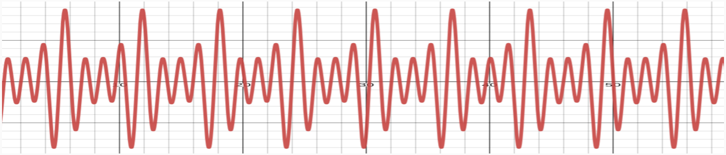
It’s time to get cereal – a whole grain of truth about GP for SR



Prof. Fabrício Olivetti de França

Federal University of ABC

11 July, 2024

Prologue

Please allow me to introduce myself

Hi, my name is Fabricio!

I’m a professor of CS at UFABC in metropolitan region of São Paulo, Brazil.

Please allow me to introduce myself

- Working since 2004 with evolutionary and natural computing
    - combinatorial problems
    - numerical optimization
    - dynamic and uncertain environments
    - clustering
    - hydroelectric power and reservoir optimization
    - multilabel classification
    - recommender systems
    - social network analysis
    - intelligent agents
    - natural language processing
    - symbolic regression (since 2018)
    - program synthesis (since 2022)

Please allow me to introduce myself

My first experience with Genetic Programming for Symbolic Regression was trying to find an interpretable recommender system:

```
6.379515826309025e-3 + -0.00*id(x_1^-4.0 * x_2^3.0 * x_3^1.
) + -0.00*id(x_1^-4.0 * x_2^3.0 * x_3^2.0) + -0.01*id(x_1^-
* x_2^3.0 * x_3^3.0) + -0.02*id(x_1^-4.0 * x_2^3.0 * x_3^4.
+ 0.01*cos(x_1^-3.0 * x_2^-1.0) + 0.01*cos(x_1^-3.0)
+ 0.01*cos(x_1^-3.0 * x_3^1.0) + 0.01*cos(x_1^-3.0 * x_2^1.
+ 0.01*cos(x_1^-2.0 * x_2^-2.0) + -0.06*log(x_1^-2.0 * x_2
+ 0.01*cos(x_1^-2.0 * x_2^-1.0) + 0.01*cos(x_1^-2.0 * x_2^-
+ 0.01*cos(x_1^-2.0) + 0.01*cos(x_1^-2.0 * x_3^1.0) + 0.01*
* x_3^2.0) + 0.01*cos(x_1^-2.0 * x_2^1.0) + 0.01*cos(x_1^-
* x_3^1.0) + -0.00*id(x_1^-2.0 * x_2^2.0) …
```

My fault!

- I took the path of least resistance: `pip install gplearn`
- Used default hyperparameters
- Interpretability denied!

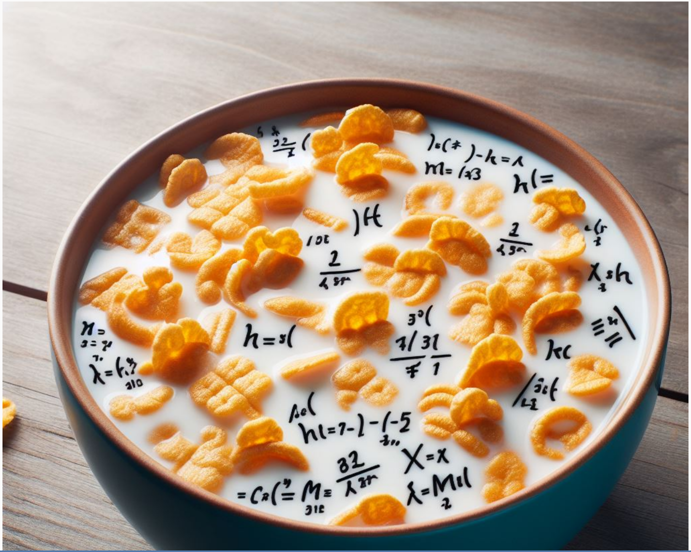
Don’t take offense at my innuendo

- This talk will go through some of my personal criticisms and complaints.
- I want to draw the attention to some things we have to do to be acknowledged outside our community
- I’ll share my view as someone with little experience in the field

In short…

It’s t-t-t-t-t-t-time….

It’s time to get cereal!



Agenda

- Interpretability
- Usability
- Additional Features
- Awareness

The holy grail of interpretable models

If it’s broken, let’s fix it

- That experience prompted me to propose an *improvement* to SR.
- Constrained representation: observing many equations from physics follows a very simple pattern.
- Interaction-Transformation Evolutionary Algorithm: weighted sum of nonlinear transformed features.
- Transformation-Interaction-Rational: extended to rational of polynomials.
- Other interesting representation: continued fraction, functional analysis, etc.

Okayish

- Competitive results
- Smaller expressions, easier to interpret
- Reviewer # 2: “Hold your horses! Can you really interpret that?!”

$$tan(1.52 - 0.20 \cdot tanh(\frac{1}{x_1 \cdot x_0}))$$

Can I?

What is interpretability anyway?

- A simple search for “symbolic regression interpretable” in Scholar retrieves 44k results.
- Looking at the top-5 by citation/year[1]:
    1. Size ~ Interpretability
    2. Found a linear model
    3. Size by default, but customizable (more on this later!)
    4. Size
    5. Description length

[1] https://colab.research.google.com/github/WittmannF/sort-google-scholar/blob/master/examples/run_sortgs_on_colab.ipynb

What is interpretability anyway?

Some sentences about interpretability:

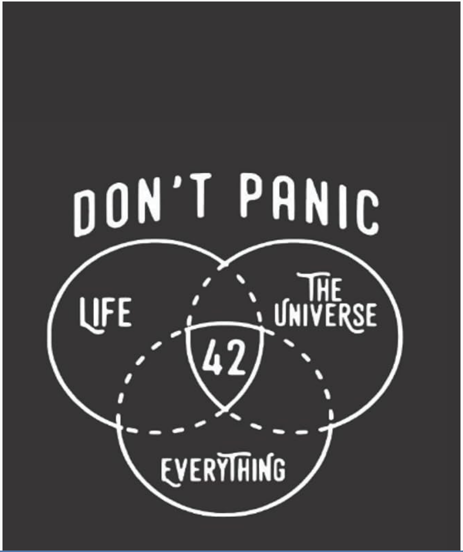
- “by virtue of its simplicity, may be easy to interpret”
- “typically measured as sparsity or model size, is but a proxy for model interpretability; a simple model may still be un-interpretable, or simply wrong”
- “…resulting expression can be readily interpreted…”
- “…our method produces closed-form mathematical formulas that have excellent interpretability…”
- “We argue that the best definition for a “simple” expression is the one that is most interpretable by the user.”

How we do it…

- Frequency analysis of selected features and nonterminals.
- Feature importance (LIME, SHAP, etc.)
- Partial dependence plot, partial effect (average or at the mean)

The meaning of life, the universe and everything else

We wish for a single number that explains everything!



A different view

In[2] we argue that interpretability comes from having a model that:

- Has few free parameters.
- Fits different data from the same phenomena.
- The parameters have meaning.

[2] Etienne Russeil, Fabrício Olivetti de França, Konstantin Malanchev, Bogdan Burlacu, Emille E. O. Ishida, Marion Leroux, Clément Michelin, Guillaume Moinard, and Emmanuel Gangler. 2024. Multiview Symbolic Regression. In Genetic and Evolutionary Computation Conference (GECCO ’24), July 14– 18, 2024, Melbourne, VIC, Australia. ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3638529.3654087

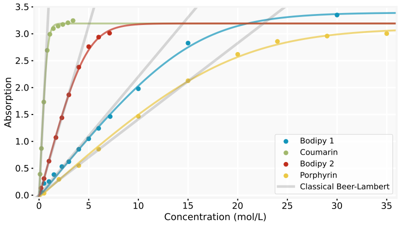
Quick example

- Absorption of light (log of transmittance) ~ concentration of dissolved substance
- Assumed to be linear up to a certain point and then it becomes a plateau
- We used 4 substances with different ϵ and fixed optical path (1 cm), we measured the relationship of the absorption with the concentration.

Beer’s law

We have found the following relation:

$$f(x;\mu,\epsilon) = \log\left(\frac{1}{\mu + exp(-\epsilon x)}\right)$$



- When $\mu = 0$ it reverts to the current Beer’s law, where ϵ turns out to be the value of extinction coefficient.
- We can rewritte the algebraic expression and see that $\mu = e^{py} - e^{-\epsilon px}$, where (px, py) is the point that the absorption starts to plateau.

Takeaways

- To interpret a model, we need context!
- The symbolic model is the start of the investigation process, not the end.
- We need human in the loop[3].

[3] Giorgia Nadizar, Luigi Rovito, Andrea De Lorenzo, Eric Medvet, and Marco Virgolin. 2024. An Analysis of the Ingredients for Learning Interpretable Symbolic Regression Models with Human-in-the-loop and Genetic Programming. ACM Trans. Evol. Learn. Optim. 4, 1, Article 5 (March 2024), 30 pages. https://doi.org/10.1145/3643688

Consumer relations

Can I has implementation?

When I started, the SR ecosystem was fragmented:

- C implementations that didn't allow to customize without changing the code.
- Java that required a certain long gone JRE version.
- MATLAB ($$).
- C++ with broken dependencies (it works! if you are using Debian 3 on an Intel 486 machine…)
- DEAP (but you had to implement the main operators).
- gplearn.
- This one is simple, just type `build` (oh…it's in Haskell…).

SRBench to the rescue!

SRBench[4] [5] is a milestone in GP benchmarking for SR as it established a standard for comparison of old and new algorithms. Apart from the large selection of datasets, it brings:

- A standard API in Python based on scikit-learn, but the participants are free to use any language as the backend.
- In a short time it is already well adopted in many comparisons (1st in rank by cite/year when searching for “symbolic regression”).

[4] La Cava, William, et al. “Contemporary Symbolic Regression Methods and their Relative Performance.” Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track. 2021.

[5] https://cavalab.org/srbench/

SRBench to the rescue!

This improved quality of life by a lot, but we still got some issues:

- Managing a single `conda` environment creates lots of conflicts.
- No clear instructions on how to test on different datasets.

Intermission

We are currently organizing a new edition of SRBench with a better selection of datasets, different tracks, and better analysis of the results.

We need your help! If you want to get involved or want to share some thoughts:

folivetti@ufabc.edu.br

Open an issue / discussion at https://github.com/cavalab/srbench/

Ecosystem ecosystem ecosystem

We need more!

- Ecosystem of support libraries that allows us to build new algrithms with easy:
  - Tree data structure
  - Evaluation, autodiff
  - Parsing
  - Fitness calculation
  - Statistics about the model
  - Simplification
- We need it in different programming languages.
- As a new idea comes up, we need to be able to implement it and validate it fast!

Supporting tools - SRTree-Opt

Support library and CLI for handling SR expressions:

- Parse different expression formats
- Refit parameters supporting different distributions (Gaussian, Poisson, and Bernoulli)
- Simplify expressions with equality saturation
- Calculate confidence interval of parameters and predictions
- Display stats about the model
- It is fast!
- It’s in Haskell!

https://github.com/folivetti/srtree-opt

Reliance on defaults

We’ve built some beliefs throughout the history of GP that were tested on certain settings:

- Rule-of-the-thumb for parameters.
- Importance of certain operators.
- Theoretical aspects such as bloat, neutrality, etc.

We need to challenge these findings and keep checking if anything changed with the new approaches.

Takeaways

- We need to put more effort into building a good end user experience, from installation to the use.
- Having an optimized ecosystem can make things easier for everybody, this is a joint effort!
- We don’t need to stick to a programming language.

What else can SR do?

SR does whatever an SR can!

SRBench showed that current state-of-the-art in GP-SR can compete with opaque methods wrt accuracy.

If they are the same, what else can SR do that those methods cannot?

Confidence intervals and statistical tools

SR is nonlinear regression! What we do with nonlinear regression is mostly valid for SR[6]:

- Profile likelihood[7]: allows us to calculate confidence intervals and identifiability of the parameters.
- Graphical summaries.
- Curvature measures.
- Statistical tests.

This is related to interpretability!

[6] Bates, D. M. “Nonlinear Regression Analysis and Its Applications.” John Wiley and Sons: New York google schola 2 (1988): 379-416.

[7] de Franca, Fabricio Olivetti, and Gabriel Kronberger. “Prediction Intervals and Confidence Regions for Symbolic Regression Models based on Likelihood Profiles.” arXiv preprint arXiv:2209.06454 (2022).

Prior knowledge

With GP-SR we can bias the search towards any side-objective. We can enjoy additional degree-of-freedom to integrate prior knowledge[8] [9], limiting the search space or penalizing non-conformant solutions. We can request for models that:

- Have a bounded co-domain (for a bounded domain)
- Is monotonic
- Is Symmetric
- many more…

[8] Kronberger, Gabriel, et al. “Shape-constrained symbolic regression—improving extrapolation with prior knowledge.” Evolutionary computation 30.1 (2022): 75-98.

[9] Kubalík, Jiří, Erik Derner, and Robert Babuška. “Symbolic regression driven by training data and prior knowledge.” Proceedings of the 2020 Genetic and Evolutionary Computation Conference. 2020.

Model simplification

We don’t need to commit with the models that are given. It’s possible to perform algebraic manipulation to accomodate for our needs:

- Reduce overparametrization.
- Simplify the expression.
- Rewrite a parameter to express a different meaning.
- Remove subexpressions that doesn’t contribute much to accuracy.

Model simplification

An interesting approach is Equality Saturation[10] [11], it allows to rewrite the expression into a more convenient form as the context ask for.

We can also infer general and local equivalence rules from our data [12].

[10] Willsey, Max, et al. "Egg: Fast and extensible equality saturation." Proceedings of the ACM on Programming Languages 5.POPL (2021): 1-29.

[11] de Franca, Fabricio Olivetti, and Gabriel Kronberger. "Reducing Overparameterization of Symbolic Regression Models with Equality Saturation." Proceedings of the Genetic and Evolutionary Computation Conference. 2023.

[12] Aldeia, Guilherme Seidyo Imai, Fabricio Olivetti de Franca, and William G. La Cava. "Inexact Simplification of Symbolic Regression Expressions with Locality-sensitive Hashing." arXiv preprint arXiv:2404.05898 (2024).

Closing Remarks! Let us preach!

Grad course experience

From February till May of 2024, I teached a 12 weeks graduate course on Symbolic Regression. It covered the essential of nonlinear regression analysis, statistics, symbolic regression and genetic programming, and related topics.

The audience was mixed: from CS students to humanities, so a good support of easy-to-use tools was important.

Students impression

In summary, the students were very interested about the possibilities of creating a nonlinear regression model. By the end of the course they got a draft paper for interesting applications:

- Non-intrusive blood pressure measurement
- Understanding what influences students grade on an online course
- Understanding what attracts tourists

Students impression

They did have some difficulties, tho:

- No MS-Windows support!! (MS-Windows users installed HeuristicLabs)
- Not always easy to install even on Linux
- Too many hyper-parameters to finetune, none of them worked well with defaults
- Lack of documentation
- Lack of customization

Slides

The slides for that course are freely available and every collaboration is welcome! The SR algorithms part is lacking! Write some lecture notes about your own algorithm!

https://github.com/folivetti/symreg-course

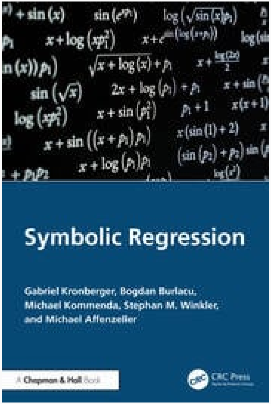
COIL

Universidade Federal do ABC participates of the Collaborative Online International Learning (COIL). We can apply to create an online course about SR!

https://collab-edu.com/hub/coil

Book

There’s a new book on Symbolic Regression hitting the shelves soon. This book offers a more practical approach aimed at data scientists.



Symbolic Regression

Gabriel Kronberger, Bogdan Burlacu, Michael Kommenda, Stephan M. Winkler, and Michael Affenzeller

A Chapman & Hall Book

CRC Press

https://www.routledge.com/Symbolic-Regression/Kronberger-Burlacu-Kommenda-Winkler-Affenzeller/p/book/9781138054813

We need to grow our community

It’s time to make people aware of SR:

- Collaborate in real-world application
- Grow our ecosystem of supporting libraries
- Offer good end user experience
- Incorporate tools beyond prediction
- Write more books
- Teach and advertise
- Stimulate your students to create awesome repositories and tutorials

Thank you!

You can download these slides in PDF format from:

https://folivetti.github.io/files/talk.pdf