Physics-Guided Symbolic Regression through Recursive Structure Discovery

Wassim Tenachi [*1 2 3] **Rodrigo Ibata** [*4] **Alejandro Miguel Illescas Giménez** [*4] **Yashar Hezaveh** [1 2 3]
Laurence Perreault Levasseur [1 2 3] **Pierre-Luc Bacon** [5 3]

Abstract

In observation-driven sciences like astrophysics, interpretable empirical models are highly valuable. Symbolic regression (SR) addresses this need by automatically discovering analytic equations from data. However, physical data often contains inherent structural properties that remain underutilized in this process. We propose a method that identifies additive and multiplicative separability in datasets—determining whether an unknown model $y = f(\mathbf{x})$ can be decomposed into simpler components $y = f_1(\mathbf{x_1}) + f_2(\mathbf{x_2})$ or $y = f_2(\mathbf{x_2}) \cdot f_2(\mathbf{x_2})$. Our approach recursively decomposes intricate datasets into simpler submodels, revealing the underlying hierarchical graph structure to enhance interpretability and enable more efficient SR. When combined with Φ-SO – a deep reinforcement learning framework for SR, our method achieves state-of-the-art performance, outperforming all baselines on SRBench's Feynman benchmark.

1. Introduction

New physical knowledge emerges from observations of natural phenomena. However, as physicists we often only observe the net result of complex systems with interacting subphenomena—especially in astrophysics, where controlled laboratory experiments are impossible, and we must rely solely on observational data. Historically, progress in our field has relied on decomposing such complex observations into simpler, interpretable physical laws.

[*]Equal contribution [1]Département de physique, Université de Montréal, Montréal, Canada [2]Ciela - Montreal Institute for Astrophysical Data Analysis and Machine Learning, Montréal, Canada [3]Mila - Quebec Artificial Intelligence Institute, Montréal, Canada [4]Université de Strasbourg, CNRS, Observatoire astronomique de Strasbourg, UMR 7550, F-67000 Strasbourg, France [5]Department of Computer Science and Operations Research, Université de Montréal, Montréal, Canada. Correspondence to: Wassim Tenachi <wassim.tenachi@mila.quebec>.

ML4Astro 2025, Vancouver, CA.

The rise of deep learning has shifted this paradigm: many phenomena are now modeled via neural networks, which, while powerful, sacrifice interpretability. This opacity hinders their integration into overarching physical theories expressed as analytical equations.

Unveiling graph structure in data To address this challenge, we propose a neural network framework that explicitly discovers hierarchical structures in physical datasets. Our approach identifies two fundamental decomposition patterns: (1) additive/multiplicative separability and (2) compositionality through nonlinear unary functions (e.g., $\frac{1}{\square}, \square^2, \exp$ ...)[1]. These patterns form the natural building blocks of analytical physical representations. For an intricate function $f : \mathbb{R}^n \to \mathbb{R}$, we decompose it into interpretable submodels $\{f_i\}_{i=1}^N$, each operating on distinct input subsets. The complete model reconstructs as $f = g \circ (f_1, \ldots, f_N)$, where the composition function g hierarchically combines submodels through either elementary operations ($+$, $\times$) or nonlinear transformations.

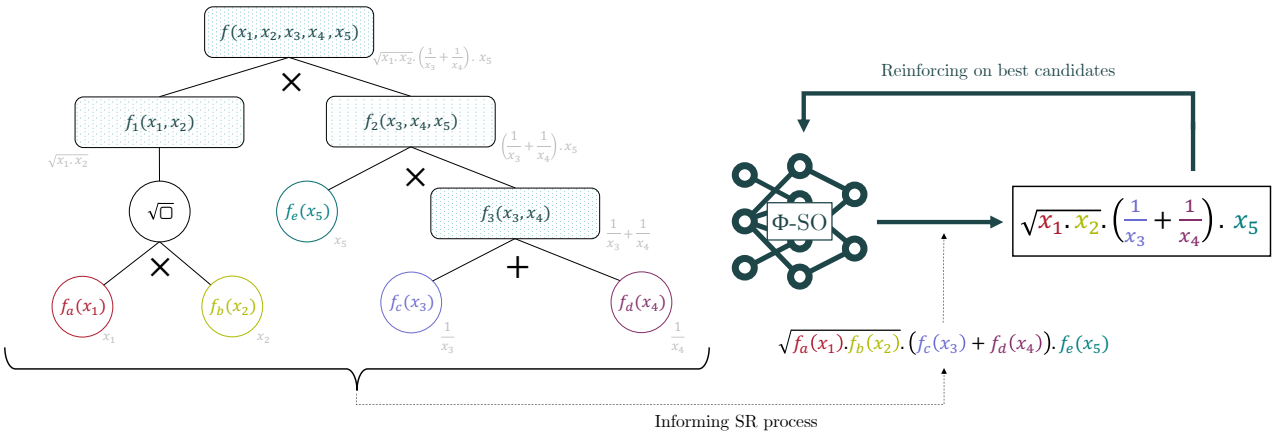
Symbolic Regression (SR) The extracted hierarchical structure directly informs a symbolic regression process, which infers an analytical form for f from data $(\mathbf{x}, y)$. Unlike numerical parameter optimization, SR explores the space of functional forms by optimizing arrangements of mathematical operators (e.g., $+$, $\times$, $\sin$, $\exp$), input variables and free constants. This combinatorial search is NP-hard, making prior knowledge of the data's structure—such as separability—critical for efficiency. Our work bridges this gap by automatically detecting and exploiting such structure.

In §1.1, we contextualize our contributions within existing literature on SR and structural decomposition.

1.1. Related Works & Contribution

Traditional SR Methods SR has historically relied on genetic programming (GP), emulating natural evolution to explore equation spaces. This approach underpins frameworks like `Eureqa` (Schmidt & Lipson, 2009),

[1]While symmetry detection represents another important structural pattern, we defer its systematic treatment to future work.



(a) Hierarchical structure discovery through recursive separability analysis (b) Structure-constrained symbolic regression with deep RL

Figure 1. **Structure-Aware Symbolic Regression.** (a) Automated decomposition of an intricate dataset $(\mathbf{x}, y)$ into interpretable submodels through recursive detection of additive/multiplicative separabilities, including those nested within nonlinear operations. (b) Structure-informed inference where the discovered hierarchy guides SR through a prior, enabling exact recovery of the ground-truth equation via deep RL.

`PySR` (Cranmer, 2023), and others (Stephens, 2015; Cava et al., 2019; Kommenda et al., 2020; Virgolin et al., 2021; Russeil et al., 2024).

Deep Learning-Based SR Recent advances employ neural networks for SR through two dominant approaches: (1) generative transformer models trained to map datasets to corresponding equations (Kamienny et al., 2022; Lalande et al., 2023; Biggio et al., 2020; 2021), and (2) deep reinforcement learning (RL) where networks iteratively generate and refine equations via policy gradient methods, as in `DSR` (Petersen et al., 2021a; Landajuela et al., 2021) and Φ-SO (Tenachi et al., 2023a; 2024). Our work extends the second paradigm, utilizing its inherent flexibility to incorporate structural priors and per-submodel length constraints.

Neuro-Symbolic Approaches Alternative approaches embed symbolic operations (e.g., $\frac{1}{\square}$, $\square^2$, exp ...) within compact neural architectures, enforcing sparsity to recover interpretable equations (Fiorini et al., 2024; Scholl et al., 2023; Martius & Lampert, 2017; Brunton et al., 2016; Sahoo et al., 2018). While we similarly incorporate nonlinearities, our method differs by hierarchically composing them through symbolic regression of submodels without explicit sparsity constraints.

Separability Detection Prior work on separability-leveraging SR includes `AIF` (Udrescu & Tegmark, 2020; Udrescu et al., 2020) and its RL hybrid `uDSR` (Landajuela et al., 2022). However, these methods are limited to additive separability and cannot handle multiplicative separabilities or nonlinear-nested structures—key limitations our approach addresses.

Gradient Estimation Accurate separability detection requires precise gradient estimation. While derivative-constrained training is well-studied in physics-informed neural networks (PINNs) (Raissi et al., 2019) and Sobolev training (Czarnecki et al., 2017), few works address derivative estimation from data alone. Our solution employs `NestyNet`[2], a novel architecture enabling high-fidelity derivative computation from potentially noisy observations.

Our Approach Our framework: (1) employs `NestyNet` as a high-fidelity function emulator, (2) leverages its precise derivatives to recursively detect additive/multiplicative separabilities—even when nested within nonlinear functions—and (3) constructs an interpretable graph of subfunctions that informs a deep RL-based symbolic regression (SR) process through structural priors. This focuses the search on analytically plausible expressions matching the discovered hierarchy.

Sections 2–4 detail our methodology, results, and conclusions respectively.

2. Method

Learning Precise Derivatives To identify separabilities, we require an emulator f that accurately fits the data $(\mathbf{x}, y) \in \mathbb{R}^{n_x \times n_y}$ while providing reliable first and second derivatives $\partial f/\partial \mathbf{x}$ and $\partial^2 f/\partial \mathbf{x}^2$. We employ the `NestyNet` architecture—a shallow network with exact

[2]The `NestyNet` architecture and its derivative estimation capabilities are described in a forthcoming paper by Rodrigo Ibata.

derivative expressions:

$$y = a \log(1 + \exp(K\mathbf{x} + b)) \tag{1}$$

where $K \in \mathbb{R}^{h \times n_x}$ is a weight matrix, $b \in \mathbb{R}^h$ a bias vector, and $a \in \mathbb{R}^{n_y \times h}$ a scaling matrix, with trainable parameters $\theta = \{K, b, a\}$.

This design enables analytical computation of derivatives, avoiding numerical inaccuracies from auto-differentiation in deeper networks[3]. The Jacobian and Hessian are given by (σ denoting the sigmoid function):

$$\nabla_{\mathbf{x}} y = K^\top \left(a\sigma(K\mathbf{x} + b)\right) \tag{2}$$

$$\frac{\partial^2 y}{\partial x_i \partial x_j} = K^\top \left(a\sigma'(K\mathbf{x} + b)\right) K \tag{3}$$

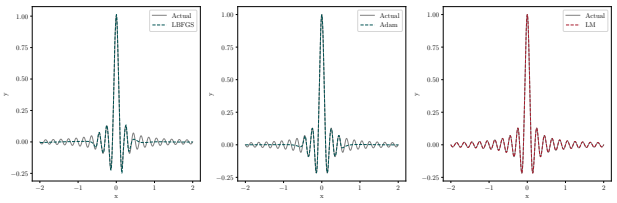


Figure 2. **Second-Order Optimization Advantage in Physics.** Training dynamics of a shallow dense network under different optimization strategies. The second-order Levenberg-Marquardt (LM) method outperforms both `Adam` (first-order) and `L-BFGS` (Zhu et al., 1997) (quasi-Newton) in final accuracy and convergence speed, especially for subtle features.

Levenberg-Marquardt Optimization While the universal approximation theorem (Hornik et al., 1989) guarantees that this shallow architecture can represent any smooth function given sufficient width h, traditional gradient descent approaches—which rely on minimizing a scalar loss function $\mathcal{L}(\theta) = \sum_i (y^{\langle i \rangle} - f(x^{\langle i \rangle}, \theta))^2$ via partial derivatives $\nabla_\theta \mathcal{L} = (\partial\mathcal{L}/\partial\theta_1, \ldots, \partial\mathcal{L}/\partial\theta_M)$—prove impractical for complex tasks with optimizers like `Adam` (Kingma & Ba, 2015) or `SGD` (Robbins & Monro, 1951). This limitation motivated the development of deep architectures (LeCun et al., 2015).

We instead employ the Levenberg-Marquardt algorithm (Levenberg, 1944; Marquardt, 1963). The method computes the Jacobian matrix $J_{ij} = \partial f(x^{\langle i \rangle})/\partial\theta_j$ and approximates the explicit Hessian as $J^\top J$, then solves the linear system $(J^\top J + \lambda I)\Delta\theta = J^\top (y - f(x, \theta))$ for parameter updates. Unlike first-order methods (e.g., SGD) that only estimate descent directions, this approach directly computes parameter updates $\Delta\theta$ to minimize χ^2 by combining gradient and curvature information. As shown in Figure 2, this second-order optimization better captures fine-scale data variations (Ranganathan, 2004).

Additive separability We detect additive separability by testing whether our emulator f decomposes into sub-functions f_1 and f_2 operating on different input subsets $\mathbf{x}_1$ and $\mathbf{x}_2$, satisfying $y = f_1(\mathbf{x}_1) + f_2(\mathbf{x}_2)$. In this case, the mixed second-order partial derivatives vanish, i.e., $\frac{\partial^2 y}{\partial x_i \partial x_j} = 0$ for $i \neq j$. To verify separability between a variable pair (x_i, x_j), we therefore evaluate the condition:

$$\text{med}\left(\left|\frac{\partial^2 y}{\partial x_i \partial x_j}\right|\right) < \epsilon_{\text{add}} \tag{4}$$

Where ϵ_{add} is an empirically determined threshold for negligible interactions (with med denoting the median operation across sample points). This test is applied across variable pairs and input partitions, when multiple valid separations exist, we select the configuration minimizing $\dim(\mathbf{x}_2)$ to prioritize the most interpretable decomposition.

Multiplicative Separability Similarly, we detect multiplicative separability by testing whether our emulator f decomposes into sub-functions f_1 and f_2 operating on distinct input subsets $\mathbf{x}_1$ and $\mathbf{x}_2$, such that $y = f_1(\mathbf{x}_1) \cdot f_2(\mathbf{x}_2)$. To verify separability between a pair of variables (x_i, x_j) where $i \neq j$, we test for the form $y = f_1(x_i) \cdot f_2(x_j)$, allowing for a potential additive constant b giving $y = f_1(x_i) \cdot f_2(x_j) + b$. This is done by verifying the mixed second-order partial derivatives relationship:

$$\text{med}\left(\left|\frac{\partial^2 y}{\partial x_i \partial x_j}\frac{1}{(y - b_{\text{med}})} - \frac{\partial y}{\partial x_i}\frac{\partial y}{\partial x_j}\frac{1}{(y - b_{\text{med}})^2}\right|\right) < \epsilon_{\text{mul}} \tag{5}$$

where ϵ_{mul} is an empirically determined threshold for negligible interactions, and $b_{\text{med}} = \text{med}(b)$, with $b = y - \frac{\partial y}{\partial x_i} \cdot \frac{\partial y}{\partial x_j} \cdot \frac{1}{\partial^2 y / \partial x_i \partial x_j}$. Since b should be constant across all input values when the separation is valid, we additionally verify that its scaled scatter is small:

$$\frac{\text{med}(|b - \text{med}(b)|)}{\text{med}(|y|)} < \epsilon_{b_{mad}} \tag{6}$$

Where $\epsilon_{b_{mad}}$ is an empirically determined threshold for negligible scatter. Again, when multiple valid separations exist, we select the configuration minimizing $\dim(\mathbf{x}_2)$ to prioritize the most interpretable decomposition.

Numerical values for trehsold parameters are given in Table 2.

Recursive Structure Search Upon detecting separability, we emulate each sub-function using distinct `NestyNet` instances. This modular approach enables iterative decomposition, where each sub-function undergoes further

[3]Where gradients must be obtained through the iterative application of the chain rule.

Criterion	Threshold Value
ϵ_{add}	10^{-4}
ϵ_{mul}	10^{-12}
$\epsilon_{b_{\text{mad}}}$	10^{-3}

Table 1. **Detection thresholds for separability analysis**

separability analysis, progressively transforming intricate datasets into interpretable graphs of simpler models. When no separability is found in $f(\mathbf{x})$, we additionally examine separabilities on transformed versions $F_{\text{NL}}(f(\mathbf{x}))$ using a library of common physical nonlinearities: $F_{\text{NL}} \in \{\square^{-1}, \square^2, \sqrt{\square}, \exp(\square)\}$, with gradients computed via the chain rule. This unique capability allows discovery of nested structures like $f(x_1, x_2) = \sqrt{f_1(x_1) + f_2(x_2)}$, as illustrated in Figure 1 (Panel a).

Structural Prior Integration The discovered graph structure guides the symbolic regression process through probabilistic priors, enhancing efficiency without rigid constraints. Our framework employs a recurrent neural network (RNN) to sequentially generate trial expressions while dynamically enforcing structural compatibility. During token generation, the algorithm tracks its position within the structure tree and modifies the RNN's output distribution to reflect the current context.

For sub-expression nodes, the probability of sampling input variables outside the permitted subset is reduced to near-zero values[4]. At separability nodes, the distribution is similarly constrained to favor relevant operators: additive separabilities preferentially select from $+, -$, while multiplicative separabilities bias toward $\times, /$. This approach maintains the RNN's exploratory capability while strongly encouraging structurally valid expressions.

We complement these constraints with length priors on subfunctions, initially based on variable counts as a proxy for complexity, with provisions for more sophisticated measures in future implementations.

Symbolic Regression Our method builds upon the Φ-SO (Physical Symbolic Optimization) algorithm, which incorporates several physics-oriented capabilities: dimensional analysis (Tenachi et al., 2023b), simultaneous fitting across multiple experimental realizations (Tenachi et al., 2024), and integration of non-differentiable constraints such as symbolic constraints on derivatives or primitives. In addition, the framework features a unique efficient free constant optimization scheme leveraging automatic differentiation of trial expressions. This established deep reinforcement learning approach trains the generator network through policy gradient methods, utilizing a risk-seeking optimization strategy (Petersen et al., 2021b) adapted from (Rajeswaran et al., 2017).

The complete workflow (Figure 1, Panel b) thus combines structural awareness with the expressive power of deep learning SR, where priors focus the search space.

[4]While maintained at non-zero to preserve theoretical exploration capacity.

3. Results & Discussion

We evaluated our method using the standardized `SRBench` framework (La Cava et al., 2021) (`https://github.com/cavalab/srbench`), comparing against 17 baseline methods on 116 ground-truth equations to be recovered from their associated data from the Feynman SR benchmark (Udrescu & Tegmark, 2020). Performance was assessed via exact symbolic recovery using `SymPy`'s (Meurer et al., 2017) equivalence checking.

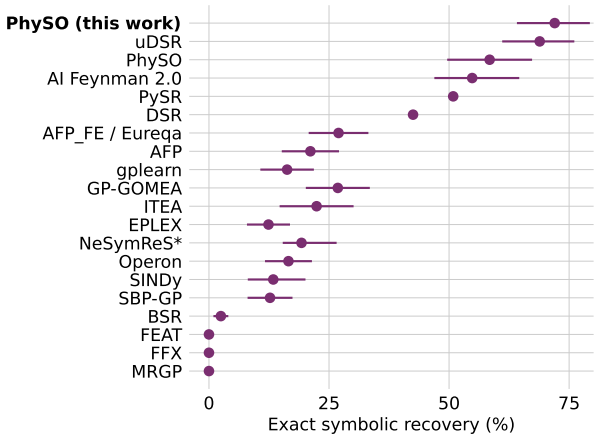


Figure 3. **State-of-the-Art Performance on SRBench.** Our method outperforms all baselines in noiseless conditions including traditional structure analysis methods, pure deep RL methods and previous hybrids.

Performance Analysis As shown on (Figure 3), our method (`PhySO` with structure analysis) achieves state-of-the-art exact recovery (72%), outperforming: (1) `AIF` (the original separability-based approach), (2) pure RL methods (`DSR`, legacy `PhySO`), and (3) hybrid approaches (`uDSR`). This advancement stems from three key innovations: (i) handling multiplicative separabilities, (ii) detecting separabilities nested within nonlinear transformations, and (iii) effectively integrating structural inference with RL through adaptive priors rather than rigid decomposition.

Limitations and Future Directions While evaluated only on noiseless data thus far, our method shows strong po-

tential for noise robustness—contrasting with traditional separability approaches. Two key innovations suggest this resilience: (1) NestyNet's capacity for stable derivative estimation with noise, and (2) our prior-based architecture's safety mechanism—when separability detection falters, performance defaults to standard RL baselines rather than failing catastrophically as AIF and uDSR that degrade severely ($< 10\%$ recovery at 10% noise). Formal noise testing and hyperparameter optimization remain for future work.

Current limitations also include detecting generalized symmetries and compositional structures as in (Udrescu et al., 2020), which we plan to incorporate. In addition, our method mostly excels in physical science applications valuing interpretable exact solutions over numerical approximation, unlike SR approaches that sacrifice interpretability for accuracy[5].

4. Conclusion

We introduced a novel symbolic regression framework that automatically discovers and exploits the inherent graph structure of physical data through identification of additive and multiplicative separabilities—including those nested within nonlinear transformations. By incorporating this structural insight as a prior within a reinforcement learning-based SR process, we achieve state-of-the-art performance (72% exact recovery) on the standard Feynman benchmark from SRBench, significantly advancing physics-capable SR.

While broadly applicable across scientific domains, our method holds particular promise for astrophysics—a field where observational data drives empirical modeling but controlled experiments are impossible. This work continues astrophysics' tradition of developing data-driven modeling techniques that ultimately benefit all physical sciences, now augmented with modern machine learning while preserving the interpretability crucial for theoretical advancement.

Acknowledgments

This work is partially supported by Schmidt Futures, a philanthropic initiative founded by Eric and Wendy Schmidt as part of the Virtual Institute for Astrophysics (VIA). The work is in part supported by computational resources provided by Calcul Quebec and the Digital Research Alliance of Canada. Y.H. and L.P. acknowledge support from the Canada Research Chairs Program, the National Sciences and Engineering Council of Canada through grants RGPIN-2020-05073 and 05102.

[5]These alternatives trade readability for faster evaluation and better generalization as an alternative to neural networks.

Impact Statement

This paper presents work whose goal is to advance the field of Machine Learning. There are many potential societal consequences of our work, none which we feel must be specifically highlighted here.

References

Biggio, L., Bendinelli, T., Lucchi, A., and Parascandolo, G. A seq2seq approach to symbolic regression. In *Learning Meets Combinatorial Algorithms at NeurIPS2020*, 2020. URL https://openreview.net/forum?id=W7jCKuyPn1.

Biggio, L., Bendinelli, T., Neitz, A., Lucchi, A., and Parascandolo, G. Neural symbolic regression that scales. In Meila, M. and Zhang, T. (eds.), *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 936–945. PMLR, 18–24 Jul 2021. URL https://proceedings.mlr.press/v139/biggio21a.html.

Brunton, S. L., Proctor, J. L., and Kutz, J. N. Discovering governing equations from data by sparse identification of nonlinear dynamical systems. *Proceedings of the National Academy of Sciences*, 113(15):3932–3937, mar 2016. doi: 10.1073/pnas.1517384113.

Cava, W. L., Singh, T. R., Taggart, J., Suri, S., and Moore, J. Learning concise representations for regression by evolving networks of trees. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=Hke-JhA9Y7.

Cranmer, M. Interpretable machine learning for science with pysr and symbolicregression. jl. *arXiv preprint arXiv:2305.01582*, 2023.

Czarnecki, W. M., Osindero, S., Jaderberg, M., Swirszcz, G., and Pascanu, R. Sobolev training for neural networks. *Advances in neural information processing systems*, 30, 2017.

Fiorini, C., Flint, C., Fostier, L., Franck, E., Hashemi, R., Michel-Dansac, V., and Tenachi, W. Generalizing the sindy approach with nested neural networks. *arXiv preprint arXiv:2404.15742*, 2024.

Hornik, K., Stinchcombe, M., and White, H. Multilayer feedforward networks are universal approximators. *Neural Networks*, 2(5):359–366, 1989. ISSN 0893-6080. doi: https://doi.org/10.1016/0893-6080(89)90020-8. URL https://www.sciencedirect.com/science/article/pii/0893608089900208.

Kamienny, P.-A., d'Ascoli, S., Lample, G., and Charton, F. End-to-end symbolic regression with transformers. In Oh, A. H., Agarwal, A., Belgrave, D., and Cho, K. (eds.), *Advances in Neural Information Processing Systems*, 2022. URL https://openreview.net/forum?id=GoOuIrDHG_Y.

Kingma, D. and Ba, J. Adam: A method for stochastic optimization. In *International Conference on Learning Representations (ICLR)*, San Diega, CA, USA, 2015.

Kommenda, M., Burlacu, B., Kronberger, G., and Affenzeller, M. Parameter identification for symbolic regression using nonlinear least squares. *Genetic Programming and Evolvable Machines*, 21(3):471–501, 2020.

La Cava, W., Orzechowski, P., Burlacu, B., de Franca, F., Virgolin, M., Jin, Y., Kommenda, M., and Moore, J. Contemporary symbolic regression methods and their relative performance. In Vanschoren, J. and Yeung, S. (eds.), *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks*, volume 1. Curran, 2021.

Lalande, F., Matsubara, Y., Chiba, N., Taniai, T., Igarashi, R., and Ushiku, Y. A transformer model for symbolic regression towards scientific discovery. *arXiv preprint arXiv:2312.04070*, 2023.

Landajuela, M., Petersen, B. K., Kim, S. K., Santiago, C. P., Glatt, R., Mundhenk, T. N., Pettit, J. F., and Faissol, D. M. Improving exploration in policy gradient search: Application to symbolic optimization. In *1st Mathematical Reasoning in General Artificial Intelligence, International Conference on Learning Representations (ICLR)*, 2021.

Landajuela, M., Lee, C. S., Yang, J., Glatt, R., Santiago, C. P., Aravena, I., Mundhenk, T., Mulcahy, G., and Petersen, B. K. A unified framework for deep symbolic regression. *Advances in Neural Information Processing Systems*, 35:33985–33998, 2022.

LeCun, Y., Bengio, Y., and Hinton, G. Deep learning. *nature*, 521(7553):436–444, 2015.

Levenberg, K. A method for the solution of certain nonlinear problems in least squares. *Quarterly of applied mathematics*, 2(2):164–168, 1944.

Marquardt, D. W. An algorithm for least-squares estimation of nonlinear parameters. *Journal of the society for Industrial and Applied Mathematics*, 11(2):431–441, 1963.

Martius, G. and Lampert, C. H. Extrapolation and learning equations, 2017. URL https://openreview.net/forum?id=BkgRp0FYe.

Meurer, A., Smith, C. P., Paprocki, M., Čertík, O., Kirpichev, S. B., Rocklin, M., Kumar, A., Ivanov, S., Moore, J. K., Singh, S., et al. Sympy: symbolic computing in python. *PeerJ Computer Science*, 3:e103, 2017.

Petersen, B. K., Larma, M. L., Mundhenk, T. N., Santiago, C. P., Kim, S. K., and Kim, J. T. Deep symbolic regression: Recovering mathematical expressions from data via risk-seeking policy gradients. In *International Conference on Learning Representations*, 2021a. URL https://openreview.net/forum?id=m5Qsh0kBQG.

Petersen, B. K., Santiago, C., and Landajuela, M. Incorporating domain knowledge into neural-guided search via in situ priors and constraints. In *8th ICML Workshop on Automated Machine Learning (AutoML)*, 2021b. URL https://openreview.net/forum?id=yAis5yB9MQ.

Raissi, M., Perdikaris, P., and Karniadakis, G. E. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational physics*, 378:686–707, 2019.

Rajeswaran, A., Ghotra, S., Ravindran, B., and Levine, S. EPOpt: Learning robust neural network policies using model ensembles. In *International Conference on Learning Representations*, 2017. URL https://openreview.net/forum?id=SyWvgP5el.

Ranganathan, A. The levenberg-marquardt algorithm. *Tutoral on LM algorithm*, 11(1):101–110, 2004.

Robbins, H. and Monro, S. A stochastic approximation method. *The annals of mathematical statistics*, pp. 400–407, 1951.

Russeil, E., de França, F. O., Malanchev, K., Burlacu, B., Ishida, E. E. O., Leroux, M., Michelin, C., Moinard, G., and Gangler, E. Multi-view symbolic regression, 2024.

Sahoo, S., Lampert, C., and Martius, G. Learning equations for extrapolation and control. In *International Conference on Machine Learning*, pp. 4442–4450. PMLR, 2018.

Schmidt, M. and Lipson, H. Distilling free-form natural laws from experimental data. *science*, 324(5923):81–85, 2009.

Scholl, P., Bieker, K., Hauger, H., and Kutyniok, G. Parfam–symbolic regression based on continuous global optimization. *arXiv preprint arXiv:2310.05537*, 2023.

Stephens, T. Gplearn, 2015. URL https://gplearn.readthedocs.io/en/stable/index.html.

Tenachi, W., Ibata, R., and Diakogiannis, F. I. Deep Symbolic Regression for Physics Guided by Units Constraints: Toward the Automated Discovery of Physical Laws. *ApJ*, 959(2):99, December 2023a. doi: 10.3847/1538-4357/ad014c.

Tenachi, W., Ibata, R., and Diakogiannis, F. I. Deep symbolic regression for physics guided by units constraints: toward the automated discovery of physical laws. *arXiv e-prints*, art. arXiv:2303.03192, March 2023b. doi: 10.48550/arXiv.2303.03192.

Tenachi, W., Ibata, R., François, T. L., and Diakogiannis, F. I. Class Symbolic Regression: Gotta Fit 'Em All. *ApJL*, 969(2):L26, July 2024. doi: 10.3847/2041-8213/ad5970.

Udrescu, S.-M. and Tegmark, M. Ai feynman: A physics-inspired method for symbolic regression. *Science Advances*, 6(16):eaay2631, 2020.

Udrescu, S.-M., Tan, A., Feng, J., Neto, O., Wu, T., and Tegmark, M. Ai feynman 2.0: Pareto-optimal symbolic regression exploiting graph modularity. *Advances in Neural Information Processing Systems*, 33:4860–4871, 2020.

Virgolin, M., Alderliesten, T., Witteveen, C., and Bosman, P. A. N. Improving Model-Based Genetic Programming for Symbolic Regression of Small Expressions. *Evolutionary Computation*, 29(2):211–237, 06 2021. ISSN 1063-6560. doi: 10.1162/evco_a_00278. URL https://doi.org/10.1162/evco_a_00278.

Zhu, C., Byrd, R. H., Lu, P., and Nocedal, J. Algorithm 778: L-bfgs-b: Fortran subroutines for large-scale bound-constrained optimization. *ACM Transactions on mathematical software (TOMS)*, 23(4):550–560, 1997.