New frontiers in machine learning interpretability
Our research team

Fergus Imrie  Alan Jeffares  Alex Chan  Alicia Curth  Alihan Hüyük  Boris van Breugel  Dan Jarrett
Hao Sun  Jeroen Berrevoets  Jonathan Crabbé  Krzysztof Kacprzyk  Nicolas Huynh  Nabeel Seedat  Paulius Rauba
Sam Holt  Tennison Liu  Yangming Li  Yuchao Qin  Zhaozhi Qian

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

vanderschaar-lab.com
Machine learning interpretability is essential

- **Understanding**: Users need to understand, quantify and manage risk
- **Transparency**: Users need to comprehend how the model makes certain predictions
- **Trustworthiness**: Users can debug the model based on their knowledge
- **Discovery**: Users need to distil insights and new knowledge from the learned model
- **Avoid implicit bias**: Users need to be able to check whether the model does not learn biases
We need to go beyond interpretability of static prediction models.
What do clinicians want from an explanation?

www.vanderschaar-lab.com/making-machine-learning-interpretable-a-dialog-with-clinicians/
## 5 classes of explanation methods

<table>
<thead>
<tr>
<th>Explanation class</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>Feature-based</td>
<td>Provides the importance of each feature to model predictions</td>
</tr>
<tr>
<td>Example-based</td>
<td>Explains model predictions with reference to other examples</td>
</tr>
<tr>
<td>Concept-based</td>
<td>Explains model predictions with reference to a human-defined concept</td>
</tr>
<tr>
<td>Model-based</td>
<td>Explains model predictions via an auxiliary meta-model</td>
</tr>
<tr>
<td>Counterfactual</td>
<td>Explains model predictions by generating synthetic example(s) that are similar but with a different prediction</td>
</tr>
</tbody>
</table>
Today’s talk: Four types of interpretability

<table>
<thead>
<tr>
<th>Explanation class</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>Feature-based</td>
<td>Provides the importance of each feature to model predictions</td>
</tr>
<tr>
<td>Example-based</td>
<td>Explains model predictions with reference to other examples</td>
</tr>
<tr>
<td>Concept-based</td>
<td>Explains model predictions with reference to a human-defined concept</td>
</tr>
<tr>
<td>Model-based</td>
<td>Explains model predictions via an auxiliary meta-model</td>
</tr>
<tr>
<td>Counterfactual</td>
<td>Explains model predictions by generating synthetic example(s) that are similar but with a different prediction</td>
</tr>
</tbody>
</table>
Interpretability Resources

Overview of our lab’s work related to interpretability

vanderschaar-lab.com/
Research pillars
Interpretable ML

vanderschaar-lab.com
# Interpretability Resources

## Explainers

Different model architectures can require different interpretability models, or "Explainers". Below are all the explainers included in this repository, with links to their source code and the papers that introduced them. SimplEx, Dynamask, shap, and Symbolic Pursuit have a common python interface implemented for them for ease of implementation (see **Interface** above and **Implementation and Notebooks** below). But any of the other methods can also be implemented by using the code in the GitHub column of the table below.

<table>
<thead>
<tr>
<th>Explain</th>
<th>Affiliation</th>
<th>GitHub</th>
<th>Paper</th>
<th>Date of Paper</th>
</tr>
</thead>
<tbody>
<tr>
<td>CARs</td>
<td>van der Schaar Lab</td>
<td>CARs source Code</td>
<td>CARs Paper</td>
<td>2022</td>
</tr>
<tr>
<td>ITERpretability</td>
<td>van der Schaar Lab</td>
<td>ITERpretability Source Code</td>
<td>ITERpretability Paper</td>
<td>2022</td>
</tr>
<tr>
<td>Label-Free XAI</td>
<td>van der Schaar Lab</td>
<td>Label-Free XAI Source Code</td>
<td>Label-Free XAI Paper</td>
<td>2022</td>
</tr>
<tr>
<td>SimplEx</td>
<td>van der Schaar Lab</td>
<td>SimplEx Source Code</td>
<td>SimplEx Paper</td>
<td>2021</td>
</tr>
<tr>
<td>Dynamask</td>
<td>van der Schaar Lab</td>
<td>Dynamask Source Code</td>
<td>Dynamask Paper</td>
<td>2021</td>
</tr>
<tr>
<td>INVASE</td>
<td>van der Schaar Lab</td>
<td>INVASE Source Code</td>
<td>INVASE Paper</td>
<td>2019</td>
</tr>
<tr>
<td>SHAP</td>
<td>University of Washington</td>
<td>SHAP Source Code (pytorch implementation: Captum GradientShap)</td>
<td>SHAP Paper</td>
<td>2017</td>
</tr>
</tbody>
</table>

## Open Source Code

github.com/vanderschaarlab/Interpretability

## Implementation and Notebooks

This repository includes a common python interface for the following interpretability methods: SimplEx, Dynamask, shap, and Symbolic Pursuit. The Interface provides the same methods for each of the methods such that you can use the same python methods in your scripts to set up an explainer for each interpretability method. The methods that are:

- init: Instantiate the class of explainer of your choice.
- fit: Performs and trains for the explainer (This is not required for Shap explainers).
- explain: Provide the explanation of the data provided.
- summary_plot: Visualize the explanation.

There are also Notebooks in this GitHub repository to demonstrate how each create the explainer object. These explainers can be saved and uploaded into the interpretability Suite user interface.
SimplEx

SimplEx is a case-based interpretability method. It can work with either tabular or time series data. You can read more about it in the paper.

For clinically focused examples go to the bespoke SimplEx Demonstrator. And for further information, here is a video demonstration of the clinical SimplEx app.

Examples

Upload your own Explainer

Data type:
Tabular

Dataset:
ids

Model:
MLP

Test record:

0 20

Test record:

<table>
<thead>
<tr>
<th></th>
<th>sepal length (cm)</th>
<th>sepal width (cm)</th>
<th>petal length (cm)</th>
<th>petal width (cm)</th>
<th>Test Prediction</th>
<th>Test Label</th>
</tr>
</thead>
<tbody>
<tr>
<td>Test Record</td>
<td>7.700000</td>
<td>3.000000</td>
<td>6.000000</td>
<td>2.300000</td>
<td>2</td>
<td>2</td>
</tr>
</tbody>
</table>

Corpus:

<table>
<thead>
<tr>
<th>Corpus number</th>
<th>sepal length (cm)</th>
<th>sepal width (cm)</th>
<th>petal length (cm)</th>
<th>petal width (cm)</th>
<th>Example Importance</th>
<th>Corpus Prediction</th>
<th>Corpus Label</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>7.600000</td>
<td>3.000000</td>
<td>6.000000</td>
<td>2.000000</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>7.700000</td>
<td>3.600000</td>
<td>6.200000</td>
<td>2.300000</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>7.700000</td>
<td>2.600000</td>
<td>6.500000</td>
<td>2.300000</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>7.700000</td>
<td>3.600000</td>
<td>6.700000</td>
<td>2.500000</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>7.500000</td>
<td>3.600000</td>
<td>6.000000</td>
<td>2.500000</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

github.com/vanderschaarlab/Interpretability
Our Resources to go Further

Our Papers
vanderschaar-lab.com/interpretable-machine-learning/

Our Code
github.com/vanderschaarlab/Interpretability

vanderschaar-lab.com
Four types of interpretability

1. Feature-based interpretability
   - Static (global/personalized)
   - Time-series
   - Causal effect inference
From Global to Individual Feature Importance

Age, Gender, Diabetes, Hypertension, SBP, 

Black Box Predictive Model

Mortality due to Covid-19: 0.78
## Limitations of other methods for model interpretability

<table>
<thead>
<tr>
<th>Method</th>
<th>Feature importance</th>
<th>Individualized feature importance</th>
<th>Model-independent</th>
<th>Identifying the set of relevant features for each instance</th>
</tr>
</thead>
<tbody>
<tr>
<td>LASSO [Tibshirani, 1996]</td>
<td>✓</td>
<td></td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>Knock-off [Candes et al, 2016]</td>
<td>✓</td>
<td></td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>L2X [Chen et al, 2018]</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>LIME [Ribeiro et al, 2016]</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>SHAPE [Lundberg et al, 2017]</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>DeepLIFT [Shrikumar et al, 2017]</td>
<td>✓</td>
<td></td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>Saliency [Simonyan et al, 2013]</td>
<td>✓</td>
<td></td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>TreeSHAP [Lundberg et al, 2018]</td>
<td>✓</td>
<td></td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>Pixel-wise [Batch et al, 2015]</td>
<td>✓</td>
<td></td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td><strong>INVASE</strong> [Yoon, Jordon and van der Schaar, 2019]</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
</tbody>
</table>

INVASE discovers the number of relevant features for each instance.
Which features of an individual are relevant for a prediction?

Mary

Age, Gender, Diabetes, Hypertension, SBP, ....

Black Box Predictive Model

Mortality due to Covid-19: 0.78

INVASE

[Yoon, Jordon, vdS, ICLR 2019]

van_der_Schaar
LAB

vanderschaar-lab.com
• How can we learn individualized feature importance?

• Key idea: Use Reinforcement Learning (RL)
  • Make observations
  • Select “actions” on the basis of these observations
  • Determine “rewards” for these actions

  • Ultimately learn a policy which selects the best actions
    • i.e. actions that maximize rewards given observations

• We use the Actor-Critic approach to RL
• **Selector network (actor)** takes instances and outputs vector of selection probabilities.
• **Predictor network (critic)** receives the selected features, makes predictions and provides **feedback to the actor**.
Feature-based explanation – in medicine, we need to go beyond interpretability of static predictions

Time-series forecasting - Dynamask [ICML 2021]

Unsupervised learning methods – Label-free explainability [ICML 2022]

Causal effect inference – ITErpretability [NeurIPS 2022]
Time-series forecasting –
Do standard interpretability methods work?

NO! [Ismail et al., NeurIPS 2020]
How to take the time context into account? [Crabbé, vdS, ICML 2021]

**Challenge: Time context matters!**

Standard methods treat each input $x_{t,i}$ as a feature

$\Rightarrow$ Time dependency is ignored

**Dynamic Perturbation Operator**

Idea: perturb each $x_{t^*,i}$ by using neighbouring times:

$$
\pi(x_{t^*,i}; t^*, i) = \sum_{t=t^* - W_1}^{t^* + W_2} c_t(t^*, i) \times x_{t,i}
$$

$\Rightarrow$ Time dependency is integrated in perturbation
Dynamask [Crabbé, vdS, ICML 2021]

Input Time Series $X \in \mathbb{R}^{T \times d_x}$

Mask $M \in [0,1]^{T \times d_x}$

Perturbation Operator $\Pi_M$

Black Box $f$

Output $f(X)$

Perturbed Output $f \circ \Pi_M(X)$

Error $\mathcal{L}_e(M)$

Adapt Saliency Scores (Backpropagate)
We need “parsimonious” explanations

What do we mean by parsimonious?

Masks should not highlight more features than necessary

⟹ Feature selection

How to enable parsimony?

User selects desired fraction $a$ of most important features

Dynamask adds a regularization to enforce sparsity:

$$\mathcal{L}_a(M) = \|\text{vecsort}(M) - r_a\|^2$$
We need “congruous” explanations

What do we mean by congruous?

Masks should avoid quick time variations of the saliency

(Robustness)

How to enable congruity?

Dynamask adds a regularization to penalize saliency jumps over time:

\[ L_c(M) = \sum_{t=1}^{T-1} \sum_{i=1}^{d_x} |m_{t+1,i} - m_{t,i}| \]
Dynamask enables the saliency map to be “legible”

[Crabbé, vdS, ICML 2021]

How to we know if the “legibility” is achieved by an interpretability method?
We need a quantitative measure
We use Information Theory!

Introduced
Mask information
&
Mask entropy

van_der_Schaar
\LAB

vanderschaar-lab.com
Dynamask - Example

True saliency

Baseline saliency

Dynamask saliency

Example number 5

[Crabbé, vdS, ICML 2021]
Explaining *Unsupervised* Models

- **Unsupervised learning**: e.g. clustering/phenotyping
- **Self-supervised learning**: e.g. polygenic risk scores [VIME, NeurIPS 2020]

**Desiderata**
- Both feature and example-based explanations
- Understand and compare representations learned by different ML encoders
- Work with a variety of explanation methods (e.g. SHAP, Influence Functions)
- Work with a variety of neural network architectures (e.g. Autoencoder, SimCLR)

[Crabbé, vdS, ICML 2022]
Four types of interpretability

1. Feature-based interpretability
2. Example-based interpretability
Example-based explanations

- select particular instances of the dataset to explain the behavior of ML models
Personalized example-based explanations –

- select particular instances of a dataset selected by the user (a corpus) to explain the behavior of ML models

![Diagram](attachment:diagram.png)
Desiderata

Personalized explanations with reference to a freely selected set of examples, called the corpus

✓ Which corpus examples explain the prediction issued for a given test example?
✓ What features of these corpus examples are relevant for the model to relate them to the test example?
Our solution: **SimplEx** [Crabbe, Qian, Imrie, vdS, NeurIPS 2021]

- **SimplEx** – able to reconstruct the test latent representation as a mixture of corpus latent representations

- Novel approach (**Integrated Jacobian**) allows SimplEx to make explicit the contribution of each corpus feature in the mixture
  - Bridge between feature importance & example-based explanations

- **SimplEx** gives the user freedom to **choose** the corpus of examples to explain model predictions in a **user-centric** way

- **SimplEx** provides user-centric explanations for any ML methods on **diverse data** (tabular, imaging, time-series, multi-modal)
SimplEx: Problem set-up

Objective: explain the prediction for the example in terms of the corpus predictions.

- **Feature space** ($x$)
  - Example $x^1$, $x^2$, $x^3$
  - Corpus $C$

- **Latent space** ($\mathcal{H}$)
  - Representation: $g$
  - $h^1$, $h^2$, $h^3$

- **Prediction space** ($y$)
  - Black-box $f$
  - Linear $l$
  - $y^1$, $y^2$, $y^3$, $y$

**van_der_Schaar \ LAB**

vanderschaar-lab.com
SimplEx: Key idea

- Corpus C
- Example x
- Feature space
- Representation
- Linear
- Prediction space
Corpus Decomposition

• Find the best corpus decomposition of the example

\[ \hat{h} = \arg \min \| h - \tilde{h} \|_H \quad s.t. \quad \tilde{h} \in CH(C) \]
How to transfer corpus explanations in the input space?

Idea:
fix a baseline input \(x^0\) with representation \(h^0 = g(x^0)\)

\[
h - h^0 \approx \sum_{c=1}^{C} w^c (h^c - h^0)
\]

Compare each corpus member \(h^c\) to the baseline \(h^0\)

Understand total shift in latent space in terms of individual contributions from each corpus member
Integrated Jacobian & Projection

\[ j_i^c = \int_0^1 \frac{\partial g \circ \gamma^c}{\partial x_i}(t) \, dt \]

\[ p_i^c = \frac{\langle h-h^0, j_i^c \rangle}{\langle h-h^0, h-h^0 \rangle} \]
SimplEx: Feature sensitivity analysis

Feature space

Corpus $C

Example $x

$x^1$

$x^2$

$x^3$

Latent space

$\mathcal{H}$

$g$

Representation

$\mathbf{h}^1$

$\mathbf{h}^2$

$\mathbf{h}^3$

$\mathbf{h}$

Prediction space

$\mathbf{y}^1$

$\mathbf{y}^2$

$\mathbf{y}^3$

$\mathbf{y}$

$f$

$l$

Linear
SimplEx Explanations: Going beyond current interpretability

Expanding the picture: SimplEx unifies example and feature-based explanations

Enhancing the picture: SimplEx captures insights from the model’s latent space
Four types of interpretability

1. Feature-based interpretability
2. Example-based interpretability
3. Concept-based interpretability
What do we mean by *concept*?

A concept is

Defined by the user with concept positive and negative examples

A binary human annotation on the examples fed to ML models

Deducible from the ML model input features
Concept-Based Explainability

Image $x$ → Neural Network $f$ → # Label $y = \text{zebra}$

Is the prediction sensitive to the stripe concept?
Is the prediction sensitive to the prostate cancer grading system?

Concept-Based Explainability
Concept Activation Vectors (Kim et al, 2017)
Concept Activation Vectors (Kim et al, 2017)
Concept Activation Regions
(Crabbe, vdS, NeurIPS 2022)
Concept Activation Regions
(Crabbe, vdS, NeurIPS 2022)
CAR Formalism

Idea. Borrow the smoothness assumption from semi-supervised learning
CAR Formalism


A concept $c$ is well encoded in $\mathcal{H}$ if we can split $\mathcal{H} = \mathcal{H}^c \cup \mathcal{H}^{\neg c}$, where

1. The CAR $\mathcal{H}^c$ mostly overlaps with positives $\mathcal{P}^c$
2. The region $\mathcal{H}^{\neg c}$ mostly overlaps with negatives $\mathcal{N}^c$
3. If two $h_1, h_2 \in \mathcal{H}$ are close and in a high-density region, then $h_1, h_2 \in \mathcal{H}^c$ xor $h_1, h_2 \in \mathcal{H}^{\neg c}$

vanderschaar-lab.com
CAR Formalism

**Idea.** Borrow the smoothness assumption from semi-supervised learning

A concept \( c \) is well encoded in \( \mathcal{H} \) if we can split \( \mathcal{H} = \mathcal{H}^c \sqcup \mathcal{H}^{\neg c} \), where

1. The CAR \( \mathcal{H}^c \) mostly overlaps with positives \( \mathcal{P}^c \)

2. The region \( \mathcal{H}^{\neg c} \) mostly overlaps with negatives \( \mathcal{N}^c \)

3. If two \( h_1, h_2 \in \mathcal{H} \) are close and in a high-density region, then \( h_1, h_2 \in \mathcal{H}^c \) xor \( h_1, h_2 \in \mathcal{H}^{\neg c} \)

---

vanderson-schaar-lab.com
CAR Formalism

**Concept Density.** Define a signed density to measure the presence of a concept

\[ \rho^c(h) = \sum_{h' \in \mathcal{P}} \kappa(h, h') - \sum_{h' \in \mathcal{N}} \kappa(h, h') \]

**Concept Activation Region.** Use concept density with SVMs to infer the CAR \( \mathcal{H}^c \)

\[ \mathcal{H}^c = (s_κ^c)^{-1}(1) \]
CAR Formalism

**Concept Density.** Define a signed density to measure the presence of a concept

\[ \rho^c(h) = \sum_{h' \in \mathcal{P}^c} \kappa[\mathbf{h}, \mathbf{h}'] - \sum_{h' \in \mathcal{N}^c} \kappa[\mathbf{h}, \mathbf{h}'] \]

**Concept Activation Region.** Use concept density with SVMs to infer the CAR \( \mathcal{H}^c \)

\[ \mathcal{H}^c = (s_k^c)^{-1}(1) \]

**Global Explanation.** Measure the relationship between class \( k \) and concept \( c \) with score

\[ \text{TCAR}_k^c \equiv \frac{|\mathcal{D}_k \cap \mathcal{H}^c|}{|\mathcal{D}_k|} \]
CAR Formalism

Concept Density. Define a signed density to measure the presence of a concept

\[ \rho_c(h) = \sum_{h' \in \mathcal{P}_c} g[h, h'] - \sum_{h' \in \mathcal{N}_c} g[h, h'] \]

Concept Activation Region. Use concept density with SVMs to infer the CAR \( \mathcal{H}^c \)

\[ \mathcal{H}^c = (s_k^c)^{-1}(1) \]

Global Explanation. Measure the relationship between class \( k \) and concept \( c \) with score

\[ \text{TCAR}^c_k \equiv \frac{|g(D_k) \cap \mathcal{H}^c|}{|D_k|} \]

Feature Importance. Use any attribution method \( \alpha \) to assign concept importance to features

\[ \text{Importance}(x_i) \text{ for } c \equiv a_i(\rho_c \circ g, x) \]

vanderschaar-lab.com
CAR Advantages

What do we get by allowing $\mathcal{H}^c$ and $\mathcal{H}^{\neg c}$ to be nonlinearly separable?

More precision. CAR classifiers better capture how concepts are spread in $\mathcal{H}$

Better agreement with humans. TCAR scores better correlate with human annotations

Consistent feature importance. CAR feature importance captures concept associations
CAR Applications

- Doctors use 5 grades (5 concepts) to determine the likelihood of prostate cancer spreading
CAR Applications

- Doctors use 5 grades (5 concepts) to determine the likelihood of prostate cancer spreading
- DNNs implicitly encode prostate grading system (CAR classifiers with > 90% ACC)
- In DNNs representations, higher grade is associated with higher mortality
CAR Applications

- Doctors use 5 grades (5 concepts) to determine the likelihood of prostate cancer spreading
- DNNs implicitly encode prostate grading system (CAR classifiers with > 90% ACC)
- In DNNs representations, higher grade is associated with higher mortality
CAR – Other advantages not covered in this talk

- CAR explanations are invariant to latent isometries
- CAR explanations are robust to adversarial perturbations and background shifts
- CAR explanations can be used to understand abstract concepts discovered
- CAR explanations can be used with a wide variety of modalities (images, time series, tabular)
Four types of interpretability

1. Feature-based interpretability
2. Example-based interpretability
3. Concept-based interpretability
4. Discovering governing laws - Explicit-functions
Discover the governing models of medicine

- Discover powerful models!

- Why?
  Models are needed to
  ✓ understand variables, relationships, components
  ✓ experiment
  ✓ act

We need to go beyond feature & example interpretability
Discovery of governing models using ML

Our focus: governing equations – compact and closed-form equations

Benefits:
- Concise
- Generalizable
- Amenable to further analysis (e.g., identifying stable equilibria)
- Transparent
- Interpretable to human experts
Clinical Risk Prediction
[Alaa, Gurdasani, Harris, Rashbass & vdS, Nature MI, 2021]

Example: Predicting breast cancer risk survival (5 years)

Breast cancer data → NHS Predict/AutoPrognosis → Prediction

Nearly 1 million patients involved in the analysis.

NCRAS > 390,000

vanderschaar-lab.com
Clinical Risk Prediction
[Alaa, Gurdasani, Harris, Rashbass & vdS, Nature MI, 2021]

Example: Predicting breast cancer risk survival (5 years)

<table>
<thead>
<tr>
<th>Method</th>
<th>AUC-ROC</th>
</tr>
</thead>
<tbody>
<tr>
<td>PREDICT</td>
<td>0.75 ± 0.0033</td>
</tr>
<tr>
<td>AutoPrognosis</td>
<td>0.84 ± 0.0032</td>
</tr>
</tbody>
</table>

Breast cancer data → AutoPrognosis → Prediction

Metamodeling

vanderschaar-lab.com
Turning black boxes into white boxes using symbolic metamodelling [Alaa & vdB, NeurIPS 2019] [Crabbe, Zhang, vdB, NeurIPS 2020]

Metamodels
Operates on a trained machine learning model and outputs a symbolic formula describing the model’s prediction surface
Building transparent risk equations of black-box ML

Metamodel representation

\[ g(x) = G(x; \theta^*) \]

\[ \theta^* = \arg\min_{\theta \in \Theta} \ell(f(x), G(x; \theta)) \]

White-box model

\[ \alpha_1 X_1 + \alpha_2 X_2^2 + \alpha_3 X_1 X_2 \]
\[ \alpha_4 X_3^3 + \alpha_5 \log(X_4) \]

Black-box ML model

Model space (uninterpretable)

[Alaa & vdS, NeurIPS 2019]
[Crabbe, Zhang, vdS, NeurIPS 2020]

vanderschaar-lab.com
Interpretability using symbolic metamodeling in practice
[Alaa, Gurdasani, Harris, Rashbass & vdS, Nature MI, 2021]

Example: Predicting breast cancer risk survival (5 years)

Breast cancer data → AutoPrognosis → Prediction

Risk equations

\[ f(Age, ER, HER2, Tumor\ size, Grade, Nodes, Screening) \]

\[
\exp \left( \frac{Age}{5} - \log \left( \frac{Tumor\ size}{100} \right) + \frac{1}{10} \log(Nodes) \right) \times \\
\exp \left( \frac{ER \cdot Nodes}{20} + \frac{ER \cdot Tumor\ size}{23} \right)
\]

van_derschaar-lab.com
Interpretability using symbolic metamodeling in practice
[Alaa, Gurdasani, Harris, Rashbass & vdS, Nature MI, 2021]

Example: Predicting breast cancer risk survival (5 years)

Breast cancer data → AutoPrognosis → Prediction

Risk equations

<table>
<thead>
<tr>
<th>Method</th>
<th>AUC-ROC</th>
</tr>
</thead>
<tbody>
<tr>
<td>PREDICT</td>
<td>0.75 ± 0.0033</td>
</tr>
<tr>
<td>AutoPrognosis</td>
<td>0.84 ± 0.0032</td>
</tr>
<tr>
<td>Metamodel</td>
<td>0.83 ± 0.0020</td>
</tr>
</tbody>
</table>

vanderschaar-lab.com
\[ g(x) = \alpha_0 \text{Age} + \alpha_1 \text{BMI}^2 + \alpha_2 \text{Age} \cdot \text{BMI} + \alpha_3 \text{Age} \cdot \text{Gender} \\
\alpha_4 \text{Gender} \cdot (1 + \alpha_5 \text{Diabetes}) + \alpha_6 \log(\text{Age} \cdot \text{Diabetes} + 1) \]

\[ \frac{\partial g(x)}{\partial \text{Age}} = \alpha_0 + \alpha_2 \text{BMI} + \alpha_3 \text{Gender} + \frac{\alpha_6 \text{Diabetes}}{\text{Age} + 1} \]
## Discovery of governing equations using ML

<table>
<thead>
<tr>
<th>Explicit function</th>
<th>Implicit function</th>
<th>Ordinary differential equation</th>
<th>Partial differential equation</th>
</tr>
</thead>
<tbody>
<tr>
<td>$y = f(x)$</td>
<td>$f(x, y) = c$</td>
<td>$\frac{dx}{dt} = f(x, t)$</td>
<td>$\frac{\partial u}{\partial t} = f(u, x)$</td>
</tr>
</tbody>
</table>

- **Symbolic Metamodels**
  - [NeurIPS ’19, ’20]

- **D-Code**
  - [ICLR ’22]

- **D-CIPHER**
  - [archive]
Our Resources to go Further

Our Papers
vanderschaar-lab.com/interpretable-machine-learning/

Our Code
github.com/vanderschaarlab/Interpretability
Engagement sessions: Inspiration Exchange

Online engagement sessions for ML researchers in healthcare; themed presentations & Q&A

Engagement sessions

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

van_der_Schaar
LAB

vanderschaar-lab.com