

Shape-constrained Symbolic Regression with NSGA-III

Christian Haider

Josef Ressel Center for Symbolic Regression
Heuristic and Evolutionary Algorithms Laboratory
University of Applied Sciences Upper Austria, Hagenberg, Austria

Abstract. Shape-constrained symbolic regression (SCSR) allows to include prior knowledge into data-based modeling. This inclusion allows to ensure that certain expected behavior is better reflected by the resulting models. The expected behavior is defined via constraints, which refer to the function form e.g. monotonicity, concavity, convexity or the models image boundaries. In addition to the advantage of obtaining more robust and reliable models due to defining constraints over the functions shape, the use of SCSR allows to find models which are more robust to noise and have a better extrapolation behavior. This paper presents a mutlicriterial approach to minimize the approximation error as well as the constraint violations. Explicitly the two algorithms NSGA-II and NSGA-III are implemented and compared against each other in terms of model quality and runtime. Both algorithms are capable of dealing with multiple objectives, whereas NSGA-II is a well established multi-objective approach performing well on instances with up-to 3 objectives. NSGA-III is an extension of the NSGA-II algorithm and was developed to handle problems with "many" objectives (more than 3 objectives). Both algorithms are executed on a selected set of benchmark instances from physics textbooks. The results indicate that both algorithms are able to find largely feasible solutions and NSGA-III provides slight improvements in terms of model quality. Moreover, an improvement in runtime can be observed using the many-objective approach.

Keywords: symbolic regression, shape-constraints, many-objective

1 Introduction

Due to their high complexity, the behavior and certain phenomena of systems and processes in critical application areas such as medical engineering or financial systems cannot be feasibly modeled from first principles alone. Thus, to handle the permanently increasing demand of high efficiency and accuracy as well as scale of experiments, data-based modeling approaches have become a standard.

Submitted manuscript to be published in *Computer Aided Systems Theory - EUROCAST 2022: 18th International Conference, Las Palmas de Gran Canaria, Feb. 2022*.

The increase of computational power and the rising popularity of machine learning algorithms for physical systems has given birth to a fourth paradigm of science "big data-driven science". Traditionally, machine learning algorithms are designed to train models, purely from data and achieve accurate predictions. These often leads to models represented as highly complex functions forms (e.g. neural networks) whose inner structure and working cannot be easily understood. Such models are called black-box models. However, it must be ensured that models found by data-driven algorithms conform to the functional principles of the modeled system, to ensure correct functionality after deployment. This is a topic especially relevant for scientific machine learning [2]. For example, physical laws such as conservation of energy, thermodynamics, or laws of gravity can be enforced for the modeling process when only a few data points are given or when they are biased to prescribe some expected behavior. Since a desired shape of the function form is enforced, these constraints are called *shape-constraints* [4,5,14].

A well-suited approach to incorporate shape constraints into modeling is symbolic regression (SR) [7,15,6]. Symbolic regression allows to find models represented as short closed-form functions. This increases the interpretability and transparency ("Explainable AI") of the models. SR finds the functional form as well as suitable parameters, which distinguishes this approach from conventional regression methods, where the functional form is already predefined and only the numerical coefficients have to be adjusted and optimized (e.g. neural networks).

In [8] the authors proposed shape-constrained symbolic regression (SCSR) a supervised machine learning approach that aims at both fitting trainings data and compliance with the given shape-constraints. It is shown that the resulting models, fit the training data well and even achieve slightly better training errors than standard genetic programming (GP) models in some cases, and that the models conform to the specified constraints.

The paper is structured as follows: In section 2, shape-constrained regression is presented in detail as well as the use of a many-objective approach. In section 3, similar work in this area is highlighted and presented. Section 4 and 5 present the experiments performed and results obtained, and conclude with a summary of the findings.

2 Shape-constrained regression

Shape-constrained regression (SCR) allows to enforce desired properties and behavior by specifying constraints which refer to the shape of the function, whereby the model is a function mapping real-valued inputs to real-valued outputs. These constraints are formulated based on prior knowledge stemming from empirical observations, domain experts knowledge, or physical principles. Mathematically these constraints can be expressed through partial derivatives. E.g. a monotonic increasing function can be enforced by a first order partial derivative of a specific variable. Table 1 highlights all mathematical definition of constraints considered in this work.

Property	Mathematical definition
Positivity	$f(X) > 0$
Negativity	$f(x) < 0$
Model bounds	$l \leq f(x) \leq u$
Monotonic-increasing	$\frac{\partial}{\partial x_1} f(x) > 0$
Monotonic-decreasing	$\frac{\partial}{\partial x_1} f(x) < 0$

Table 1: Mathematical formulation of constraints used in this work.

The evaluation of the constraints often requires approximation methods due to the need of finding the extrema of a non-linear function. For example a monotonic increasing constraint - partial derivative w.r.t the variable needs to be equal or greater than zero over the whole input domain. Therefore, the minimum value of the partial derivative in the given domain has to be found. If the model is non-linear, a non-linear optimization problem has to be solved, which is often a NP-hard problem which results in the need of approximation methods.

Considering approximation methods, it can be distinguished between pessimistic and optimistic approaches [3]. Pessimistic approaches guarantee that the found solution is indeed valid but often discard valid solution due to bound overestimation. Optimistic approaches can generate false-positives but guarantee that solutions classified as infeasible are indeed not feasible.

In this work interval arithmetic (IA) is used as a pessimistic approximation method for constraint evaluation.

2.1 Many-objective approach

In this work two multi/many-objective evolutionary search frameworks, as described in [5], are used. Both presented algorithm start with a random initialized population, represented as expression trees. Followed by a repeating main-loop: fitness evaluation, parental selection, recombination, and mutation. Both algorithms are configured equally: max number of evaluation 500000, tournament selection with a crowded group size of 5, single-subtree crossover, and a mutator that either changes a single node or a subtree with a randomly initialized subtree. Further parameter settings can be taken from [5].

The problem is modeled using a $1+n$ objectives approach - one main objective and one objective for each constraint specified. The first objective is handled in the data-based loss function and is minimizing the approximation error. In this paper the error is calculated using the normalized mean squared error (NMSE) in percent, as show in equation1, where y represents the target vector and \hat{y} the prediction vector.

$$NMSE(y, \hat{y}) = \frac{100}{var(y)N} \sum_{i=1}^N (y_i - \hat{y}_i)^2 \quad (1)$$

The following objectives are minimizing the constraint violations, whereas each constraint is treated in a separate objective:

$$P_i = P_i^{inf} + P_i^{sup} \quad (2)$$

where

$$\begin{aligned} P_i^{inf} &= |\min(\inf(f_i(x)) - \inf(c_i), 0)| \\ P_i^{sup} &= |\max(\sup(f_i(x)) - \sup(c_i), 0)| \end{aligned} \quad (3)$$

Where $\sup(x)$ and $\inf(x)$ are functions returning the superior and inferior bounds of the intervals.

3 Related work

The most related article for this work is presented in [8]. The authors introduce SCSR a method, which allows to use prior knowledge in a data-based modelling approach. The authors presented a single objective approach, which uses a feasible/infeasible population split. The proposed method was tested on a set of instances taken from physics textbook. The results showed that using a-priori knowledge helps with finding feasible solutions even on highly noise data and on extrapolation. Currently, the inclusion of additional domain knowledge in data-based modelling gets more and more attention in literature and there are some recent paper targeting this topic [2,16,11]. Auguste et al. presented two new methods to include monotonic constraints in regression and classification trees [1]. In [9] the authors present a multi-objective symbolic regression approach to minimize the approximation error on the training data as well as the constraint violations on the constraint dataset. Therefore, they extended the NSGA-II algorithm and used sampling to evaluate the constraints. The inclusion of prior knowledge in data-based modeling also plays an increasingly important role in modeling with neural networks as some recent articles show [18,12,10,11,17].

4 Experiments and results

In this paper we compare the multi-objective algorithm NSGA-II to the many-objective algorithm NSGA-III. The solution quality as well as the runtime of both algorithms are used for comparison. The experiments are executed on a set of different test instances.

4.1 Problem instances

To test both algorithms, expression from physics textbooks are used. Specifically, instances from the *Feynman Symbolic Regression Database* [19] are used. Therefore, only a subset of all instances are used, selected by the reported difficulty in [19] and only instances where shape-constraints could be derived. To generate training data, 300 points are sampled uniformly at random out of the expressions from table 2. Afterwards we split the points so that the first 10% and last 10%

are used as test set and the rest is used for training, this represents a test set outside the hull of the training data. In table 3 all derived constraints for each instance are shown. The constraint's column shows in the first tuple the desired domain constraint the following values define the constraints on each variable, where 0 means no constraint, 1 defines a monotonic increasing constraint, and -1 a monotonic decreasing function. The constraint definition for the *Pagie-1* instance follows the same principle, with the additional restriction that the defined constraints are valid only in certain range of the input variables, which is shown within the brackets.

Instance	Expression
I.6.20	$\exp\left(\frac{-\left(\frac{\theta}{\sigma}\right)^2}{2}\right) \frac{1}{\sqrt{2\pi}\sigma}$
I.9.18	$\frac{G m1 m2}{(x2-x1)^2+(y2-y1)^2+(z2-z1)^2}$
I.30.5	$\text{asin}\left(\frac{\text{lambda}}{nd}\right)$
I.32.17	$\frac{1}{2}\epsilon c E f^2 \frac{8\pi r^2}{3} \frac{\omega^4}{(\omega^2-\omega_0^2)^2}$
I.41.16	$\frac{h \omega^3}{\pi^2 c^2 (\exp(\frac{h\omega}{kbT})-1)}$
I.48.20	$\frac{m c^2}{\sqrt{1-\frac{v^2}{c^2}}}$
II.35.21	$n_{rho} mom \tanh\left(\frac{mom B}{kbT}\right)$
III.9.52	$\frac{p_d E f t}{h} \sin\left(\frac{(\omega-\omega_0)t}{2}\right)^2$
III.10.19	$mom \sqrt{Bx^2 + By^2 + Bz^2}$

Table 2: Problem instances taken from [19]

Additionally, a regression problem from [13] is added, it is called *Pagie-1* in the following. It's a problem instance with two variables as follows:

$$f(x, y) = \frac{1}{(1+x^{-4})} + \frac{1}{(1+y^{-4})} \quad (4)$$

It is evaluated over the range of:

$$-5 \leq x \leq 5 \quad \text{and} \quad -5 \leq y \leq 5 \quad (x, y \neq 0) \quad (5)$$

To generate trainings data for the *Pagie-1* instance the training set consists of data points spaced 0.4 apart between the limits, which results with a set of 676 distinct x, y data points.

4.2 Results

Table 4 shows the result of 10 independent runs over all test instances. The error is represented as normalized mean squared error (NMSE) in percent. The left most column shows the instance and in the following two columns the results of

Instance	Input space	Constraints
I.6.20	$(\sigma, \theta) \in [1..3]^2$	$([0..\infty], 0, -1)$
I.9.18	$(x1, y1, z1, m1, m2, G, x2, y2, z2) \in [3..4]^3 \times [1..2]^6$	$([0..\infty], -1, -1, -1, 1, 1, 1, 1, 1)$
I.30.5	$(\text{lambda}, n, d) \in [1..5]^2 \times [2..5]$	$([0..\infty], 1, -1, -1)$
I.32.17	$(\epsilon, c, Ef, r, \omega, \omega_0) \in [1..2]^5 \times [3..5]$	$([0..\infty], 1, 1, 1, 1, -1)$
I.41.16	$(\omega, T, h, kb, c) \in [1..5]^5$	$([0..\infty], 0, 1, -1, 1, -1)$
I.48.20	$(m, v, c) \in [1..5] \times [1..2] \times [3..20]$	$([0..\infty], 1, 1, 1)$
II.35.21	$(n_{rho}, mom, B, kb, T) \in [1..5]^5$	$([0..\infty], 1, 1, 1, -1, -1)$
III.9.52	$(pa, Ef, t, h, \omega, \omega_0) \in [1..3]^4 \times [1..5]^2$	$([0..\infty], 1, 1, 0, -1, 0, 0)$
III.10.19	$(mom, Bx, By, Bz) \in [1..5]^4$	$([0..\infty], 1, 1, 1, 1)$
Pagie-1	$(x, y) \in [-5..5]^2$	$([0..2], -1(x < 0), 1(x > 0), -1(y < 0), 1(y > 0))$

Table 3: Shape constraints used for each problem instance. *Input space* column refers to the variable domains. *Constraints* column represents the defined constraints over each variable. The first tuple represents the allowed model range the following values represent either no constraint over the variable for value 0, monotonic decreasing for value -1 or monotonic increasing for value 1.

both algorithms NSGA-II and NSGA-III are shown. It can be observed that both algorithms give similar results, but NSGA-III gives slightly better results over all instances, but without being statistically significant. In table 5 the runtime performance of both algorithms is compared. The left most columns shows again the instance followed by the two columns showing the median runtime measured in seconds. The results are similar to the error comparison in table 4, whereas the NSGA-III has advantage over all instances.

	NSGA-II	NSGA-III
I.6.20	20.88	19.14
I.30.5	7.32	6.24
I.32.17	7.17	6.38
I.41.16	18.50	15.21
I.48.20	24.19	22.58
II.35.21	14.60	14.54
III.9.52	89.03	89.00
III.10.19	11.30	10.62
Pagie-1	46.41	40.71

Table 4: Median test error (NMSE in %)

5 Summary

In this paper two multi-objective methods for shape-constrained symbolic regression are evaluated and compared to each other. The comparison of both methods was mainly motivated by the article [] in which the advantages of many-objective

	NSGA-II	NSGA-III
I.6.20	1798.02	1407.67
I.30.5	3621.99	3604.91
I.32.17	5812.10	4504.23
I.41.16	3858.61	2879.05
I.48.20	2825.38	1647.43
II.35.21	3217.67	3045.41
III.9.52	3009.62	2064.16
III.10.19	3939.14	2254.29
Pagie-1	4800.77	4105.86

Table 5: Median runtime (in seconds)

algorithms over multi-objective algorithms, when having more than 3 objectives is shown.

The two algorithms have been integrated in the heuristic and evolutionary algorithm framework (HeuristicLab) and are benchmarked on a set of equation from physics textbooks *Feynman Symbolic Regression Database*.

The results showed that using many-objective algorithms over multi-objective algorithms has slightly advantages in some instances but for a significant difference the instances had too less objectives. It is also shown that using the many-objective approach can help with runtime.

Although no statistical significant differences were achieved, it is shown that many-objective algorithms can help with performance increases on instances with more than 3 objectives. A more detailed comparison on instances with more than 10 objectives will be investigated in further research.

Acknowledgement

The financial support by the Christian Doppler Research Association, the Austrian Federal Ministry for Digital and Economic Affairs and the National Foundation for Research, Technology and Development is gratefully acknowledged.

References

1. Auguste, C., Malory, S., Smirnov, I.: A better method to enforce monotonic constraints in regression and classification trees (2020). <https://doi.org/10.48550/ARXIV.2011.00986>, <https://arxiv.org/abs/2011.00986>
2. Baker, N., Alexander, F., Bremer, T., Hagberg, A., Kevrekidis, Y., Najm, H., Parashar, M., Patra, A., Sethian, J., Wild, S., et al.: Workshop report on basic research needs for scientific machine learning: Core technologies for artificial intelligence. Tech. rep., USDOE Office of Science (SC), Washington, DC (United States) (2019)
3. Gupta, M., Cotter, A., Pfeifer, J., Voevodski, K., Canini, K., Mangylov, A., Moczydlowski, W., van Esbroeck, A.: Monotonic calibrated interpolated

- look-up tables. *Journal of Machine Learning Research* **17**(109), 1–47 (2016), <http://jmlr.org/papers/v17/15-243.html>
4. Gupta, M., Louidor, E., Mangylov, O., Morioka, N., Narayan, T., Zhao, S.: Multi-dimensional shape constraints. In: *International Conference on Machine Learning (ICML 2020)*. pp. 3918–3928. PMLR (2020)
 5. Haider, C., de França, F.O., Burlacu, B., Kronberger, G.: Using shape constraints for improving symbolic regression models. *arXiv preprint arXiv:2107.09458* pp. 1–32 (2021)
 6. Koza, J.R., Poli, R.: Genetic programming. In: *Search methodologies*, pp. 127–164. Springer (2005)
 7. Koza, J.R., et al.: *Genetic programming II*, vol. 17. MIT press Cambridge, MA (1994)
 8. Kronberger, G., de Franca, F.O., Burlacu, B., Haider, C., Kommenda, M.: Shape-Constrained Symbolic Regression—Improving Extrapolation with Prior Knowledge. *Evolutionary Computation* pp. 1–24 (10 2021)
 9. Kubalík, J., Derner, E., Babuška, R.: Symbolic regression driven by training data and prior knowledge. In: *Proceedings of the 2020 Genetic and Evolutionary Computation Conference*. pp. 958–966 (2020)
 10. Li, L., Fan, M., Singh, R., Riley, P.: Neural-guided symbolic regression with asymptotic constraints (2019). <https://doi.org/10.48550/ARXIV.1901.07714>, <https://arxiv.org/abs/1901.07714>
 11. Liu, X., Han, X., Zhang, N., Liu, Q.: Certified monotonic neural networks. In: *Advances in Neural Information Processing Systems*. vol. 33, pp. 15427–15438. Curran Associates, Inc (2020)
 12. Muralidhar, N., Islam, M.R., Marwah, M., Karpatne, A., Ramakrishnan, N.: Incorporating prior domain knowledge into deep neural networks. In: *2018 IEEE International Conference on Big Data (Big Data)*. pp. 36–45 (2018). <https://doi.org/10.1109/BigData.2018.8621955>
 13. Pagie, L., Hogeweg, P.: Evolutionary Consequences of Coevolving Targets. *Evolutionary Computation* **5**, 401–18 (02 1997)
 14. Papp, D., Alizadeh, F.: Shape-constrained estimation using nonnegative splines. *Journal of Computational and Graphical Statistics* **23**(1), 211–231 (2014)
 15. Poli, R., Langdon, W.B., McPhee, N.F., Koza, J.R.: *A field guide to genetic programming*. Lulu. com (2008)
 16. Rai, A.: Explainable AI: From black box to glass box. *Journal of the Academy of Marketing Science* **48**(1), 137–141 (2020)
 17. Raissi, M., Perdikaris, P., Karniadakis, G.: 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). <https://doi.org/https://doi.org/10.1016/j.jcp.2018.10.045>, <https://www.sciencedirect.com/science/article/pii/S0021999118307125>
 18. Stewart, R., Ermon, S.: Label-free supervision of neural networks with physics and domain knowledge. In: *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*. p. 2576–2582. AAAI’17, AAAI Press (2017)
 19. Udrescu, S.M., Tegmark, M.: AI Feynman: A physics-inspired method for symbolic regression. *Science Advances* **6**(16), eaay2631 (2020). <https://doi.org/10.1126/sciadv.aay2631>