Systems & Control Transactions

Original Research Article
Peer Reviewed Conference Proceeding
ESCAPE 36 - European Symposium on Computer Aided Process Engineering
Sheffield, UK. 21-24 June 2026

PSE PRESS

Predicting Ecotoxicity (HC50) Values Using Symbolic Regression for Transparent Life Cycle Assessment

Abdulhakeem Ahmed[a], Nitya Kasera[a], and Ana I. Torres[ab]*

[a] Department of Chemical Engineering, Carnegie Mellon University, Pittsburgh, PA, 15213, USA
[b] Wilton E. Scott Institute for Energy Innovation, Carnegie Mellon University, Pittsburgh, PA, 15213, USA
* Corresponding author: aitorres@cmu.edu

ABSTRACT

Accurate life cycle assessment (LCA) depends on robust characterization factors (CFs), which quantify impacts such as ecotoxicity through the integration of fate (FF), exposure (XF), and effect (EF) factors. While databases such as USEtox and Ecoinvent provide essential CFs, significant data gaps remain, particularly in ecotoxicity endpoints like hazardous concentration 50% (HC_{50}), which directly inform effect factor calculations. Existing machine learning models can predict such values, but they often lack interpretability, which limits trust and transparency in environmental modeling. To address this, a machine learning framework is applied that utilizes symbolic regression (SR) and genetic programming (GP) to predict missing HC_{50} values from physicochemical descriptors. A dataset with 14 descriptors was used to train SR models capable of generating interpretable mathematical expressions that link chemical properties to HC_{50} values. SR models were benchmarked against prominent black-box models such as random forest (RF) and neural network (NN) models. SR performance was consistent across multiple parameter configurations, while revealing recurring patterns in variable selection. These results demonstrate that symbolic regression can both predict ecotoxicity values with comparative accuracy and provide transparent functional relationships, thereby filling existing data gaps in characterization factors needed for a complete LCA.

Keywords: Life Cycle Assessment, Symbolic Regression, Machine Learning

INTRODUCTION

Life Cycle Assessment is a cornerstone of environmental modeling, enabling quantification of the potential impacts of chemical substances across their life cycles. Among the various impact categories, ecotoxicity assessment is particularly important for evaluating risks to aquatic and terrestrial ecosystems. Ecotoxicity characterization within LCA depends on effect factors such as the *Hazardous Concentration for 50% of species* (HC_{50}), which represents the concentration at which half of the tested species experience a measurable toxic effect. Reliable HC_{50} data for chemicals are essential for computing characterization factors used in assessment frameworks such as USEtox and TRACI [1, 2]. However, measured HC_{50} values are missing for many chemicals, limiting the completeness of LCA inventories and impact models.

Machine learning (ML) models are powerful tools for estimating missing ecotoxicity values based on physicochemical properties of substances. While black-box algorithms such as random forests or neural networks provide accurate predictions, their limited interpretability can hinder informed decision-making. Symbolic regression (SR) offers a promising alternative, deriving analytical expressions that directly relate input descriptors to target variables. The use of explainable data-driven models in chemical engineering is well established, with a review of past efforts, developments, and future directions presented in [3]. Most interpretable SR methods rely on genetic programming, as discussed in [4], where it was applied to mechanism discovery. Researchers have also explored optimization-based formulations; for example, in [5], a mixed-integer nonlinear programming (MINLP) model is proposed for SR to compete with traditional, stochastic genetic programming methods. The benefit of optimization is stronger optimality guarantees, unlike

genetic algorithms, which can be computationally less expensive but may yield lower solution quality [6].

Despite growing interest in data-driven methods, few studies have applied ML models to predict chemical properties in the context of life-cycle assessment or environmental impact evaluation. One such example is [7], where the authors employed ML models to estimate environmental endpoint indicators such as human health, resource utilization, and climate change, though these efforts relied exclusively on black-box models.

The present study applies symbolic regression to predict HC_{50} values using a dataset of 14 physicochemical descriptors (e.g., molecular weight, partition coefficients, solubility, and bioconcentration factor). The overall goal is to derive an interpretable model that connects the property to the descriptors. The applied SR workflow encompasses data preprocessing, model training via evolutionary optimization, and performance evaluation across multiple configurations of population size and generation count. Model interpretability is investigated through the analysis of recurrent variable patterns and expression structures.

By comparing predicted HC_{50} values against baseline models also assessed in [8] for the same dataset, we evaluate whether symbolic regression can achieve comparative predictive performance while preserving explicit analytical expressions. These equations can help bridge existing data gaps in effect factor estimation, thereby improving the robustness of LCA-based ecotoxicity characterization. The remainder of this paper details the symbolic regression framework, presents the results and interpretability analysis, and examines implications for model generalization, descriptor roles, and ecotoxicity characterization.

METHODS

Dataset

The ecotoxicity dataset used in this study originates from [8] and contains 2, 122 chemicals with experimentally determined values for HC_{50} and 14 descriptors. These descriptors include molecular weight, boiling point, and water solubility and are listed in Table 1. The descriptors were compiled into a numerical matrix and HC_{50} values were log-transformed ($\log(HC_{50})$) to form the target vector.

Table 1: Description of molecular features

Variable	Descriptor	Abbreviation
X_0	Molecular Weight	MW
X_1	Atmospheric Hydroxylation Rate	AOH
X_2	Bioconcentration Factor	BCF
X_3	Biodegradation half-life	BioHL
X_4	Boiling point	BP
X_5	Henry's law constant	HL
X_6	Fish biotransformation half-life	KM
X_7	Octanol/air partition coefficient	KOA
X_8	Octanol/water partition coefficient	LogP
X_9	Melting point	MP
X_{10}	Soil adsorption coefficient	KOC
X_{11}	Vapor pressure	VP
X_{12}	Water solubility	WS
X_{13}	Mode of action	MoA

Baseline Models

Two common ML models are used to benchmark the symbolic regression approach: a Random Forest (RF) regressor and a Neural Network (NN). The RF model consists of an ensemble of decision trees constructed using bootstrap samples of the training data. Each tree is built by recursively partitioning the descriptor space using impurity-minimizing splits, while random feature selection at each node promotes decorrelation between trees. Final predictions are obtained by averaging the outputs of all trees. This structure enables RF to capture nonlinear interactions between descriptors while maintaining robustness to noise and outliers [9].

The NN model is a fully connected feed-forward network trained using backpropagation. The network maps the standardized descriptor vector to the $\log(HC_{50})$ target through a sequence of nonlinear transformations defined by hidden layers and activation functions. By adjusting its weights during training, the NN model can capture complex continuous relationships. Although it requires careful regularization to avoid overfitting on medium-sized datasets [10].

Symbolic Regression Framework

Symbolic regression was implemented using a genetic programming (GP) framework developed in [11]. SR searches for explicit analytical expressions that relate the input descriptors to the target $\log(HC_{50})$. The approach evolves populations of candidate mathematical expressions using operations such as mutation, crossover, and selection.

The GP search was configured to explore functional structures while preventing unnecessary algebraic complexity. The operator set was restricted to the four arithmetic operations {+, −, ×, ÷}. Candidate expressions were initialized as shallow trees with depths between 0 and 3, and the maximum allowable depth during evolution was

constrained to 10. Population sizes ranged from 4 to 12 individuals, and each evolutionary run proceeded for 100–2, 000 generations.

At every generation, each candidate expression was evaluated using the root-mean-square error (RMSE) between predicted and measured $\log(\mathrm{HC}_{50})$ values. To balance accuracy and interpretability, adjusted R^2 was incorporated into the fitness function to penalize unnecessarily complex expressions. After the evolutionary search identified a set of high-performing symbolic structures, the coefficients associated with each expression were re-optimized using a post-evolution linear regression step.

The final SR model takes the general form shown in Equation (1), where each term $f_i(X)$ represents an evolved symbolic transformation of the descriptor set $\{X_0, X_1, \ldots, X_{13}\}$, and the coefficients β_i correspond to optimized weights.

$$\hat{y} = \sum_i^k \beta_i f_i(X) \qquad (1)$$

Model development

All baseline models were trained using a three-way data partition consisting of training (70%), validation (10%), and testing subsets (20%). The training set was used for model fitting, the validation set for hyperparameter tuning, and the test set for final out-of-sample evaluation. The SR development strategy described in [11] is applied here to develop the SR models. The strategy involves training and validation on multiple data splits and GP parameters. To address the probabilistic nature of the algorithm, multiple runs are made for each data split and configuration. For the study, 100 expressions are generated per configuration and data split. Five data splits are evaluated for six parameter configurations varying in population size and number of generations, yielding a total of 3000 expressions.

Population size and number of generations control how thoroughly the genetic program explores the search space. Higher values for both increase the computational budget, and larger populations increase the set of candidate solutions maintained and evolved in each generation. By assessing various configurations, we can evaluate the effect of these parameters on predictive performance. This is done by performing statistical tests on the validation set to compare different data splits and model configurations. The Friedman and Wilcoxon Rank tests are employed to measure statistical significance between configurations and data splits. The Friedman test determines whether configurations differ significantly within each split. The Wilcoxon test evaluates statistical significance in performance differences between configurations. All tests are performed for a 99% confidence level.

The development of the RF and NN baselines is streamlined by the scikit-learn [12] and hyperopt [13] Python packages for model construction and hyperparameter optimization, respectively. Model accuracy was evaluated using the standard regression metrics (RMSE and R^2). The coefficient of determination (R^2) quantifies the proportion of variance in the measured HC_{50} values explained by the model, while the root-mean-square error (RMSE) measures the average magnitude of prediction error in the units of the log-transformed target variable. These metrics were computed for the training, validation, and test subsets to evaluate model fit and generalization behavior.

Feature Importance and Structural Analysis

The contribution of each feature to the predicted target value for all models is evaluated using SHAP (SHapley Additive exPlanations) [14]. SHAP is an approach for explaining individual predictions of an ML model by attributing a contribution value to each input feature. It builds on Shapley values from cooperative game theory, which fairly distribute total gains or costs among collaborating players. In this context, each feature acts as a player, and SHAP fairly allocates the model's prediction difference (from a baseline) across them.

In addition to understanding how individual features contribute to model predictions, we examine the structural diversity of the expressions generated via SR. This structural analysis provides complementary insights into the models and their underlying patterns. To characterize the diversity of the discovered expressions, we perform structural pattern mining by extracting abstract syntax tree (AST)-based features for each unique mathematical expression. Each expression is parsed into its AST representation, from which quantitative features, including operation counts, tree structural properties (depth, total nodes), and variable usage patterns, are extracted. Two feature extraction modes are considered: a variable-agnostic mode that treats all variables as generic placeholders (focusing purely on the expression skeletal structure) and a variable-specific mode that encodes which molecular descriptors are used with binary indicator features. AST features are compiled into a feature matrix with rows representing unique expressions and columns representing the extracted structural characteristics. Once these feature vectors are obtained, a systematic comparison of the structural characteristics of the different expressions follows.

To quantify structural similarity between expressions, we compute pairwise Euclidean similarity matrices from the standardized feature vectors. Euclidean similarity measures the distance between feature vectors with values ranging from 0 (completely dissimilar) to 1 (identical) after taking the normalized inverse distance. By computing separate similarity metrics for both modes, we assess whether expressions share similar structural motifs and/or molecular descriptors.

To illustrate this, consider the analytical expressions

in Equation (2). In the variable-agnostic case, we evaluate the number of variables used, the types of operations, and the overall structure. The variables, X_0,X_1, and X_2 are treated as interchangeable placeholders, as are X_5,X_7, and X_9. Consequently, $f_1$and f_2 are structurally similar despite their different variable names, resulting in a Euclidean similarity score of 1. In contrast, in the variable-specific case we also consider exact variable usages. Since $f_1$and f_2 use different variables, the measured similarity score is low, with a score of 0.143.

$$\begin{aligned} f_1 &= X_0 + X_1X_2 \\ f_2 &= X_5 + X_7X_9 \end{aligned} \quad (2)$$

RESULTS & DISCUSSION

SR Hyperparameter Tuning

Symbolic regression experiments were performed for six configurations of generation and population size pairs; performance was measured by the RMSE values on the validation set. Figure 1 depicts the RMSE distribution for each configuration across all data splits. Model statistics averaged across the splits for the validation set are recorded in Table 2.

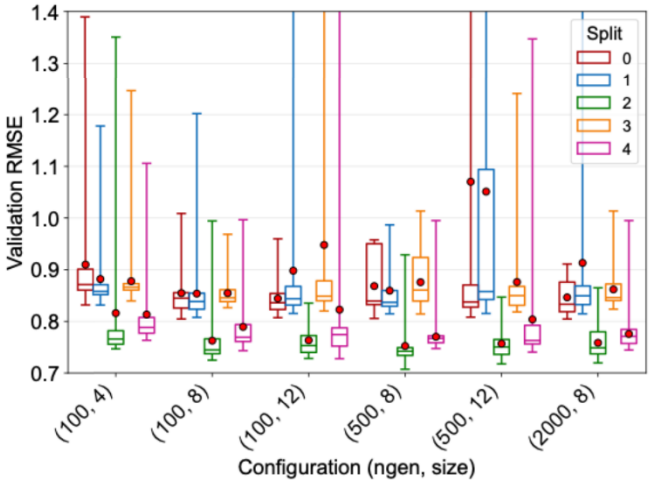


Figure 1: RMSE distribution across evolutionary parameter configurations and data splits. The red dot is the mean. In the box, the middle line is the median, the bottom line is the 1st quartile, and the top line is the 3rd quartile. The tails are the minimum and maximum values.

Friedman and Wilcoxon tests were conducted across data splits and configurations. In all splits, the Friedman test yielded p-values ≤ 0.01, indicating statistically significant differences among configurations. Building on these results, of the 75 Wilcoxon pairwise comparisons, 32 were significant, with the parameter set (ngen = 100, size = 8) showing statistical dominance. Notably, this configuration also exhibits a stable RMSE distribution and the lowest mean validation RMSE (see Table 2); therefore, it was selected as the optimal SR model parameter set. Consequently, all subsequent SR model results and baseline comparisons are based on this configuration, and use Split 1 to ensure consistent training, validation, and testing sets. The SR model with the lowest validation RMSE under these specifications is presented in Equation (3).

Table 2: RMSE validation statistics averaged across split sets.

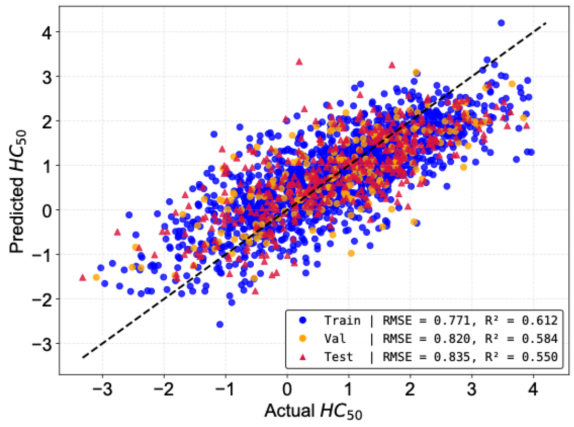
Config.	Mean	Median	Std
(100, 4)	0.860	0.856	0.0985
(100, 8)	0.823	0.826	0.0652
(100, 12)	0.855	0.831	0.3044
(500, 8)	0.825	0.829	0.0699
(500, 12)	0.912	0.828	0.9603
(2000, 8)	0.831	0.827	0.1161

$$\begin{aligned} \log(\mathrm{HC}_{50}) = \; & 0.241X_{12} + 0.279X_{13} - 0.259X_2 \\ & -0.113X_9 - 0.000413\frac{X_8^2}{X_2X_9} \\ & + 0.106\frac{X_8}{X_2X_5} + 0.145\frac{-X_1 + \frac{X_{11}}{X_4} - X_{13}}{X_0} \\ & + 27.1\frac{-X_1 + \frac{X_{11}}{X_4} - X_{13}}{X_1X_{13}(X_1 - X_7)} \end{aligned} \quad (3)$$

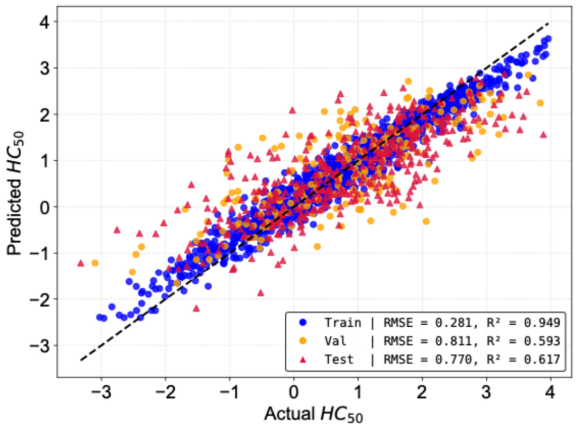
As shown, Equation (3) is a compact interpretable model with high variable diversity, using 11 out of the 14 available descriptors; several appear multiple times. The optimal weights (β_i) are the coefficients preceding each subexpression. Prominent monomials include water solubility (X_{12}), mode of action (X_{13}), bioconcentration factor (X_2), and melting point (X_9). Based on the signs of each variable, water solubility and mode of action appear to be positively correlated with $\log(\mathrm{HC}_{50})$, while bioconcentration factor and melting point are inversely correlated with $\log(\mathrm{HC}_{50})$. However, this is not the full picture, considering the presence of polynomial terms. These relationships are made explicit via feature importance analysis in the subsequent section.

Comparative Performance of SR, RF, and NN

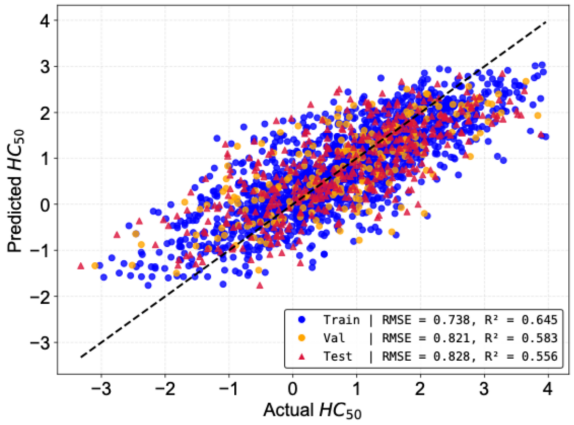
All models were evaluated using the same train/validation/test sets and assessed using the coefficient of determination (R^2) and the root-mean-square error (RMSE). The performance results for training and testing are summarized in Table 3. Furthermore, parity plots (Figure 2) provide a visual assessment of model generalization, comparing SR, RF, and NN models.



(a) SR model



(b) RF model



(c) NN model

Figure 2: Parity plots comparing predicted and actual $\log(\mathrm{HC}_{50})$ values for the SR, RF, and NN models on split set 1.

As shown in Table 3, the RF model achieves the highest performance on the training set, but its accuracy drops considerably on the test set- suggesting overfitting. This is evident from the large differences in both RMSE and R^2 between the training and test results. The NN model displays lower performance than the RF model in training and testing, while the SR model has the lowest training scores but performs about equally to the NN model on the test set. While RF achieves the strongest apparent fit on the training data, this advantage does not translate to unseen samples. On average, all three models perform roughly equally on the test set across all data splits.

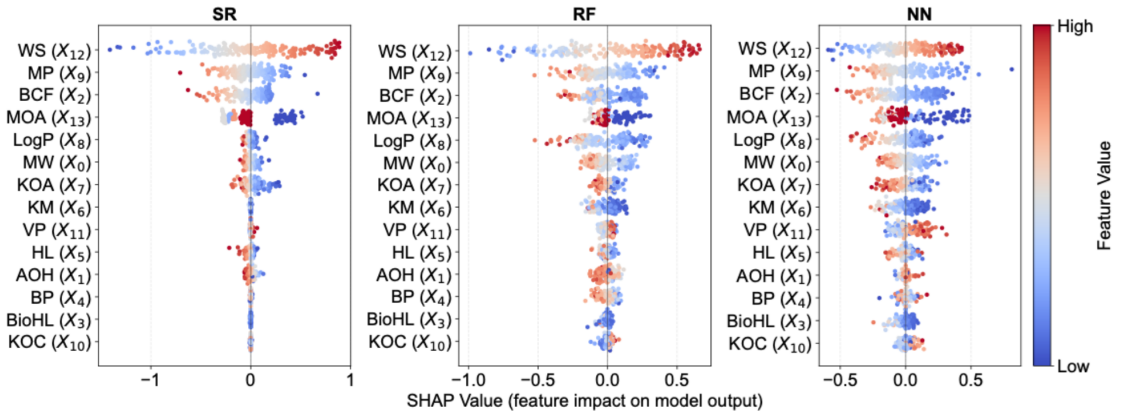


Figure 3. SHAP-based feature importances for SR, RF, and NN models. Particles are samples representing the impact distribution of each feature on the target output, indicating the direction that each feature value drives the predicted $\log(\mathrm{HC}_{50})$ output. High positive values are positively correlated with the target, and high negative values are negatively correlated.

Table 3: Average model performance metrics for RF, NN, and SR models on 5 data splits. SR corresponds to configuration (*ngen*=100, *size*=8). Standard deviations in parentheses.

Model	R^2 (Train)	RMSE (Train)	R^2 (Test)	RMSE (Test)
RF	0.930 (0.0446)	0.322 (0.0874)	0.609 (0.0136)	0.766 (0.0142)
NN	0.719 (0.0598)	0.659 (0.0707)	0.573 (0.0269)	0.800 (0.0225)
SR	0.606 (0.004)	0.785 (0.012)	0.573 (0.019)	0.801 (0.030)

Feature Importance Across Models

The feature importances for the three models, obtained using the SHAP methodology, are shown in Figures 3 and 4. Figure 3 describes the effect of each feature on $\log(\mathrm{HC}_{50})$ prediction, while Figure 4 shows the cumulative mean absolute SHAP value for all features of the compared models. The two most important features across all models are water solubility (X_{12}) and melting point (X_9). The next prominent features include bioconcentration factor (X_2), mode of action (X_{13}), and the octanol/water partition coefficient (X_8).

Figure 3 reveals that for the top features, higher water solubility increases the predicted $\log(\mathrm{HC}_{50})$ value, while higher melting point reduces the predicted value. A negative correlation is also observable for bioconcentration factor, mode of action, and octanol/water partition coefficient with higher feature values reducing the predicted $\log(\mathrm{HC}_{50})$ value.

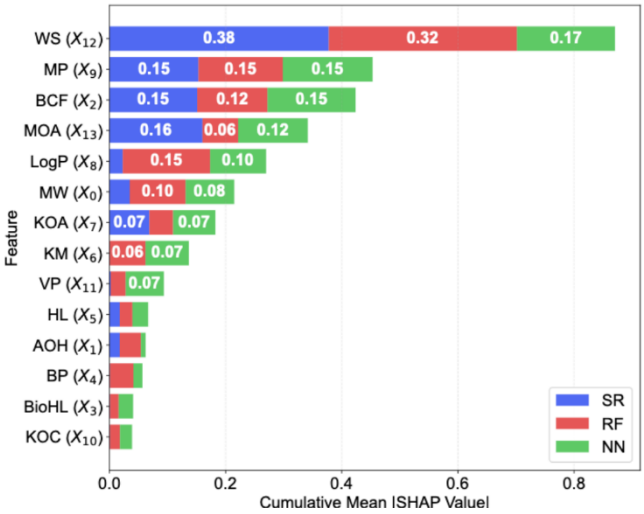


Figure 4. Mean absolute Shapley values for SR, RF, and NN models.

Overall, the findings of the feature importance analysis agree with the visual inspection of the expression in Equation (3). It clarifies the contributions of the octanol/water partition coefficient (X_8) and mode of action (X_{13}), which are ambiguous in Equation (3) due to their repeated occurrences. The analysis also reveals that vapor pressure (X_{11}) and boiling point (X_4) primarily act as structural constructors in Equation (2), as they do not exert a direct influence on the predicted value, as shown by Figure 4. While feature importances can be used to effectively extract variable information from black box models, an explicit expression remains desirable. It can be inspected against domain knowledge to verify if the discovered relationships are plausible and consistent with established theory or intuition. Furthermore, when an expression satisfies these criteria, it may uncover new

mechanistic insights.

Structural Expression Analysis

Having established how the individual features in each model contribute to predicting the target, we next examine the structural characteristics of the expressions obtained by symbolic regression. The goal is to see if a common canonical form emerges among the generated expressions. From this data-driven assessment, we seek to uncover mechanistic insights and identify the elements of these expressions that drive their convergence.

Across all evolutionary configurations, 3000 symbolic expressions were generated, of which 999 are unique and used to quantify structural similarity. Figure 5 summarizes the distribution of these expressions. The most frequent expression (Equation (4)) occurs 16 times, while roughly 300 expressions appear only once, indicating that most derived expressions are recurrent.

$$\begin{aligned}\log(\mathrm{HC}_{50}) = \; & 0.161X_1 - 0.000167X_{10} \\ & - 0.265X_{12} - 0.170X_2 \\ & - 0.00133X_8 - 0.161X_9 + \frac{0.654}{X_{13}}\end{aligned} \tag{4}$$

Comparing the expression with the most frequent appearance against Equation (3), there is a clear divergence in the structures. Equation (4) is more compact and uses fewer features than Equation (3). Except for the soil adsorption coefficient (X_{10}), which is accompanied with a low coefficient value, all variables present in (4) also appear in (3).

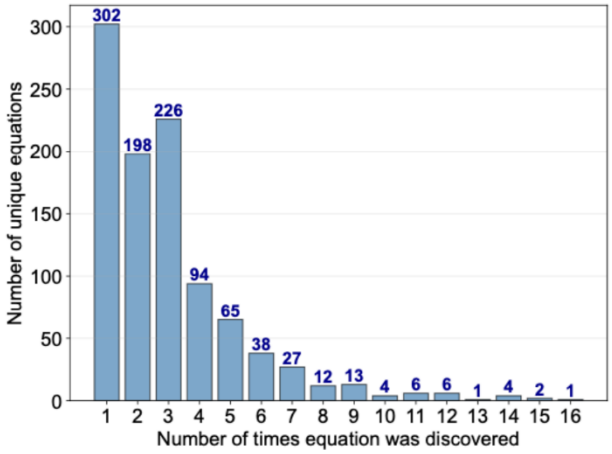


Figure 5. Frequency of unique expressions from all SR models. Labels above each bar represent the number of unique expressions for the corresponding appearance count on the x-axis.

After identifying the unique expressions, we

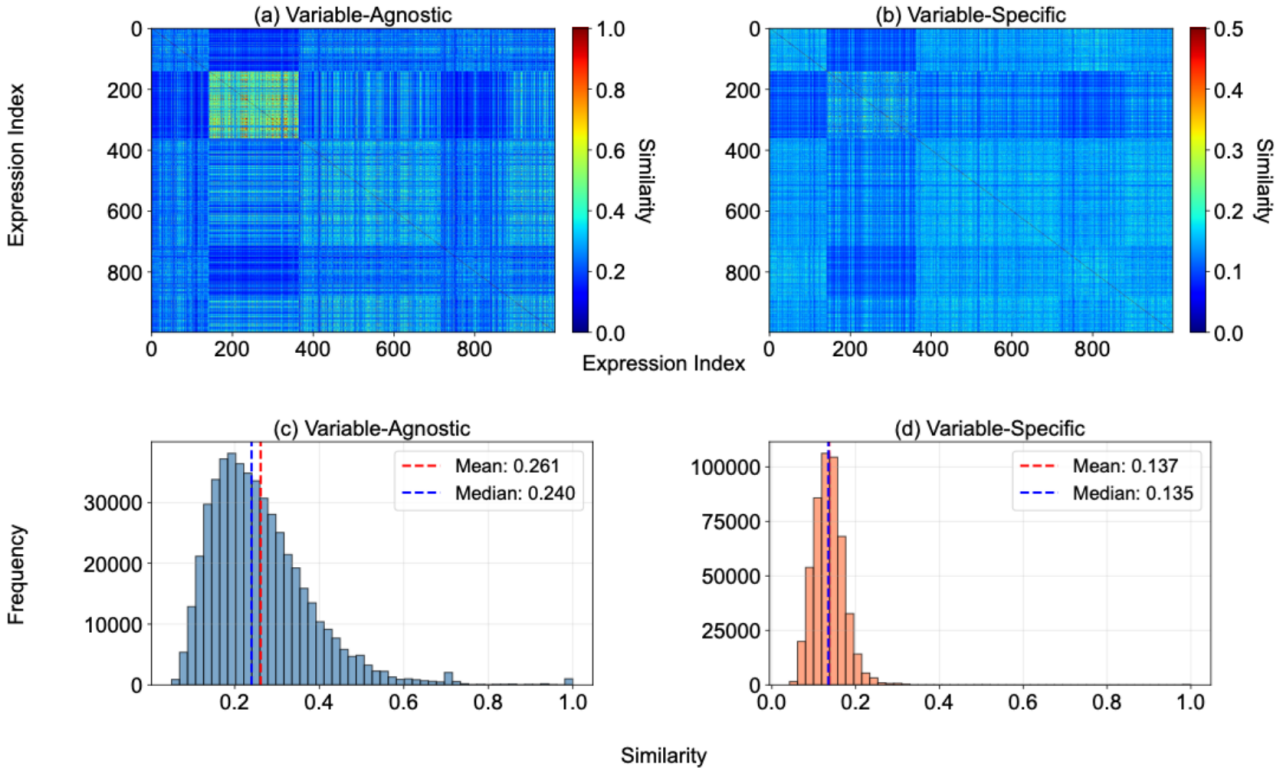


Figure 6. Characterization of structural diversity of SR expressions for variable-agnostic (left) and variable-specific (right) structural signatures. Frequencies in (c) and (d) are the number pairs appears with the similarity score on the x-axis.

proceed to characterize the structural diversity of the SR expressions for both variable-agnostic and variable-specific evaluations by measuring the similarity between expression pairs. The findings of this analysis are presented in Figure 6. Figure 6 (a and c) shows that for the variable-agnostic case (variables not explicitly considered), most expressions are not structurally similar, with a mean similarity score of 0.261. This means that expressions are extremely diverse in their use of number/type of operators, as well as in expression lengths. If we consider Equations (3) and (4), for example, the similarity score is 0.138, which falls below the mean.

The variable-specific case shown in Figure 6 (b and d) also indicates high feature diversity with a mean similarity score of 0.137. Again, comparing Equations (3) and (4), the similarity score in this case is 0.110. The near-zero mean and the distribution of the similarity scores shown in Figure 6 suggest that most expressions have little to no overlap in the overall structure. Yet, performance remains consistent across all models.

CONCLUSION

This study demonstrates that symbolic regression is a strong alternative to conventional black-box models for predicting ecotoxicity effect factors used within life cycle assessment frameworks. The symbolic regression models showed stable performance across multiple evolutionary configurations. When benchmarked against random forest and neural network models, symbolic regression achieved comparable predictive accuracy while offering an explicit analytical expression. This is particularly valuable for LCA, where traceability and scientific justification of characterization factors are paramount. Furthermore, feature importances obtained with SHAP were consistent across models, reinforcing SR's ability to compete with traditional black-box methods. Lastly, the structural analysis of the derived SR expressions revealed substantial variation, indicating that caution is needed when assigning causality. At the same time, this variation can be viewed positively, as it provides multiple pathways to accurately compute a property based on the data available. In summary, our results highlight symbolic regression as a promising method for reducing data gaps in CF estimation.

ACKNOWLEDGEMENTS

Funding received under the GEM Fellowship is gratefully acknowledged.

REFERENCES

1. Rosenbaum RK, Bachmann TM, Gold LS, Huijbregts MAJ, Jolliet O, Juraske R, Koehler A, Larsen HF, MacLeod M, Margni M, McKone TE, Payet J, Schuhmacher M, van de Meent D, Hauschild MZ. Usetox—the UNEP-SETAC toxicity model: recommended characterisation factors for human toxicity and freshwater ecotoxicity in life cycle impact assessment. Int J Life Cycle Assess 13:532-546 (2008). https://doi.org/10.1007/s11367-008-0038-4
2. Bare J. TRACI 2.0: the tool for the reduction and assessment of chemical and other environmental impacts 2.0. Clean Techn Environ Policy 13:687-696 (2011). https://doi.org/10.1007/s10098-010-0338-9
3. Venkatasubramanian V. The promise of artificial intelligence in chemical engineering: is it here, finally?. AIChE Journal 65:466-478 (2018). https://doi.org/10.1002/aic.16489
4. Chakraborty A, Sivaram A, Venkatasubramanian V. AI-DARWIN: a first principles-based model discovery engine using machine learning. Computers & Chemical Engineering 154:107470 (2021). https://doi.org/10.1016/j.compchemeng.2021.107470
5. Cozad A, Sahinidis NV. A global MINLP approach to symbolic regression. Math. Program. 170:97-119 (2018). https://doi.org/10.1007/s10107-018-1289-x
6. Talbi E. Metaheuristics. Wiley (2009). https://doi.org/10.1002/9780470496916
7. Aboagye E, Longo J, Conway M, Pazik J, Barkow M, McMahon M, Weil B, Appiah HD, Hesketh R, Yenkie KM. Leveraging machine learning algorithms to predict life cycle inventory assessments (LCIA) to facilitate sustainable process design. Computers & Chemical Engineering 201:109217 (2025). https://doi.org/10.1016/j.compchemeng.2025.109217
8. Hou P, Jolliet O, Zhu J, Xu M. Estimate ecotoxicity characterization factors for chemicals in life cycle assessment using machine learning models. Environment International 135:105393 (2020). https://doi.org/10.1016/j.envint.2019.105393
9. Breiman L. Random forests. Machine Learning 45:5-32 (2001). https://doi.org/10.1023/a:1010933404324
10. Bishop CM. Neural networks for pattern recognition. Oxford University PressOxford (1995). https://doi.org/10.1093/oso/9780198538493.001.0001
11. Ferreira J, Pedemonte M, Torres AI. Development of a machine learning-based soft sensor for an oil refinery's distillation column. Computers & Chemical Engineering 161:107756 (2022). https://doi.org/10.1016/j.compchemeng.2022.107756

12. Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., Duchesnay, É. Scikit-learn: machine learning in Python. *J. Mach. Learn Res.* 12:2825-2830 (2011).
13. Bergstra J, Yamins D, Cox D. Making a science of model search: hyperparameter optimization in hundreds of dimensions for vision architectures. *Proc. Int. Conf. Mach. Learn. PMLR* 28:115-123 (2013). https://doi.org/10.48550/arXiv.1209.5111
14. Lundberg SM, Lee SI. A unified approach to interpreting model predictions. *Adv. Neural Inf. Process. Sys. 30* (2017). https://doi.org/10.48550/arXiv.1705.07874