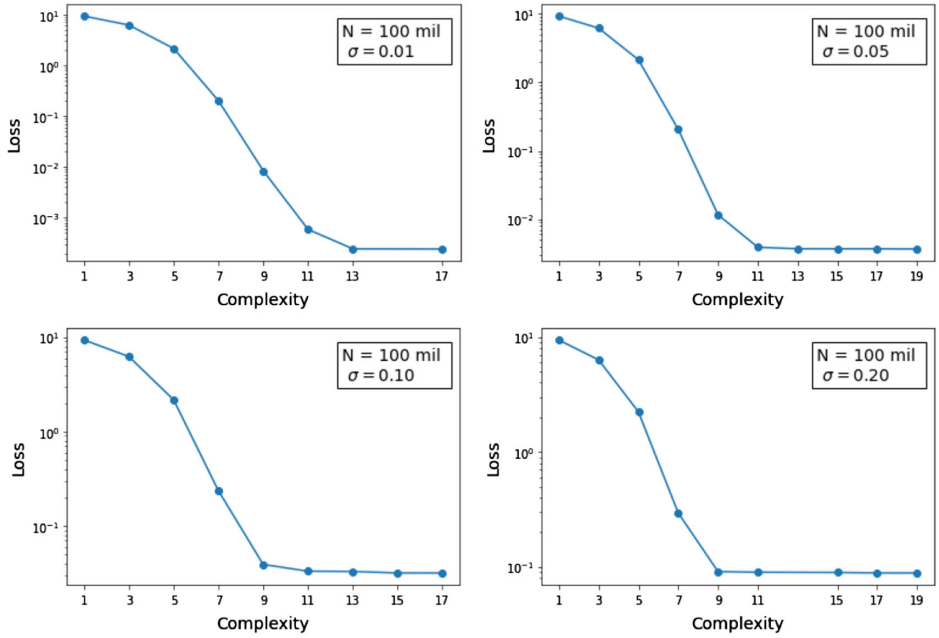

FIG. 11. The same as Fig. 7, but for the exercise performed in Sec. III C 1 with the added detector smearing. Results are shown for several values of the detector resolution parameter σ as labeled in the plots.

N_{events}, and at some point will start to dominate the error budget. As a result, as illustrated in Fig. 12, the MSE loss will start to deviate from the benchmark case of $\sigma = 0$. The exact point where this deviation occurs, depends on the size of the detector smearing parameter—the larger the smearing, the earlier the loss saturates.

2. Data generated with *MadGraph*

Finally, we repeat the exercise from Sec. III B 2 with the addition of calorimeter detector resolution typical of the ILC, $\delta E/E = 0.17/\sqrt{E}$ [57–60]. The results are displayed in Table VIII and Fig. 13, which are the analogs of Table VI and Fig. 9 from Sec. III B 2. The results are as expected,

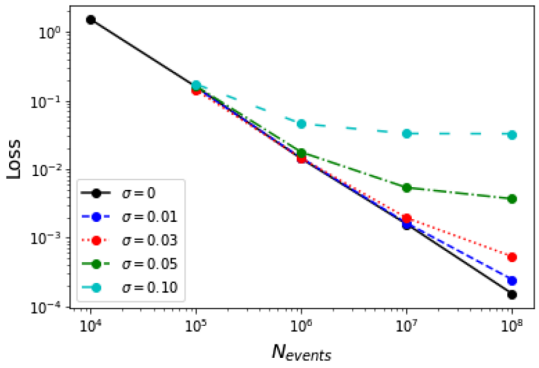

FIG. 12. Loss as a function of the number of events in the training data, for several values of the detector resolution parameter σ. In each case, we chose to show the PySR result whose complexity is at the "knee" of the corresponding plot from Fig. 11.

based on what we have observed in the previous subsections. The corresponding predicted differential distribution is shown in the right panel of Fig. 10.

IV. CONCLUSIONS AND OUTLOOK

This study adds to the already wide range of applications of modern machine learning to event generation and simulation-based inference in collider phenomenology [61]. We demonstrated the use of symbolic regression for two common problems in high-energy particle physics. First, in the case of kinematic or event variables which are defined through some kind of an algorithm, the symbolic regression produces analytical formulas whose accuracy is limited only by the desired functional complexity. In Sec. II we showed how to do this in the example of the stransverse mass variable M_{T2}—we were able to rederive all known analytical formulas for M_{T2} in certain special transverse momentum configurations. Second, the

TABLE VIII. Results from the symbolic regression exercise performed in Sec. III C 2 including detector effects in the training data.

Complexity	Fitted function	MSE	Score
5	$0.724/(x + 0.015)$	6.82	0.051
7	$18.07 + 98.855446x$	1.97	0.623
9	$-11.72 + \frac{2.42-0.057/x}{x}$	0.115	1.419
11	$x - 11.97 + \frac{2.44-0.057/x}{x}$	0.114	0.007
13	$2x - 12.23 + \frac{2.46-0.058/x}{x}$	0.112	0.007
15	$-x - 9.90 + \frac{2.04+(-0.03-0.0004/x)/x}{x}$	0.091	0.102

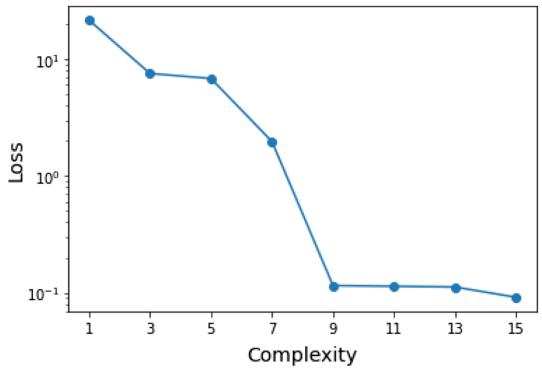

FIG. 13. The same as Fig. 9, but adding the effects of the calorimeter resolution as in Sec. III C 2.

symbolic regression can also produce analytical formulas for certain kinematic distributions of interest, for which theoretical results are unknown or difficult to obtain. In fact, parametrizing the observed distributions in the data with analytical formulas is a standard task in many analyses which attempt to measure the background from data. In Sec. III we demonstrated that this fit can be done either at the gen-level (Sec. III B) or at the detector level (Sec. III C). Note that this last exercise is a nontrivial result, which involves the convolution of the parton-level analytical result with the transfer function describing the detector. To the best of our knowledge, such analytical expressions are rarely discussed in the literature.

The work presented here can be extended in several directions. For example, the M_{T2} concept can be readily applied to more complex event topologies, where one has several choices of designating parent and daughter particles, leading to a menagerie of different "subsystem" M_{T2} variables [62,63]. It would be interesting to see whether the symbolic regression can "derive" the correct answer for M_{T2} in the general case, for which no analytical formula is known. One could also explore other modern techniques for symbolic regression that are adaptable to high-dimensional data [26,64–66].

ACKNOWLEDGMENTS

We thank A. Roman for collaboration in the early stages of this work. K. Kong and K. Matchev would like to thank the Aspen Center for Physics for hospitality during the completion of this work, supported in part by National Science Foundation Grant No. PHY-1607611. This work is supported in parts by US Grants No. DE-SC0019474 and No. DOE DE-SC0022148.

[1] Pat Langley, Bacon: A production system that discovers empirical laws, in *Proceedings of the 5th International Joint Conference on Artificial Intelligence (IJCAI), MIT, Cambridge, MA, USA* (1977).
[2] Pat Langley, Herbert A. Simon, and Gary L. Bradshaw, Heuristics for empirical discovery, in *Computational Models of Learning*, edited by Leonard Bolc (Springer Berlin Heidelberg, Berlin, Heidelberg, 1987), pp. 21–54.
[3] Mieczyslaw Kokar, Determining arguments of invariant functional descriptions, Mach. Learn. **1**, 403 (1986).
[4] Pat Langley and Jan M. Zytkow, Data-driven approaches to empirical discovery, Artif. Intell. **40**, 283 (1989).
[5] Robert Zembowicz and Jan M. Żytkow, Discovery of equations: Experimental evaluation of convergence, in *Proceedings of the Tenth National Conference on Artificial Intelligence, AAAI'92* (AAAI Press, Washington, DC, 1992), pp. 70–75.
[6] Ljupco Todorovski and Saso Dzeroski, Declarative bias in equation discovery, in *Proceedings of the Fourteenth International Conference on Machine Learning* (Morgan Kaufmann Publishers, San Francisco, CA, USA, 1997), pp. 376–384.
[7] Josh Bongard and Hod Lipson, From the cover: Automated reverse engineering of nonlinear dynamical systems, Proc. Natl. Acad. Sci. U.S.A. **104**, 9943 (2007).
[8] Michael Schmidt and Hod Lipson, Distilling free-form natural laws from experimental data, Science **324**, 81 (2009).
[9] Peter W. Battaglia, Razvan Pascanu, Matthew Lai, Danilo Rezende, and Koray Kavukcuoglu, Interaction networks for learning about objects, relations and physics, arXiv:1612.00222.
[10] Michael B. Chang, Tomer Ullman, Antonio Torralba, and Joshua B. Tenenbaum, A compositional object-based approach to learning physical dynamics, arXiv:1612.00341.
[11] Roger Guimerà, Ignasi Reichardt, Antoni Aguilar-Mogas, and Francesco A. Massucci, Manuel Miranda, Jordi Pallarès, and Marta Sales-Pardo, A Bayesian machine scientist to aid in the solution of challenging scientific problems, Sci. Adv. **6**, eaav6971 (2020).
[12] Silviu-Marian Udrescu and Max Tegmark, AI Feynman: A physics-inspired method for symbolic regression, Sci. Adv. **6**, eaay2631 (2020).
[13] Miles Cranmer, Alvaro Sanchez-Gonzalez, Peter Battaglia, Rui Xu, Kyle Cranmer, David Spergel, and Shirley Ho, Discovering symbolic models from deep learning with inductive biases, arXiv:2006.11287.
[14] Ziming Liu, Varun Madhavan, and Max Tegmark, AI Poincaré 2.0: Machine learning conservation laws from differential equations, Phys. Rev. E **106**, 045307 (2022).

[15] Yoshitomo Matsubara, Naoya Chiba, Ryo Igarashi, Tatsunori Taniai, and Yoshitaka Ushiku, Rethinking symbolic regression datasets and benchmarks for scientific discovery, arXiv:2206.10540.
[16] Miles D. Cranmer, Rui Xu, Peter Battaglia, and Shirley Ho, Learning symbolic physics with graph networks, arXiv:1909.05862.
[17] Ana Maria Delgado, Digvijay Wadekar, Boryana Hadzhiyska, Sownak Bose, Lars Hernquist, and Shirley Ho, Modeling the galaxy-halo connection with machine learning, Mon. Not. R. Astron. Soc. **515**, 2733 (2022).
[18] Raban Iten, Tony Metger, Henrik Wilming, Lídia del Rio, and Renato Renner, Discovering Physical Concepts with Neural Networks, Phys. Rev. Lett. **124**, 010508 (2020).
[19] Pablo Lemos, Niall Jeffrey, Miles Cranmer, Shirley Ho, and Peter Battaglia, Rediscovering orbital mechanics with machine learning, arXiv:2202.02306.
[20] Konstantin T. Matchev, Katia Matcheva, and Alexander Roman, Analytical modeling of exoplanet transit spectroscopy with dimensional analysis and symbolic regression, Astrophys. J. **930**, 33 (2022).
[21] Suyong Choi, Construction of a kinematic variable sensitive to the mass of the Standard Model Higgs boson in $H \to WW^* \to \ell^+\nu\ell^-\nu^-$ using symbolic regression, J. High Energy Phys. 08 (2011) 110.
[22] Anja Butter, Tilman Plehn, Nathalie Soybelman, and Johann Brehmer, Back to the formula—LHC edition, arXiv:2109.10414.
[23] Aurélien Dersy, Matthew D. Schwartz, and Xiaoyuan Zhang, Simplifying polylogarithms with machine learning, arXiv:2206.04115.
[24] Abdulhakim Alnuqaydan, Sergei Gleyzer, and Harrison Prosper, SYMBA: Symbolic computation of squared amplitudes in high energy physics with machine learning, Mach. Learn. Sci. Tech. **4**, 015007 (2023).
[25] Yiqun Wang, Nicholas Wagner, and James M. Rondinelli, Symbolic regression in materials science, MRS Commun. **9**, 793 (2019).
[26] Nikos Arechiga, Francine Chen, Yan-Ying Chen, Yanxia Zhang, Rumen Iliev, Heishiro Toyoda, and Kent Lyons, Accelerating understanding of scientific experiments with end to end symbolic regression, arXiv:2112.04023.
[27] P. Phillips, Carina Hahn, Peter Fontana, Amy Yates, Kristen Greene, David Broniatowski, and Mark Przybocki, Four principles of explainable artificial intelligence, NIST Interagency/Internal Report (NISTIR), National Institute of Standards and Technology, Gaithersburg, MD, 2021.
[28] Oscar Fajardo-Fontiveros, Ignasi Reichardt, Harry R. De Los Rios, Jordi Duch, Marta Sales-Pardo, and Roger Guimera, Fundamental limits to learning closed-form mathematical models from data, arXiv:2204.02704.
[29] Miles Cranmer, PySR: Fast & parallelized symbolic regression in Python/Julia, 10.5281/zenodo.4041459 (2020).
[30] Tao Han, Collider phenomenology: Basic knowledge and techniques, in *Theoretical Advanced Study Institute in Elementary Particle Physics: Physics in* $D \geqq 4$ (World Scientific Publishing Company, Singapore, 2005), pp. 407–454.
[31] Alan J. Barr and Christopher G. Lester, A review of the mass measurement techniques proposed for the large hadron collider, J. Phys. G **37**, 123001 (2010).
[32] 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).
[33] Roberto Franceschini, Doojin Kim, Kyoungchul Kong, Konstantin T. Matchev, Myeonghun Park, and Prasanth Shyamsundar, Kinematic variables and feature engineering for particle phenomenology, arXiv:2206.13431.
[34] Andrea Banfi, Gavin P. Salam, and Giulia Zanderighi, Phenomenology of event shapes at hadron colliders, J. High Energy Phys. 06 (2010) 038.
[35] Iain W. Stewart, Frank J. Tackmann, and Wouter J. Waalewijn, N-Jettiness: An Inclusive Event Shape to Veto Jets, Phys. Rev. Lett. **105**, 092002 (2010).
[36] Jesse Thaler and Ken Van Tilburg, Identifying boosted objects with N-subjettiness, J. High Energy Phys. 03 (2011) 015.
[37] C. G. Lester and D. J. Summers, Measuring masses of semiinvisibly decaying particles pair produced at hadron colliders, Phys. Lett. B **463**, 99 (1999).
[38] Alan Barr, Christopher Lester, and P. Stephens, A variable for measuring masses at hadron colliders when missing energy is expected; m_{T2}: The Truth behind the glamour, J. Phys. G **29**, 2343 (2003).
[39] Won Sang Cho, James S. Gainer, Doojin Kim, Konstantin T. Matchev, Filip Moortgat, Luc Pape, and Myeonghun Park, On-shell constrained M_2 variables with applications to mass measurements and topology disambiguation, J. High Energy Phys. 08 (2014) 070.
[40] Won Sang Cho, James S. Gainer, Doojin Kim, Sung Hak Lim, Konstantin T. Matchev, Filip Moortgat, Luc Pape, and Myeonghun Park, OPTIMASS: A package for the minimization of kinematic mass functions with constraints, J. High Energy Phys. 01 (2016) 026.
[41] Won Sang Cho, James S. Gainer, Doojin Kim, Konstantin T. Matchev, Filip Moortgat, Luc Pape, and Myeonghun Park, Improving the sensitivity of stop searches with on-shell constrained invariant mass variables, J. High Energy Phys. 05 (2015) 040.
[42] Graham G. Ross and Mario Serna, Mass determination of new states at hadron colliders, Phys. Lett. B **665**, 212 (2008).
[43] Christopher G. Lester, The stransverse mass, M_{T2}, in special cases, J. High Energy Phys. 05 (2011) 076.
[44] Colin H. Lally and Christopher G. Lester, Properties of MT2 in the massless limit, arXiv:1211.1542.
[45] Christopher Lester and Alan Barr, m_{TGen}: Mass scale measurements in pair-production at colliders, J. High Energy Phys. 12 (2007) 102.
[46] Won Sang Cho, Kiwoon Choi, Yeong Gyun Kim, and Chan Beom Park, Gluino Stransverse Mass, Phys. Rev. Lett. **100**, 171801 (2008).
[47] Won Sang Cho, Kiwoon Choi, Yeong Gyun Kim, and Chan Beom Park, Measuring superparticle masses at hadron collider using the transverse mass kink, J. High Energy Phys. 02 (2008) 035.

[48] Christopher G. Lester and Benjamin Nachman, Bisection-based asymmetric M_{T2} computation: A higher precision calculator than existing symmetric methods, J. High Energy Phys. 03 (2015) 100.
[49] Andreas Birkedal, Konstantin Matchev, and Maxim Perelstein, Dark matter at colliders: A model independent approach, Phys. Rev. D **70**, 077701 (2004).
[50] Jonathan L. Feng, Shufang Su, and Fumihiro Takayama, Lower Limit on Dark Matter Production at the Large Hadron Collider, Phys. Rev. Lett. **96**, 151802 (2006).
[51] Shrihari Gopalakrishna, Maxim Perelstein, and James D. Wells, Extra dimensions vs. supersymmetric interpretation of missing energy events at a linear collider, *eConf* C010630, P311 (2001).
[52] W. Oller, H. Eberl, and W. Majerotto, Full one loop corrections to neutralino pair production in e^+e^- annihilation, Phys. Lett. B **590**, 273 (2004).
[53] Kentarou Mawatari and Bettina Oexl, Monophoton signals in light gravitino production at e^+e^- colliders, Eur. Phys. J. C **74**, 2909 (2014).
[54] J. Kalinowski, K. Mekala, P. Sopicki, A. F. Zarnecki, and W. Kotlarski, Sensitivity of future e^+e^- colliders to processes of dark matter production with light mediator exchange, Acta Phys. Pol. B Proc. Suppl. **15**, A10 (2022).
[55] Johan Alwall, Michel Herquet, Fabio Maltoni, Olivier Mattelaer, and Tim Stelzer, MadGraph 5: Going beyond, J. High Energy Phys. 06 (2011) 128.
[56] Brenden K. Petersen, Deep symbolic regression: Recovering mathematical expressions from data via policy gradients, arXiv:1912.04871.
[57] C. Adloff *et al.* (CALICE Collaboration), Response of the CALICE Si-W electromagnetic calorimeter physics prototype to electrons, Nucl. Instrum. Methods Phys. Res., Sect. A **608**, 372 (2009).
[58] Philip Bambade *et al.*, The international linear collider: A global project, arXiv:1903.01629.
[59] Moritz Habermehl, Mikael Berggren, and Jenny List, WIMP dark matter at the international linear collider, Phys. Rev. D **101**, 075053 (2020).
[60] Halina Abramowicz *et al.* (ILD Concept Group), International large detector: Interim design report, arXiv:2003.01116.
[61] Simon Badger *et al.*, Machine learning and LHC event generation, arXiv:2203.07460.
[62] K. Kawagoe, M. M. Nojiri, and G. Polesello, A new SUSY mass reconstruction method at the CERN LHC, Phys. Rev. D **71**, 035008 (2005).
[63] Michael Burns, Kyoungchul Kong, Konstantin T. Matchev, and Myeonghun Park, Using subsystem M_{T2} for complete mass determinations in decay chains with missing energy at hadron colliders, J. High Energy Phys. 03 (2009) 143.
[64] Stéphane d'Ascoli, Pierre-Alexandre Kamienny, Guillaume Lample, and François Charton, Deep symbolic regression for recurrent sequences, arXiv:2201.04600.
[65] Pierre-Alexandre Kamienny, Stéphane d'Ascoli, Guillaume Lample, and François Charton, End-to-end symbolic regression with transformers, arXiv:2204.10532.
[66] Jiachen Li, Ye Yuan, and Hong-Bin Shen, Symbolic expression transformer: A computer vision approach for symbolic regression, arXiv:2205.11798.