Symbolic Regression: Algorithms, Applications, and Theory

Philipp Scholl
(Ludwig-Maximilians-Universität München)

MIAPbP
August 2025

Outline

1. Introduction Symbolic Regression
2. ParFam
3. Application in Robotics
4. Identifiability

Overview Symbolic Regression (SR)

Goal of SR:

- Given data $(x_i, y_i)_i$ with $y_i \in \mathbb{R}$ find a function f such that $f(x_i) \approx y_i$.

Overview Symbolic Regression (SR)

Goal of SR:

- Given data $(x_i, y_i)_i$ with $y_i \in \mathbb{R}$ find an interpretable function f such that $f(x_i) \approx y_i$.

Overview Symbolic Regression (SR)

Goal of SR:

- Given data $(x_i, y_i)_i$ with $y_i \in \mathbb{R}$ find an interpretable function f such that $f(x_i) \approx y_i$ with as little assumptions as possible.

Overview Symbolic Regression (SR)

Goal of SR:

- Given data $(x_i, y_i)_i$ with $y_i \in \mathbb{R}$ find an interpretable function f such that $f(x_i) \approx y_i$ with as little assumptions as possible.

Current approaches:

Overview Symbolic Regression (SR)

Goal of SR:

- Given data $(x_i, y_i)_i$ with $y_i \in \mathbb{R}$ find an interpretable function f such that $f(x_i) \approx y_i$ with as little assumptions as possible.

Current approaches:

- Genetic programming (GP)

LMU LUDWIG-MAXIMILIANS-UNIVERSITÄT MÜNCHEN

Overview Symbolic Regression (SR)

Goal of SR:

- Given data $(x_i, y_i)_i$ with $y_i \in \mathbb{R}$ find an interpretable function f such that $f(x_i) \approx y_i$ with as little assumptions as possible.

Current approaches:

- Genetic programming (GP)
- Sparse regression

Overview Symbolic Regression (SR)

Goal of SR:

- Given data $(x_i, y_i)_i$ with $y_i \in \mathbb{R}$ find an interpretable function f such that $f(x_i) \approx y_i$ with as little assumptions as possible.

Current approaches:

- Genetic programming (GP)
- Sparse regression
- Recursive problem simplifications using known physical symmetries

Overview Symbolic Regression (SR)

Goal of SR:

- Given data $(x_i, y_i)_i$ with $y_i \in \mathbb{R}$ find an interpretable function f such that $f(x_i) \approx y_i$ with as little assumptions as possible.

Current approaches:

- Genetic programming (GP)
- Sparse regression
- Recursive problem simplifications using known physical symmetries
- Continuous optimization

Overview Symbolic Regression (SR)

Goal of SR:

- Given data $(x_i, y_i)_i$ with $y_i \in \mathbb{R}$ find an interpretable function f such that $f(x_i) \approx y_i$ with as little assumptions as possible.

Current approaches:

- Genetic programming (GP)
- Sparse regression
- Recursive problem simplifications using known physical symmetries
- Continuous optimization
- Meta learning

Overview Symbolic Regression (SR)

Goal of SR:

- Given data $(x_i, y_i)_i$ with $y_i \in \mathbb{R}$ find an interpretable function f such that $f(x_i) \approx y_i$ with as little assumptions as possible.

Current approaches:

- Genetic programming (GP)
- Sparse regression
- Recursive problem simplifications using known physical symmetries
- Continuous optimization
- Meta learning
- Pre-training using deep learning

Overview Symbolic Regression (SR)

Goal of SR:

- Given data $(x_i, y_i)_i$ with $y_i \in \mathbb{R}$ find an interpretable function f such that $f(x_i) \approx y_i$ with as little assumptions as possible.

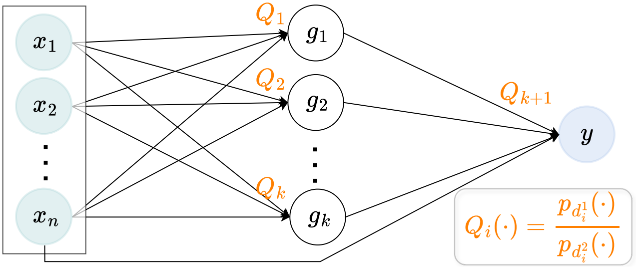
Current approaches:

- Genetic programming (GP)
- Sparse regression
- Recursive problem simplifications using known physical symmetries
- Continuous optimization
- Meta learning
- Pre-training using deep learning
- LLMs + GP

Algorithm: ParFam

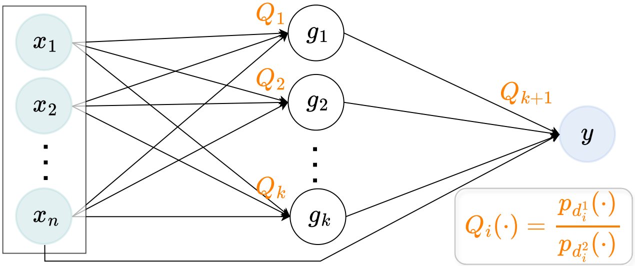
Scholl, Bieker, Hauger, and Kutyniok. "ParFam – (Neural-guided) Symbolic Regression Based on Continuous Global Optimization". ICLR. 2025.

ParFam Architecture



- $g_1, \ldots, g_k$ are physically interesting functions, e.g., sin, exp, $\sqrt{}$
- $Q_1, \ldots, Q_{k+1}$ are rational functions

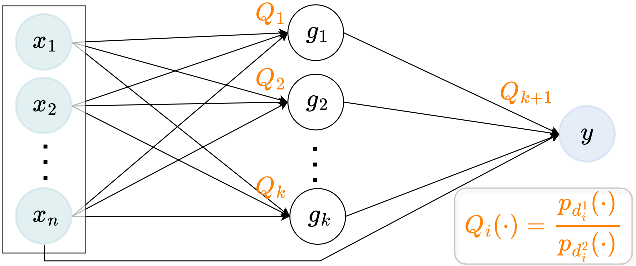
ParFam Architecture



- $g_1, \ldots, g_k$ are physically interesting functions, e.g., sin, exp, $\sqrt{}$
- $Q_1, \ldots, Q_{k+1}$ are rational functions

$\rightsquigarrow$ Translates the SR problem into a continuous optimization problem

ParFam Architecture

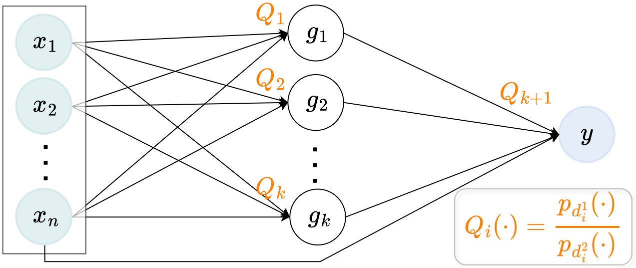


- $g_1, \ldots, g_k$ are physically interesting functions, e.g., sin, exp, $\sqrt{}$
- $Q_1, \ldots, Q_{k+1}$ are rational functions

$\leadsto$ Translates the SR problem into a continuous optimization problem

$\leadsto$ Avoids nesting functions such as $\sin(\cos(x))$:

ParFam Architecture



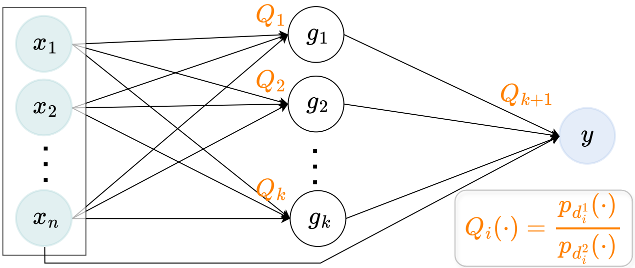
- $g_1, \ldots, g_k$ are physically interesting functions, e.g., $\sin$, $\exp$, $\sqrt{}$
- $Q_1, \ldots, Q_{k+1}$ are rational functions

$\rightsquigarrow$ Translates the SR problem into a continuous optimization problem

$\rightsquigarrow$ Avoids nesting functions such as $\sin(\cos(x))$:

- Hard-to-interpret and rarely encountered formulas

ParFam Architecture



- $g_1, \ldots, g_k$ are physically interesting functions, e.g., sin, exp, $\sqrt{}$
- $Q_1, \ldots, Q_{k+1}$ are rational functions

$\rightsquigarrow$ Translates the SR problem into a continuous optimization problem

$\rightsquigarrow$ Avoids nesting functions such as $\sin(\cos(x))$:

- Hard-to-interpret and rarely encountered formulas
- Better to optimize, due to easier loss landscape, less issues exploding/vanishing gradient

ParFam: Optimization

Loss function:

$$L(\theta) = \frac{1}{N} \sum_{i=1}^{N} \left(y_i - f_\theta(x_i)\right)^2 + \lambda ||\theta||_1,$$

ParFam: Optimization

Loss function:

$$L(\theta) = \frac{1}{N}\sum_{i=1}^{N}\left(y_i - f_\theta(x_i)\right)^2 + \lambda||\theta||_1,$$

Optimizer:

1. Global optimization with Basin-hopping

ParFam: Optimization

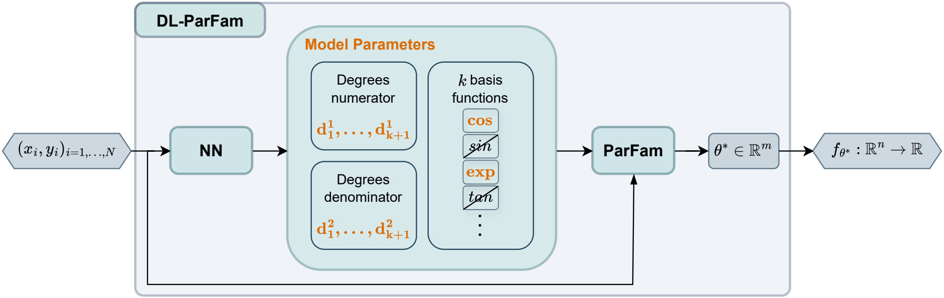
Loss function:

$$L(\theta) = \frac{1}{N}\sum_{i=1}^{N}\left(y_i - f_\theta(x_i)\right)^2 + \lambda||\theta||_1,$$

Optimizer:

1. Global optimization with Basin-hopping
2. Additional fine-tuning routine with BFGS to sparsify the weights by setting small coefficients to 0.

DL-ParFam



Training details:

- Architecture: SET-Transformer as decoder and FCNN as encoder
- Dataset size: 4M functions, generation time was 1.3h on 1 RTX3090
- Training time: 31h on 3 RTX3090

SRBench: Feynman dataset

119 physical formulas from the Feynman lectures

Name	Formula
Feynman-III-4-33	$y = \frac{h\omega}{2\pi\left(\exp\left(\frac{h\omega}{2\pi Tkb}\right)-1\right)}$
Feynman-III-8-54	$y = \sin^2\left(\frac{2\pi E_n t}{h}\right)$
Feynman-II-24-17	$y = \sqrt{-\frac{\pi^2}{d^2}+\frac{\omega^2}{c^2}}$
Feynman-I-18-4	$y = \frac{m_1 r_1 + m_2 r_2}{m_1 + m_2}$
Feynman-I-38-12	$y = \frac{\epsilon h^2}{\pi m q^2}$
Feynman-I-40-1	$y = n_0 \exp\left(-\frac{gmx}{Tkb}\right)$
Feynman-I-8-14	$y = \sqrt{(-x_1+x_2)^2+(-y_1+y_2)^2}$
Feynman-I-9-18	$y = \frac{Gm_1 m_2}{(-x_1+x_2)^2+(-y_1+y_2)^2+(-z_1+z_2)^2}$
Feynman-test-17	$y = \frac{m^2\omega^2 x^2\left(\frac{\alpha x}{y}+1\right)+p^2}{2m}$

SRBench: Strogatz dataset

7 two-dimensional ODEs

Name	Formula
Bacterial Respiration	$\dot{x} = -\frac{xy}{0.5x^2+1} - x + 20$ $\dot{y} = -\frac{xy}{0.5x^2+1} + 10$
Bar Magnets	$\dot{x} = -\sin(x) + 0.5\sin(x-y)$ $\dot{y} = -\sin(y) - 0.5\sin(x-y)$
Glider	$\dot{x} = -0.05x^2 - \sin(y)$ $\dot{y} = x - \frac{\cos(y)}{x}$
Predator Prey	$\dot{x} = x\left(-x - \frac{y}{x+1} + 4\right)$ $\dot{y} = y\left(\frac{x}{x+1} - 0.075y\right)$
Shear Flow	$\dot{x} = \cos(x)\cot(y)$ $\dot{y} = \left(0.1\sin^2(y) + \cos^2(y)\right)\sin(x)$

Results

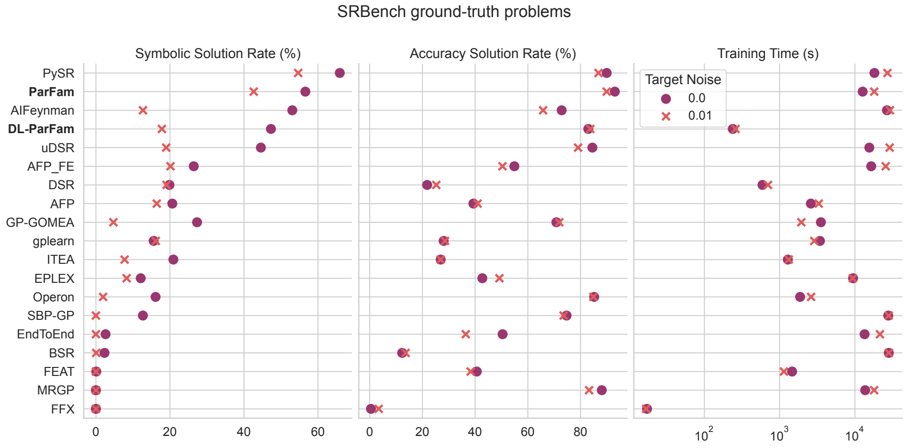


Figure: Mean over both SRBench ground-truth data sets combined.

LMU LUDWIG-MAXIMILIANS-UNIVERSITÄT MÜNCHEN

Application: Modeling the friction in robot joints

Scholl, Dietrich, Wolf, Lee, Schäffer, Kutyniok, and Iskandar. "Interpretable Robotic Friction Learning via Symbolic Regression." arXiv preprint arXiv:2505.13186 (2025).

Background

$$\mathbf{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \mathbf{C}(\boldsymbol{q}, \dot{\boldsymbol{q}})\dot{\mathbf{q}} + \boldsymbol{\tau}_{\mathbf{g}}(\mathbf{q}) = \boldsymbol{\tau}_{\mathbf{m}} + \boldsymbol{\tau}_{\mathbf{f}} + \boldsymbol{\tau}_{\mathbf{ext}}$$

Background

$$\mathbf{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \mathbf{C}(\boldsymbol{q}, \dot{\boldsymbol{q}})\dot{\mathbf{q}} + \boldsymbol{\tau}_{\mathbf{g}}(\mathbf{q}) = \boldsymbol{\tau}_{\mathbf{m}} + \boldsymbol{\tau}_{\mathbf{f}} + \boldsymbol{\tau}_{\mathbf{ext}}$$

Background

$$\mathbf{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \mathbf{C}(\boldsymbol{q}, \dot{\boldsymbol{q}})\dot{\mathbf{q}} + \boldsymbol{\tau}_{\mathbf{g}}(\mathbf{q}) = \boldsymbol{\tau}_{\mathbf{m}} + \boldsymbol{\tau}_{\mathbf{f}} + \boldsymbol{\tau}_{\mathbf{ext}}$$

Good friction estimation is necessary for:

- Force control
- Safe-human robot interaction
- External torque estimation

Background

$$\mathbf{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \mathbf{C}(\boldsymbol{q},\dot{\boldsymbol{q}})\dot{\mathbf{q}} + \boldsymbol{\tau}_{\mathbf{g}}(\mathbf{q}) = \boldsymbol{\tau}_{\mathbf{m}} + \boldsymbol{\tau}_{\mathbf{f}} + \boldsymbol{\tau}_{\mathbf{ext}}$$

Good friction estimation is necessary for:

- Force control
- Safe-human robot interaction
- External torque estimation

Current approaches:

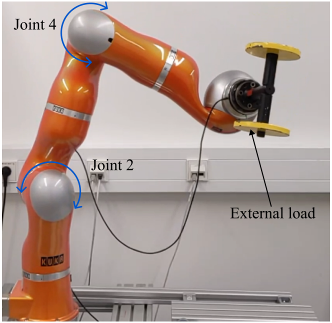
- Model-based: Complex due to the dependence on many factors, hard to adapt to new settings, etc.
- Data-driven: Accurate and flexible, but lack interpretability and, thus, trustworthiness

LuGre Model

$$
\begin{aligned}
\tau_{\mathsf{f,s}}(\dot{q}) &= g(\dot{q}) + s(\dot{q}) \\
g(\dot{q}) &= \operatorname{sgn}(\dot{q})\left(F_{\mathsf{c}} + (F_{\mathsf{s}} - F_{\mathsf{c}})\, e^{-|\dot{q}/v_{\mathsf{s}}|^{\delta_{\mathsf{s}}}}\right) \\
s(\dot{q}) &= F_v \dot{q}
\end{aligned}
$$

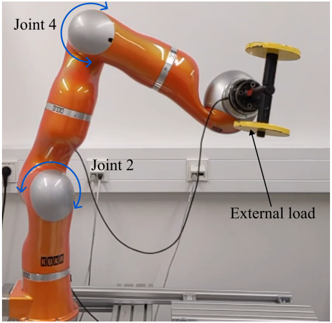
- s represents the velocity strengthening term
- g represents the Stribeck effect (velocity weakening term)
- F_{c}, F_{s}, F_{v}, v_{s}, and δ_{s} are contants that have to be identified for each robot

Data



Joints 2 and 4 of a DLR KUKA LWR IV+ robot:

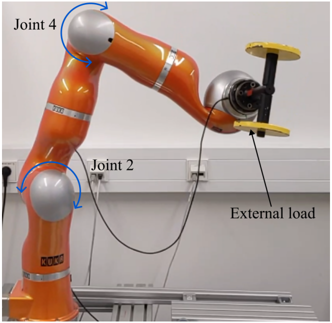
Data



Joints 2 and 4 of a DLR KUKA LWR IV+ robot:

- Measuring position, velocity, gravitational torque, and friction torque
- Dataset A
  - Constant, single-joint velocities

Data

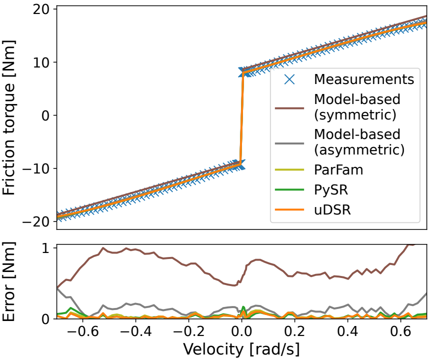


Joints 2 and 4 of a DLR KUKA LWR IV+ robot:

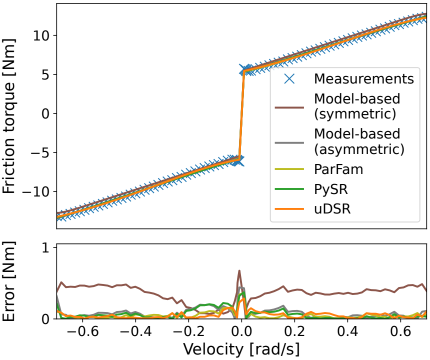
- Measuring position, velocity, gravitational torque, and friction torque
- Dataset A
  - Constant, single-joint velocities
- Dataset B
  - Simultaneous joint movement
  - With and without external load

Dataset A (without gravitational torque dependence)

Averaged over the friction values for constant velocity ⇒ No gravitational torque dependence



(a) Joint 2



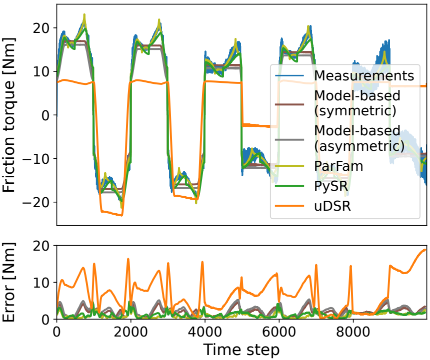
(b) Joint 4

Dataset A (without gravitational torque dependence)

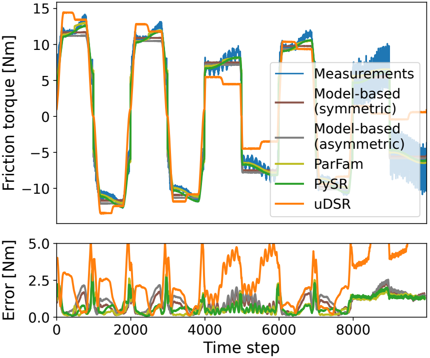
	Estimation error [Nm]		Complexity [-]		$\hat{\tau}_f$
	Joint 2	Joint 4	Joint 2	Joint 4	Joint 2
Model-based (symmetric)	0.737	0.350	20	20	— (symmetric LuGre model)
Model-based (asymmetric)	0.134	0.093	40	40	— (asymmetric LuGre model)
ParFam	0.037	0.058	22	23	$\hat{\tau}_f = \left(3.66\dot{q} + 8.52 sign(\dot{q}) - 0.44\right) \cdot \exp\left(-0.78\dot{q}^2 + 1.26\lvert\dot{q}\rvert + 0.02\dot{q} - 0.02 sign(\dot{q})\right)$
PySR	0.041	0.074	15	11	$\hat{\tau}_f = -4.283\dot{q}^4 sign(\dot{q}) + \left(8.452 - 0.215 sign(\dot{q})\right)\left(1.842\dot{q} + sign(\dot{q}) - 0.049\right)$
uDSR	0.033	0.059	36	43	$\hat{\tau}_f = -8.083\dot{q}^3 + 4.643\dot{q}^2 sign(\dot{q}) - 0.068\dot{q}^2 - 0.639\lvert\dot{q}\rvert + 13.609\dot{q} + 0.827 sign(\dot{q}) + 0.827 sign(\dot{q}) + e^{\dot{q}} + e^{e^{sign(\dot{q})}} - 9.899$

Table: Results of the different approaches on Dataset A without load dependency: Mean absolute error of the friction estimation in Nm and complexity of the computed formulas for Joint 2 and 4, and the computed formula for $\hat{\tau}_f$ for Joint 2.

Dataset A (with gravitational torque dependence)



(a) Joint 2



(b) Joint 4

Dataset A (with gravitational torque dependence)

	Estimation error [Nm]		Complexity [-]	
	Joint 2	Joint 4	Joint 2	Joint 4
Model-based (symmetric)	3.663	0.914	20	20
Model-based (asymmetric)	4.073	1.068	40	40
ParFam	1.025	0.724	63	32
PySR	1.294	0.800	13	8
uDSR	7.168	2.671	12	12

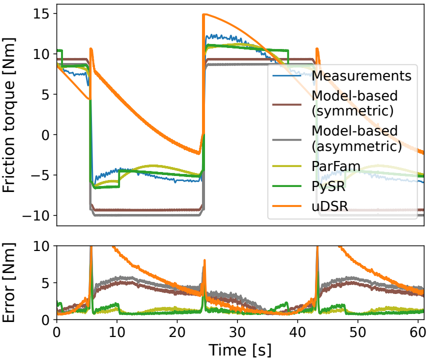
Dataset A (with gravitational torque dependence)

	Estimation error [Nm]		Complexity [-]	
	Joint 2	Joint 4	Joint 2	Joint 4
Model-based (symmetric)	3.663	0.914	20	20
Model-based (asymmetric)	4.073	1.068	40	40
ParFam	1.025	0.724	63	32
PySR	1.294	0.800	13	8
uDSR	7.168	2.671	12	12

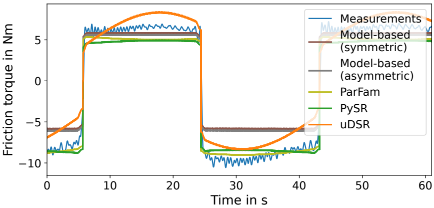
Example formula learned by ParFam:

$$
\begin{aligned}
\hat{\tau}_f = & -1.12\dot{q}^2 + 0.01\dot{q}\tau_g + 1.095|\dot{q}| - 0.01\dot{q}\,\mathrm{sgn}(\tau_g) - 14.87\dot{q} + 0.026|\tau_g| \\
& - 0.089\tau_g - 0.154\,\mathrm{sgn}(\dot{q})\,\mathrm{sgn}(\tau_g) + 7.162\,\mathrm{sgn}(\tau_g) + 0.532 \\
& + \left(-0.663\dot{q} + 0.014\tau_g + 1.64\,\mathrm{sgn}(\dot{q}) - 4.245\,\mathrm{sgn}(\tau_g) - 10.159\,\mathrm{sgn}(\dot{q}) - 0.226\right) \\
& \cdot \sqrt{\left|0.167\dot{q} - 0.076\tau_g - 0.467\,\mathrm{sgn}(\dot{q}) + 2.687\,\mathrm{sgn}(\tau_g)\right|}.
\end{aligned}
$$

Dataset B



(a) Joint 2



(b) Joint 4

Dataset B

	Without load [Nm]		With load [Nm]		Complexity [-] (adaption only)		Formula (adaption only)
	Joint 2	Joint 4	Joint 2	Joint 4	Joint 2	Joint 4	Joint 2
Model-based (symmetric)	3.663	1.473	3.171	1.628	20	20	(no adaption)
Model-based (asymmetric)	4.073	1.318	3.550	1.648	40	40	(no adaption)
ParFam	0.611	0.742	1.007	1.189	20	11	$\hat{\tau}_f = -0.012g^2 + 0.016g\,\mathrm{sgn}(\dot{q}) + 0.176\lvert g\rvert - 0.169g + 0.712\,\mathrm{sgn}(\dot{q}) + 1.036$
PySR	0.911	0.936	0.953	1.428	3	2	$\hat{\tau}_f = \left(\mathrm{sgn}(g) - 0.9893261\right)^2$
uDSR	4.824	2.238	4.502	1.636	5	5	$\hat{\tau}_f = 0.330\,\mathrm{sgn}(\dot{q})\left(g-1\right)$

Theory: Identifiability

Scholl, Bacho, Boche, and Kutyniok. "Symbolic recovery of differential equations: The identifiability problem." arXiv:2210.08342, 2022.

Scholl, Bacho, Boche, and Kutyniok. "The uniqueness problem of physical law learning." In ICASSP, IEEE, 2023.

Hauger, Scholl, and Kutyniok. "Robust identifiability for symbolic recovery of differential equations." In ICASSP, IEEE, 2025.

PDE learning problem

Input: Data $(u(t_i, x_j))_{i,j}$ about the solution $u : \mathbb{R} \times \mathbb{R}^m \to \mathbb{R}^d$ of some PDE

$$\frac{\partial^n u}{\partial^n t} = F(u, \frac{\partial u}{\partial x_1}, \dots, \frac{\partial u}{\partial x_m}, \frac{\partial^2 u}{\partial^2 x_1}, \frac{\partial^2 u}{\partial x_1 \partial x_2}, \dots).$$

PDE learning problem

Input: Data $(u(t_i,x_j))_{i,j}$ about the solution $u\colon \mathbb{R}\times\mathbb{R}^m \to \mathbb{R}^d$ of some PDE

$$\frac{\partial^n u}{\partial^n t} = F\left(u, \frac{\partial u}{\partial x_1}, \dots, \frac{\partial u}{\partial x_m}, \frac{\partial^2 u}{\partial^2 x_1}, \frac{\partial^2 u}{\partial x_1 \partial x_2}, \dots\right).$$

Output: The function F describing the PDE solved by u

PDE learning problem

Input: Data $(u(t_i, x_j))_{i,j}$ about the solution $u: \mathbb{R} \times \mathbb{R}^m \to \mathbb{R}^d$ of some PDE

$$\frac{\partial^n u}{\partial^n t} = F(u, \frac{\partial u}{\partial x_1}, \ldots, \frac{\partial u}{\partial x_m}, \frac{\partial^2 u}{\partial^2 x_1}, \frac{\partial^2 u}{\partial x_1 \partial x_2}, \ldots).$$

Output: The function F describing the PDE solved by u

$\rightsquigarrow$ When is F uniquely defined by u?

Ambiguity of the PDE

One-way wave equation

$$u_t = -cu_x$$

Korteweg–De Vries (KdV) equation

$$u_t = 6uu_x - u_{xxx}$$

Ambiguity of the PDE

One-way wave equation

$$u_t = -cu_x$$

Korteweg–De Vries (KdV) equation

$$u_t = 6uu_x - u_{xxx}$$

Solved by the same function u.

Ambiguity of the PDE

One-way wave equation

$$u_t = -cu_x$$

Korteweg–De Vries (KdV) equation

$$u_t = 6uu_x - u_{xxx}$$

Solved by the same function u.

$\rightsquigarrow$ Theoretical understanding when there is only one PDE solved by a function u?

Ambiguity of the PDE

One-way wave equation

$$u_t = -cu_x$$

Korteweg–De Vries (KdV) equation

$$u_t = 6uu_x - u_{xxx}$$

Solved by the same function u.

$\rightsquigarrow$ Theoretical understanding when there is only one PDE solved by a function u?

$\rightsquigarrow$ How to numerically check when the PDE is the unique explaining one?

Uniqueness problem

For preselected derivatives $u_{\alpha^1}, \ldots, u_{\alpha^l}$ and fixed $n \in \mathbb{N} \setminus \{0\}$, is there a unique function $F : \mathbb{R}^k \to \mathbb{R}$ in some set of functions V such that

- $\frac{\partial^n u}{\partial^n t}(t,x) = F(u_{\alpha^1}(t,x), \ldots, u_{\alpha^l}(t,x))$

Uniqueness problem

For preselected derivatives $u_{\alpha^1}, \ldots, u_{\alpha^l}$ and fixed $n \in \mathbb{N} \setminus \{0\}$, is there a unique function $F: \mathbb{R}^k \to \mathbb{R}$ in some set of functions V such that

- $\frac{\partial^n u}{\partial^n t}(t, x) = F(u_{\alpha^1}(t, x), \ldots, u_{\alpha^l}(t, x))$

or

- $\frac{\partial^n u}{\partial^n t}(t, x) = F(t, x_1, \ldots, x_m, u_{\alpha^1}(t, x), \ldots, u_{\alpha^l}(t, x))$

Uniqueness problem

For preselected derivatives $u_{\alpha^1}, ..., u_{\alpha^l}$ and fixed $n \in \mathbb{N} \setminus \{0\}$, is there a unique function $F: \mathbb{R}^k \to \mathbb{R}$ in some set of functions V such that

- $\frac{\partial^n u}{\partial^n t}(t,x) = F(u_{\alpha^1}(t,x), ..., u_{\alpha^l}(t,x))$

or

- $\frac{\partial^n u}{\partial^n t}(t,x) = F(t, x_1, ..., x_m, u_{\alpha^1}(t,x), ..., u_{\alpha^l}(t,x))$

or

- $u_{tt}(t) = F(u(t), u_t(t))$?

LMU LUDWIG-MAXIMILIANS-UNIVERSITÄT MÜNCHEN

Uniqueness problem

For preselected derivatives $u_{\alpha^1}, \dots, u_{\alpha^l}$ and fixed $n \in \mathbb{N} \setminus \{0\}$, is there a unique function $F : \mathbb{R}^k \to \mathbb{R}$ in some set of functions V such that

- $\frac{\partial^n u}{\partial^n t}(t,x) = F(u_{\alpha^1}(t,x), \dots, u_{\alpha^l}(t,x))$

or

- $\frac{\partial^n u}{\partial^n t}(t,x) = F(t, x_1, \dots, x_m, u_{\alpha^1}(t,x), \dots, u_{\alpha^l}(t,x))$

or

- $u_{tt}(t) = F(u(t), u_t(t))$?

General: Let $\boldsymbol{g_1}, \dots, \boldsymbol{g_k} : U \to \mathbb{R}$ be either a **projection on one of the coordinates**, any **derivative of** $\boldsymbol{u}$ that exists, or the function $\boldsymbol{u}$ and let $\boldsymbol{n} \in \mathbb{N} \setminus \{0\}$. Is there a **unique** function $F : \mathbb{R}^k \to \mathbb{R}$ in some set of functions $\boldsymbol{V}$ such that

$$\frac{\partial^n u}{\partial^n t} = F(g)$$

for $g = (g_1, \dots, g_k)$ holds?

Trivially annihilated

Definition

Let $g: U \to \mathbb{R}^k$, $U \subset \mathbb{R}^{m+1}$ open, and $0 \in V \subset \{f: \mathbb{R}^k \to \mathbb{R}\}$. We say that g is **non-trivially annihilated** in V, if there exists some $H \in V \setminus \{0\}$ such that $H(g) = 0$.
Otherwise we say that g is **only trivially annihilated** in V.

Trivially annihilated

Definition

Let $g: U \to \mathbb{R}^k$, $U \subset \mathbb{R}^{m+1}$ open, and $0 \in V \subset \{f: \mathbb{R}^k \to \mathbb{R}\}$. We say that g is **non-trivially annihilated** in V, if there exists some $H \in V \setminus \{0\}$ such that $H(g) = 0$.
Otherwise we say that g is **only trivially annihilated** in V.

Proposition

Let $V \subset \{f: \mathbb{R}^k \to \mathbb{R}\}$ closed under addition and subtraction such that there exists $F \in V$ with $F(g) = \frac{\partial^n u}{\partial^n t}$. Then, F is the ***unique*** *function in V with $F(g) = \frac{\partial^n u}{\partial^n t}$ if and only if g is* ***only trivially annihilated*** *in V.*

Trivially annihilated

Definition

Let $g: U \to \mathbb{R}^k$, $U \subset \mathbb{R}^{m+1}$ open, and $0 \in V \subset \{f : \mathbb{R}^k \to \mathbb{R}\}$. We say that g is **non-trivially annihilated** in V, if there exists some $H \in V \setminus \{0\}$ such that $H(g) = 0$.
Otherwise we say that g is **only trivially annihilated** in V.

Proposition

*Let $V \subset \{f : \mathbb{R}^k \to \mathbb{R}\}$ closed under addition and subtraction such that there exists $F \in V$ with $F(g) = \frac{\partial^n u}{\partial^n t}$. Then, F is the **unique** function in V with $F(g) = \frac{\partial^n u}{\partial^n t}$ if and only if g is **only trivially annihilated** in V.*

Linear PDEs

The function $g = (g_1, \ldots, g_k)$ is **only trivially annihilated** in the set of linear functions if and only if $g_1, \ldots, g_k$ are **linearly independent** over $\mathbb{R}$.

Linear PDEs

The function $g = (g_1, \ldots, g_k)$ is **only trivially annihilated** in the set of linear functions if and only if $g_1, \ldots, g_k$ are **linearly independent** over $\mathbb{R}$.

Corollary (Uniqueness for linear PDEs)

Assume there exists at least one linear function $F : \mathbb{R}^k \to \mathbb{R}$ with $F(g) = \frac{\partial^n u}{\partial^n t}$. Then F is the ***unique linear function*** *such that $F(g) = \frac{\partial^n u}{\partial^n t}$ holds if and only if $g_1, \ldots, g_k$ are* ***linearly independent****.*

Polynomial PDEs

The function $g = (g_1, \ldots, g_k)$ is **only trivially annihilated** in the set of polynomial functions if and only if $g_1, \ldots, g_k$ are **algebraically independent** over $\mathbb{R}$.

Polynomial PDEs

The function $g = (g_1, \dots, g_k)$ is **only trivially annihilated** in the set of polynomial functions if and only if $g_1, \dots, g_k$ are **algebraically independent** over $\mathbb{R}$.

Theorem

Let $g_1, \dots, g_k : U \to \mathbb{R}$, $U \subset \mathbb{R}^{m+1}$ open, be k algebraic functions. Assume that there is polynomial $F : \mathbb{R}^k \to \mathbb{R}$ such that $F(g) = \frac{\partial^n u}{\partial^n t}$:

- *If $k > m+1$: F is **not** the **unique** polynomial with $F(g) = \frac{\partial^n u}{\partial^n t}$.*
- *If $k \leq m+1$: F is the **unique** polynomial with $F(g) = \frac{\partial^n u}{\partial^n t}$ if and only if there exists one point $(t, x) \in U$ with $\boldsymbol{rank(J_g(t,x)) = k}$.*

Analytic PDEs

Proposition (?)

Let $F : \mathbb{R}^k \to \mathbb{R}$ be an analytic function and $\mathscr{D} \subset \mathbb{R}^k$ a set with $\lambda^k(\mathscr{D}) > 0$, where λ^k is the k-dimensional Lebesgue-measure. Then $F|_{\mathscr{D}} = 0$ implies $F = 0$.

Analytic PDEs

Proposition (?)

Let $F: \mathbb{R}^k \to \mathbb{R}$ be an analytic function and $\mathscr{D} \subset \mathbb{R}^k$ a set with $\lambda^k(\mathscr{D}) > 0$, where λ^k is the k-dimensional Lebesgue-measure. Then $F|_{\mathscr{D}} = 0$ implies $F = 0$.

Corollary (Measure criterion for analytic functions)

*Assume there exists at least one analytic function $F: \mathbb{R}^k \to \mathbb{R}$ with $F(g) = \frac{\partial^n u}{\partial^n t}$. Set $\mathscr{D} := g(U)$. If $\lambda^k(\mathscr{D}) > 0$, then F is the **unique analytic function** with $F(g) = \frac{\partial^n u}{\partial^n t}$.*

Analytic PDEs

Theorem (Jacobi criterion for analytic functions)

*Assume there exists at least one analytic function $F: \mathbb{R}^k \to \mathbb{R}$ with $F(g) = \frac{\partial^n u}{\partial^n t}$ and $g_1, \ldots, g_k$ are continuously differentiable. If there exists one point $(t,x) \in U$ with $rank(J_g(t,x)) = k$, then F is the **unique analytic function** with $F(g) = \frac{\partial^n u}{\partial^n t}$.*

Numerical example

$u(t,x) = 1/(t+x)$, solving the PDE $u_t = u_x$.

Numerical example

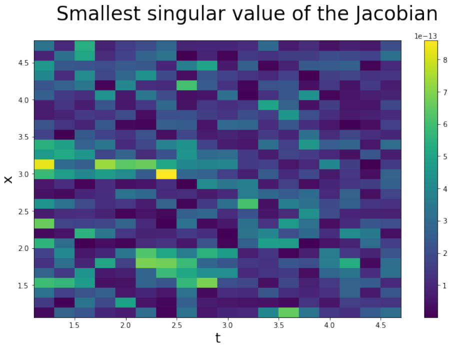
$u(t,x) = 1/(t+x)$, solving the PDE $u_t = u_x$.

Is $u_t = u_x$ the unique polynomial PDE in the form $u_t = F(u, u_x)$ solved by u?

Numerical example

$u(t,x) = 1/(t+x)$, solving the PDE $u_t = u_x$.

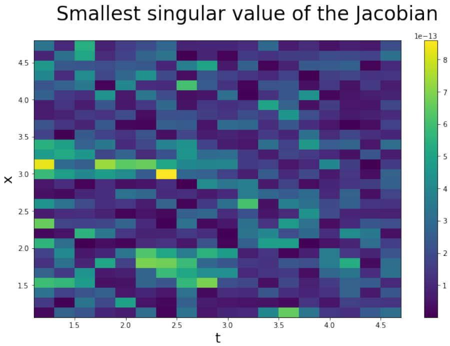
Is $u_t = u_x$ the unique polynomial PDE in the form $u_t = F(u, u_x)$ solved by u?



Numerical example

$u(t,x) = 1/(t+x)$, solving the PDE $u_t = u_x$.

Is $u_t = u_x$ the unique polynomial PDE in the form $u_t = F(u, u_x)$ solved by u?



No, the singular values are close to zero at every point. $(u_t = -u^2)$

Extensions

- To noise

Extensions

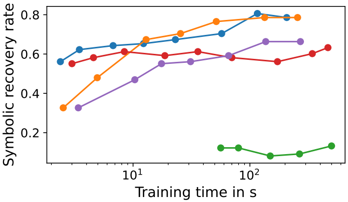
- To noise
- Regularization + measurement error

Extensions

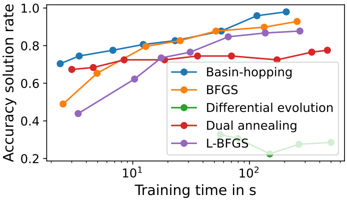
- To noise
- Regularization + measurement error
- Dynamical systems

THANK YOU!

Additional results



(a) Symbolic recovery rate



(b) Accuracy solution rate

Figure: Symbolic recovery and accuracy solution rate (percentage of data sets with $R^2 > 0.999$ for the test set) of ParFam with different optimizers on a random subset of the Feynman problems.