From Data to Discovery: Machine Learning's Role in Advancing Science

Mihaela van der Schaar

John Humphrey Plummer Professor of Machine Learning, Artificial Intelligence and Medicine, University of Cambridge
Director, Cambridge Center for AI in Medicine

van_der_Schaar \LAB
vanderschaar-lab.com

UNIVERSITY OF CAMBRIDGE

mv472@cam.ac.uk
@MihaelaVDS
linkedin.com/in/mihaela-van-der-schaar/

Meet our group!

https://www.vanderschaar-lab.com/
→ Research Team

☆ = joined us in 2024


Alan Jeffares

☆ 
Antonin Berthon

☆ 
Claudio Fanconi


Hao Sun

☆ 
Harry Amad


Julianna Piskorz


Kasia Kobalczyk


Krzysztof Kacprzyk


Max Ruiz Luyten


Nabeel Seedat


Nicolás Astorga


Nicholas Huyn


Paulius Rauba


Qiyao Wei

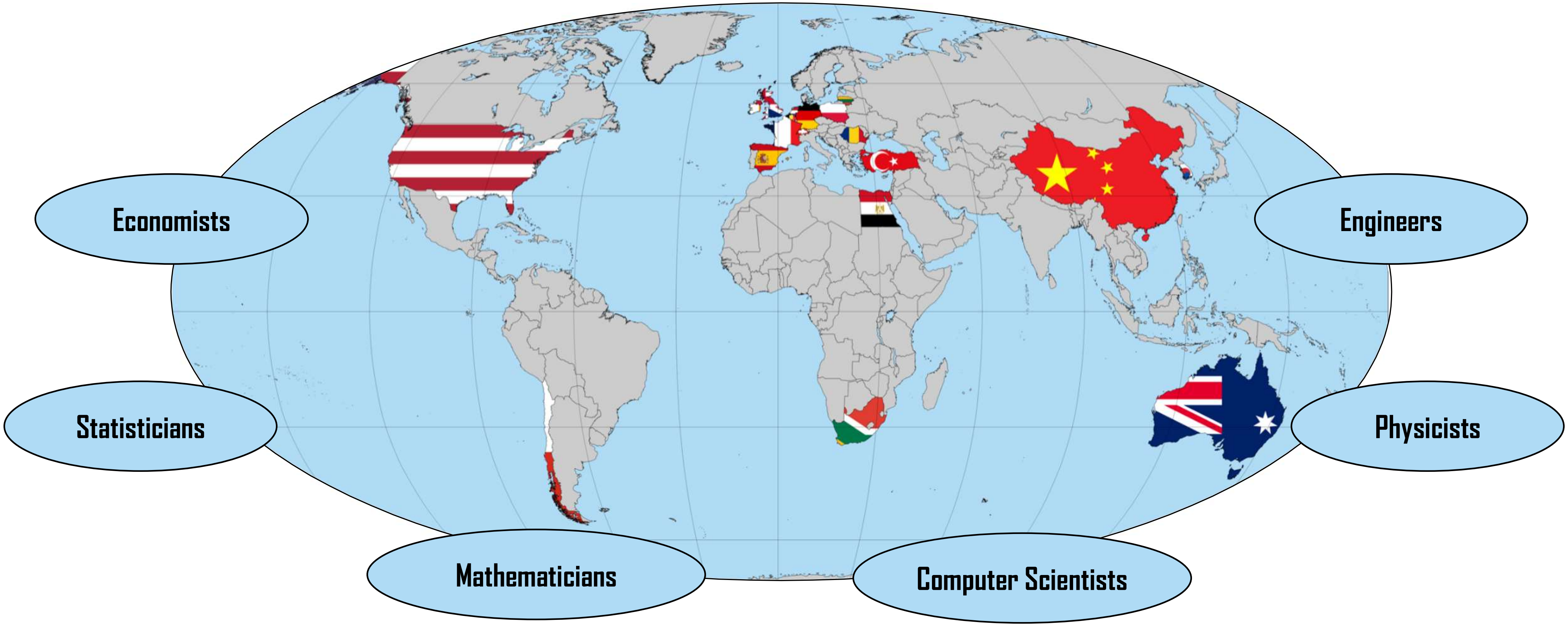

Sam Holt


Tennison Liu

☆ 
Thomas Pouplin

☆ 
Victor Baillet

Our lab – diverse and international



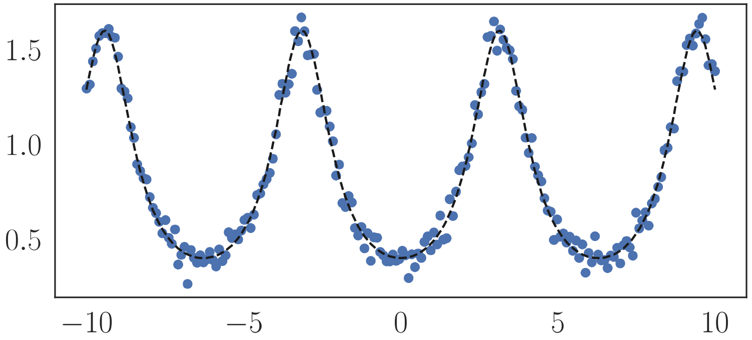
Today's Agenda: Scientific Discovery

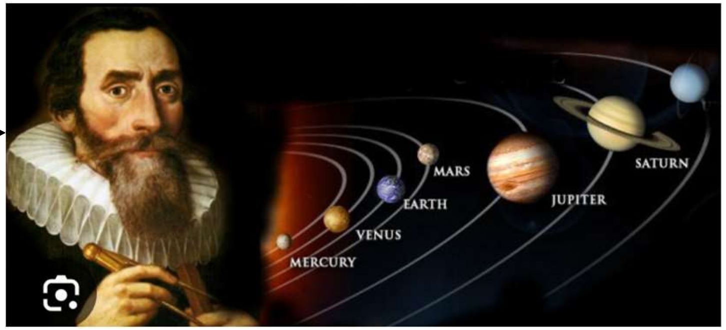
Input dataset $\mathcal{D}$

x_1	x_2	x_3	$f(x_1, x_2, x_3)$
0.8000000000000000	1.3999999999999999	1.1000000000000001	1.3080276559966466
1.2444444444444445	1.8444444444444446	1.5444444444444445	1.3049231475479603
1.6888888888888889	2.2888888888888888	1.9888888888888889	0.1193534271097364
2.1333333333333329	2.7333333333333329	2.4333333333333331	-2.2953685012633303
2.5777777777777775	3.1777777777777776	2.8777777777777778	-5.3304157991987680
3.0222222222222221	3.6222222222222222	3.3222222222222224	-7.7468365793855964
3.4666666666666663	4.0666666666666664	3.7666666666666666	-7.9655006446337717
3.9111111111111105	4.5111111111111111	4.2111111111111104	-4.5673917555329293
4.3555555555555552	4.9555555555555557	4.6555555555555550	3.1295119644306908
4.7999999999999998	5.4000000000000004	5.0999999999999996	14.5971830911204492

Symbolic formula f

$$f(x_1, x_2, x_3) = x_1(1 + x_2 \cos(x_3))$$





Johannes Kepler

$$r = \frac{a(1-\epsilon^2)}{1+\epsilon\cos(\theta)}$$

Overview

1. Scientific Discovery in the Era of Machine Learning
   - Discovering ODEs and PDEs from Data
   - Can LLMs help?

2. Causal Discovery: The Next Step in Causality

3. A powerful application: Digital Twins

From Data to Discovery:
Machine Learning’s Role in Advancing Science

Human experts (scientists) discover governing equations

Science of Medicine
Other sciences

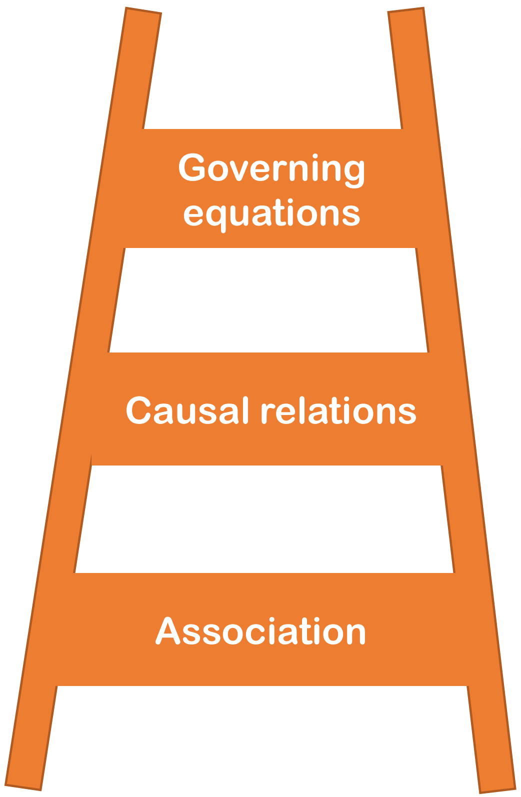
	Explicit function	Ordinary differential equation	Partial differential equation
Typical form	$y = f(x)$	$\frac{dx}{dt} = f(x,t)$	$\frac{\partial u}{\partial t} = f(u,x)$
Examples	Relativity $E = m \cdot c^2$	Newton’s law $m\frac{d^2x}{dt^2} = F(x)$	Heat equation $\frac{\partial u}{\partial t} = \Delta u$

Benefits:
- Concise
- Generalizable
- Amenable to further analysis (e.g., identifying stable equilibria)
- Transparent

- Tumor growth models
- Pharmacological models
- Epidemiological models
- Spatiotemporal physiological systems
- Population models
- Age-structured epidemiological models

★ Discover Transparent Time-series (ICLR 2024)
★ Discover ODEs – D-CODE (ICLR 2022) & DGSR (ICLR 2023)
★ Discover PDEs – D-CIPHER (NeurIPS 2023)

The “Discovery” Ladder



Dynamical systems

★ Discovery of governing eq.

★ **ODE Discovery for Longitudinal Heterogeneous Treatment Effect Inference [ICLR 2024]**

★ **Digital Twins [NeurIPS 2024]**

Discovery of governing equations using ML

	Explicit function	Ordinary differential equation	Partial differential equation
Typical form			
Examples	Relativity	Newton’s law	Heat equation

A hard problem

ODE

$x(t) \longleftrightarrow x'(t)$

To describe dynamical systems, we need

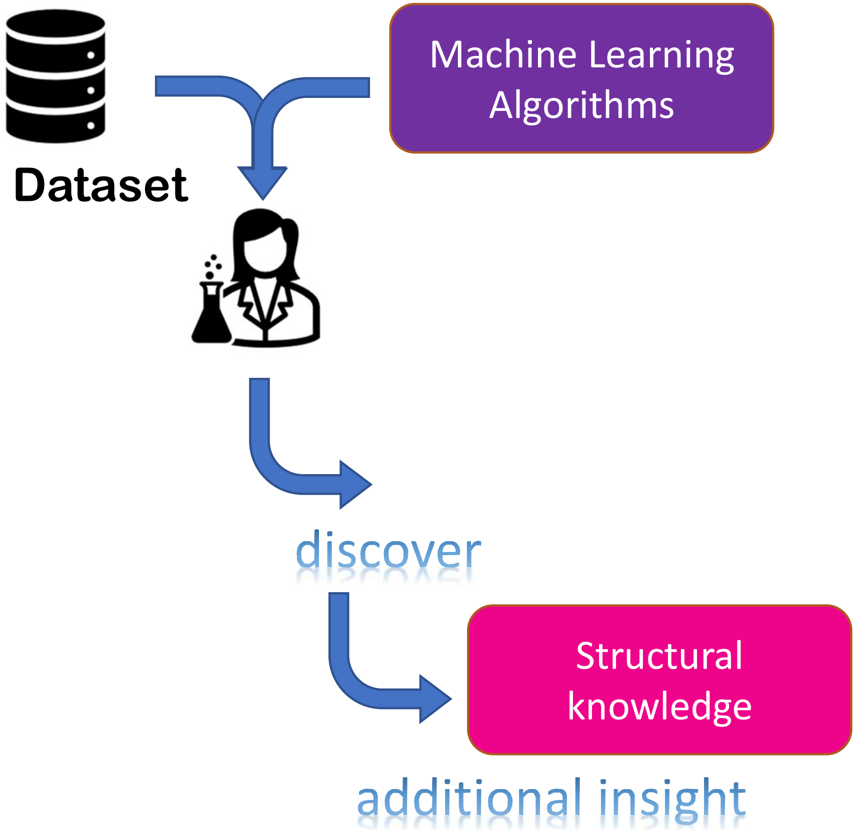
Differential equations

- Equations that involve derivatives
- Commonly used to describe continuous-time dynamical systems
- Describe the change in infinitesimal time (time derivative)
- E.g. Ordinary DE

$$x(t) \xleftrightarrow{\text{ODE}} x'(t)$$

Learning ODEs from data: A hard problem

Problem formulation



Dataset

$$\left\{\begin{array}{c} \mathbf{y}_1(t) \\ \vdots \\ \mathbf{y}_D(t) \end{array}\right\}$$

$t \in \{t_1, t_2, \dots, T\}$

$\mathbf{y}_i(t) \in \mathbb{R}^J$

Goal: Discover $\longrightarrow$

System of J ODEs

$$\left\{\begin{array}{c} f_1(\boldsymbol{x}) = \dot{x}_1 \\ \vdots \\ f_J(\boldsymbol{x}) = \dot{x}_J \end{array}\right\}$$

$x_j : [0, T] \to \mathbb{R}$

$\boldsymbol{x}(t) = [x_1, \dots, x_J]^T$

$f_j : \mathbb{R}^J \to \mathbb{R}$

Symbolic Regression

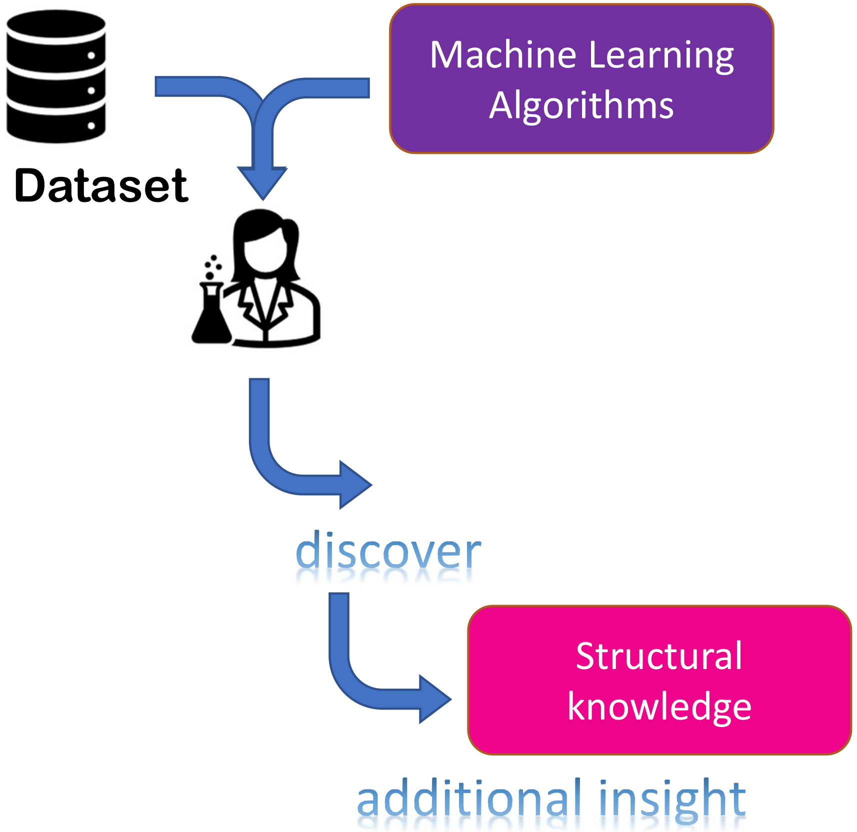
Unique challenges in discovering ODEs

1. **The time derivative is not observed**
   - **Only observe the states over time**
   - **Conventional *symbolic regression* methods are not applicable**

2. **It is difficult to estimate the time derivative**
   - **States are observed sporadically with noise**
   - **Naïve two-step symbolic regression is likely to fail**

3. **Difficulty in directly solving the initial value problem of ODE**
   - **The true initial condition is unknown & difficult to infer**
   - **Sensitive to initial condition**
   - **Computationally challenging**

Discover closed-form ordinary differential equations (ODEs) from observed trajectories - *D-CODE*



Z. Qian, K. Kacprzyk, M. van der Schaar, ICLR 2022

Zhaozhi Qian

Krzysztof Kacprzyk

D-CODE: Discovering Closed-Form ODEs
[Qian, Kacprzyk, vdS, ICLR 2022]

Variational formulation of ordinary differential equations

$$\dot{x}_j(t) = f_j(\boldsymbol{x}(t)), \; \forall j = 1, \dots, J, \; \forall t \in [0, T]$$

Hackbusch, W. (2017)
Variational Formulation

Characterize an ODE without referring to the derivative!

D-CODE: motivation

Variational formulation of ordinary differential equations

$$\dot{x}_j(t) = f_j(\boldsymbol{x}(t)),\ \forall j = 1, \ldots, J,\ \forall t \in [0, T] \tag{1}$$

Definition 1. Consider $J \in \mathbb{N}^+$, $T \in \mathbb{R}^+$, continuous functions $\boldsymbol{x} : [0, T] \to \mathbb{R}^J$, $f : \mathbb{R}^J \to \mathbb{R}$, and $g \in \mathcal{C}^1[0, T]$, where $\mathcal{C}^1$ is the set of continuously differentiable functions. We define the functionals

$$C_j(f, \boldsymbol{x}, g) := \int_0^T f(\boxed{\boldsymbol{x}(t)})g(t)dt + \int_0^T \boxed{x_j(t)}\dot{g}(t)dt; \quad \forall j \in \{1, 2, \ldots, J\}$$

D-CODE: motivation

Variational formulation of ordinary differential equations

$$\dot{x}_j(t) = f_j(\boldsymbol{x}(t)),\ \forall j = 1, \ldots, J,\ \forall t \in [0, T] \tag{1}$$

Definition 1. Consider $J \in \mathbb{N}^+$, $T \in \mathbb{R}^+$, continuous functions $\boldsymbol{x} : [0, T] \to \mathbb{R}^J$, $f : \mathbb{R}^J \to \mathbb{R}$, and $g \in \mathcal{C}^1[0, T]$, where $\mathcal{C}^1$ is the set of continuously differentiable functions. We define the functionals

$$C_j(f, \boldsymbol{x}, g) := \int_0^T f\big(\boldsymbol{x}(t)\big)g(t)dt + \int_0^T x_j(t)\boxed{\dot{g}(t)}dt; \quad \forall j \in \{1, 2, \ldots, J\}$$

Hackbusch, W. (2017)
Variational Formulation

D-CODE: motivation

Variational formulation of ordinary differential equations

$$\boxed{\dot{x}_j(t) = f_j(\boldsymbol{x}(t))}, \ \forall j = 1, \ldots, J, \ \forall t \in [0, T] \quad (1)$$

Definition 1. Consider $J \in \mathbb{N}^+$, $T \in \mathbb{R}^+$, continuous functions $\boldsymbol{x} : [0, T] \to \mathbb{R}^J$, $f : \mathbb{R}^J \to \mathbb{R}$, and $g \in \mathcal{C}^1[0, T]$, where $\mathcal{C}^1$ is the set of continuously differentiable functions. We define the functionals

$$C_j(f, \boldsymbol{x}, g) := \int_0^T f\big(\boldsymbol{x}(t)\big) g(t) dt + \int_0^T x_j(t) \dot{g}(t) dt; \quad \forall j \in \{1, 2, \ldots, J\}$$

Proposition 1. *(Hackbusch, 2017) Consider $J \in \mathbb{N}^+$, $T \in \mathbb{R}^+$, a continuously differentiable function $\boldsymbol{x} : [0, T] \to \mathbb{R}^J$, and continuous functions $f_j : \mathbb{R}^J \to \mathbb{R}$ for $j = 1, \ldots, J$. Then $\boldsymbol{x}$ is the solution to the system of ODEs in Equation 1 if and only if*

$$\boxed{C_j(f_j, \boldsymbol{x}, g) = 0,} \ \forall j \in \{1, \ldots, J\}, \ \forall g \in \mathcal{C}^1[0, T], \ g(0) = g(T) = 0$$

D-CODE: theory

$$d_{\boldsymbol{x}}(f, f^*) := ||f \circ \boldsymbol{x} - f^* \circ \boldsymbol{x}||_2 = ||(f - f^*) \circ \boldsymbol{x}||_2$$

Theorem 1. *Consider $J \in \mathbb{N}^+$, $j \in \{1, \ldots, J\}$, $T \in \mathbb{R}^+$. Let $f^* : \mathbb{R}^J \to \mathbb{R}$ be a continuous function, and let $\boldsymbol{x} : [0, T] \to \mathbb{R}^J$ be a continuously differentiable function satisfying $\dot{x}_j(t) = f^*(\boldsymbol{x}(t))$. Consider a sequence of functions $(\widehat{\boldsymbol{x}}_k)$, where $\widehat{\boldsymbol{x}}_k : [0, T] \to \mathbb{R}^J$ is a continuously differentiable function. If $(\widehat{\boldsymbol{x}}_k)$ converges to $\boldsymbol{x}$ in L^2 norm. Then for any Lipschitz continuous function f*

$$\lim_{S\to\infty} \lim_{k\to\infty} \sum_{s=1}^{S} C_j(f, \widehat{\boldsymbol{x}}_k, g_s)^2 = d_{\boldsymbol{x}}(f, f^*)^2, \tag{7}$$

where $\{g_1, g_2, \ldots\}$ is a Hilbert (orthonormal) basis for $L^2[0, T]$ such that $\forall i$, $g_i(0) = g_i(T) = 0$ and $g_i \in C^1[0, T]$.

D-CODE: theory

$$d_{\boldsymbol{x}}(f, f^*) := ||f \circ \boldsymbol{x} - f^* \circ \boldsymbol{x}||_2 = ||(f - f^*) \circ \boldsymbol{x}||_2$$

Theorem 1. *Consider $J \in \mathbb{N}^+$, $j \in \{1, \dots, J\}$, $T \in \mathbb{R}^+$. Let $f^* : \mathbb{R}^J \to \mathbb{R}$ be a continuous function, and let $\boldsymbol{x} : [0, T] \to \mathbb{R}^J$ be a continuously differentiable function satisfying $\dot{x}_j(t) = f^*(\boldsymbol{x}(t))$. Consider a sequence of functions $(\widehat{\boldsymbol{x}}_k)$, where $\widehat{\boldsymbol{x}}_k : [0, T] \to \mathbb{R}^J$ is a continuously differentiable function. If $(\widehat{\boldsymbol{x}}_k)$ converges to $\boldsymbol{x}$ in L^2 norm. Then for any Lipschitz continuous function f*

$$\lim_{S \to \infty} \lim_{k \to \infty} \sum_{s=1}^{S} C_j(f, \widehat{\boldsymbol{x}}_k, g_s)^2 = d_{\boldsymbol{x}}(f, f^*)^2, \tag{7}$$

where $\{g_1, g_2, \dots\}$ is a Hilbert (orthonormal) basis for $L^2[0, T]$ such that $\forall i$, $g_i(0) = g_i(T) = 0$ and $g_i \in C^1[0, T]$.

Natural choice

$$g_s(t) = \sqrt{2/T} \sin(s\pi t/T)$$

D-CODE: algorithm

Preprocessing

$$\begin{Bmatrix} \mathbf{y}_1(t) \\ \cdot \\ \cdot \\ \cdot \\ \mathbf{y}_D(t) \end{Bmatrix} \xrightarrow{\text{Estimate}} \begin{Bmatrix} \hat{\boldsymbol{x}}_1(t) \\ \cdot \\ \cdot \\ \cdot \\ \hat{\boldsymbol{x}}_D(t) \end{Bmatrix}$$

$t \in \{t_1, t_2, \dots, T\}$

$\mathbf{y}_i(t) \in \mathbb{R}^J$

$t \in [0, T]$

$\boldsymbol{x}_i(t) \in \mathbb{R}^J$

We estimate trajectories, ***not*** derivatives!

Optimization

$$C_j(f, \boldsymbol{x}, g) := \int_0^T f(\boldsymbol{x}(t)) g(t) dt + \int_0^T x_j(t) \dot{g}(t) dt$$

$$\hat{f}_j = \arg\min_f \sum_{i=1}^{N} \sum_{s=1}^{S} C_j(f, \hat{\boldsymbol{x}}_i, g_s)^2$$

Prespecified testing functions

$$g_s(t) = \sqrt{2/T} \sin(s\pi t/T)$$

Symbolic regression

D-CODE: experiments

Dynamical systems:
- **Gompertz model**
- **Generalized logistic model**
- **Glycolytic oscillator**
- **Lorenz system**

Benchmarks:
Two-step symbolic regression with

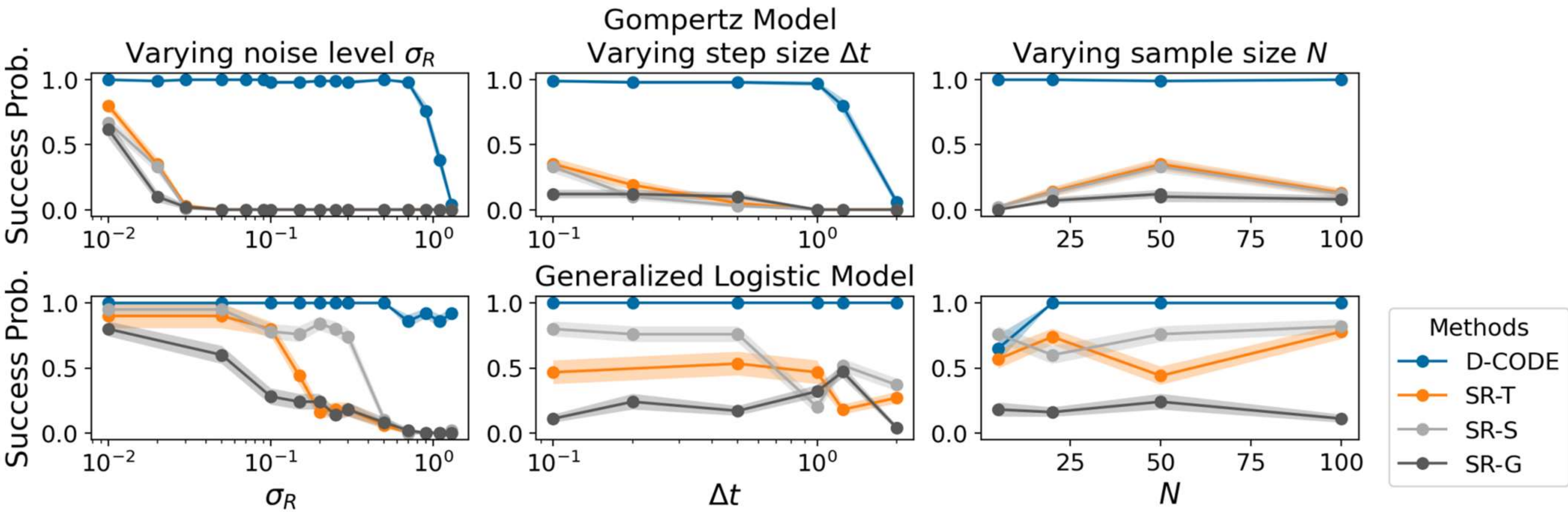
a) **total variation regularized differentiation (SR-T)**
b) **spline-smoothed differentiation (SR-S)**
c) **Gaussian process smoothed differentiation (SR-G)**

D-CODE: Experiments

$$\dot{x}(t) = -\theta_1 x(t) \cdot \log\left(\theta_2 x(t)\right) \quad \text{Gompertz Model}$$

$$\dot{x}(t) = \theta_1 x(t) \cdot \left(1 - x(t)^{\theta_2}\right) \quad \text{Generalized Logistic Model}$$

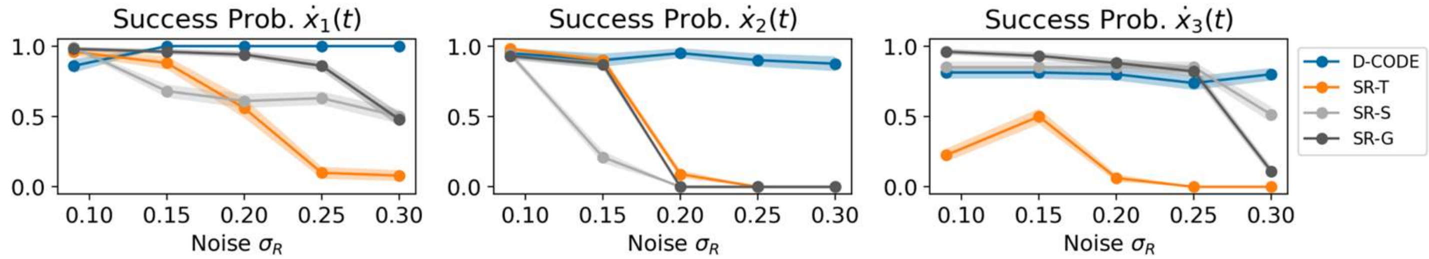
asymmetric growth with saturation

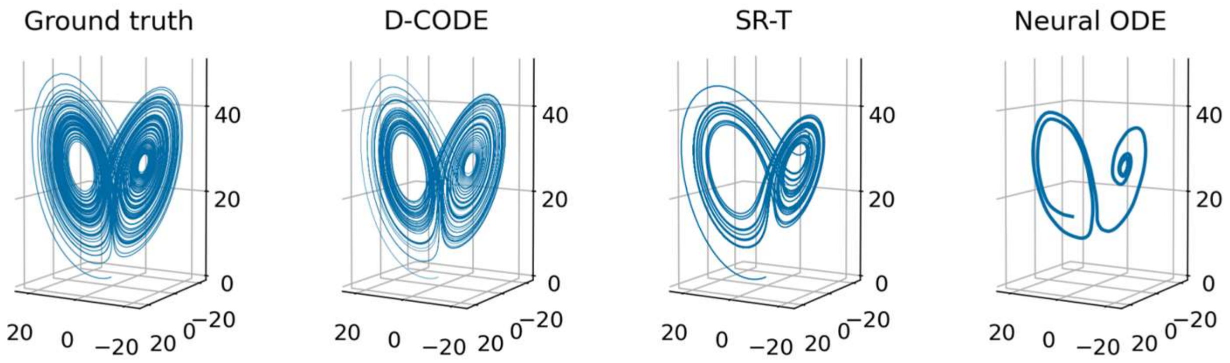


D-CODE: Experiments

Chaotic Lorenz system. The Lorenz system is a model system for chaotic dynamics, defined as:

$$\dot{x}_1(t) = \theta_1\big(x_2(t) - x_1(t)\big); \;\; \dot{x}_2(t) = x_1(t)\big(\theta_2 - x_3(t)\big) - x_2(t); \;\; \dot{x}_3(t) = x_1(t)x_2(t) - \theta_3 x_3(t)$$

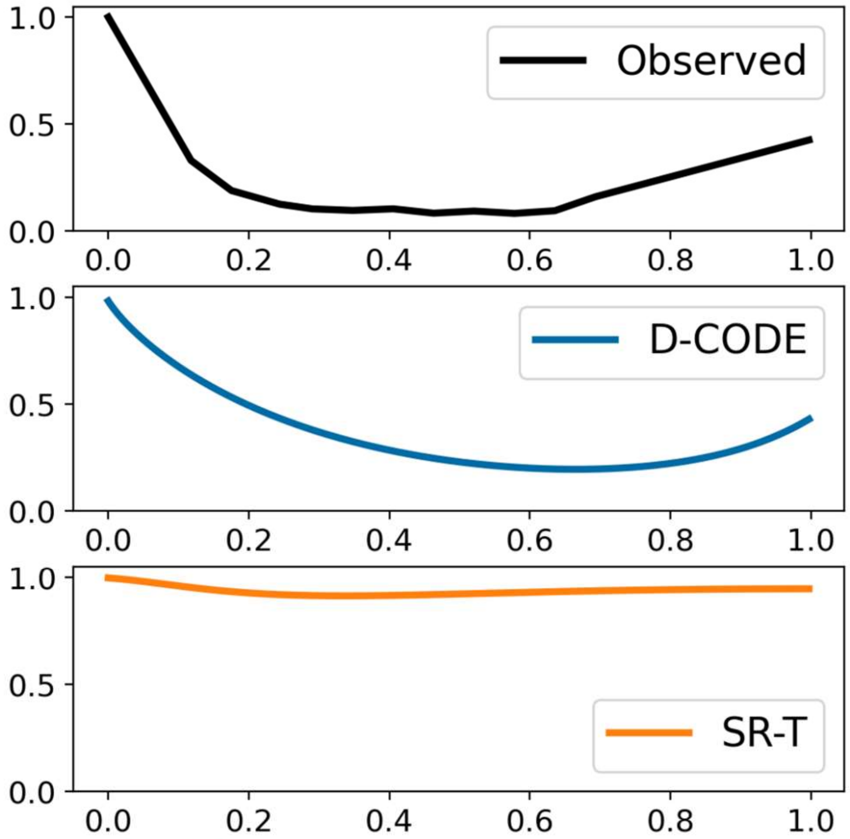




chaotic &
non-periodic systems

D-CODE in action

Discover temporal effects of chemotherapy on tumor volume



Dataset: 8 clinical trials on cancer patients

The following two ODEs are discovered by D-CODE and SR-T.

$$\dot{x}(t) = 4.48t^2 x(t) + \log(t); \quad \text{D-CODE}$$

$$\dot{x}(t) = 4x(t) \log\big(tx(t)\big) \log\big(tx(t) + 2t\big); \quad \text{SR-T}$$

Discovery of governing equations using ML

	Explicit function	Implicit function	Ordinary differential equation	Partial differential equation
Typical form	$y = f(x)$	$f(x, y) = c$	$\frac{dx}{dt} = f(x, t)$	$\frac{\partial u}{\partial t} = f(u, x)$

Why do we care?

- Spatiotemporal physical & physiological systems
- Population models
- Age-structured epidemiological models

<u>**A SUPER hard problem**</u>

What about higher order ODEs and PDEs?

$$\frac{du}{dt}$$

$$\frac{\partial u}{\partial t} \quad \frac{\partial u}{\partial x} \quad \frac{\partial u}{\partial y}$$

$$\frac{\partial^2 u}{\partial t^2} \quad \frac{\partial^2 u}{\partial x^2} \quad \frac{\partial^2 u}{\partial y^2}$$

$$\frac{\partial^2 u}{\partial t \partial x} \quad \frac{\partial^2 u}{\partial t \partial y} \quad \frac{\partial^2 u}{\partial x \partial y}$$

$$u\frac{\partial u}{\partial t} \quad u^2\frac{\partial u}{\partial t} \quad u\frac{\partial u}{\partial x}$$

Difficult to search

Variational trick may not work

Kacprzyk, K., Qian, Z. & vdS
D-CIPHER: Discovery of Closed-form Partial Differential Equations
(NeurIPS 2023)

Krzysztof Kacprzyk

Zhaozhi Qian

van_der_Schaar
\ LAB

Relax assumptions and still allow for variational formulation?

Current methods that utilize variational formulation
- **make the evolution assumption and**
- **assume the PDE to be in a linear combination form or**
- **work only for explicit first order ODEs (D-CODE)**

Relaxing Linear Combination assumption –
not trivial as not all PDEs admit variational formulation

Any PDE: Derivative-bound and derivative-free part

$$\underbrace{f\left(\boldsymbol{x}, \boldsymbol{u}(\boldsymbol{x}), \partial^{[K]}\boldsymbol{u}(\boldsymbol{x})\right)}_{\text{derivative-bound}} - \underbrace{g\left(\boldsymbol{x}, \boldsymbol{u}(\boldsymbol{x})\right)}_{\text{derivative-free}} = 0$$

Requires some technical constraints for the variational trick to work!

Can be evaluated directly from data – No additional constraints!

Variational-Ready PDEs

Currently the broadest family of PDEs that admit variational formulation

Kacprzyk, K., Qian, Z. & van der Schaar, M. D-CIPHER: Discovery of Closed-form Partial Differential Equations. (NeurIPS 2023)

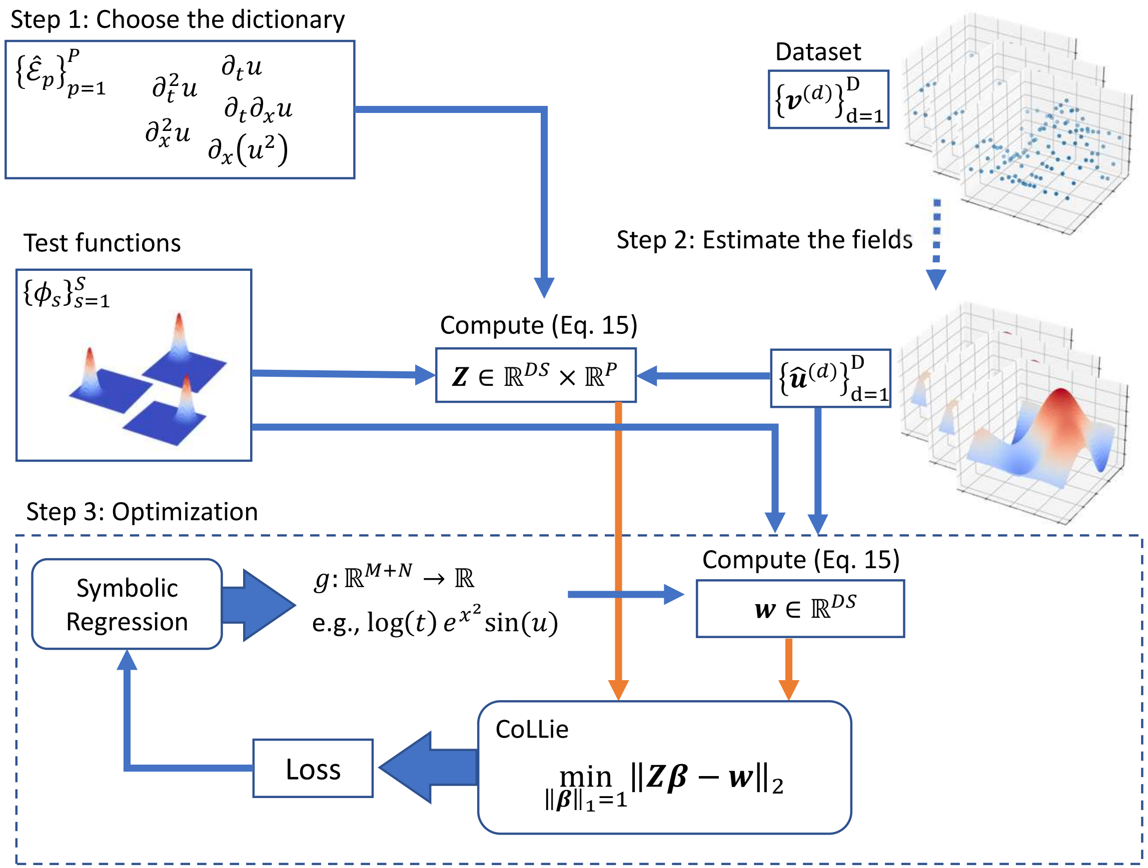
D-CIPHER

- **Assumptions:**
  - **No linear combination assumption**
  - **No evolution assumption**

D-CIPHER

Kacprzyk, K., Qian, Z. & van der Schaar, M. D-CIPHER: Discovery of Closed-form Partial Differential Equations. (NeurIPS 2023)

- **Algorithm**
  - **Uses variational formulation**
  - **Searches through all closed-form derivative-free parts**
  - **Searches through a linear subspace of derivative-bound parts**



van_der_Schaar \LAB

vanderschaar-lab.com

UNIVERSITY OF CAMBRIDGE

Overview

1. Scientific Discovery in the Era of Machine Learning
   - Discovering ODEs and PDEs from data
   - Can LLMs help?

Data-Driven Discovery of Dynamical Systems in Pharmacology using Large Language Models

NeurIPS 2024, Spotlight


Samuel Holt

Zhaozhi Qian


Tennison Liu


Jim Weatherall


Mihaela van der Schaar

van_der_Schaar \LAB

vanderschaar-lab.com

UNIVERSITY OF CAMBRIDGE

Problems with Symbolic Regression

- **Only applicable to problems with few input variables (e.g., three)**
- **Very computationally expensive**
- **Space of equations grows *super exponentially* with equation length and has both discrete and continuous components.**

Our solution: Leveraging Large Language Models (LLMs) to iteratively discover and refine pharmacological dynamics

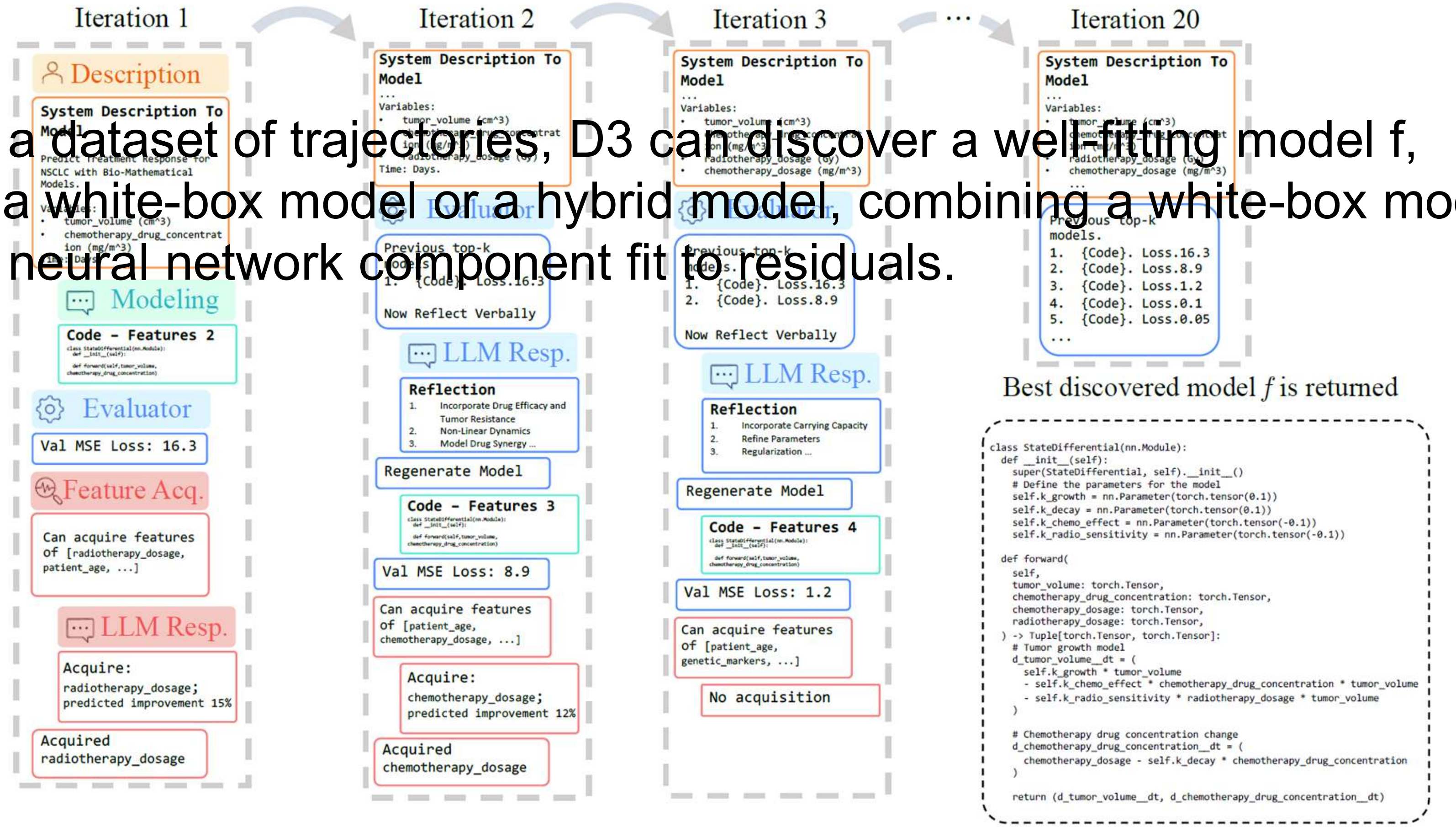
Data-Driven Discovery (D3) framework

Capabilities:

- Proposes, acquires, and integrates new features
- Validates and compares pharmacological dynamical system models
- Insights: Uncovers new insights into pharmacokinetic processes
- Demonstration: Identifies well-fitting, interpretable models across diverse pharmacokinetic datasets

Our solution: Leveraging Large Language Models (LLMs) to iteratively discover and refine pharmacological dynamics

Given a dataset of trajectories, D3 can discover a well-fitting model f, either a white-box model or a hybrid model, combining a white-box model with a neural network component fit to residuals.



New Discovered PK Warfarin Model

Experiments on a real pharmacokinetic Warfarin dataset

- D3 uncovers a new plausible pharmacokinetic model
- Outperforms existing literature
- Highlighting its potential for precision dosing in **clinical applications**

New Discovered PK Warfarin Model

- D3-white-box discovered a new warfarin PK white-box model with a test loss of 0.39, of the following:

$$\frac{dC}{dt} = \sqrt{D} - k_{\text{eff}} \cdot \frac{C}{K_m + C},$$

$$\begin{aligned}
k_{\text{eff}} = {} & k_{e,\text{base}} + k_{e,\text{age}} \cdot (A - \overline{A}) + k_{e,\text{sex}} \cdot (S - \overline{S}) \\
& + k_{\text{decay}} \cdot C + k_{ds} \cdot D \cdot (S - \overline{S}) \\
& + k_{as} \cdot (A - \overline{A}) \cdot (S - \overline{S}) + k_{ad} \cdot D \cdot (A - \overline{A})
\end{aligned}$$

Table 3: Warfarin Modeling Comparison

Method	Warfarin Best Model Test MSE
Existing Warfarin PK	0.646
D3-white-box	**0.39**
D3-hybrid	**0.271**

New Discovered PK Warfarin Model

- Discover well-fitting dynamical system models, achieving low MSE in test predictions on the held-out test dataset of individual trajectories

Method	Lung Cancer MSE ↓	Lung Cancer (with Chemo.) MSE ↓	Lung Cancer (with Chemo. & Radio.) MSE ↓	Plankton Microcosm MSE ↓	COVID-19 MSE ↓	Warfarin PK MSE ↓
DyNODE	326±5.96	55.7±52.8	16.2±6.35	0.000397±0.000883	74±2.69	0.726±0.17
SINDy	325±5.95	11.8±0.442	13.7±0.635	0.00135±0	93.5±0.509	6.84±1.76
ZeroShot	5.78e+03±7.6e+03	304±86.1	6.44e+03±4.27e+03	0.333±0.274	2.47e+03±2.52e+03	1.81±8.53
ZeroOptim	225±204	33.8±50.8	6.38±8.97	0.0133±0.0013	7.88±0.0468	398±5.05e+03
RNN	1.16e+06±3.21e+04	719±94.3	137±5.88	0.0306±0.0459	1.39e+04±2.47e+03	**0.0495±0.0406**
Transformer	7.07±0.558	0.346±0.0701	0.207±0.0318	3.42e-05±1.97e-05	**0.261±0.0915**	1.33±0.941
D3-white-box	59.4±101	4.8±11.8	2.42±2.02	0.000245±0.00022	5.92±1.17	19.6±40.3
D3-hybrid	**4.72±9.16**	**0.0978±0.0463**	**0.135±0.225**	**1.86e-06±1.87e-06**	1.88±2.57	0.647±0.167

New Discovered PK Warfarin Model: Expert commentary

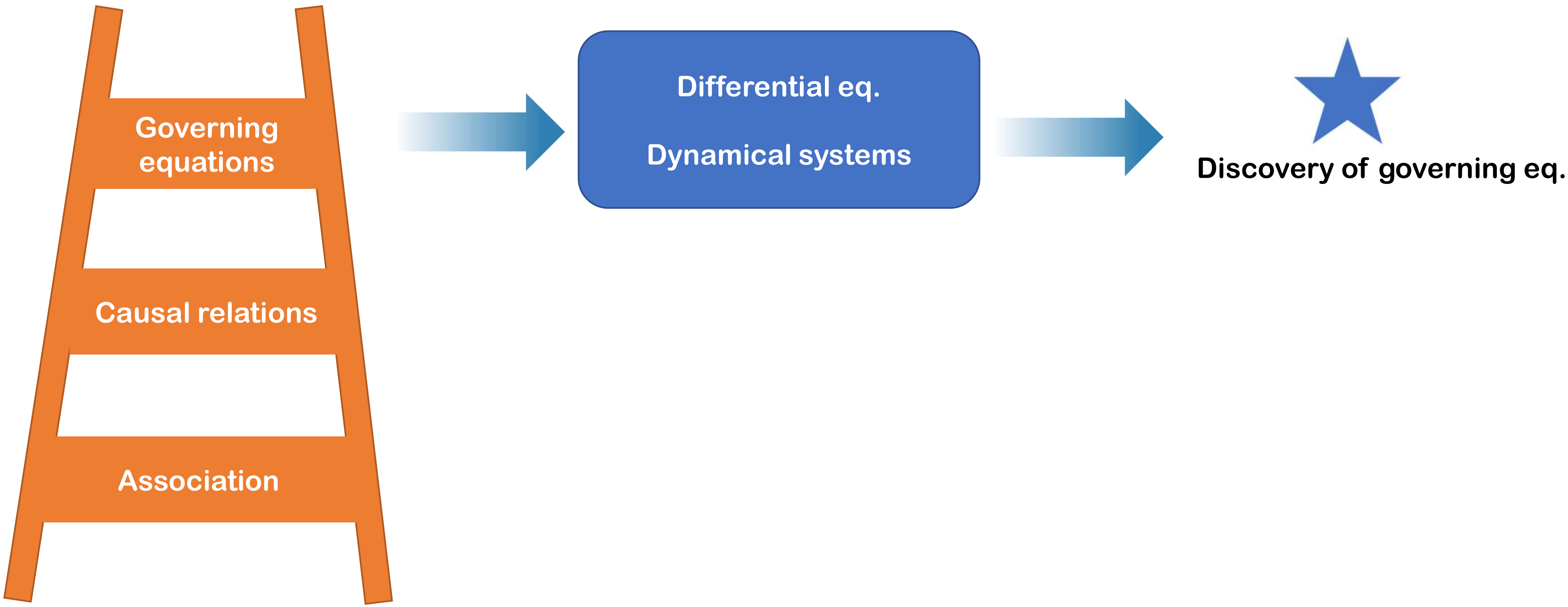
- **Prof. Jean-Baptiste Woillard, Pharmacologist.** "The **model is promising** and **pharmacokinetically plausible**. The next step is to apply D3 to other clinically relevant PK drug datasets."
- **Prof. Richard Peck, Clinical Pharmacologist.** "This model is reasonable and potentially superior. It represents a **significant advance in clinical pharmacology** by automatically identifying robust PK models."
- **Prof. Eoin McKinney, Clinician.** "**This model is significant**, as consortiums are dedicated to improving Warfarin [Consortium, 2009]. The **model adds novel components**, such as the Michaelis component for time-varying changes and novel interaction terms like age-sex."

Overview

1. **Scientific Discovery in the Era of Machine Learning**
   - Discovering ODEs and PDEs from data
   - Can LLMs help?

2. **Causal Discovery: The Next Step in Causality**

The “Discovery” Ladder



Causal treatment effects inference over time

Goal

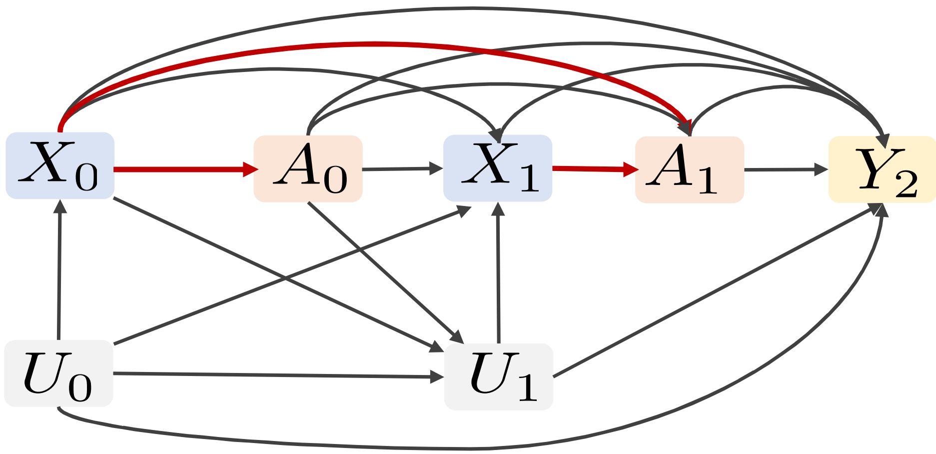
Of interest is estimating the expected potential outcome $Y_{t:t+\tau}(\bar{\boldsymbol{a}}_{t:t+\tau})$, for some $\tau > 0$ under *hypothetical* future treatments $\bar{\boldsymbol{a}}_{t:t+\tau}$ given the *historical* features $\boldsymbol{X}_{0:t}$ and the previous treatments $\boldsymbol{A}_{0:t}$ (Neyman, 1923; Rubin, 1980):

$$\mathbb{E}[Y_{t:t+\tau}(\bar{\boldsymbol{a}}_{t:t+\tau})|\boldsymbol{V}, \boldsymbol{X}_{0:t}, \boldsymbol{A}_{0:t}] \quad (1)$$

Challenges in causal treatment effects inference over time

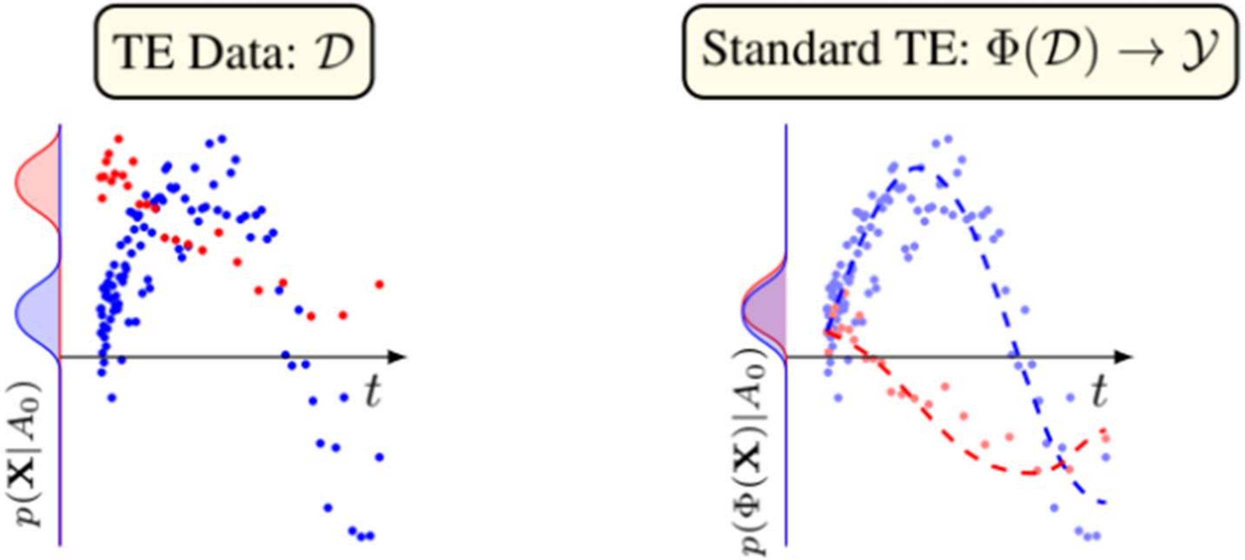
The patient history $\bar{\mathbf{H}}_t = (\bar{\mathbf{X}}_t, \bar{\mathbf{A}}_{t-1}, \mathbf{V})$ contains time-dependent confounders which bias the treatment assignment $\mathbf{A}_t$ in the observational dataset.

Patient covariates - affected by past treatments which then influence future treatments and outcomes



Bias from time-dependent confounders.

Causal treatment effects inference over time: An ML perspective

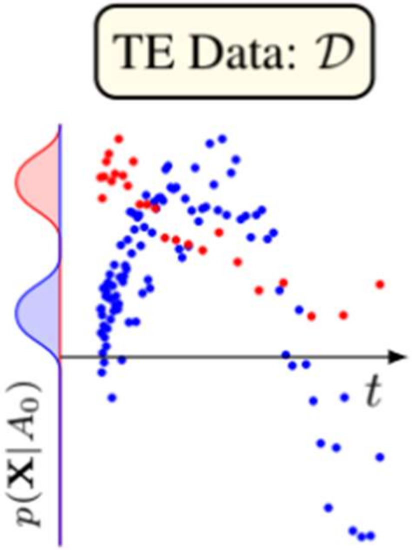


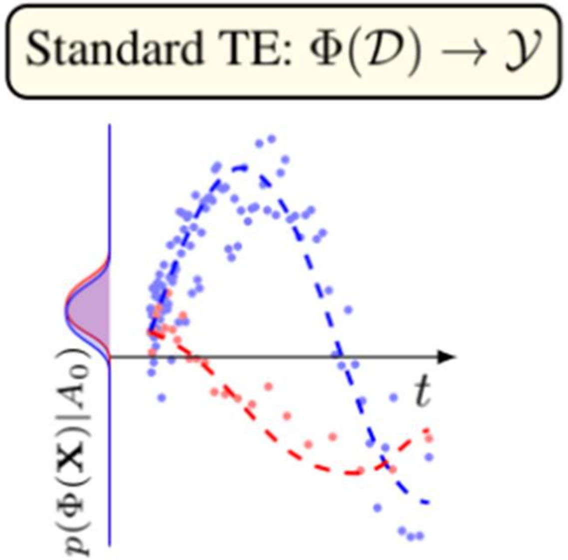
- **RMSN (NeurIPS 2018)**
- **CRN (ICLR 2020)**
- **DTR (NeurIPS 2020)**
- **TE-CDE (ICML 2022)**
- **Informative Sampling (ICML 2023)**

Learns a representation of the data and uses the representation

A Deep Learning perspective

Causal treatment effects inference over time: An ML perspective





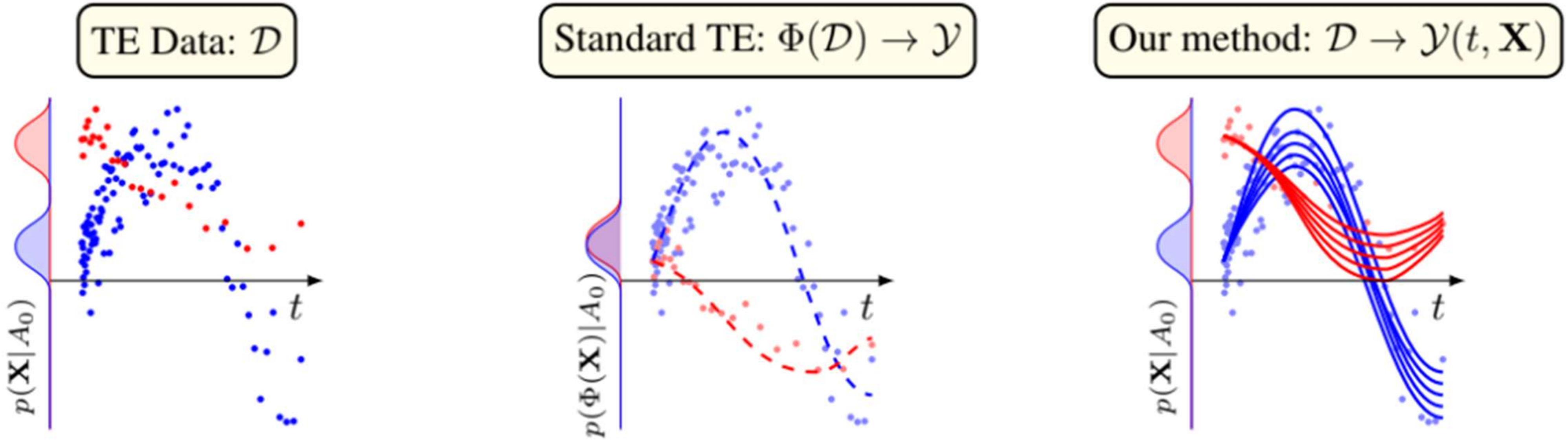
Learns a representation of the data and uses the representation

A Deep Learning perspective

Limitations:

1. **Not interpretable**
2. **Sampling**
3. **(Assumptions)**

Causal treatment effects inference over time: An ML perspective



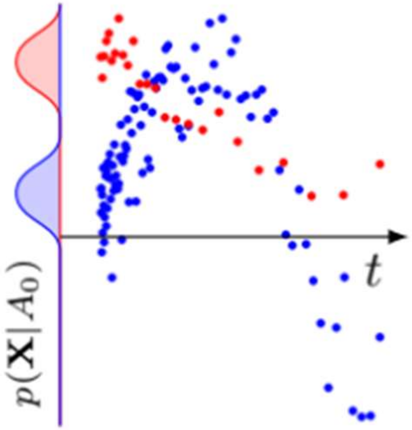
Learns a representation of the data and uses the representation

Learns an ODE, refined for each specific patient

A Dynamical Systems perspective

Causal treatment effects inference over time: An ML perspective

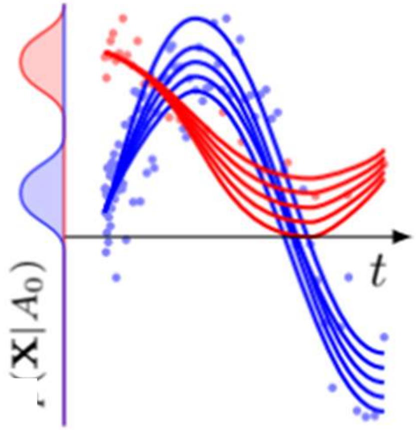
TE Data: $\mathcal{D}$



$$\frac{d\boldsymbol{x}(t)}{dt} = \dot{\boldsymbol{x}}(t) = \boldsymbol{F}(\boldsymbol{v}, \boldsymbol{x}(t), \boldsymbol{a}(t)) \quad \text{and} \quad y(t) = g(\boldsymbol{x}(t)),$$

g is prespecified by the user

Our method: $\mathcal{D} \rightarrow \mathcal{Y}(t, \mathbf{X})$



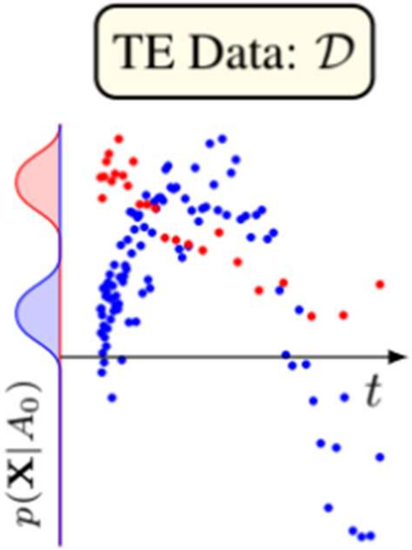
Goal:

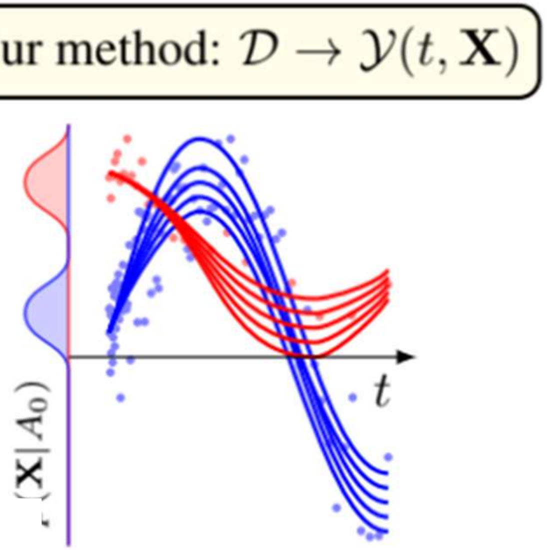
recover the underlying system of ODEs $\boldsymbol{F}$ based on the observed dataset $\mathcal{D}$

Learns an ODE, refined for each specific patient

A Dynamical Systems perspective

Causal treatment effects inference over time: An ML perspective





Addressing Limitations:

1. Interpretable
2. Irregular Sampling
3. New Assumptions

Learns an ODE, refined for each specific patient

A Dynamical Systems perspective

Why structural equations?

Advantages over neural networks

- Interpretable
- Naturally works for irregular sampling and continuous trajectories
- Smaller hypothesis space
- Better performance in certain scenarios

Challenges

- Different ways to learn from data
- Different problem descriptions
- Static features are not considered in ODE discovery
- ODE discovery methods find only a single equation for a whole dataset
- Diverse types of treatment: continuous, binary, categorical, multiple

Our Solution:
ODE Discovery for Longitudinal Heterogeneous Treatment Effect Inference

[Kacprzyk, Holt, Berrevoets, Qian & vdS, ICLR 2024]


Krzysztof Kacprzyk


Sam Holt


Jeroen Berrevoets

Zhaozhi Qian

Our Solution:
ODE Discovery for Longitudinal Heterogeneous Treatment Effect Inference

[Kacprzyk, Holt, Berrevoets, Qian & vdS, ICLR 2024]

- We provide a general framework which connects ODE discovery with TE
- Reconcile the differences
- We propose INSITE, an illustrative TE method based on ODE discovery

Three discrepancies

(1) different assumptions,
(2) discrete (not continuous) treatment plans, and
(3) variability across subjects

Each discrepancy is explained and reconciled with actionable steps.

CATE Assumptions

extensions to continuous-time causal effects (Lok, 2008; Saarela & Liu, 2016; Ryalen et al., 2019)

Assumption 2.1 (Consistency) *For an observed treatment process $\boldsymbol{A}_{0:T^{(i)}} = \boldsymbol{a}$, the potential outcome is the same as the factual outcome $\boldsymbol{Y}(\boldsymbol{a}) = \boldsymbol{Y}_{0:T^{(i)}}$.*

Assumption 2.2 (Overlap) *The treatment intensity process $\lambda(t|\mathfrak{F}_t)$ is not deterministic given any filtration $\mathfrak{F}_t$[2] (Klenke, 2008) and time point $t \in [0, T]$, i.e.,*

$$\gamma < \lambda(t|\mathfrak{F}_t) = \lim_{\delta t \to 0} \frac{p(A_{t+\delta t} - A_t \neq 0|\mathfrak{F}_t)}{\delta t} < 1 - \gamma, \quad \textit{with} \quad \gamma \in (0, 1)$$

Assumption 2.3 (Ignorability) *The intensity process $\lambda(t|\mathfrak{F}_t)$ given the filtration $\mathfrak{F}_t$ is equal to the intensity process that is generated by the filtration $\mathfrak{F} \cup \{\sigma(\boldsymbol{Y}_s) : s > t\}$ that includes the σ-algebras generated by future outcomes $\{\sigma(\boldsymbol{Y}_s) : s > t\}$.*

Our Assumptions

Assumption 3.1 (Existence and Uniqueness) *The underlying process can be modelled by a system of ODEs* $\dot{\boldsymbol{x}}(t) = \boldsymbol{F}(\boldsymbol{v}, \boldsymbol{x}(t), \boldsymbol{a}(t))$,[3] *and for every initial condition* $\boldsymbol{x}_0$, $\boldsymbol{v}$ *and treatment plan* $\boldsymbol{a}$ *at* t_0, *there exists a unique continuous solution* $\boldsymbol{x} : [t_0, T] \to \mathbb{R}^d$ *satisfying the ODEs for all* $t \in (t_0, T)$*(Lindelöf, 1894; Ince, 1956).*

Assumption 3.2 (Observability) *All dimensions of all variables in* $\boldsymbol{F}$ *are observed for all individuals, ensuring sufficient data to identify the system's dynamics and infer the ODE's structure and parameters (Kailath, 1980).*

Assumption 3.3 (Functional Space) *Each ODE in* $\boldsymbol{F}$ *belongs to some subspace of closed-form ODEs. These are equations that can be represented as mathematical expressions consisting of binary operations* $\{+, -, \times, \div\}$, *input variables, some well-known functions (e.g.,* $\{\log, \exp, \sin\}$*), and numeric constants (e.g.,* $\{-0.2, \ldots, 5.2\} \in \mathbb{R}$*) (Schmidt & Lipson, 2009).*

Our Assumptions

Assumptions 3.1 and 3.2 play a crucial role in ODE discovery:
needed to correctly identify the underlying equation.

- Assumption 3.1 ensures that the discovered ODE has a unique solution
- Assumption 3.2 is necessary such that the observed data can be used to accurately identify the underlying ODE

The assumptions made in the treatment effects literature (assumptions 2.1 to 2.3) serve a similar purpose as they allow us to interpret the estimand as a causal effect, i.e., they are necessary for identification.

Assumption 3.3 defines space of equations to be consider for the optimization algorithm

	ODE discovery	**Treatment effects**		**Explanation**
ref	*assumption*	*assumption*	*ref*	
3.1	*existence & uniqueness*	*consistency*	2.1	2.1 is *implicit* through 3.2.
3.2	*observability*	*overlap*	2.2	2.2 can be relaxed by 3.1 and 3.3
3.3	*functional spaces*	*ignorability*	2.3	2.3 is *similar* as 3.2.

Reconcilition in 3 steps

1. New identification assumptions
- accept ODE discovery assumptions

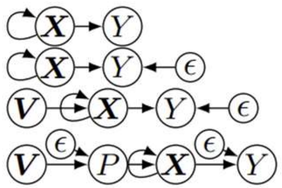
2. Diverse treatment types
- decide how treatments are represented

3. Variability across subjects
- decide on the the desired BSV level

ODE discovery *ref*	*assumption*		Treatment effects *assumption*	*ref*	Explanation
3.1	*existence & uniqueness*		*consistency*	2.1	2.1 is *implicit* through 3.2.
3.2	*observal*				and 3.3
3.3	*function*				

Treatment	S/D	Domain of $\boldsymbol{a}$	Constant	$\boldsymbol{F}(\boldsymbol{x}(t), \boldsymbol{v}, \boldsymbol{a}(t))$
Binary	S D	$a(t) \in \{0, 1\}$	Yes Piece-wise	$\boldsymbol{f}_{a(t)}(\boldsymbol{x}(t), \boldsymbol{v})$ or $\boldsymbol{f}_0(\boldsymbol{x}(t), \boldsymbol{v}) + a(t)\boldsymbol{f}_1(\boldsymbol{x}(t), \boldsymbol{v})$
Categorical	S D	$a(t) \in [1, K]$	Yes Piece-wise	$\boldsymbol{f}_{a(t)}(\boldsymbol{x}(t), \boldsymbol{v})$
Multiple	S D	$\boldsymbol{a}(t) \in \{0, 1\}^K$	Yes Piece-wise	$\boldsymbol{f}_{\boldsymbol{a}(t)}(\boldsymbol{x}(t), \boldsymbol{v})$ or $\sum_{i=1}^{K} a_i(t)\boldsymbol{f}_i(\boldsymbol{x}(t), \boldsymbol{v})$
Continuous	S D	$\boldsymbol{a}(t) \in \mathbb{R}^K$	Yes No	$\boldsymbol{f}(\boldsymbol{x}(t), \boldsymbol{v}, \boldsymbol{a}(t))$

	(i) ODE.	(ii) +RUV	+Cov.	+Dist		$y(t)$	Parameters (eq. (5))
A	✓	✗	✗	✗		$x(t)$	$C_0 = c_0, C_1 = c_1$
B	✓	✓	✗	✗		$x(t) + \epsilon$	$C_0 = c_0, C_1 = c_1$
C	✓	✓	✓	✗		$x(t) + \epsilon$	$C_0 = q(c_0), C_1 = q(c_1)$
D	✓	✓	✓	✓		$x(t) + \epsilon$	$C_0 \sim \mathcal{N}(q(c_0), \sigma_0), C_1 \sim \mathcal{N}(q(c_1), \sigma_1)$



van_der_Schaar \LAB

vanderschaar-lab.com

UNIVERSITY OF CAMBRIDGE

Limitations

1. A correct set of candidate functions (tokens) is necessary for correct model recovery.
2. ODE discovery works best in sparse settings. The reason is two-fold: from a technical point of view, sparse equations are much less complex and simply easier to recover; from a usability point of view, the usefulness of non-sparse equations is limited as interpretability is negatively affected by non-sparse (or non-parsimonious) equations (Crabbe & vdS, 2020).
3. ODEs are noise free. Since we recover ODEs, the found equations do not model a source of noise as is typically the case in structural equation modelling. To model noise terms explicitly, our framework should be extended into stochastic DEs.

Overview

1. Scientific Discovery in the Era of Machine Learning
   - Discovering ODEs and PDEs from data
   - Can LLMs help?

2. Causal Discovery: The Next Step in Causality

3. A powerful application: Digital Twins

Automatically Learning Hybrid Digital Twins of Dynamical Systems

Spotlight @ NeurIPS 2024

Samuel Holt*, Tennison Liu*, Mihaela van der Schaar

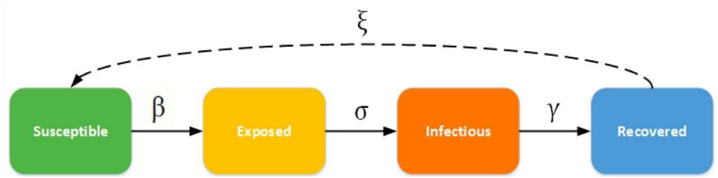
van_der_Schaar \ LAB
vanderschaar-lab.com

UNIVERSITY OF CAMBRIDGE

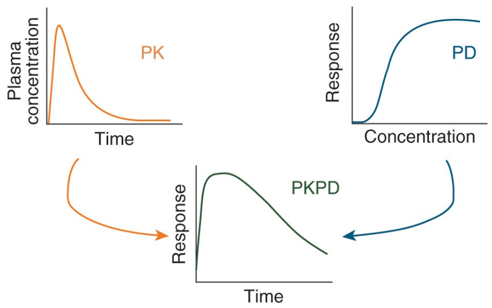
tl522@cam.ac.uk
linkedin.com/in/tennison-liu/

Dynamical Systems

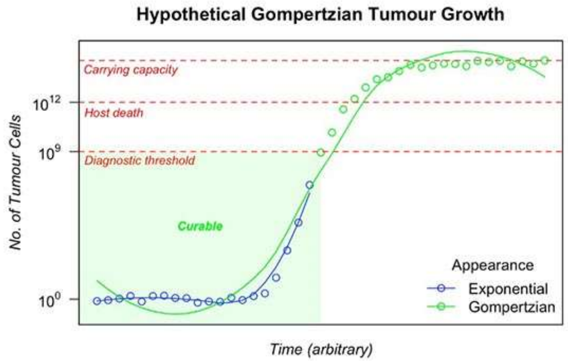
- **Models to describe how variables evolve over time (e.g. to simulate complex physiological processes)**
- **Critical to predicting disease progression, treatment strategies, and improving patient care**
- **Dynamical system $\mathcal{S} \coloneqq (\mathcal{X}, \mathcal{U}, \mathbf{\Phi})$, where $\mathcal{X}$ is the state space, $\mathcal{U}$ is the action space (e.g. treatments), and $\mathbf{\Phi}\colon \mathcal{X} \times \mathcal{U} \times \mathcal{T} \to P(\mathcal{X})$ is the dynamics function**



SEIR Model



PKPD Models



Tumour Growth

Digital Twins

Digital Twins (DTs): Computational models $\boldsymbol{f}_{\boldsymbol{\theta},\boldsymbol{\omega}(\boldsymbol{\theta})} \in \mathcal{F}$ that aim to approximate the dynamics model $\boldsymbol{\Phi}$

- **$\boldsymbol{\theta} \in \boldsymbol{\Theta}$ denotes the model specification, and $\boldsymbol{\omega}(\boldsymbol{\theta}) \in \boldsymbol{\Omega}(\boldsymbol{\theta})$ denotes the model parameterization**

Useful for answering questions:

- **Simulate future outcomes (*what is the future disease spread?*)**
- **Understand system changes (*how does disease dynamics vary in different demographics?)***
- **Evaluate the impact of control/intervention policy (*how to curb disease transmission?*)**

Digital Twins: Desiderata

Effective DTs should satisfy the following desiderata:

[P1] Generalisation to unseen state-action distributions. The DT should robustly model state-action distributions not observed during training

Example:
Can a DT trained on adult patient data reliably predict drug responses for paediatric cases?

Digital Twins: Desiderata

Effective DTs should satisfy the following desiderata:

[P1] Generalisation to unseen state-action distributions. The DT should robustly model state-action distributions not observed during training

[P2] Sample-efficient learning. Learn accurate dynamics given the limited volume of empirical data

> **Example:**
> **Can a DT model the progression of rare diseases and its response to treatment with <100 samples?**

Digital Twins: Desiderata

Effective DTs should satisfy the following desiderata:

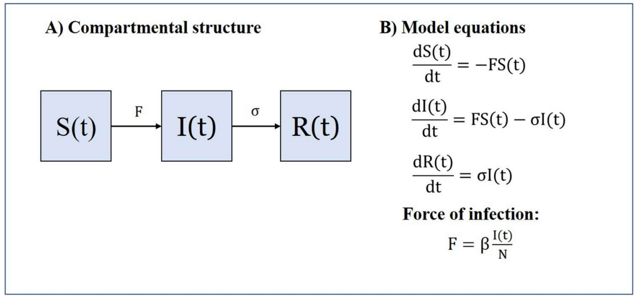
[P1] Generalisation to unseen state-action distributions. The DT should robustly model state-action distributions not observed during training

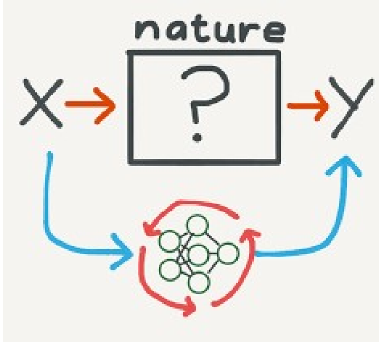
[P2] Sample-efficient learning. Learn accurate dynamics given the limited volume of empirical data

[P3] Evolvability. The twin should be easily evolvable to model the changing dynamics of the underlying system

Example:
Is the DT model easily 'updatable' to incorporate new bacterial strains (or evolving resistance patterns) without requiring complete retraining?

Existing Approaches





Mechanistic

Closed-form equations, grounded in biological/physical principles

Strengths: strong generalization [P1] (when correctly specified)

Weaknesses: fail catastrophically when incorrect, limited by domain knowledge [P3]

Neural

Learns dynamics directly from data using neural/black box models

Strengths: requires minimal assumptions, learns complex dynamics that elude mechanistic modelling

Weaknesses: sample-inefficient [P2], over-parameterised and monolithic black-box [P3]

Our Approach: Motivation

Combine their strengths to develop Hybrid Digital Twins (HDTwins), $f = f_{mech} \circ f_{neural}$

- **f_{mech} symbolically encodes domain-grounded priors, improving generalisation**
- **f_{neural} models complex temporal patterns where f_{mech} might be incomplete/incorrect**

Traditionally: relied heavily on human expertise to craft hybrid DTs

Our work: automatically specify and optimize hybrid DT models

Our Approach: Formulation

Conceptually, hybrid modelling $f_{\boldsymbol{\theta},\boldsymbol{\omega}(\boldsymbol{\theta})}$ involves two stages:

- **Specification of the model structure (neural architecture, functional form), $\boldsymbol{\theta} \in \boldsymbol{\Theta}$**
- **Parameterisation the model (neural weights, coefficients), $\boldsymbol{\omega}(\boldsymbol{\theta}) \in \boldsymbol{\Omega}(\boldsymbol{\theta})$**

Mathematically, this process can be formulated as a bi-level optimisation problem:

$$\min_{\boldsymbol{\theta} \in \boldsymbol{\Theta}} \mathcal{L}_{outer}\left(\boldsymbol{\theta}, \boldsymbol{\omega}^*(\boldsymbol{\theta})\right)$$

$$\text{where} \quad \boldsymbol{\omega}^*(\boldsymbol{\theta}) = \operatorname{argmin}_{\boldsymbol{\omega} \in \boldsymbol{\Omega}(\boldsymbol{\theta})} \mathcal{L}_{inner}\left(\boldsymbol{\theta}, \boldsymbol{\omega}(\boldsymbol{\theta})\right)$$

- **Upper-level: optimal specification that maximises generalisation performance**
- **Lower-level: optimal parameters that maximises training performance**

Easier said than done?

Automatically learning HDTwins is challenging:

Encoding domain priors

Automatically encoding the correct domain knowledge into hybrid DTs (crucial to improving generalisation and sample efficiency)

Combinatorial search space

Space of possible models is discrete/combinatorial, intractable to manually specify

Our Approach: Method Overview

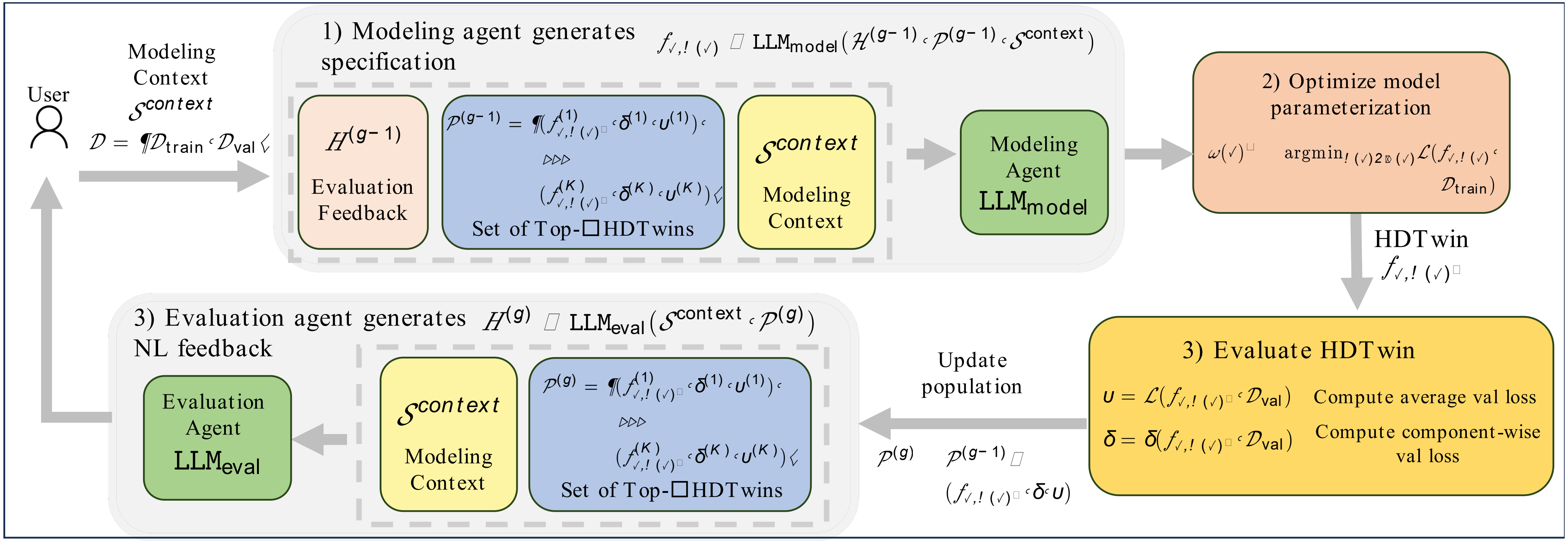
HDTwinGen: Novel evolutionary framework that efficiently designs DTs using large language models (LLMs)

Three steps:

- Utilising LLMs as a generative model to iteratively propose DT specification (represented as code)
- Offline optimization of model parameters from training data
- Model performance is automatically evaluated and fed back to the LLM for iterative improvements

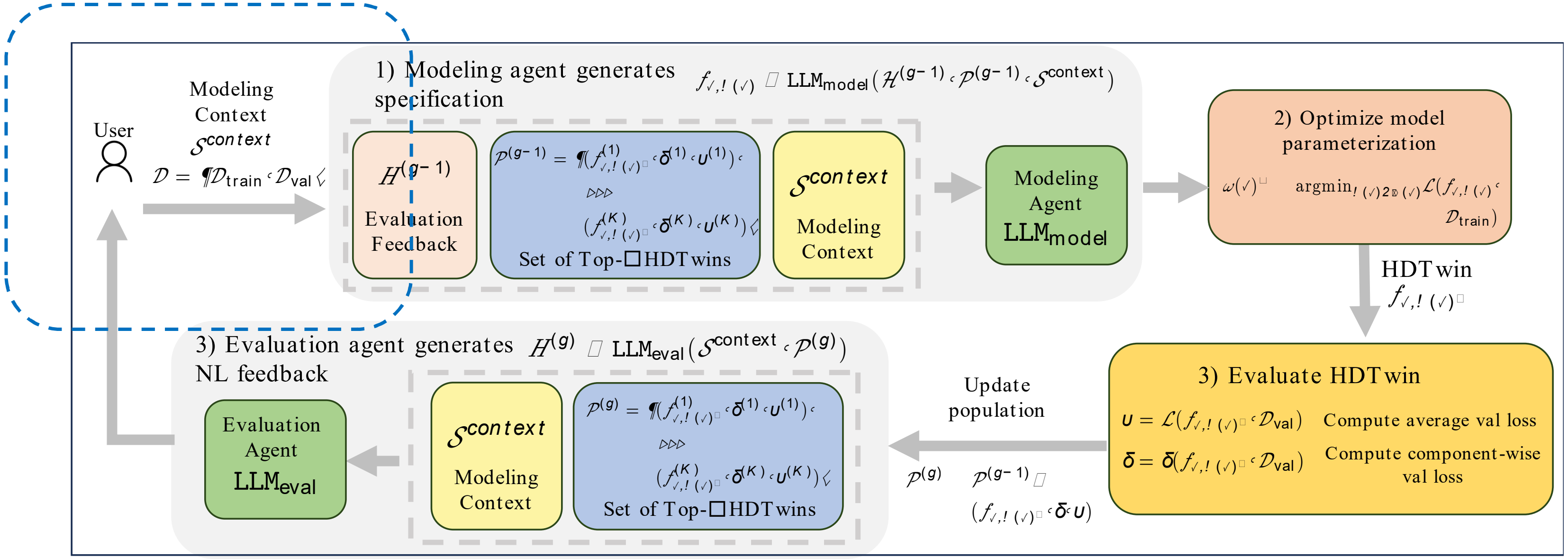
This process is repeated over multiple generations until we have a model that we are satisfied with

Our Approach: In Detail



Our Approach: In Detail

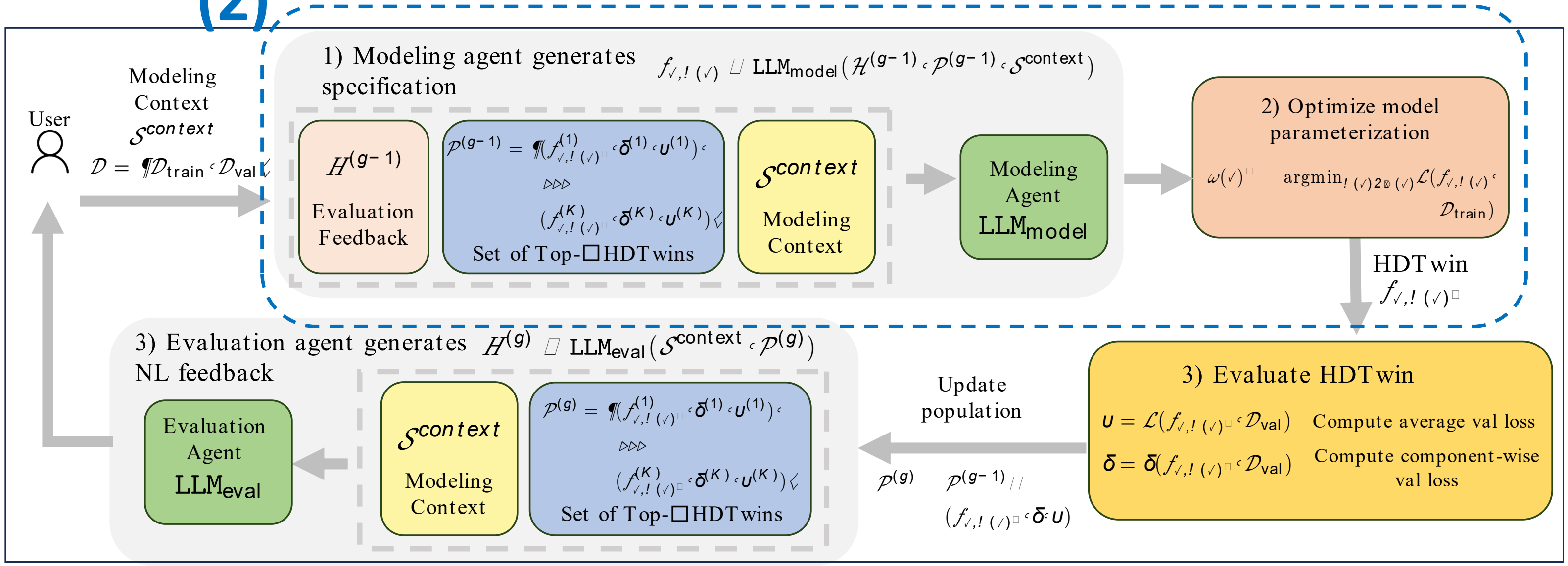
(1)



Initialisation. The process begins with the user providing:

- **Modelling context $\mathcal{S}^{context}$, which semantically describes the system**
- **Data $\mathcal{D}$ used for training/evaluation**

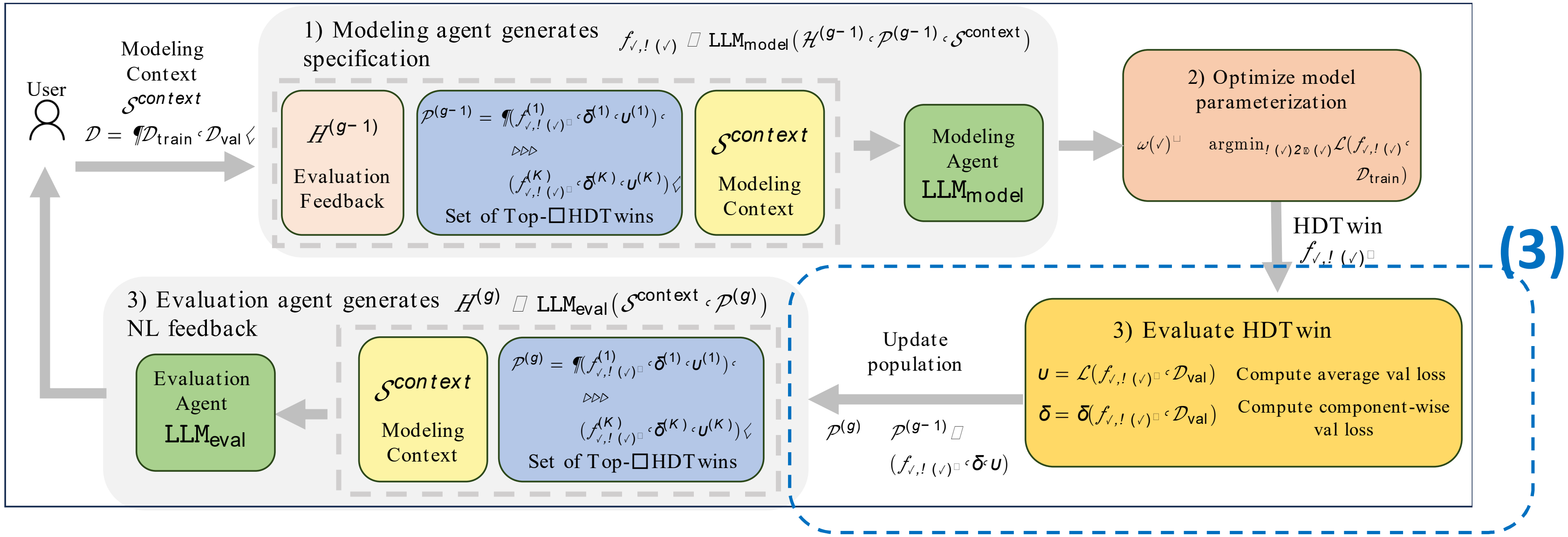
Our Approach: In Detail (2)



Model generation. In iteration $g \in G$:

- **The modelling agent (LLM) generates a novel model specification $f_{\theta,\omega(\theta)}$ (with placeholder parameters)**
- **It has access to the modelling context $\mathcal{S}^{context}$, set of Top-K past models $\mathcal{P}^{(g-1)}$, and most recent evaluation feedback $H^{(g-1)}$**
- **The parameters are optimised to yield new candidate model**

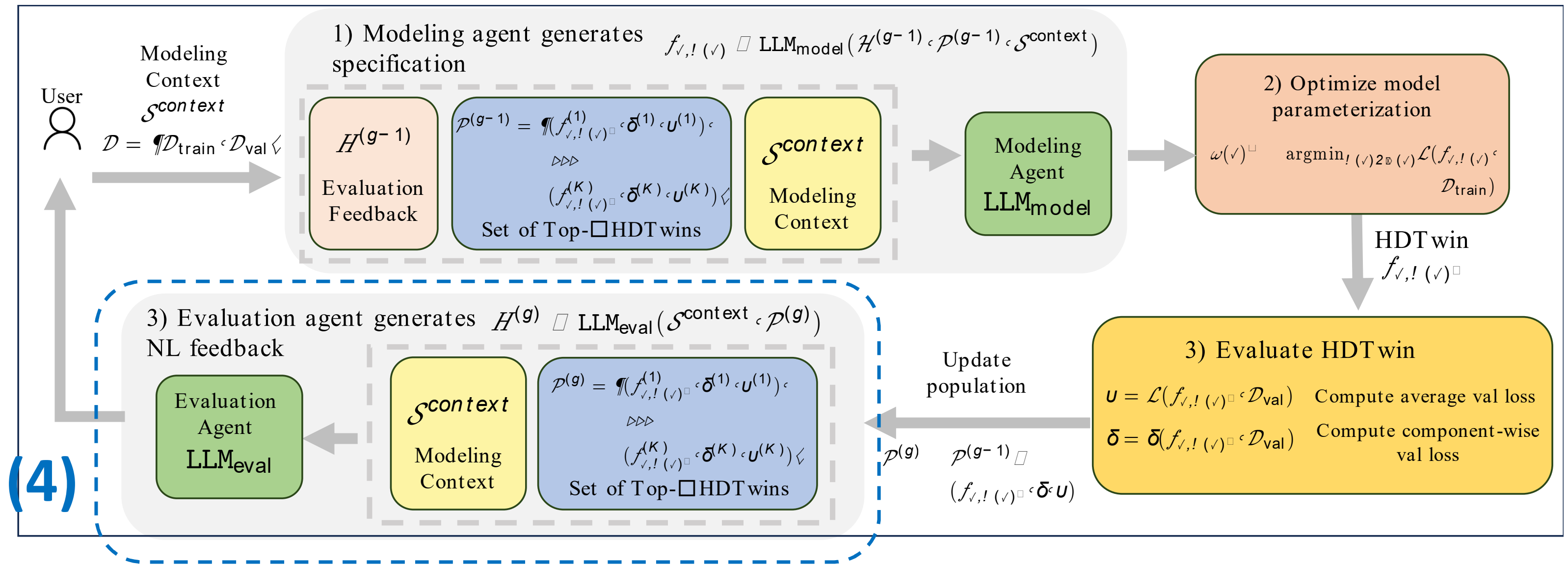
Our Approach: In Detail



Model evaluation and selection

- **The newly generated model is evaluated using the provided modelling objective (e.g. MSE)**
- **Model pool updated: $\mathcal{P}^{(g)}$ is updated with new top-K models**

Our Approach: In Detail



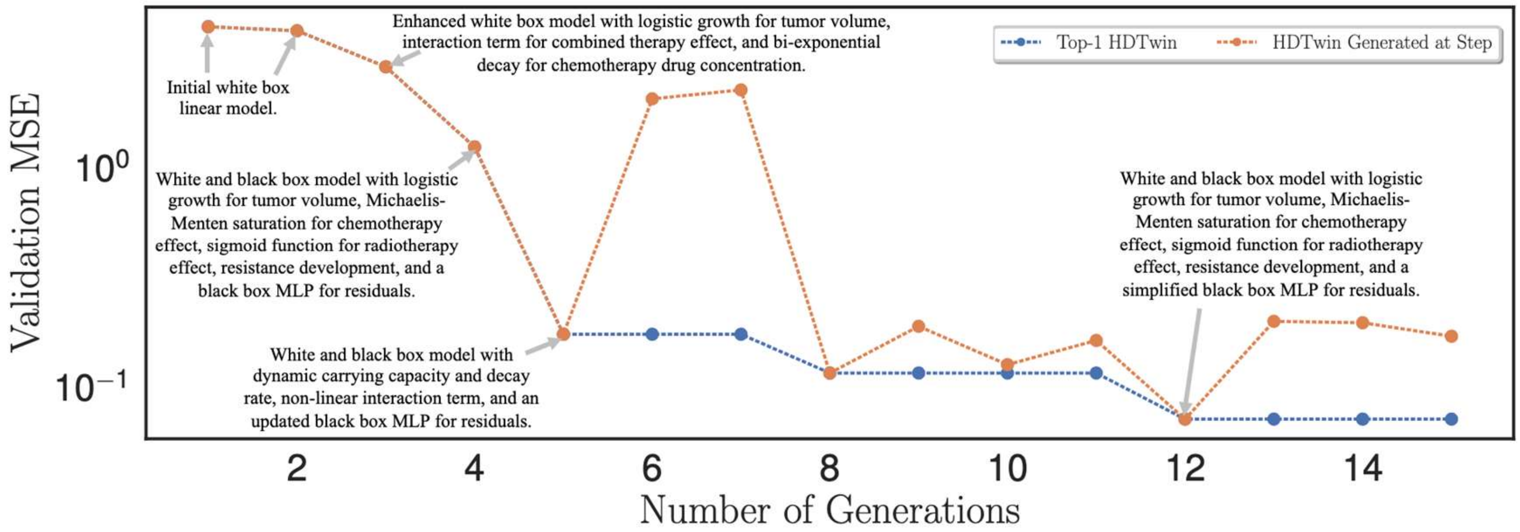
Model improvement

- The evaluation agent generates $H^{(g)}$ textual feedback based on the current pool of models $\mathcal{P}^{(g)}$ and evaluation instructions in $\mathcal{S}^{context}$

Empirical Investigation

Examined [P3] evolvability: the ability to easily update the model with minimal retraining

- **Pareto-front of the evolved HDTwin, underscoring efficiency in understanding and evolving the candidate models to achieve better HDTwins**
- **Also capable of incorporating expert feedback to steer model development!**



Engagement sessions: Inspiration Exchange

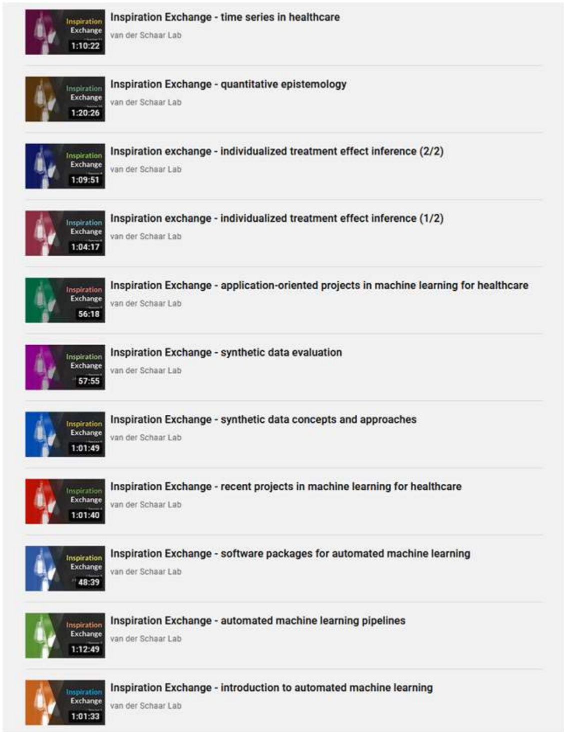
www.vanderschaar-lab.com/
→Engagement sessions
→Inspiration Exchange

Inspiration Exchange is a series of engagement sessions aiming to share ideas and discuss topics that will define the future of machine learning in healthcare. These events will target machine learning students, and will emphasize sharing of new ideas and development of new methods, approaches, and techniques.

As a lab, our purpose is to create new and powerful machine learning techniques and methods that can revolutionize healthcare. This doesn't happen in a vacuum. At inception, we are inspired by ideas and discussions; in implementation, we need connections, trust, and partnership to make a real difference.

While you can learn about our work at major conferences in machine learning or in our papers, we think it's a better idea to create a community and keep these conversations going. We're also aware that many people—both in healthcare and machine learning—have questions about what we do, and how they can contribute.

For more information about Inspiration Exchange—and to sign up to join in—please have a look at the sections below, and keep checking for new updates.



November 2024
4pm UK time/5pm CEST time
Digital Twins

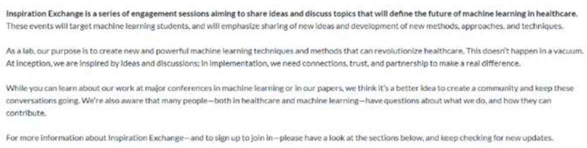
van_der_Schaar \LAB

vanderschaar-lab.com

UNIVERSITY OF CAMBRIDGE

Engagement sessions: Inspiration Exchange

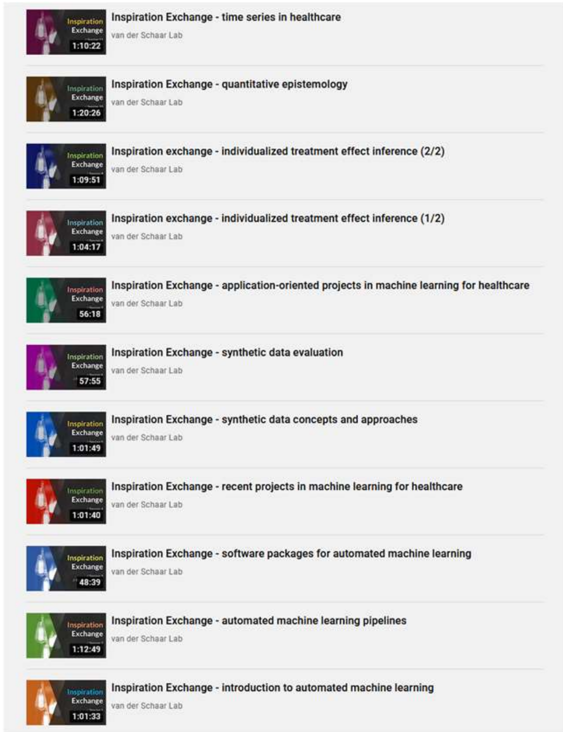
www.vanderschaar-lab.com/
→Engagement sessions
→Inspiration Exchange


Inspiration Exchange is a series of engagement sessions aiming to share ideas and discuss topics that will define the future of machine learning in healthcare. These events will target machine learning students, and will emphasize sharing of new ideas and development of new methods, approaches, and techniques.

As a lab, our purpose is to create new and powerful machine learning techniques and methods that can revolutionize healthcare. This doesn't happen in a vacuum. At inception, we are inspired by ideas and discussions; in implementation, we need connections, trust, and partnership to make a real difference.

While you can learn about our work at major conferences in machine learning or in our papers, we think it's a better idea to create a community and keep these conversations going. We're also aware that many people—both in healthcare and machine learning—have questions about what we do, and how they can contribute.

For more information about Inspiration Exchange—and to sign up to join in—please have a look at the sections below, and keep checking for new updates.



February 10, 2025
4pm UK time/5pm CEST time
Meta-Learning

March 2025
4pm UK time/5pm CEST time
Discovery from Data Using AI

van_der_Schaar
\ LAB

vanderschaar-lab.com

UNIVERSITY OF CAMBRIDGE