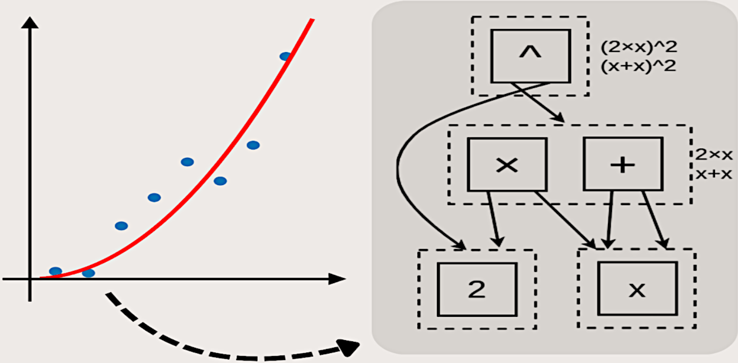
Symbolic Regression – Discovering Equations from Observational Data



Fabrício Olivetti de França

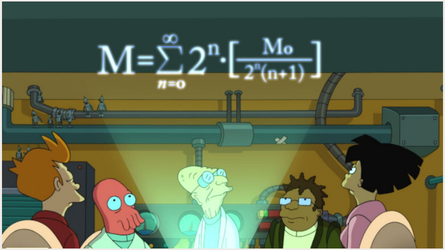
Federal University of ABC

Symbolic Regression

Let’s do science!

The core of our modern scientific knowledge is based on careful production and analysis of experimental data.

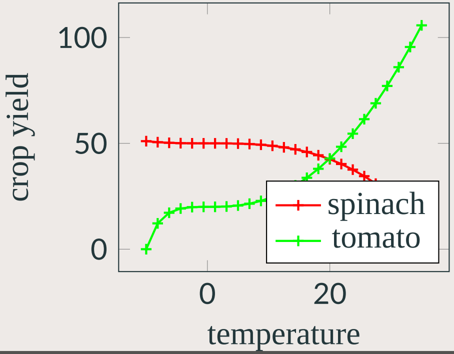
- Specify a hypothesis.
- Collect data through experiments.
- Describe the measurements with a mathematical function.
- Replicate.



Parametric Functions

- Parametric functions on the form $f(x;\theta)$ compacts the data.
- Parameters can be used to adjust such function to the observations.
- Often determined by prior knowledge or well established models.
- Parameters summarize the data and highlight differences.

$$f(\mathbf{x},) = \frac{\theta_1 \, x^3}{\theta_2 \, + x} + \theta_3$$



Generic Models

Generic models are common patterns that can be used to describe a wide range of phenomena...or an overparameterized model that can fit any data.

Examples

- Linear models.
- Quadratic models.
- Exponential decays.
- Neural Networks.
- Random Forests.

Generic Models (examples)

Linear Models

They capture the tendency of the data: ”as x increases, y tends to increase too by a certain amount”.

Neural Networks

Flexible function that can *mold* to the data, but can also be too flexible and capture noise. It creates a smooth interpolation of the data.

Generic Models (examples)

Linear Models

They capture the tendency of the data: ”as x increases, y tends to increase too by a certain amount”.

We want something in between!

Neural Networks

Flexible function that can *mold* to the data, but can also be too flexible and capture noise. It creates a smooth interpolation of the data.

Equation Discovery

Automating the process

- Relying on pre-determined parametric functions can capture just part of the behavior (underfit).
- In other situations, it may capture the noise in the data collection (overfit) or be too obscure.
- Ideally we should have a function that is capable of fitting that particular data and only that data (not entirely true).
- **Equation Discovery** is the task of automatically finding such function.

aka Symbolic Regression

This is also known as Symbolic Regression, and can be formalized as the minimization of a loss function:

$$\min_{\theta, f(x;\theta)} \mathcal{L}(f(x;\theta), y)$$

Genetic Programming

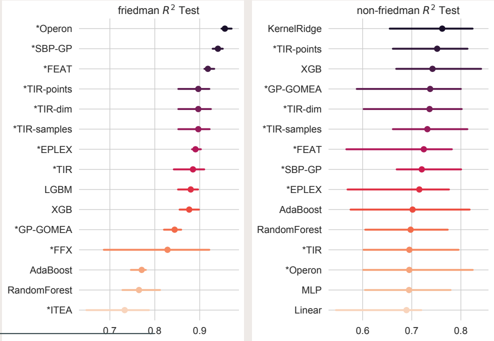
Genetic Programming (GP) searches for omputer programs to solve problems, including symbolic regression.

GP

```
gp nPop = do
  p <- initialPopulation(nPop)
  until convergence repeat
      parents   <- select-from(p)
      children  <- recombine(parents)
      children' <- perturb(children)
      p         <- reproduce(p, children')
  return best(p)
```

Against opaque models[1]

Symbolic Regression is capable of achieving similar accuracy as other ML models, while being more compact.

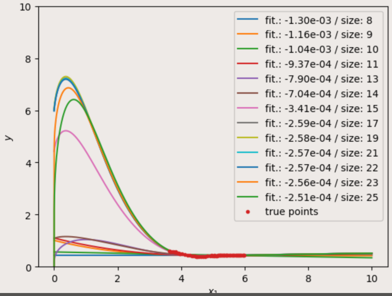


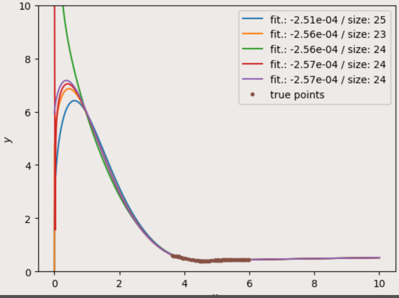
[1]La Cava, William, et al. "Contemporary Symbolic Regression Methods and their Relative Performance." Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 1). 2021.

What do we really want?

Not just accuracy!

- Accuracy-size tradeoff: simplest model with a good accuracy.
- The limiting behavior and smoothness of the function is also important.
- Many desiderata that are not captured by a single loss function, but easily incorporated in SR.





Additional Constraints

I got the knowledge!

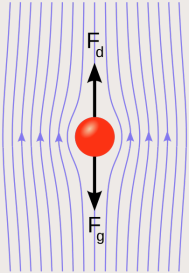
We can incorporate prior knowledge to the search, such as:

- Shape constraints (e.g., $f'(x) \geq 0$).
- Limiting behavior (e.g., $f(x) \to 0$ as $x \to \infty$).
- Units (e.g., $f(x)$ must be in meters).
- Physical constraints (e.g., include conservation laws).
- Incorporating multiple views of the data.

Success Cases

Particle-Laden Flows[2]

- GP alone is capable of identifying models for two-particle systems.
- Coupling GP and Graph Neural Networks, it is possible to extend to *n* particles.
- Equal or better than human-created solutions for the Stoken flows.
- $\frac{1}{r}\sum Ar + B\sin(\theta) + C$.



[2] Reuter, Julia, et al. “Graph networks as inductive bias for genetic programming: Symbolic models for particle-laden flows.” European Conference on Genetic Programming (Part of EvoStar). Cham: Springer Nature Switzerland, 2023.

Twitter[3]

- GP was capable of creating a logistic model that models the dynamics of conflict on Twitter.
- Better accuracy than Random Forests and Decision Trees.
- They observed a compound effect of number of tweets around a certain topic during conflicts.
- A more homogeneous distribution of replies per tweets is associated with non-conflicting topics.
- The skewness of the distribution of interactions act as a phase transition.

[3] De França, Fabricio Olivetti, et al. "Understanding conflict origin and dynamics on Twitter: A real-time detection system." Expert Systems with Applications 212 (2023): 118748.

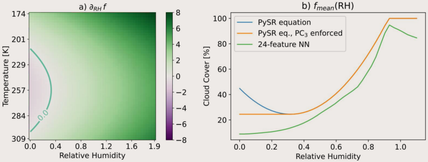
Material Science[4]

- Hybrid models for stress-strain curves in aluminum alloys.
- SR is used to predict the calibration parameters of a known physical model.
- Insightful analysis on the effect of temperature and force on the material, while keeping the expression simple.

[4]Kabliman, Evgeniya, et al. "Application of symbolic regression for constitutive modeling of plastic deformation." Applications in Engineering Science 6 (2021): 100052.

Weather[5]

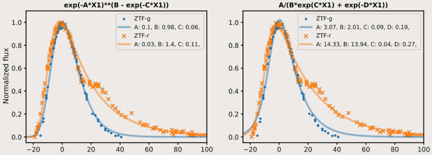
- Parametric models of cloud cover.
- Pareto front of models: linear models, traditional, SR, and NN.
- SR produced a good balance between model complexity and accuracy that could be further improved by manual inspection.
- Physically plausible and understand the relationship of the variables.



[5] Grundner, Arthur, et al. "Data-driven equation discovery of a cloud cover parameterization." Journal of Advances in Modeling Earth Systems 16.3 (2024): e2023MS003763.

Supernovae[6]

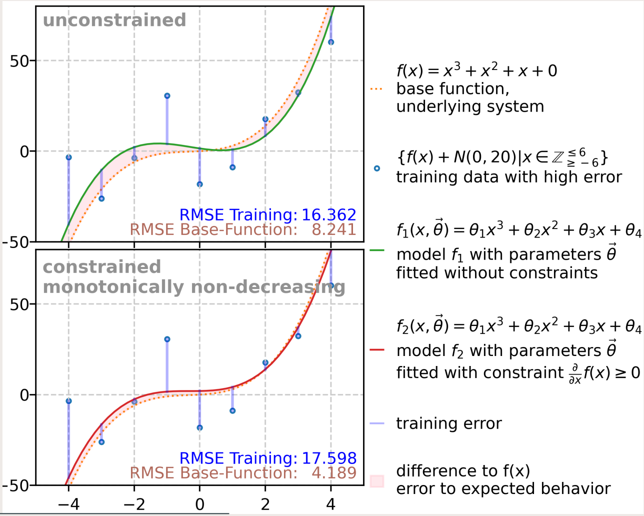
- Modeling the peak luminosity of type Ia supernovae.
- Different data with different photometrics (red and green filters).
- Multi-view symbolic regression: find a single model that fits every dataset independently.



[6]Russeil, Etienne, et al. "Multiview symbolic regression." Proceedings of the Genetic and Evolutionary Computation Conference. 2024.

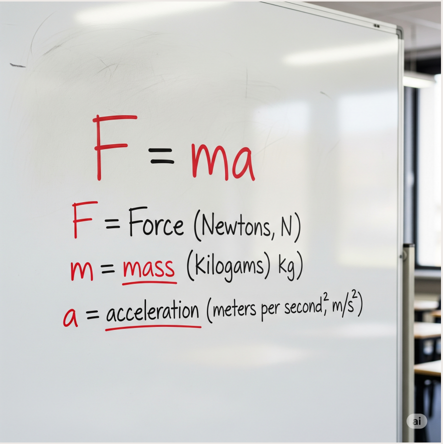
Extensions

Shape-constraints[7]



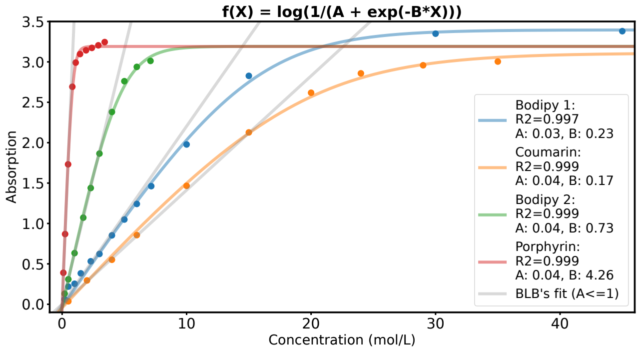
[7] Kronberger, Gabriel, et al. "Shape-constrained symbolic regression—improving extrapolation with prior knowledge." Evolutionary computation 30.1 (2022): 75-98.

Unit-aware [8]



[8]Reuter, Julia, et al. "Unit-Aware Genetic Programming for the Development of Empirical Equations." International Conference on Parallel Problem Solving from Nature. Cham: Springer Nature Switzerland, 2024.

Multi-view[9]



[9] Russeil, Etienne, et al. “Multiview symbolic regression.” Proceedings of the Genetic and Evolutionary Computation Conference. 2024.

Recommendations of softwares

PyOperon[10]

High-performance C++ library with Python bindings

- Competitive runtime, good accuracy
- Supports multi-objective optimization, many hyper-parameters to adjust to your liking.
- May overparameterize the model

[10] https://github.com/heal-research/pyoperon/

PySR [11]

Customizable SR

- Good balance between runtime performance and accuracy.
- Extremely flexible with lots of customization options.
- Not the optimal accuracy, need to perform post selectno process.

[11] https://github.com/MilesCranmer/PySR

SINDy[12]

Nonlinear dynamical systems

- Sparse Identification of Nonlinear Dynamical systems.
- Fast and accurate for ODE systems.

[12] https://pysindy.readthedocs.io/en/latest/examples/index.html

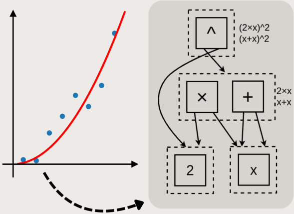
eggp [13]

Efficiency of search

- Improve efficiency of search using e-graphs.
- Returns a good balance of accurate, simple, and with small number of parameters.
- Perform equally or better than Operon and PySR.
- Integration with exploration tool (rEGGression).
- The creator is in front of you!

[13] https://github.com/folivetti/eggp

Conclusion



symbolic regression + e-graphs = ❤️

🥚

- Symbolic Regression is a powerful tool for discovering equations from data.
- Help move science forward by automating the process of equation discovery.
- It still needs a post-search finetuning of the obtained model.
- A long way to go, but we are getting there! We need your help!

Questions

Python library and CLI

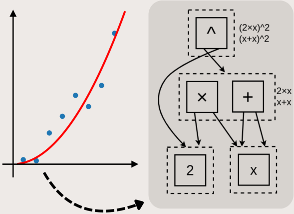
- pip install eggp
- pip install reggression
- pip install symregg

Open-source:

- **https://github.com/folivetti/eggp**
- **https://github.com/folivetti/reggression**
- **https://github.com/folivetti/symregg**

rEGGression

rEGGression



symbolic regression + e-graphs = ❤

🥚

- e-graph brings the essence of relational databases into symbolic regression.
- rEGGression can help us navigate the set of visited expressions during a search .
- many new features on the way.

Pattern Matching

```
(egraph.top(3,
    filters=["size <= 7"],
    pattern="v0 + x0")
 .style.format(fmt))
```

	Id	Latex	Fitness
0	7202	$\left(\theta_0 \cdot \lvert(x_0 + x_0)\rvert^{\theta_1}\right)$	-0.001309
1	12425	$\lvert(\theta_0 \cdot (x_0 + x_0))\rvert^{\theta_1}$	-0.001309
2	198550	$\left\lvert\frac{x_1}{(x_0 + x_0)}\right\rvert^{\theta_0}$	-0.003112

- Expressions **not** matching a certain pattern.
- Or that matches a pattern at the root.

Breaking up the expression

Subtrees, optimize the unevaluated and insert new.

```
egraph.subtrees(100)
```

Expression	Fitness
x_0	-20.23
θ_0	-14.12
$\theta_0 x_0$	-6.53
$x_0^{\theta_0 x_0}$	NaN
$x_0^{\theta_0 x_0} + \theta_1$	NaN
$\theta_1 x_1$	NaN
$x_0^{\theta_0 x_0} + \theta_1 x_1$	$-1.32 \cdot 10^{-3}$

```
egraph.optimize(93)
```

Expression	Fitness
$\theta_1 x_1$	-5.43

```
egraph.insert("x0 ^
      (t0 + x1)")
```

Expression	Fitness
$x_0^{\theta_0 + x_1}$	-0.43

Modularity

```
egfinal.modularity(2, filters=["> 3"])
```

$$\left(\left(|z_0|^{\theta_0} + |z_0|^{\theta_1}\right) \cdot \theta_2\right)$$
$$z_0 = (log_{Re} - r_k)$$

$$|z_0|^{\left(\theta_0 \cdot \left|\frac{1}{z_0}\right|^{\theta_1}\right)}$$
$$z_0 = \frac{log_{Re}}{r_k}$$

$$((f_0(\theta_{0...2}) \cdot (f_0(\theta_{3...5}) \cdot r_k)) + \theta_6)$$
$$f_0(\theta) = \left(\left(\frac{((\theta_0 + r_k) + r_k)}{r_k} \cdot \theta_1\right) + \theta_2\right)$$