Thesis for The Degree of Doctor of Philosophy

Interpretable Machine Learning for Prediction with Missing Values at Test Time

Lena Stempfle

Department of Computer Science and Engineering
Chalmers University of Technology
Gothenburg, Sweden, 2025

Interpretable Machine Learning for Prediction with Missing Values at Test Time

Lena Stempfle

ISBN 978-91-8103-246-8
Doktorsavhandlingar vid Chalmers tekniska högskola, Ny serie nr 5704.
ISSN 0346-718X

Department of Computer Science and Engineering
Division of Data Science and Engineering
Healthy AI Lab
Chalmers University of Technology
SE-412 96 Göteborg,
Sweden
Phone: +46(0)31 772 1000

Cover: Sparse, interpretable linear rule model for predicting diagnostic change in Alzheimer's disease. Each present feature contributes via a learned coefficient. The clinician weighs whether to include the missing FDG value, highlighting ambiguity in decision-making with missing data.

Printed by Chalmers Digitaltryck,
Gothenburg, Sweden 2025.

To my family.

Interpretable Machine Learning for Prediction with Missing Values at Test Time

Lena Stempfle

Department of Computer Science and Engineering
Chalmers University of Technology

Abstract

This thesis addresses the challenge of making interpretable predictions when feature values may be missing at deployment (at “test time”). Although imputation is a common strategy for handling missing values, it can obscure the relationship between inputs and predictions, thereby undermining interpretability and trust– especially in safety-critical domains such as healthcare. Alternatively, incorporating missingness indicators may introduce complexity and further reduce model interpretability. Tree-based models can handle missing values natively but are limited to specific model classes, potentially restricting flexibility and generalizability. To overcome these limitations, this thesis develops methods that (i) retain or improve predictive performance, (ii) handle missing values effectively at test time, and (iii) produce models that are simple and interpretable.

We first leverage missingness patterns by introducing *Sharing Pattern Submodels*, where a separate interpretable submodel is trained for each unique missingness pattern, with parameters shared across submodels via sparsity to enhance generalization. Next, we investigate training models that rarely require the values of missing (or imputed) features at test time. We introduce *MINTY*, a linear rule-based model that avoids imputation by allowing logical substitutions for missing features. We then generalize this idea through a *missingness-avoiding framework*, which extends to multiple model classes, including decision trees, sparse linear models, and ensembles, by incorporating classifier-specific regularization terms into their learning objectives to discourage reliance on missing values. To support the development of clinically valuable models, we conducted a clinician survey revealing that medical professionals favor models that natively handle missingness. Finally, we explore interpretable patient history representations for modeling policies in sequential clinical decision-making, shifting the focus from missingness to temporal modeling. Collectively, this work establishes methods for interpretable machine learning with test-time missingness, supported by both technical innovations and human-centered insights, to enable transparent and practical decision support.

Keywords

Machine learning, Interpretability, Missing Values, Healthcare, Decision Making

List of Publications

Appended publications

This thesis is based on the following publications:

[**Paper A**] **Lena Stempfle**, Ashkan Panahi, Fredrik D. Johansson, *Sharing pattern submodels for prediction with missing values.*
Proceedings of the AAAI Conference on Artificial Intelligence 37 (8), 9882-9890 (2023).

[**Paper B**] **Lena Stempfle**, Fredrik D. Johansson, *MINTY: Rule-based models that minimize the need for imputing features with missing values.*
International Conference on Artificial Intelligence and Statistics. PMLR 964-972 (2024).

[**Paper C**] **Lena Stempfle***, Anton Matsson*, Newton Mwai, Fredrik D. Johansson, *Prediction models that learn to avoid missing values.*
To appear in Proceedings of the 42nd International Conference on Machine Learning (ICML) PMLR 267 (2025).

[**Paper D**] **Lena Stempfle**, James Arthur, Julie Josse, Tobias Gauss, Fredrik D. Johansson, *Handling missing values in clinical machine learning: Insights from an expert study.*
Findings of the 4th Machine Learning for Health Symposium (ML4H), (2024).

[**Paper E**] Anton Matsson, **Lena Stempfle**, Yaochen Rao, Zachary R. Margolin, Heather J. Litman, Fredrik D. Johansson, *How should we represent history in interpretable models of clinical policies?*
Proceedings of the 4th Machine Learning for Health Symposium, PMLR 259:714-734, (2025).

*Equal contribution.

Other publications

The following publications were published during my PhD studies, or are currently under review. However, they are not included in this thesis due to content overlapping with that of the included publications or content not related to the thesis.

[**F**] Hákon Valur Dansson, **Lena Stempfle**, Hildur Egilsdóttir, Alexander Schliep, Erik Portelius, Kaj Blennow, Henrik Zetterberg and Fredrik D. Johansson for the Alzheimer's Disease Neuroimaging Initiative (ADNI). *Predicting progression and cognitive decline in amyloid-positive patients with Alzheimer's disease. Alzheimer's Research and Therapy 13, 151 (2021)*

[**G**] **Lena Stempfle**, Fredrik D. Johansson. *Learning replacement variables in interpretable rule-based models. ICML 3rd Workshop on Interpretable Machine Learning in Healthcare (IMLH) (2024)*

[**H**] Christoffer Ivarsson Orrelid*, Oscar Rosberg*, Sophia Weiner, Johan Gobom, Fredrik Daniel Johansson, Newton Mwai, **Lena Stempfle**. *Applying Machine Learning to High-Dimensional Proteomics Datasets for the Identification of Alzheimer's Disease Biomarkers. Fluids and Barriers of the CNS 22, no. 1 (2025): 23.*

[**I**] Martin Gillstedt*, **Lena Stempfle***, John Paoli, Fredrik Daniel Johansson, Sam Polesie. *Predicting melanoma in the adult Swedish population using machine learning on registry data. Under Review*

[**J**] **Lena Stempfle***, Lucia Huo*, Olawale Salaudeen, Marzyeh Ghassemi *Who Said LLMs Are Better? The Missing Human Baseline and Judge. Under Review*

[**K**] Sana Tonekaboni*, **Lena Stempfle***, Adibvafa Fallahpour*, Walter Gerych, Marzyeh Ghassemi *An Investigation of Memorization Risk in Healthcare Foundation Models. Under Review*

[**L**] Casandra Parent, **Lena Stempfle**, Nathan Zekarias, Sara Beery, Walter Gerych, Evan Fricke, Marzyeh Ghassemi *Machine learning to predict human health metrics using census-tract environmental data.*, *Under Review*

*Equal contribution.

Summary of contributions

The contributions to the appended papers by the author of this thesis are listed below:

[**Paper A**] Co-designed the study, implemented baselines and proposed methods, performed the experiments, managed the submission, and presented the findings at the respective conference. All authors contributed to analyzing the results, responding to reviewer feedback, and writing the manuscript.

[**Paper B**] Co-designed the study, implemented baselines and proposed methods, performed the experiments, jointly analyzed the results, managed the submission, and presented the findings at the respective conference. All authors contributed to analyzing the results, responding to reviewer feedback, and writing the manuscript.

[**Paper C**] The first two authors contributed equally to this work. Together, they co-designed the study, implemented both baseline and proposed methods, and conducted the experiments. All authors contributed to analyzing the results, writing the manuscript, and responding to reviewer feedback.

[**Paper D**] Co-designed the survey in collaboration with clinicians and academic partners, distributed the pilot and main studies with support from clinical collaborators, created visualizations, jointly analyzed the results, managed the submission, and wrote most of the manuscript. All authors contributed to revising the manuscript and responding to reviewer feedback.

[**Paper E**] Contributed to dataset preparation, supported the implementation of some baselines, performed parts of the experiments, helped analyze the results, and contributed to writing the manuscript and responding to reviewers.

Acknowledgments

Being a PhD student often feels like a rollercoaster, filled with highs, lows, and unforgettable experiences. What made this journey truly meaningful were the people who supported, challenged, and inspired me along the way. I'm deeply grateful to everyone who helped shape this path.

First and foremost, I want to thank my supervisor, Fredrik Johansson. Fredrik, this thesis would not have been possible without your unwavering guidance and belief in me, even when I doubted myself. I really admire your energy, your ability to balance so many projects, and your attention to detail, all while supporting each of us with tireless dedication. I look forward to continuing our collaboration.

To my co-supervisor, Devdatt Dubhashi, thank you for your thoughtful feedback and for the inspiring discussions. And to David Sands, I truly appreciated your kind and encouraging words in your role as examiner.

It's been a privilege to work in the Healthy AI Lab with Anton, Newton, Ahmet, Herman, Marc, Alessandro, and Adam. Thank you for your valuable discussions, and camaraderie during paper deadlines, study trips, and conferences. We built a truly supportive and healthy environment. I'm especially grateful to Anton and Newton, working with you taught me so much.

I'm deeply thankful to Julie Josse and Marzyeh Ghassemi for hosting me at Inria and MIT. Your mentorship and research have been truly inspiring and I was warmly welcomed to your groups. Julie, thank you for your ongoing feedback on the survey paper. Marzyeh, I'm excited for the work ahead and grateful for this opportunity. I also want to thank Finale Doshi-Velez for including me in your group meetings during my time in Boston. It was inspiring to see such a thriving, supportive environment in your research group.

Lovisa and Christopher, office days were always more fun with you around. I appreciated your thoughtful advice, the laughs, and the way we navigated PhD life, organized study trips, and planned research visits together. To Nicolas, Alec, and Ahmet, thank you for your thoughtful questions, genuine curiosity, and steady support in and beyond research. Kelsey, co-organizing the Women in ML workshop with you (and others) has been a joy. Your energy and passion for a cause that is so close to our hearts are inspiring.

A special thanks to Arthur James and Sam Polesie for generously sharing your clinical insights and patiently answering my many questions along the way. The real-world challenges you face in practice continue to motivate me to develop clinically meaningful ML models.

To all my co-authors and collaborators, thank you for every discussion, exchange, and shared effort along the way. I'm especially grateful for the time spent with my MIT collaborators: Wale, Lucia, Cassie, Adib, Vinith, and Sana. I've learned so much from you. Vinith, it was a real joy to share the office with you and Sana, your drive and dedication were a constant reminder that going the extra mile is always worth it. I also want to thank Martin at Sahlgrenska for our collaboration on a real-world clinical project.

I'm sincerely grateful to the administrative staff, Andrea, Clara, and Wolfgang, for keeping everything running so smoothly. Special thanks to Fatima, whose support and kind words so often extended well beyond work.

To Rocío Mercado, Niklas Kühl, and Adel Daoud, you have become true mentors. Your guidance and example as thoughtful academics helped me navigate key decisions, including the one to accept the postdoc at MIT.

Now outside of work, ...
I'm grateful for the friendships that brought joy and balance to everyday life. Chiara, Maria, Erik, Camilla, Clarissa, Pierluigi, Alessio, Anna, Anton, Emma, and Andreas–thank you for the laughs, the sports, and for making me feel at home in Gothenburg, especially during the pandemic. For those also doing PhDs, sharing the ups and downs of academia was a true source of comfort.

Eliott, I honestly can't tell if we've spent more time having great conversations or actually climbing in the hall. In any case, I really value our friendship and your consistent "yes" to spontaneous adventures. Verena, my dearest friend for so many years, thank you for your constant support, your care for others, and the memories we've made traveling together. To Theresa, Lisa, Kathi, Marlou and Wies–thank you for being my greatest cheerleaders. Even with the distance, it still feels like we're just living next door. To my book club–Hanna, Katharina, Anna, and Angie–thank you for the fresh perspectives you bring with every book. Our monthly discussions were true highlights.

My deepest gratitude goes to my parents, Elisabeth and Karl. Your love and support have been my foundation. I owe everything to you. To Johanna, my wonderful sister, your strength in every situation, your instinctive problem-solving and your many visits to Sweden meant the world. I'm proud to call you my sister.

And finally, to Amr, my partner and best friend. You've been my steady anchor through the most intense and chaotic moments, always helping me stay grounded. I'm deeply thankful for your love, your patience, and for sharing this journey, through our PhD years and everything beyond. Your quiet strength and the small things you do every day mean more to me than I can say. You make my life better in every way.

Finally, I gratefully acknowledge support from the Wallenberg AI, Autonomous Systems and Software Program (WASP), funded by the Knut and Alice Wallenberg Foundation, including grants for international research visits. Thanks also to the Barbro Osher Endowment Foundation for supporting my research exchange. Computations were enabled by SNIC and NAISS resources, partially funded by the Swedish Research Council (grant no. 2018-05973).

Contents

Part I

Introductory Chapters

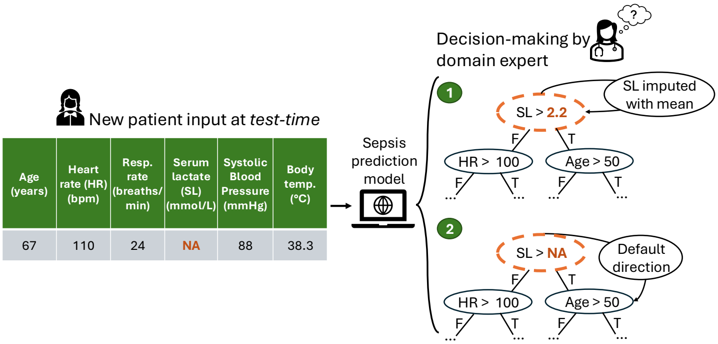
Chapter 1

Introduction

Most machine learning methods assume that all relevant input features will be available at deployment (at "test time") (Little & Rubin, 2019). However, in practice, missing values are common: they arise whenever a required observation is missing, underscoring the imperfect and evolving nature of data collection in real-world settings (Emmanuel et al., 2021). This issue frequently arises in tabular data across scientific domains (Chourib, 2025). In healthcare, missing values in data sets may result from incomplete patient records, missed appointments, or delays in diagnostic tests (Marston et al., 2010; Wells et al., 2013); in industrial settings, it can be due to sensor failures, communication errors, or data corruption (Dasu & Johnson, 2003; Ehrlinger et al., 2018). These missing values are not rare anomalies; they are persistent, arising from diverse and domain-specific causes (Schafer & Graham, 2002). However, in real-world settings, missing values often arise not only during training but also at test time, posing challenges for predictive performance and reliability. Users must either assign values to missing inputs or rely on the model's internal handling, both of which can affect its behavior and interpretability. This is particularly relevant in settings where humans provide input and interpret the output, such as clinical decision support systems or risk scoring tools.

To illustrate this, consider the example shown in Figure 1.1: A 67-year-old woman arrives at the emergency department with weakness and fatigue. A machine learning model is used to estimate her risk of developing sepsis within six hours. At triage, four vital signs are available, but serum lactate, a key biomarker for sepsis, is missing because it has not yet been ordered, which is common early in care. Faced with test-time missingness, the clinician must decide how to handle the uncertainty: Should they impute, meaning fill in a plausible value for the missing entry, leave it blank, or rely on the model's internal handling of missingness? This scenario exemplifies the real-world complexity of making interpretable predictions when key values are missing at test time.

A common strategy is to impute missing values before prediction, thereby restoring the input to a fully observed form (Rubin, 1976). Simple techniques like zero, mean, or median imputation are computationally efficient but often



Age (years)	Heart rate (HR) (bpm)	Resp. rate (breaths/ min)	Serum lactate (SL) (mmol/L)	Systolic Blood Pressure (mmHg)	Body temp. (°C)
67	110	24	NA	88	38.3

Figure 1.1: Early sepsis prediction with missing serum lactate (SL = NA): NA indicates that the value is not available at test time. (1) Impute the missing SL value using a population average (e.g., 2.2 mmol/L), or (2) follow a default direction learned during training, such as the majority decision path in a tree-based model (e.g., follow the right side of the tree without replacing the missing lactate value). In practice, clinicians may rely on these model outputs to guide urgent interventions. However, imputing with an average can obscure important individual variation, as it is chosen without consideration of the outcome, e.g., a patient with truly elevated lactate may be misclassified as low risk. Default paths, despite being informed by the smallest prediction error on the training data, still bypass personalized information and may lead clinicians to overlook patients who deviate from the norm.

fail to reflect the true data distribution (Little & Rubin, 2019). More advanced methods, such as Multiple Imputation by Chained Equations (MICE) (Van Buuren & Groothuis-Oudshoorn, 2011), iteratively model each variable conditional on the others, generating multiple completed datasets to capture the uncertainty of the imputation. These methods may introduce bias and rely on assumptions such as Missing-at-Random (MAR) (Pedersen et al., 2017; Rubin, 1976). In this example, clinicians might impute a missing serum lactate value using the population average. However, if lactate is typically measured only when sepsis is suspected, its absence carries clinical meaning. Since the population used for imputation may reflect only a specific subset of patients, such as those with suspected sepsis, the resulting estimate can be misleading. Yet imputation may be suboptimal when missingness arises at prediction time (Le Morvan, Prost et al., 2020). Indeed, in such settings, the Bayes-optimal predictor depends not only on the observed inputs but also on the missingness itself (Le Morvan et al., 2021).

Alternatively, one can include *missingness indicators* in the model (Rubin, 1976), which are additional binary features denoting whether a value is missing or observed. For example, if a test like lactate was not ordered, the corresponding indicator would be set to 1 (missing); if ordered, it would be 0 (observed). These indicators can be used with a wide range of models and help capture patterns in the missingness structure (Van Ness et al., 2023). However, they do not provide any information about the actual missing values themselves.

Beyond imputation and indicator-based strategies, a small class of methods avoids both by handling missing values *natively*. For example, XGBoost (T. Chen & Guestrin, 2016) and other tree-based models can learn "default" decision paths during training, determining how to route instances when a feature is missing. This allows the model to operate directly on incomplete data. In the example shown in Figure 1.1, a decision tree trained to predict sepsis risk could learn to send patients with missing lactate values along the right-side path. However, native handling of missing values remains limited to a few specialized models, and the decision paths are often derived from approximate or ad hoc learning procedures. Although this enables prediction without imputation, heuristics can misrepresent underlying data patterns and introduce bias. Additionally, because the logic behind these paths is not always transparent, it can be difficult to interpret or validate the model's behavior, particularly in critical applications like healthcare. Overall, it is crucial to design models that prioritize the information actually observed at test time, rather than relying heavily on imputation or learned defaults that may obscure uncertainty.

Since the goal of predictive models is often to support actionable decision-making, it is essential to focus not only on accuracy but also on interpretability (Vellido, 2020). By interpretability, we mean the ability to understand and reason about how a model arrives at its predictions in ways that are comprehensible to humans (Biran & Cotton, 2017; Kim, Khanna & Koyejo, 2016). For instance, in healthcare, this is essential for enabling domain experts to assess trust, identify errors, and consider model outputs in clinical decision-making (Ahmad, Eckert & Teredesai, 2018; Bénard et al., 2021a; Liu, Kumara & Reich, 2021; Ustun & Rudin, 2019). Current interpretable machine learning (IML) models, such as logistic regression, decision trees, and rule-based models (Molnar & Freiesleben, 2024) struggle to handle missing values at the point of prediction. Like many other machine learning methods, they typically rely on imputation strategies or default handling mechanisms learned during training. For instance, rule-based models, despite being interpretable, face this challenge: if a rule relies on a missing feature, the model must either ignore the rule or make assumptions that may not hold, reducing both reliability and interpretability (as seen in clinical risk scores (Afessa et al., 2005)).

This thesis is guided by two central questions: *How can we achieve high predictive accuracy in the presence of missing values at test time, and how can we ensure that the resulting models remain simple and interpretable?*

This thesis aims to develop methods that (i) retain or improve predictive performance, (ii) effectively handle missing values at test time, and (iii) yield models that are simple and interpretable. We pursue this through two complementary methodological directions. First, we leverage pattern missingness and enforce parameter sharing through sparsity in pattern coefficient specializations via regularization. Second, we reduce the reliance on frequently missing inputs through regularization, allowing the model to make accurate predictions despite missing values at test time. We complement these contributions by conducting an expert survey, which shows that clinicians prefer models that handle missingness directly at test time, prioritizing interpretability over complex imputation strategies or the use of missingness indicators. Finally, we turn to the challenge of capturing temporal patterns in patient data and evaluating interpretable representations of patient histories to support clinical policy learning, beyond a particular focus on the missing-value setting considered earlier.

The main contributions of the thesis can be summarized as follows:

- Paper A (Stempfle, Panahi & Johansson, 2023) introduces Sharing Pattern Submodels (`SPSM`), a method that leverages the structure of missingness by learning both global parameters shared across all data and pattern-specific parameters for groups of samples with the same observed features. This is achieved through sparsity-inducing regularization, encouraging efficient parameter sharing while adapting to the missingness pattern. `SPSM` handles missing values at test time while maintaining or improving predictive performance compared to baselines that rely on imputation. This method enhances interpretability by producing concise model descriptions and is theoretically proven to lead to consistent estimation.

- Paper B (Stempfle & Johansson, 2024b) proposes `MINTY`, a rule-based learning method that avoids reliance on missing values by leveraging disjunctions between variables that can replace each other. This results in a sparse linear rule model that balances interpretability and achieves comparable predictive performance to baselines that rely on imputation or missingness indicators. This paper extends the workshop version presented by Stempfle and Johansson (2024a).

- Paper C (Stempfle et al., 2025) extends the idea from Paper B of reducing reliance on missing values by introducing a general *missingness-avoiding* (MA) machine learning framework that minimizes the need to access missing (or imputed) features at test time. We develop tailored MA algorithms for decision trees, tree ensembles, and sparse linear models by incorporating classifier-specific regularization terms into their learning objectives. Empirical experiments demonstrate that these models effectively reduce reliance on features with missing values while maintaining predictive performance compared to their unregularized counterparts across various datasets.

- Paper D (Stempfle et al., 2024) surveyed 55 clinicians from 29 French trauma centers to examine their interaction with interpretable ML models to predict hemorrhagic shock with missing values. Our findings show that clinicians prefer models that natively handle missing values over imputation-based approaches, aligning better with their decision-making process.

- Finally, we shift the focus from handling missing values to modeling time-series data by exploring how to efficiently represent patient histories in an interpretable way when learning clinical policies. Paper E (Matsson et al., 2025) analyzes learned policies across patient subgroups, critical states, and treatment stages, showing that interpretable sequence models using learned representations perform comparably to black-box models, while models relying solely on hand-crafted representations require minimal historical context to remain competitive.

During my PhD, I have co-authored the following publications, which are not included in this thesis: Dansson et al. (2021), Ivarsson Orrelid et al. (2025) and Stempfle and Johansson (2024a).

The thesis is structured as follows. Chapter 2 explores learning with missing values, beginning with an overview of prediction challenges when missingness occurs not only during but also at test time. We introduce missingness mechanisms and common strategies for handling missing values, highlighting their limitations. Chapter 3 defines interpretable machine learning in the context of this thesis and highlights its importance in safety-critical applications. It presents interpretable-by-design techniques such as sparse linear models, decision trees, and rule-based methods used in the proposed algorithms. Evaluation methods for interpretability are briefly discussed. Chapter 4 brings the background together by outlining three key challenges in supervised prediction under test-time missingness, showing why current methods fail to provide the interpretability needed for trust and accountability in high-stakes decision-making. Chapter 5 summarizes the papers on which this thesis is based, followed by Chapter 6, which presents conclusions and directions for future research. The original papers are included in Part II, reformatted for consistency but otherwise unchanged.

Chapter 2

Learning with missing values

In this section, we introduce the mathematical notation for prediction with missing values and outline the challenges of handling missingness at test time. We then present missingness mechanisms and briefly discuss missingness patterns. Finally, we review common strategies for handling missing values during both training and testing, highlighting their assumptions and limitations. This background provides the necessary context for understanding the methodological choices and contributions of the subsequent papers.

2.1 Prediction with missing values at test time

In supervised learning, the goal is to predict an outcome $Y \in \mathcal{Y}$ from an input vector $X = (X_1, \ldots, X_d) \in \mathcal{X} \subseteq \mathbb{R}^d$, where the value of any of the features X_j may be missing, either *at training time* or *at test time*.

Missingness is indicated by a binary mask $M = (M_1, \ldots, M_d)^\top \in \{0,1\}^d$ applied to a complete feature vector X^*, such that:

$$X_j = \begin{cases} X_j^* & \text{if } M_j = 0 \\ \texttt{NA} & \text{if } M_j = 1 \end{cases}$$

as introduced in (Little & Rubin, 2019; Rubin, 1976). To learn a predictor, we are given a training dataset $D = \{(x_i, m_i, y_i)\}_{i=1}^n$, drawn from a distribution p, assumed to be the same for both training and test data. Here, $x_i = (x_{i1}, \ldots, x_{id}) \in (\mathbb{R} \cup \{\texttt{NA}\})^d$ is a partially observed feature vector, $m_i = (m_{i1}, \ldots, m_{id}) \in \{0,1\}^d$ is the corresponding missingness mask (with $m_{ij} = 1$ if x_{ij} is observed and $m_{ij} = 0$ if x_{ij} is missing), and $y_i \in \mathbb{R}$ is the outcome.

Our objective is to learn a function h that minimizes the expected loss over the distribution p:

$$R(h) := \mathbb{E}_{D \sim p}[L(h(X), Y)], \tag{2.1}$$

where $h : (\mathbb{R} \cup \{\texttt{NA}\})^d \times \{0,1\}^d \to \mathbb{R}$ and L is a suitable loss function (e.g., squared loss or logistic loss) (Hastie, Tibshirani & Friedman, 2009).

The Bayes-optimal predictor minimizes the conditional risk:

$$h^*(x, m) = \operatorname*{argmin}_{a \in \mathcal{A}} \mathbb{E}[L(a, Y) \mid X = x, M = m], \tag{2.2}$$

where $\mathcal{A}$ is the set of possible predictions (e.g., $\mathbb{R}$ for regression or $\{0, 1\}$ for classification), and a denotes a candidate prediction. For squared loss, the Bayes-optimal predictor corresponds to the conditional expectation, i.e., $h^*(x, m) = \mathbb{E}[Y \mid X = x, M = m]$.

In practice, we learn h by minimizing the empirical risk:

$$\hat{R}(h) = \frac{1}{n} \sum_{i=1}^{n} L(h(x_i, m_i), y_i). \tag{2.3}$$

The function h in Equation 2.3 may be implemented as a composition of an imputation step followed by a prediction model, or as a model that directly incorporates the missingness pattern M. We discuss classical approaches to learning with missingness in Section 2.3.

2.2 Missingness mechanisms

We introduce missingness mechanisms, as they guide how to handle missing values by informing assumptions about the data-generating process and influence modeling or imputation strategies (Little & Rubin, 2019).

The missingness mechanisms are commonly categorized into three types by Rubin (1976): *missing completely at random (MCAR)*, where the probability of missingness is independent of the data; *missing at random (MAR)*, where the probability that a variable X_j is missing depends only on the observed data; and *missing not at random (MNAR)*, where missingness may depend on the unobserved value of the variable itself or other unobserved factors. Note that under MCAR, while missingness is independent of the data, correlations between missingness indicators (e.g., M_j and $M_{j'}$) can still exist if they stem from external, data-independent sources.

The mechanisms behind missingness are defined by the relationship between missing and observed values, which addresses the following: *What is causing the data to be missing?*

The notation used in this thesis will be close to Rubin (1976) and Schafer (1997), with some exceptions to ensure consistent notation throughout the thesis. As defined before, the observations of X (both observed and missing) can be partitioned into X^o, indicating the observed features, and X^m, representing the missing feature part, such that $X = (X^o, X^m)$. As defined in the previous section, M is the missingness mask and is assumed to be known.

The missingness mechanisms express the probability that a set of values is missing given the values taken by the observed and missing observations. It can be denoted by: $\mathbb{P}(M \mid X^o, X^m)$.

We define MCAR as the case where the probability of missingness is independent of both observed and unobserved measurements:

$$\mathbb{P}(M \mid X^o, X^m) = \mathbb{P}(M).$$

This implies that the missingness is unrelated to the data values, meaning any pattern of missing data arises purely by chance. While the missingness does not depend on the data, dependencies between missingness indicators themselves (e.g., M_j and $M_{j'}$) are still possible—for instance, due to a lab device failure affecting several variables simultaneously. A typical example of MCAR is a clinical trial where patients are randomly excluded due to scheduling conflicts, unrelated to any of their medical characteristics. Another example is random dropout in survey data due to unrelated technical issues (van Buuren, 2018).

MAR is a broader class than MCAR, allowing the probability of missingness to depend on observed variables, but not on the missing values themselves or other unobserved factors $\mathbb{P}(M \mid X^o, X^m) = \mathbb{P}(M) \mid X^o$. For example, consider a university that surveys its alumni to gather information on their current employment and income levels. Suppose alumni working in a particular sector are less likely to disclose their income, but they do report their sector of employment. In this case, the missingness pattern is related to an observed variable (sector), making it MAR rather than MCAR (van Buuren, 2018). MAR is more general and more realistic than MCAR. Modern missing value methods generally start from the MAR assumption.

If neither MCAR nor MAR holds, we speak of missing not at random. MNAR means that the probability of missing varies for reasons that are unknown to us. The reason why a variable is missing still depends on the unobserved variables themselves. For example, in clinical settings, patients with more severe depression symptoms may be less likely to attend follow-up visits due to fatigue or lack of motivation. As a result, depression severity scores may be missing precisely when symptoms are worst, leading to a biased outcome distribution that underrepresents the most affected individuals. Another example of MNAR arises in public opinion research, where individuals with weaker opinions may be less likely to respond, making their views underrepresented. MNAR is the most difficult case to handle.

Identifying the underlying missingness mechanism is crucial because each requires different handling strategies. Although simple imputation may suffice under MCAR or MAR, it can lead to significant bias under MNAR (Sperrin et al., 2020). However, the true mechanism is rarely known and cannot be directly verified from observed data alone. MCAR and MAR can sometimes be distinguished using statistical tools such as Little's MCAR test (Little, 1988), which assesses whether missingness is independent of the data. In practice, missingness is often suspected to be MNAR. Recent work seeks to better understand these mechanisms, identify their causes, and mitigate the risks introduced by imputation algorithms using high-accuracy glass-box explainable booster machines (EBMs) (Z. Chen et al., 2023). Their focus lies on helping users detect, understand, and mitigate missing value issues, rather than automatically fixing datasets. However, identifying the type of missingness mechanism is notoriously difficult, as it is typically not testable from observed data alone.

Several works have investigated this challenge by proposing statistical tests for specific assumptions—especially MCAR (Little, 1988); developing modeling approaches to account for or infer MNAR structure (Mohan, Pearl & Tian, 2013); and exploring the fundamental limits of identifiability (Mealli & Rubin, 2015; Mohan & Pearl, 2021).

Remark on missingness patterns. Missingness patterns describe the structure of observed and missing entries in the previously defined missingness mask M (Little, 1993). It is important to distinguish between *missingness patterns* and *missingness mechanisms*: mechanisms explain *why* values are missing, but not *where* missing values occur, such as empty cells or invalid entries. Missingness patterns arise in data-generating processes with structural reasons for why certain variables are measured. As a result, samples in the datasets can be grouped by recurring patterns of observed and missing variables (Little, 1993). For instance, samples may contain only a subset of observed variables when different measurements are systematically taken using different sensors or instruments, such as in hospital settings.

Visualization plays a central role in exploring missingness patterns, especially in large datasets where structure is difficult to detect (Josse & Husson, 2012). Case-variable matrix plots are commonly used to display one column per variable, highlighting where values are missing. Tools like `visdat` and the scalable `visna` plot in R enable such visualizations (Unwin, 2020). These methods group rows by recurring missingness patterns, helping structure emerge even in high-dimensional settings such as genomics or EHRs. Additional R tools, such as the *missMDA package* (Josse & Husson, 2016), combine visualization with imputation techniques for exploratory multivariate analysis. The *CRAN Task View* on missing values (Josse et al., 2025) provides an overview of available tools for visualization, modeling, and imputation. Together, these visual summaries help assess whether missingness is random or concentrated in specific subsets (Molenberghs et al., 2014), guiding downstream modeling decisions.

Some prediction methods explicitly exploit structured missingness patterns to improve learning. For example, Mercaldo and Blume (2020) fit separate models per missingness pattern, though this may lead to inefficient data use. Building on this, the algorithm in Paper A introduces sparsity-based parameter sharing across pattern-specific models. It leverages missingness patterns by allowing each to use a sparse subset of parameters drawn from a shared model, capturing both common structure and pattern-specific effects. Pattern mixture models (Little, 1993) also explicitly model the missingness pattern M by decomposing the joint distribution $P(Y, X, M)$, allowing Y to depend on M; however, they often require strong assumptions for identifiability. Similarly, Zaffran et al. (2023) propose a latent variable model that disentangles missingness-related artifacts from signal, explicitly incorporating structured missingness into representation learning.

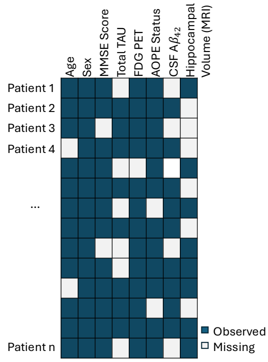


Figure 2.1: Simplified visualization of missingness patterns showing the presence (blue) and absence (gray) of eight clinical features across n patients. Features include cognitive assessments (MMSE), biomarkers (Total Tau, CSF Aβ_{42}, FDG PET), demographics (Age, Sex), genetic risk (APOE status), and brain imaging (MRI Hippocampal Volume). Visualizing missingness helps identify dependencies between features, detect features frequently missing alone, and potentially inform modeling decisions such as imputation strategies.

2.3 Classical strategies for learning with test-time missingness

This section presents commonly used strategies for making predictions with missing values in supervised learning. These include: *complete-case analysis* (Janssen et al., 2009); *impute-then-predict* approaches, where a standard machine learning model is trained on imputed data (Rubin, 1976); the use of *missingness indicators* to capture informative missingness patterns (Little & Rubin, 2019); and models that handle missing values *natively*, such as XGBoost (T. Chen & Guestrin, 2016).

One of the simplest approaches to handling missing values is complete case analysis (also known as list-wise deletion), where rows with any missing values are excluded prior to model fitting. This method can perform adequately when data are MCAR or when the extent of missingness is negligible (Janssen et al., 2010; Knol et al., 2010). However, these assumptions are often violated in practice, which can introduce bias and lead to loss of statistical power (Janssen et al., 2009). Moreover, complete case analysis is inefficient in terms of data usage, especially problematic in domains like healthcare, where data are often scarce (Z. Chen et al., 2023). It also fails to leverage the potential informativeness of the missingness itself, and, critically, it does not address missing values that may occur at test time. For these reasons, we do not consider this method further in this work.

2.3.1 Impute-then-regress

A widely used strategy for handling missing values in supervised learning is the *impute-then-regress* approach. It first imputes the missing entries in the partially observed feature vector $x \in (\mathbb{R} \cup \{\texttt{NA}\})^d$ using a function $\phi : (\mathbb{R} \cup \{\texttt{NA}\})^d \times \{0,1\}^d \to \mathbb{R}^d$, yielding a completed vector $x^I = \phi(x,m) \in \mathbb{R}^d$.

The prediction function $f : \mathbb{R}^d \to \mathbb{R}$ is then trained on the imputed data. Given a chosen, fixed imputation function ϕ, the learning objective becomes:

$$\arg\min_{f}; \mathbb{E}_{D \sim p}\left[L\left(f\left(\phi(X,M)\right), Y\right)\right], \tag{2.4}$$

where L is the same loss function as before (e.g., squared error or logistic loss), and $\phi(X,M)$ denotes a transformation of the incomplete data. Note that the objective is *not* optimized jointly with respect to f and ϕ; ϕ is fixed a priori and only f is learned from the data. This formulation typically assumes that the missingness mechanism is MAR, allowing us to rely on observed data during training (Carpenter et al., 2023; Seaman et al., 2013) and that the imputation function generalizes well to the test-time missingness patterns. The MAR assumption is widely used because it makes model estimation feasible without modeling the missingness mechanism directly. However, it cannot be verified from the observed data alone, unlike MCAR, which is testable via statistical procedures such as Little's test (Little, 1988). Thus, MAR is often a practical rather than verifiable assumption. Even though powerful, this impute-then-regress approach under the MAR assumption may be suboptimal under distribution shifts or when the missingness pattern itself is informative (Le Morvan, Prost et al., 2020). Josse et al. (2019) reviewed approaches to handling missing values in supervised (non-deep learning) settings and showed that, under certain assumptions, even simple imputation strategies like mean imputation can be consistent. Complementing this, Le Morvan, Prost et al. (2020) studied linear predictors with missing values in covariates and demonstrated that the optimal predictor may no longer be linear. They further showed how constant imputation of each feature can be optimized with respect to the model loss.

For an extensive review of imputation strategies, see Shadbahr et al. (2023). Imputation techniques can be broadly categorized into *single imputation methods*, where each missing value is imputed once, and *multiple imputation methods*, which create several completed datasets to reflect uncertainty about the missing data. Single imputation methods often fill in missing values with deterministic estimates, such as zeros or means, which tend to underestimate variability and can bias results, even under MCAR (Jamshidian & Schott, 2007; van Buuren, 2018). Regression-based imputation improves accuracy by leveraging observed relationships, but still distorts uncertainty. Stochastic variants, in contrast, incorporate random noise to better reflect the underlying data distribution (Buck, 1960). In contrast, multiple imputation (e.g., MICE) generates several completed datasets, capturing uncertainty through pooled estimates (de Goeij et al., 2013; Van Buuren, 2007). Other state-of-the-art imputation methods include MissForest (Stekhoven & Bühlmann, 2012), KNN Imputer (Troyanskaya et al., 2001), matrix completion techniques (Mazumder, Hastie & Tibshirani, 2010;

Yu, Rao & Dhillon, 2016), and generative approaches such as deep generative models (Mattei & Frellsen, 2019; Yoon, Jordon & Schaar, 2018).

While our proposed methods aim to reduce reliance on imputed values, some level of imputation remains necessary and is incorporated in different ways. In Paper A, we obtain shared coefficients through a main model that requires imputed inputs, for which we use both zero imputation and MICE(Van Buuren & Groothuis-Oudshoorn, 2011). Papers B and C focus more explicitly on minimizing dependence on imputed features, yet still rely on imputed data (zero, mean, or mode imputation) to enable comparison and integration within standard pipelines. To benchmark our approaches, we compare them against commonly used imputation-based baselines, including logistic regression, LASSO (Tibshirani, 1996), decision trees, and MLPs (Rumelhart, Hinton & Williams, 1986), using several imputation strategies. This allows us to assess whether comparable predictive performance can be achieved while limiting or avoiding reliance on imputed values. In Paper E, the absence of prior history necessitates imputation, especially since the behavior policy model requires a fixed-size input. Here, missing values are primarily imputed at the patient level using the last observation carried forward, followed by mean or frequent-category imputation.

2.3.2 Missingness indicators

A common strategy for handling missing values in supervised learning is to use the input with *missingness indicators*, where a binary mask $M \in \{0,1\}^d$ denotes whether each feature is missing. The model then learns a function $f(X^I, M)$ that captures the conditional distribution $\mathbb{E}[Y \mid X^I, M]$, where X^I is a simply imputed version of X (for example, zero- or mean-imputation). Then, when trained on data with similar missingness patterns, such models can better adapt to test-time missing values—particularly when the missingness itself carries predictive information (i.e., is informative) (Rubin, 1976).

If the missingness mechanism is *informative*, that is, if $P(M \mid Y) \neq P(M)$, then the missingness pattern M contains predictive information about the target Y that is not captured by the imputed features X^I alone. In this setting, conditioning on both X^I and M can yield more accurate predictions. Formally, let $f_1^*(X^I) = \mathbb{E}[Y \mid X^I]$ and $f_2^*(X^I, M) = \mathbb{E}[Y \mid X^I, M]$ denote the Bayes optimal predictors for each set of conditions. Then:

$$\mathbb{E}\left[(Y - f_1^*(X^I))^2\right] > \mathbb{E}\left[(Y - f_2^*(X^I, M))^2\right]. \tag{2.5}$$

This reflects the fact that missingness patterns can help explain part of the variability in Y, and ignoring them may result in suboptimal models. Considering the mutual information $I(Y; M \mid X^I) > 0$, omitting M means discarding a relevant signal. This phenomenon, known as *informative missingness* (Rubin, 1976), implies that even perfect recovery of the missing entries in X cannot substitute the predictive value carried by M. For example, in clinical data, the presence or absence of a measurement often reflects underlying decision processes or patient states (Groenwold, 2020). This means that even a theoretically perfect imputation model cannot fully recover predictive signals that are inherently encoded in the pattern of missingness itself. Ignoring M in such cases leads to a systematic loss of information, and may result in suboptimal predictions even with accurate imputations (Little & Rubin, 2019).

Several methods incorporate missingness directly into predictive models. A common approach is the Missing Indicator Method (MIM), which augments inputs with binary flags for missingness, enabling models to exploit missingness patterns and improve performance under informative missingness (Van Ness et al., 2023). NeuMiss(Le Morvan, Josse et al., 2020) further integrates missingness by multiplying zero-imputed inputs with the missingness mask, directly encoding absence into the model. Liu, Kumara and Reich (2021) take an interpretable approach, using a mixed-integer programming framework that encodes missing survey responses as binary indicators rather than imputing them, allowing the model to learn from both observed and intentionally skipped answers. More recently,McTavish et al. (2024) introduced M-GAM, a generalized additive model that learns sparse interactions between features and the missingness mask, which we compare to the MA-methods in Paper C. In `MINTY`, missing values are replaced using zero-imputation, and a binary mask is simultaneously stored to capture the missingness pattern. This setup allows the model to process complete input vectors while retaining information about which values were originally missing–information later used to assess the model's reliance on missingness

2.3.3 Native Strategies for Handling Missingness

Tree-based models offer native strategies to handle missing values without relying on imputation, integrating missingness into their decision processes. For example, the Classification and Regression Trees (CART) algorithm handles missing data using *surrogate splits*. When evaluating splits, CART uses only the subset of data where values are observed for a candidate feature, computing split criteria accordingly. Once the best splitting feature is chosen, if its value is missing for a specific observation during inference, CART identifies alternative features, known as surrogates, that most closely replicate the primary split. These surrogates are ranked by their predictive agreement with the primary split and used to guide the decision path (Lewis, 2000). Figure 2.2 illustrates this mechanism in a clinical example predicting sepsis risk, where heart rate serves as a surrogate split when temperature is missing.

In contrast, XGBoost handles missing values by learning optimal *default directions* during training. When evaluating a potential split on a feature with missing values, XGBoost considers only the non-missing observations to compute gain. Simultaneously, it learns whether instances with missing values should be assigned to the left or right child node, choosing the direction that minimizes training loss. This mechanism allows XGBoost to integrate missingness directly into the learned tree structure (T. Chen & Guestrin, 2016). Another notable approach is Missingness Incorporated in Attributes (MIA) (Josse et al., 2024; Kapelner & Bleich, 2015; Twala, Jones & Hand, 2008), which treats missingness as an informative signal. For continuous variables, MIA expands the decision space by allowing explicit branching for missing values, for instance, using rules such as $X_j < t$, $X_j \geq t$, or $X_j = \text{NA}$. For categorical variables, the missing value is treated as an additional category. This allows the model to leverage potential predictive information in the missingness pattern itself, making it effective under test-time missingness. While all these strategies are effective within their respective model classes, they are inherently model-specific and thus less generalizable to other architectures.

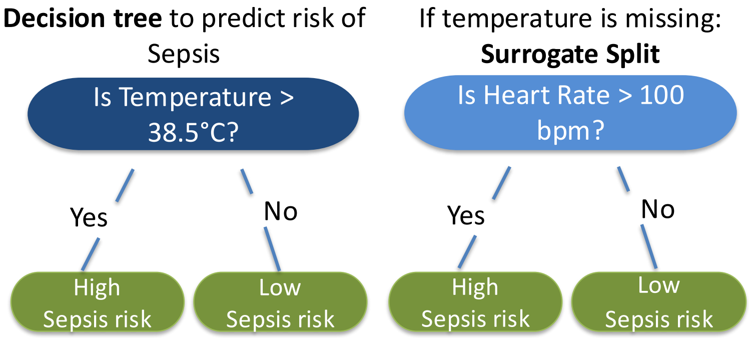


Figure 2.2: Decision tree for sepsis risk prediction using surrogate splits. The model first splits on temperature. If the temperature is missing, it uses heart rate as a surrogate to guide the decision. This illustrates how CART maintains predictive paths despite missing values.

Chapter 3

Interpretable Machine Learning

Interpretability in machine learning is a nuanced and context-dependent concept, and defining it precisely, especially from a mathematical standpoint, remains challenging (Molnar & Freiesleben, 2024). In general, interpretability refers to the degree to which a human can understand the cause of a decision or predict the behavior of a model (Biran & Cotton, 2017; T. Miller, 2019). Kim, Khanna and Koyejo (2016) defines:

> *A method is interpretable if a user can correctly and efficiently predict the method's results.*

The more interpretable a machine learning model, the easier it is for someone to understand why certain decisions or predictions were made. A model is considered more interpretable if its decisions are easier for humans to comprehend (Molnar, 2020). In this thesis, interpretability refers to models and methods that make the behavior of machine learning systems comprehensible to humans (Doshi-Velez & Kim, 2017). This includes extracting relevant knowledge about relationships present in the data or learned by the model itself (Murdoch et al., 2019). Importantly, interpretability does not require full transparency of internal mechanisms, but rather a sufficient understanding to support informed decision-making, especially in high-stakes domains such as healthcare, credit scoring, and criminal justice (Rudin et al., 2022).

Why and when do we need interpretability? Interpretability is crucial in high-stakes domains, where prediction errors can have serious consequences, such as in healthcare or criminal justice. In contrast, it may be less important in low-risk settings like ad serving or postal code sorting (Rudin, 2019). In such sensitive contexts, interpretability supports model auditing, justification, and trust. Designing interpretable models remains challenging, as it requires balancing simplicity, transparency, and actionable explanations without oversimplification (Rudin et al., 2022). Moreover, recent methods have proven effective for model debugging and identifying dataset issues (Adebayo et al., 2020; Koh & Liang, 2017). When high predictive performance alone is insufficient, interpretability becomes essential for validating and refining models (Doshi-Velez & Kim, 2017).

Interpretability also plays a critical role in scientific discovery. Molnar and Freiesleben (2024) argue that for supervised learning to genuinely support understanding of real-world phenomena, models must be equipped with tools such as causal reasoning, domain knowledge, interpretability, and uncertainty estimation. These tools help transform predictions into actionable insights, support decision-making, and generate new hypotheses (Wysocki et al., 2023).

How to achieve interpretability in machine learning? Lipton (2018) examines model properties and techniques commonly associated with interpretability, distinguishing between transparency and post-hoc explanations, which provide insight after model training. Broadly, the field differentiates between *interpretable-by-design models*, which are inherently transparent and allow users to directly understand or reason about the model's predictions, and *post-hoc methods*, which provide approximations or explanations after the fact to gain insight into their decision-making processes (see Molnar (2020) for a comprehensive overview).

This thesis focuses on interpretable-by-design models, which embed explanations directly into their structure (e.g., decision trees, linear models). These models are preferred in high-stakes contexts because they allow users to reason about predictions without relying on post-processing steps. In contrast, post-hoc methods attempt to explain black-box models after training. Common techniques include LIME (Local Interpretable Model-agnostic Explanations)(Ribeiro, Singh & Guestrin, 2016), which fits a simple interpretable model locally around a prediction to approximate the black-box behavior, and SHAP (SHapley Additive exPlanations)(Lundberg & Lee, 2017), which assigns feature importance scores based on Shapley values from cooperative game theory. These methods can offer useful insights, but they often approximate rather than faithfully represent the model's behavior, which may reduce their reliability in certain settings (Covert, Lundberg & Lee, 2021; Slack et al., 2020). Moreover, their explanations rely on perturbations or extrapolations in regions with little or no training data, and methods like SHAP can be computationally expensive for complex models.

3.1 Examples of interpretable machine learning

We focus on interpretable-by-design models to ensure transparency and enable direct insights into both the data and model behavior. Prior work (Kaur et al., 2024) has emphasized the cognitive burden that can arise when users must adapt to unfamiliar model formats–even if they are interpretable. To mitigate this challenge, we prioritize familiar model representations, such as risk scores and decision trees, which are widely used in healthcare and other domains for their balance of simplicity and clarity (Molnar, 2020). In the following, we highlight several well-established interpretable models from the literature that have informed the design of the methods presented in this thesis.

3.1.1 Linear Models

Linear models are widely regarded as interpretable due to their transparent structure and ease of understanding. A linear regression model predicts an outcome y as a weighted sum of input features:

$$y = a_0 + a_1 x_1 + \cdots + a_d x_d + \epsilon,$$

where a_j are feature weights and $\epsilon \sim \mathcal{N}(0, \sigma^2)$ represents additive Gaussian noise, capturing the error between the prediction and the actual outcome (Hastie, Tibshirani & Friedman, 2009). The coefficients a_j provide direct insight into how each feature influences the prediction, which supports both transparency and trust in decision-critical domains.

Sparsity. Sparsity enhances interpretability by reducing the number of features and parameters used in a model, making it easier to understand and analyze. This is typically achieved via regularization methods such as *LASSO* (Tibshirani, 1996), which solves:

$$\min_a \frac{1}{n} \sum_{i=1}^{n} (a^\top x^{(i)} - y^{(i)})^2 + \alpha \|a\|_1,$$

where $\|a\|_1 = \sum_{j=1}^{d} |a_j|$ is the ℓ_1-norm, and α controls the level of sparsity in the parameters. This penalty encourages many coefficients to shrink to zero, leaving only the most influential features and thus reducing model complexity. Compared to dense models, sparse models are easier to inspect and communicate, especially in high-stakes domains such as healthcare or criminal justice.

To further promote interpretability, Takada, Suzuki and Fujisawa (2020) modifies LASSO by penalizing the selection of correlated features, encouraging diverse and informative predictors. Integer programming approaches such as Supersparse Linear Integer Models (Ustun & Rudin, 2016; Ustun, Traca & Rudin, 2013) offer feature sparsity with integer coefficients, producing concise scoring systems interpretable by non-experts. integrate sparsity with handling of missingness indicators to mitigate overfitting and avoid combinatorial explosions raised in earlier work (Van Ness et al., 2023). In practice, sparse models have been used to build transparent decision rules relying on just a handful of variables, such as age, blood pressure, and comorbidities in clinical triage (Ustun & Rudin, 2019).

We use sparsity in different capacities in our work to develop interpretable models. For instance, in Paper A, we introduced sparsity into patterns-specific modeling (`SPSM`) to allow concise descriptions of specialized submodels. By limiting the number of differences (nonzero coefficients) between submodels, we allowed clinicians to meaningfully interpret variable relevance, directionality, and magnitude. In LASSO-MA, introduced in Paper C, we train sparse models that avoid relying on frequently missing features, instead prioritizing features that are both predictive and commonly available.

Generalized Linear Models. Generalized Linear Models (GLMs) extend linear regression to handle non-Gaussian outcomes and nonlinear relationships through a link function (Nelder & Wedderburn, 1972). A GLM assumes the conditional distribution of the response variable $Y \mid X$ belongs to the exponential family and models the expected outcome via:

$$g(\mathbb{E}[Y \mid X]) = a_0 + a_1 x_1 + \cdots + a_d x_d,$$

where $g(\cdot)$ is the link function. The distribution of $Y \mid X$ introduces a noise model appropriate for the outcome type (e.g., Bernoulli noise for binary data, Poisson noise for counts).

For continuous targets with additive Gaussian noise, the identity link $g(z) = z$ is used, reducing the GLM to a classical linear regression model:

$$\mathbb{E}[Y \mid X] = a_0 + a_1 x_1 + \cdots + a_d x_d, \quad \text{with } Y \mid X \sim \mathcal{N}(\mathbb{E}[Y \mid X], \sigma^2)$$

(McCullagh, 2019).

As GLMs remain interpretable in low dimensions, complexity increases with interactions and high-dimensional inputs (Wei et al., 2019).

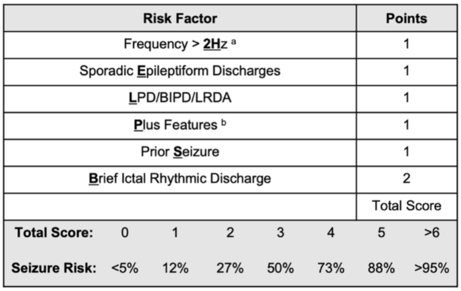
Risk Scores. When learned directly from data, risk scores yield sparse, interpretable models that are widely used in healthcare (Ustun & Rudin, 2019). They assign integer points to features based on logistic regression coefficients and are often used in screening tools (e.g., sleep apnea (Ustun et al., 2016)) or diagnosis (e.g., Alzheimer's (Souillard-Mandar et al., 2016)). These models benefit from integer coefficients, enabling mental calculation and facilitating clinical adoption (Rudin, Wang & Coker, 2020).

As the example in Figure 3.1 shows, interpretable models output humanly understandable summaries of their calculations that help us understand how they produce predictions (Rudin, 2019). This transparency can help build trust in machine learning systems (Molnar, 2020).

In Paper B, we introduced `MINTY`, a sparse linear rule model designed for test-time missingness. The model builds on generalized linear models and incorporates rule-based logic using disjunctions (logical ORs) across substitute features, an approach that aligns with how clinicians often reason about risk factors. `MINTY` uses *LASSO* regularization to produce a compact and interpretable rule set, particularly in the presence of redundant features. Its output can be visualized and is inspired by the structure of widely used clinical risk scores.

3.1.2 Decision Trees

Decision trees are interpretable models that capture nonlinearities and interactions by recursively splitting data according to feature thresholds (Souza et al.,



Risk Factor	Points
Frequency > 2Hz [a]	1
Sporadic Epileptiform Discharges	1
LPD/BIPD/LRDA	1
Plus Features [b]	1
Prior Seizure	1
Brief Ictal Rhythmic Discharge	2
	Total Score

Total Score:	0	1	2	3	4	5	>6
Seizure Risk:	<5%	12%	27%	50%	73%	88%	>95%

Figure 3.1: Illustration of variables used to calculate the 2HELPS2B risk score (Struck et al., 2020). The total score is calculated by summing over the points on the right column, associated with a particular seizure risk. The rules explain medical details such as the brief independent periodic discharge (BIPD), continuous EEG (cEEG), generalized periodic discharg (GPD), lateralized periodic discharge (LPD), lateralized rhythmic delta activity (LRDA). **Plus features** are defined as superimposed rhythmic, fast, or sharp activity for LRDA, BIPDs, LPDs, or GPDs.

2022). Each path from the root to a leaf corresponds to a set of rules, and the prediction on the leaf reflects the average of training examples reaching that node (example of tree in Figure 2.2 in the previous section). Their structure allows for step-by-step traceability of predictions, especially when trees are shallow (Molnar, 2020). As shown by Grinsztajn, Oyallon and Varoquaux (2022), decision trees often outperform deep learning models on medium-sized tabular datasets and they remain competitive and often state-of-the-art on medium-sized datasets (~10K samples), even without considering their advantages in speed and computational efficiency.

Short trees remain interpretable, but as trees grow deeper, they become harder to follow and lose transparency (Souillard-Mandar et al., 2016). Ensembles like Random Forests (Breiman, 2001) and Gradient Boosted Trees (e.g., XGBoost (T. Chen & Guestrin, 2016)) further trade interpretability for accuracy by combining multiple trees to enhance predictive performance. These methods have been successfully applied in clinical settings (Caruana et al., 2015; Lundberg & Lee, 2017).

3.1.3 Rule-based Methods

Rule-based models express decision logic using a set of human-readable `IF-THEN` rules. These rules are semantically meaningful and align closely with human reasoning. For example:

```
IF age > 60 AND blood pressure > 140 THEN risk = high
```

Each rule defines a logical conjunction of feature conditions, making model decisions easy to trace and understand. This transparency is a key reason why rule-based models are considered interpretable. Unlike complex black-box models, the decision-making process in rule-based systems can be examined and validated by domain experts, which is particularly important in high-stakes areas like healthcare or policy-making. In falling rule-based models, the importance of features is implicit, features in early rules with high impact are considered more influential (Wang & Rudin, 2015).

Rule-based models can be learned algorithmically from data. One widely used method is RuleFit (Friedman & Popescu, 2008), which extracts decision rules from ensembles of decision trees and combines them with linear terms in a sparse additive model. Another approach is Bayesian Rule Lists (Letham et al., 2015), which constructs a probabilistic model over possible rule sets and selects a small list of rules guided by priors that favor sparsity. More recent work improves inference efficiency while preserving interpretability by constraining rule complexity, such as scalable Bayesian rule models (Margot & Luta, 2021) and SIRUS (Stable and Interpretable RUle Set) (Bénard et al., 2021b), a classification algorithm based on random forests that produces a concise and stable set of rules.

Despite their interpretability, rule-based models have notable limitations. Learning a concise and accurate rule set is computationally hard, especially in high-dimensional spaces (Letham et al., 2015; Wang & Rudin, 2015). There is often a trade-off between rule simplicity and predictive coverage: simpler rules may fail to capture important interactions, while more complex rules can overwhelm users and reduce clarity (Bénard et al., 2021b). Additionally, rule mining algorithms may produce redundant or overlapping rules, which obscure the decision logic and complicate validation (Mannhardt et al., 2016). These challenges highlight the need for methods that encourage sparsity, stability, and semantic coherence in rule-based learning.

3.2 Evaluation towards interpretablity

As the previous section defined interpretability, highlighted its importance, and outlined strategies to achieve it, this section focuses on how to evaluate interpretability. While interpretable-by-design models, such as decision trees or sparse linear models, are often assumed to improve understanding and trust, empirical evidence suggests this is not always the case. In a series of large-scale experiments, Poursabzi-Sangdeh et al. (2021) showed that although transparent models with few features were easier for users to simulate, this did not consistently help them follow the model's predictions or recognize its errors. In some cases, transparency even hinders user performance due to cognitive overload. These findings underscore the importance of evaluating interpretability through empirical testing, rather than relying solely on design intuition. Although there is no universal metric, researchers can assess interpretability based on task-specific requirements and decision-making contexts, or design appropriate experiments (T. Miller, 2019). Doshi-Velez and Kim (2017)

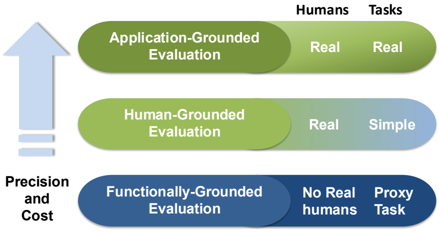

Humans
Tasks
Application-Grounded Evaluation
Real
Real
Human-Grounded Evaluation
Real
Simple
Functionally-Grounded Evaluation
No Real humans
Proxy Task
Precision and Cost

Figure 3.2: Taxonomy of evaluation approaches for interpretability. Figure design is inspired by Doshi-Velez and Kim (2017).

propose a taxonomy that classifies evaluation methods into three categories: application-grounded, human-grounded, and functionality-grounded. These complementary approaches differ in task specificity and the level of human involvement they require. As shown in Figure 3.2, the choice between human-subject experiments and automated evaluations depends on the complexity and nature of the task.

1. **Application-grounded evaluation** of interpretability involves human experiments in real-world applications. For example, in a diagnostic model detecting prostate tumors from medical images, doctors would assess the correctness of predictions. The goal is to determine how well human-generated explanations aid others in completing the task.

2. **Human-grounded evaluation** uses simplified human experiments where the task resembles a real application but does not require domain experts. This approach is useful when the target community is unavailable or abstract tasks are needed. For instance, laypeople might compare hybrid images with highlighted regions generated by a model to assess which best identifies distinguishing features. Other examples include selecting the better explanation from a pair, predicting a model's output given input and explanation, or suggesting input changes that would alter the model's prediction, such as adjusting workplace parameters to prevent employee attrition.

3. **Functionally-grounded evaluation**, in contrast, excludes human subjects. It is preferred when human involvement is costly or unnecessary. This method often measures improvements in model performance based on prior human-validated interpretability.

To achieve a high impact in the real world, our community must acknowledge the significant time and effort required for such evaluations and uphold rigorous standards in experimental design (Doshi-Velez & Kim, 2017). As recognized in the human-computer-interaction (HCI) community (Antunes et al., 2008), application-grounded evaluations are inherently challenging. However, they provide direct evidence of a system's success by measuring performance against its intended objective.

In Paper D, we adopt an application-grounded evaluation approach, acknowledging both its high potential impact and the considerable effort it requires. This framework guided the design of our expert survey with healthcare professionals. We evaluate interpretable machine learning methods for handling missing values by engaging a network of clinicians in a real-world task: predicting hemorrhagic shock.

In summary, interpretable machine learning encompasses a range of models that prioritize transparency, trust, and domain understanding. This chapter introduced core model types and outlined how recent advances improve their usability in practice, particularly in high-stakes domains like healthcare. The next chapters build on these foundations to present our proposed methods and empirical evaluations.

Chapter 4

Central Challenges of Test-Time Missingness in Interpretable Machine Learning

Chapters 2 and 3 established the foundation for supervised prediction with test-time missingness using interpretable models, with the goal of supporting autonomous, informed decision-making by domain experts. This chapter builds on that foundation by explaining why current methods for handling missing values are insufficient for delivering the level of interpretability needed for trust and accountability in high-stakes settings.

In interpretable models such as linear models or decision tree classifiers (examples in Section 3.1), imputation is typically necessary, as these models require fully observed input vectors. Complex imputation techniques, such as MICE (Van Buuren, 2007), can obscure the relationship between input features and model predictions, thereby reducing interpretability. Moreover, even under *perfect imputation*, the Bayes-optimal predictor trained solely on imputed values may still be suboptimal (Le Morvan, Prost et al., 2020). Many existing methods for handling missing values focus on recovering ground truth values, often guided by statistical inference. However, in predictive settings, the objective is not accurate reconstruction but improved downstream performance.

Challenge 1: *Imputation hides which values were missing and the reasons behind their absence, potentially obscuring informative missingness patterns and limiting the model's ability to handle missingness at test-time.*

In many applications, the pattern of missingness itself can be predictive. This is referred to as informative missingness (Rubin, 1976) (see Section 2.3.2 for details). For example, a missing lab test might signal that the clinician found it unnecessary, perhaps because a diagnosis was already clear or a substitute test was used. In such cases, the absence of data carries signal, suggesting that some features may be redundant or context-dependent. To retain this signal, a common strategy is to augment the input space with missingness indicators $M \in \{0,1\}^d$, resulting in an extended input $(X, M) \in \mathbb{R}^d \times \{0,1\}^d$ (Little & Rubin, 2019). This enables the model to learn dependencies between missingness patterns and the outcome Y, potentially boosting predictive performance (Van Ness et al., 2023).

However, this expansion increases the dimensionality: from d features to $2d$, and even further to $d(d-1)+2d$ when first-order interactions are included. This makes interpretable modeling more difficult. In linear models, it leads to more coefficients; in rule-based models, it enlarges the rule search space, increasing the risk of complexity and reduced clarity. In approaches that fit separate submodels for each missingness pattern (Mercaldo & Blume, 2020), the number of models grows exponentially with the number of features, making it infeasible in data-scarce or time-sensitive environments such as healthcare.

Challenge 2: *Maintaining model simplicity is important, as expanding the input space often comes at the expense of interpretability.*

Beyond the complexity introduced by missingness indicators, a separate challenge arises in models that handle missing data natively.

Tree-based models, such as decision trees and gradient-boosted trees (e.g., XGBoost (T. Chen & Guestrin, 2016), gradient boosting machines (Friedman, 2001)), are widely used for their predictive accuracy and, in the case of individual trees, for their intuitive structure (Molnar & Freiesleben, 2024). These models include native strategies to deal with missing data without requiring explicit imputation.

For example, gradient boosting trees learn default split directions: if a feature is missing at test time, the model assigns the instance to a default branch that minimizes the training loss (T. Chen & Guestrin, 2016). Another common strategy is the use of surrogate splits (Valdiviezo & Van Aelst, 2015), where the tree selects alternative features that mimic the primary split decision when the main feature is missing. While these mechanisms help the model function with incomplete inputs, they often rely on fixed heuristics that may not fully exploit the information embedded in missingness. Moreover, their internal logic is not explicitly surfaced to the user, which reduces transparency (Molnar & Freiesleben, 2024). In large ensembles, interpretability deteriorates further, making it difficult to understand how missingness influences predictions, or how a specific feature path led to a decision (Lundberg, Erion & Lee, 2020).

Challenge 3: *Default path assignment and surrogate splits are inherently limited to tree-based model classes, and restrict interpretability, especially in complex ensembles.*

Our goal is to address these challenges by designing interpretable models with strong predictive performance that reduce reliance on complex imputations and avoid unnecessarily inflating the input space, while effectively accounting for test-time missingness. In the next chapter, we summarize the five papers that constitute the core contributions of this thesis.

Chapter 5

Summary of Included Papers

While the full manuscripts of the appended publications are included in Part II, this chapter briefly summarizes their respective research contributions.

5.1 Paper A: Sharing Pattern Submodels for Prediction with Missing Values

In this paper, we present a method for handling missing values during deployment, especially when there are few examples for each missingness pattern. A missingness pattern refers to a specific combination of observed and missing features in the input data. For example, one group of patients may have blood pressure and age recorded but not cholesterol, while another has cholesterol and age but not blood pressure. Such patterns can arise due to differences in equipment availability across hospital sites or due to medical preconditions specific to certain patient groups.

Prior work has proposed fitting separate models for each missingness pattern to avoid imputation and enhance interpretability (Mercaldo & Blume, 2020). However, this can lead to high variance with limited data per pattern and overlooks potential relationships across patterns. Conversely, using a single model across all missingness patterns often requires imputing missing values, which can introduce bias into the prediction estimates.

We propose *Sharing Pattern Submodels* (`SPSM`), a method for accurate prediction with incomplete inputs that also yields concise, interpretable model output descriptions for domain experts. `SPSM` balances the bias–variance trade-off by combining a shared main model with sparse, pattern-specific deviations. For each missingness pattern, a submodel is trained using pattern-specific coefficients Δ_m, which are combined additively with shared coefficients θ from a main model. The final prediction is given by $\hat{y} = (\theta + \Delta_m)^\top x$, enabling shared learning across all patterns while allowing flexibility to adapt to pattern-specific characteristics. This structure promotes information sharing across patterns by balancing predictive accuracy and variance. Shared coefficients θ are regularized using ℓ_1 or ℓ_2 norms, while Δ_m is encouraged to be sparse via an ℓ_1 penalty. The regularization strength λ_m controls the degree of sharing–larger values lead to greater reliance on the shared model.

Figure 5.1 illustrates this phenomenon with patients from three different clinics, each following slightly different data collection practices. As introduced earlier, this results in distinct missingness patterns; for example, one clinic may routinely record blood pressure but omit certain lab tests, while another does the reverse. In SPSM, each clinic's data is modeled with a submodel that captures its specific pattern of missingness via coefficients Δ_m, and at the same time shares information through a global set of coefficients θ. This enables the model to adapt to clinic-specific practices while benefiting from a shared predictive structure across all clinics.

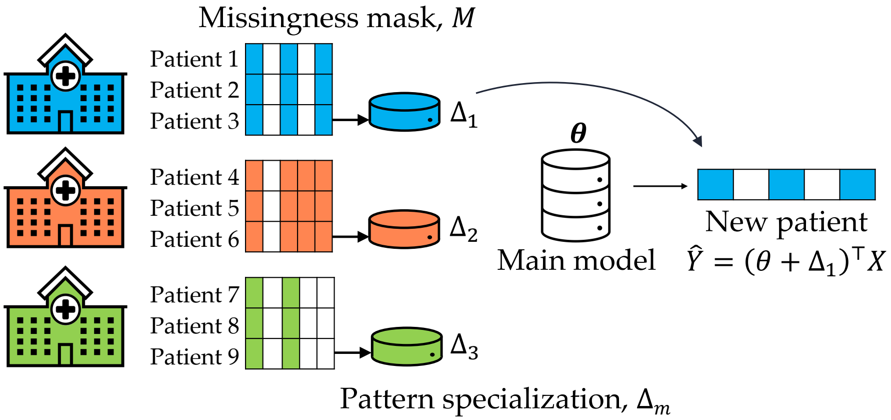


Figure 5.1: Coefficient sharing between a main model θ and pattern submodels for three clinics with different patterns in missing values. The white areas in the missingness masks represent missing features, the filled areas indicate observed ones. Without specialization Δ_m, a shared average prediction across clinics may not yield optimal performance for any clinic. Conversely, fitting separate models for each clinic leads to high variance and inefficient use of data.

We evaluated the SPSM model on simulated and real-world data. The experimental results indicate that SPSM performs comparably or slightly better than baselines across all datasets without relying on imputation (Figure 5.2). The results demonstrate that the proposed method never performs worse than non-sharing pattern submodels, which do not make efficient use of the available data. Our theoretical analysis shows that in a linear-Gaussian setting, our method also recovers the sparsity of the true process. Although this may not reduce variance in the large-sample limit, the sparsity enhances interpretability.

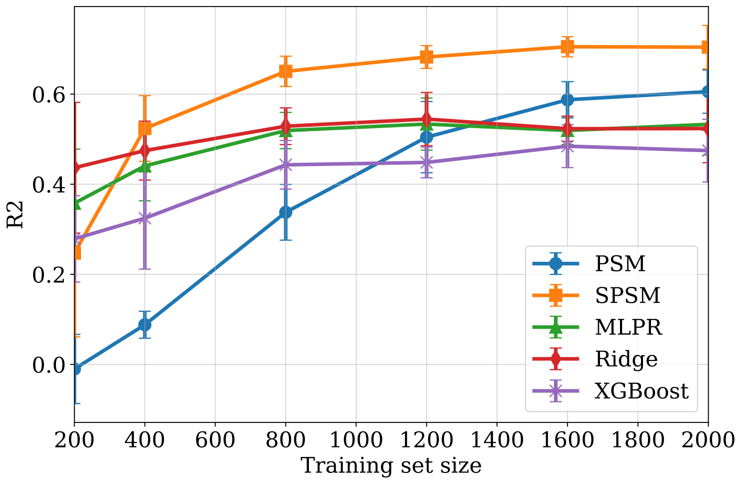


Figure 5.2: Performance on simulated data for Setting A (higher is better). Error bars represent the standard deviation over 5 random data splits. The full dataset contains $n = 2000$ samples.

Why is `SPSM` Interpretable? `SPSM` improves interpretability by allowing domain experts to compare pattern specializations and understand how similar submodels behave and are affected by missing values (Table 5.1). We argue that a set of submodels is easier to interpret if the specializations contain fewer non-zero coefficients; that is, if $\Delta_{\neg m}$ is sparse. This is achieved through regularization, resulting in a sparse model including only a subset of input features affecting predictions, thus reducing the model's effective complexity (Cowan, 2010; G. A. Miller, 1956). Note, `SPSM` learns linear models, but it is not limited to linear systems and does not assume anything about the missingness mechanism.

Table 5.1: Example of Δ_4 for regression using `SPSM` ADNI data (Weiner et al., 2010). `SPSM` uses $\gamma = 10$ and $\lambda = 13$ as parameters for a single seed. There are 10 missingness patterns in total, with 4 of them having non-zero coefficients for Δ and a pattern-specific intercept. Coefficients are for standardized variables.

Missing features in pattern 4: ABETA, TAU, and PTAU at baseline (bl)

Feature	Δ_4	θ	$\theta + \Delta_4$
Age	-0.140	0.121	-0.019
FDG-PET	-0.090	-0.039	-0.129
Whole Brain (bl)	0.000	-0.045	-0.044
Fusiform	0.016	0.021	0.037
ICV	0.001	0.093	0.094
Intercept	-0.10	0.18	

5.2 Paper B: MINTY: Rule-based models that minimize the need for imputing features with missing values

Paper A leverages patterns of missingness to improve prediction performance and produce interpretable model outputs, while Papers B and C develop models that minimize reliance on frequently missing variables. As before, we focus on settings where the observed variables follow a *fixed, unknown* distribution $p(X, M, Y)$, and handling missing values during both training and deployment challenges the maintenance of model accuracy and interpretability.

Predicting 2-year change in cognitive function (ADAS13)

Model rules	**Coef.**	**Score**
MMSE ≤ 26 OR Alzheimer's disease (AD)	+4	+4
TAU ≤ 191 OR PTAU ≤ 17	-5.2	-5.2
Married = TRUE	+3	+0
	Predicted change:	**-1.2**

Anna's features

MMSE	**TAU**	**PTAU**	**MAR.**	**AD**
24	170	N/A	N/A	No

Figure 5.3: Illustrative example of scoring system predicting cognitive decline, measured by a change in the ADAS13 cognitive function score, using the *ADNI* data, including incomplete data. The blue, underlined features indicate that these variables are observed for the specific patient, Anna, and the red shows that the observations for the variables are missing.

To address the limitations of imputation-based approaches, we propose MINTY, a method for learning **int**terpretable generalized linear rule models (GLRMs) that minimize reliance on imputed (**mis**sing) values. MINTY constructs disjunctive rules by grouping literals of single variables, allowing the rule to be evaluated as true when at least one literal is observed, regardless of others being missing. This design exploits redundancy in the covariates and supports interpretability by aligning model logic with observed inputs.

We start from GLRMs, which are not inherently designed to handle missing values. The MINTY approach extends the column-generation strategy by Wei et al. (2019), iteratively adding disjunctive rules that minimize empirical risk and at the same time controlling reliance on missing values. The objective function includes a parameter $\gamma \in [0, \infty)$ to regularize reliance on missing values in the disjunctions, effectively balancing predictive power and interpretability by penalizing rules that depend on missing data.

To find the next rule, we solve an optimization problem, either exactly using the Gurobi solver or approximately via a heuristic beam search, that selects a sparse set of features, determines which samples activate the rule, and identifies those that rely on missing values (through ρ). The objective balances three things:

- How well the rule captures relevant samples.
- How much it relies on missing values (penalized by a parameter γ).
- How complex the rule is, measured by how many features it uses (penalized by λ_1).

The first constraint ensures that a sample activates the rule if it satisfies any of the selected conditions (based on observed features). The second constraint tracks when a sample relies on missing values to activate the rule: this happens only if (i) no selected feature is observed and true, and (ii) at least one selected feature is missing.

Figure 5.3 illustrates a disjunctive linear rule model used to predict cognitive decline, measured by the ADAS13 cognitive test score. A higher score indicates lower cognitive ability, and a positive change from baseline to the 2-year follow-up reflects deterioration. Each rule (left) contributes a score (right) if at least one literal is observed and true, regardless of other missing values. For a patient, Anna, the rule `Tau` $\leq$ `191 OR PTAU` $\geq$ `27` is satisfied due to her observed TAU level, even though PTAU is missing, contributing -5.2 to the score. The rule `MMSE = 24 OR AD Diagnosis=True` is also true via her MMSE score, despite no prior AD diagnosis. In contrast, a single-literal rule like `Married = True` cannot be evaluated when missing and defaults to a zero contribution–standard in clinical risk scoring (Afessa et al., 2005), though ideally avoided by favoring rules that require only observed values.

We evaluate `MINTY` on three real-world datasets with natural and semi-natural missingness. Baselines include standard models with imputation, models with native support for missingness (e.g., `XGB` (T. Chen & Guestrin, 2016)), and models that leverage missingness patterns (e.g., `NEUMISS`) (Le Morvan, Josse et al., 2020). `MINTY` achieves competitive or superior predictive performance while substantially reducing reliance on missing values (lower ρ), all without requiring imputation. These results highlight `MINTY`'s balance between interpretability and accuracy. Beyond the empirical study, we analyze the effect of a regularization term, controlled by $\gamma \geq 0$, that penalizes reliance on missing values in disjunctive linear rule models. Setting $\gamma = 0$ offers flexibility but sacrifices interpretability, while $\gamma \to \infty$ enforces strict rules that avoid imputation but may yield trivial solutions in settings with frequent missingness. Instead, selecting a moderate γ allows us to limit reliance on missing data while maintaining both model accuracy and interpretability. In other words, requiring *perfect* variable redundancy through rules by letting $\gamma \to \infty$ is too strict for many settings. Instead, we can aim to *limit* or *minimize* the average reliance on missing values $\bar{\rho}$ by selecting a moderate γ.

Interpreting Learned Rules on *ADNI* Table 5.2 shows MINTY models learned on *ADNI*, formatted like medical or justice risk scores (Ustun & Rudin, 2019). In each table, on the left are rule definitions and on the right, their coefficients. The tables on the left and right correspond to MINTY$\gamma = 0$ and MINTY$\gamma = 0.01$, respectively. In the *ADNI* task, the goal is to predict the cognitive decline measured by a change in the cognitive test score ADAS13. The learned coefficients match expectations as, for example, diagnoses of Alzheimer's disease (AD) or mild cognitive impairment (LMCI) are associated with higher cognitive decline (positive coefficients). Similarly, MMSE $\geq$ 29 (a score indicating normal cognitive ability) is associated with a smaller decline in ADAS13 (negative coefficient). The two models with $\gamma = 0$ and $\gamma = 0.01$ learn similar rules with similar coefficients but with different reliance on features with missing values ($\bar{\rho} = 0.40$ vs $\bar{\rho} = 0.27$). The rules, TAU $\leq$ 191.1 OR Hippocampus, where hippocampus volume is denoted as V$_h$, $\geq$ 7721.0 and FDG $\leq$ 1.163 are not included in the second model ($\gamma = 0.01$), since they are missing for 0.33% and 0.27% of all individuals in the data set. By using a higher γ we achieve a more robust solution with less dependence on imputed values.

Table 5.2: MINTY models learned on *ADNI* a) using $\gamma = 0$ (left) and b) $\gamma = 0.01$ (right). The R^2 for the two models were 0.64 and 0.63 respectively, the latter with smaller reliance on features with missing values ($\bar{\rho} = 0.28$ vs $\bar{\rho} = 0.40$). Two rules in the left model are not in the right model due to more frequent missingness; the right model adds two rules with less missingness. MINTY models on *ADNI* with different γ values.

a) MINTY with $\gamma = 0$ ($R^2 = 0.64$, $\bar{\rho} = 0.40$)

Rule	Coeff.
AD OR LMCI diagnosis	+0.35
MMSE $\leq$ 26.0 OR LMCI	+0.23
LDELTOTAL $\leq$ 3.0	+0.63
AD diagnosis	+0.65
V$_h$ $\leq$ 6071.0 OR Male	+0.18
MMSE $\geq$ 29.0	−0.16
Entorhinal $\leq$ 3022.0	+0.18
LDELTOTAL 3−8	+0.27
TAU $\leq$ 191.1 OR V$_h$ $\geq$ 7721.0	−0.19
FDG $\leq$ 1.163	+0.17
Intercept	−0.57

b) MINTY with $\gamma = 0.01$ ($R^2 = 0.63$, $\bar{\rho} = 0.28$)

Rule	Coeff.
AD OR LMCI diagnosis	+0.36
MMSE $\leq$ 26.0 OR LMCI	+0.22
LDELTOTAL $\leq$ 3.0	+0.67
AD diagnosis	+0.68
V$_h$ $\leq$ 6071.0 OR Male	+0.19
MMSE $\geq$ 29.0	−0.17
Entorhinal $\leq$ 3022.0	+0.17
LDELTOTAL 3−8	+0.28
V$_h$ $\geq$ 7721.0	−0.16
APOE4 = 1	+0.08
Intercept	−0.61

While MINTY is promising, its application is limited to GLRMs. Paper C, presented in the next section, explores extending this framework to non-linear models such as ensemble methods and expanding its use to other areas of interpretable machine learning.

5.3 Paper C: Prediction models that learn to avoid missing values

Paper C builds on the idea of avoiding missing values during prediction at deployment, similar to the goal outlined in Paper B. A central question of this work is: What if we could sidestep imputation, indicators, and specialized architectures by *learning to avoid* using features with missing values in predictions? If a feature is missing but still needed for a prediction in a given instance, a good model should not rely on it in the first place.

The main contribution of Paper C is the *missingness-avoiding (MA) machine learning* framework for training models to rarely require the values of missing (or imputed) features at test time. We design tailored MA learning algorithms for decision trees, tree ensembles, and sparse linear models by incorporating classifier-specific regularization terms into their learning objectives.

We introduced *missingness reliance* with the definition that h relies on missing values in an observation $\mathbf{x}$ if there is a feature j such that 1) $x_j = \texttt{NA}$ and 2) computing $h(\mathbf{x})$ requires evaluating x_j or its imputed value x_j^I. We use a binary indicator function $\rho(h, \mathbf{x}) \in \{0, 1\}$ to indicate that computing $h(\mathbf{x})$ relies on at least one missing feature in $\mathbf{x}$,

$$\rho(h, \mathbf{x}) = \max_{j \in [d]} \mathbb{1}[a_h(\mathbf{x}, j) = 1 \wedge x_j = \texttt{NA}] \ . \tag{5.1}$$

The expected missingness reliance ρ of a hypothesis h in a distribution $p(X, M, Y)$ is then defined as $\rho(h) := \mathbb{E}_p[\rho(h, X)]$. The goal of MA learning is to find a suitable trade-off between expected predictive performance and missingness reliance:

$$\underset{h \in \mathcal{H}}{\text{minimize}} \ \mathbb{E}_p[L(Y, h(X))] + \alpha\rho(h) \ , \tag{5.2}$$

controlled by a tradeoff parameter $\alpha \geq 0$.

Our goal is to learn hypotheses h that are *missingness avoiding*, that is, they are unlikely to require the value of a missing variable at test time. Let $a_h(\mathbf{x}, j) = 1$ denote the event that computing $h(\mathbf{x})$ requires access to the value of x_j (imputed or observed) and $a_h(\mathbf{x}, j) = 0$ otherwise. For example, computing the prediction of a linear model with imputed inputs, $h_\theta(\mathbf{x}^I) = \theta^\top \mathbf{x}^I$, requires access to x_j^I whenever $\theta_j \neq 0$. A decision tree requires access to x_j^I if feature j appears on the prediction path from root to leaf for the input $\mathbf{x}$. A rule model requires access to x_j^I if the truth values of its rules are contingent on x_j^I. Figure 5.4 illustrates how different models may avoid relying on missing values. he approach is applied to sparse linear models (`MA-LASSO`) for interpretability; decision trees (`MA-DT`) to capture nonlinear interactions; and random forests (`MA-RF`) and gradient boosting (`MA-GBT`) to enhance generalization and performance.

We investigated when it is possible to achieve both low prediction error and zero reliance on missing features. The MA objective introduces a trade-off between accuracy and missingness reliance. Under observed deterministic data collection (ODDC), where features are collected in predictable ways, such

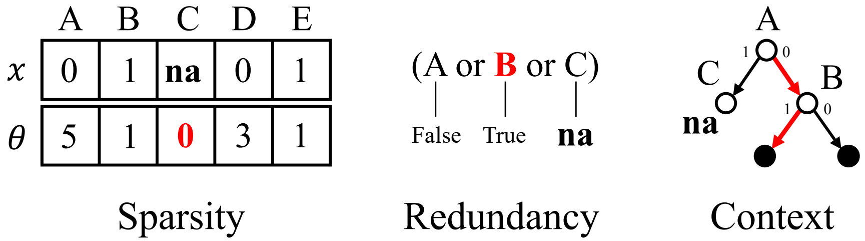


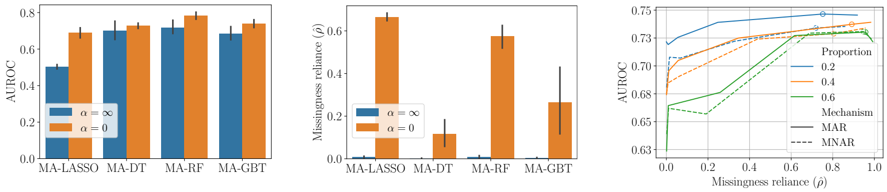
Figure 5.4: Missing values can be avoided in several ways. Sparse models (left) can be trained not to use features that are frequently missing. Disjunctive rule models (middle) can be fit to include rules that exploit redundancy in the variable set. Trees (right) can be fit so that missing values rarely occur on the decision paths.

as age always being recorded or MRIs ordered after abnormal tests, models like decision trees can exploit these patterns. If the target outcome depends only on features that are always observed under ODDC, accurate predictions can be made without relying on missing features. However, when missingness occurs randomly or due to unobserved factors, avoiding reliance becomes harder and may reduce accuracy. In such cases, especially when missingness itself is informative (e.g., tests ordered only when serious conditions are suspected), models that use missingness indicators can outperform those that ignore them.

Experiments were conducted on six real-world datasets, comparing MA models against relevant baselines. Our experiments show that MA models effectively learn to generate accurate predictions while minimizing reliance (ρ) on missing values at test time. In particular, `MA-DT` and `MA-RF` achieve AUROC scores comparable to standard DT and RF but with drastically lower missingness reliance. `MA-LASSO` provides a trade-off between sparsity and missingness avoidance. Figures 5.5(a) and 5.5(b) show this trade-off on the Alzheimer's Disease Neuroimaging Initiative[1](ADNI) dataset. All MA models reach near-zero empirical missingness reliance $\hat{\rho}$ with a large α, but this can come at a cost to AUROC. Conversely, setting $\alpha = 0$ often boosts AUROC, yet substantially increases missingness reliance, particularly for `MA-LASSO` and `MA-RF`. This highlights how tuning α affects both performance and interpretability through missingness dependence. 5.5(c) illustrates the relationship between AUROC and missingness reliance under varying proportions and mechanisms using `MA-LASSO`. L^1-regularized logistic regression models are included with circles for reference. With MAR missingness in up to 40 % of features, `MA-LASSO` maintains high performance even with 25 % missingness reliance. Similar patterns appear in the more challenging MNAR setting, though with slightly lower AUROCs.

How does MA-learning improve interpretability? In Figure 5.6, we include examples of trees with $\alpha = 0$, $\alpha = \alpha^*$, and $\alpha = \infty$ at their respective optimal depths. The value α^* is selected as the candidate model with the lowest

[1] https://adni.loni.usc.edu



(a) ADNI: AUROC vs. MA estimator.

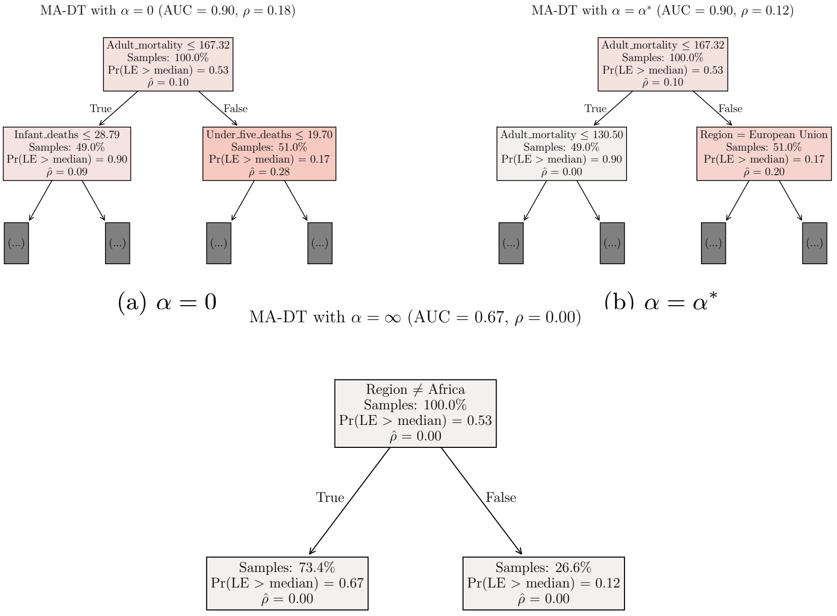
(b) ADNI: $\hat{\rho}$ vs. MA estimator.

(c) Breast Cancer: AUROC vs $\hat{\rho}$.

Figure 5.5: (a) and (b): Test-set AUROC and missingness reliance ($\hat{\rho}$) for MA estimators in ADNI when transitioning from $\alpha = \infty$ to $\alpha = 0$. Removing missingness regularization improves predictive performance, but the increase in missingness reliance is much more pronounced, especially for MA-LASSSO and MA-RF. (c): Test-set AUROC versus $\hat{\rho}$ for MA-LASSSO in Breast Cancer, where 50 % synthetic missingness is added to an increasing proportion of input features. Missingness not at random (MNAR) is more challenging than missingness at random (MAR), but MA-LASSSO demonstrates robust performance for large fractions of missingness.

$\hat{\rho}$ among those achieving at least 95 % of the maximum AUROC. When $\alpha = \infty$, the algorithm is forced to split on the always-observed feature Region instead of Adult_mortality, which has approximately 10 % missingness and is used when $\alpha = 0$ and $\alpha = \alpha^*$. In the latter case, Region appears in the first split in the right branch of the tree, reducing missingness reliance without sacrificing AUROC compared to the unregularized tree, which uses the incomplete feature Under_five_deaths.

Future work could explore alternative selection strategies for α^* and alternative definitions of ρ. Instead of only trading off predictive performance for minimal α, approaches could prioritize limiting missing features per individual, since relying on dataset averages may mask important variability. Application-specific thresholds and domain knowledge could guide these strategies, although such thresholds are often difficult to quantify.



(a) $\alpha = 0$

(b) $\alpha = \alpha^*$

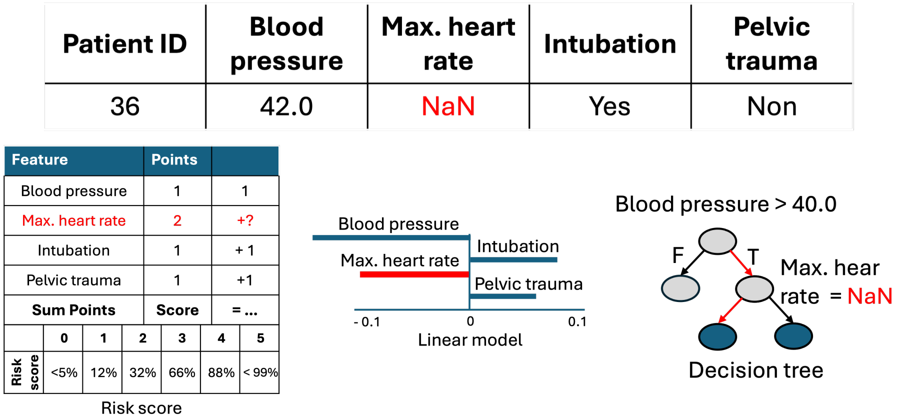
(c) $\alpha = \infty$

Figure 5.6: Example decision trees for $\alpha = 0$, $\alpha = \alpha^*$, and $\alpha = \infty$ fit to LIFE. The nodes are colored based on the missingness reliance ρ. The goal is to predict whether a country's life expectancy (LE) is above or below the median life expectancy. (a): `MA-DT` with $\alpha = 0$ behaves as a regular decision tree, splitting on highly predictive features such as `Adult_mortality` and `Infant_deaths`. (b): `MA-DT` with α selected to balance the trade-off between predictive performance and AUROC. Similar to the standard decision tree, `MA-DT` first splits on `Adult_mortality`, then by the European Union region, reducing missingness reliance. (c): With large α, the tree splits only by the region "Africa", achieving zero missingness reliance but with much worse predictive accuracy.

5.4 Paper D: Handling missing values in clinical machine learning: Insights from a large-scale expert study

Building on previous works that addressed prediction under missing values and model interpretability through methodological advancements, Paper D shifts focus to the human perspective. This work investigates how clinicians, as potential users of interpretability-by-design models, perceive and prefer to handle missing values in prediction models. The paper offers insights into the human factors that influence the practical adoption of such models and informs the design of future methods.

The main contribution includes a qualitative survey of 55 Traumabase clinicians[2], a network of trauma specialists in France, analyzing their attitudes toward artificial intelligence (AI) and machine learning (ML), as well as their strategies for handling missing values in patient records. The study also provides insights into current clinical decision-making workflows. We then evaluated the use of three IML approaches by clinicians in a real-world scenario–predicting hemorrhagic shock in trauma patients with missing values–examining their reasoning, preferences, and challenges in decision making. Lastly, based on the clinicians' feedback, we proposed design guidelines for future IML models that natively handle missing values to support clinical practice and real-world use.



Patient ID	Blood pressure	Max. heart rate	Intubation	Pelvic trauma
36	42.0	NaN	Yes	Non

Feature	Points	
Blood pressure	1	1
Max. heart rate	2	+?
Intubation	1	+ 1
Pelvic trauma	1	+1
Sum Points	**Score**	**= ...**

	0	1	2	3	4	5
Risk score	<5%	12%	32%	66%	88%	< 99%

Risk score

Figure 5.7: During the survey, we presented the clinicians with a patient sample (top), and descriptions of the three trained interpretable machine learning models (from left: risk score, logistic regression, and decision tree) applied to predict hemorrhagic shock for that sample (bottom). Clinicians were asked to compute the predictions given a missing heart rate value and to assess their understanding of and confidence in the models.

The first finding reveals that clinicians vary in their attitudes toward AI/ML

[2] https://www.traumabase.eu/en_US

and missing values in patient records. The largest group (25% of participants) used AI daily and favored interpretable methods, while another substantial group (16%) preferred black-box models like MICE despite limited familiarity. Across groups, a common current practice is that clinicians estimate clinically plausible values using patient history and context to handle missingness in clinical scores.

The second part of the survey presented an individual patient case with a missing value and predictions from three IML models, including risk scores (Ustun & Rudin, 2019), linear models (Hastie, Tibshirani & Friedman, 2009), and decision trees (Breiman, 2017), which were selected for their interpretability and likely familiarity among clinicians (e.g., risk scores) or intuitive structure (e.g., decision trees). See Figure 5.7 for an illustration of the survey interface. Clinicians were asked to estimate the risk of hemorrhagic shock and describe how they would handle the missing value. To control for performance, all models were calibrated to similar accuracy (Molnar, 2020; Stiglic et al., 2020). This setup assessed how clinicians interpret model outputs with missing values, including their preferred imputation strategies (e.g., zero, or population averages) or domain knowledge for each model. The results showed clinicians favored models that either natively handle missingness or incorporate transparent, clinically intuitive imputation. Black-box methods like MICE (Van Buuren & Groothuis-Oudshoorn, 2011) were generally disfavored for their lack of interpretability. Implicit imputation, where clinicians estimated missing values based on available features and their judgment, aligned more closely with clinical reasoning in their current workflow and was preferred over explicit methods like zero or mean imputation. Under missing value conditions, decision trees and risk scores were most trusted, and at the same time, linear models drew skepticism due to perceived limitations in handling missing values (Figure 5.8).

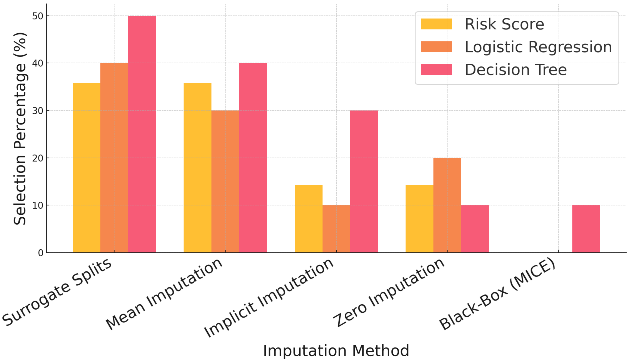


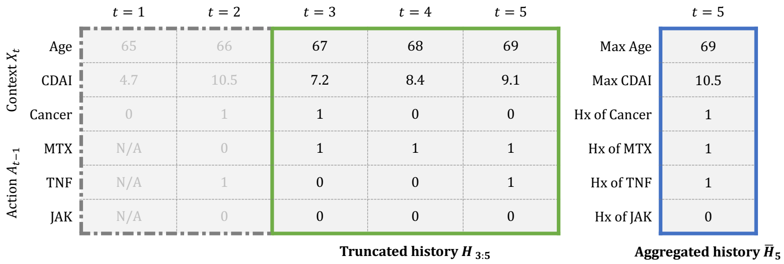
Figure 5.8: Clinician preferences for imputation methods across different IML models. We normalize by dividing the number who chose a combination by the total, as the total votes for a model can vary.

Finally, some findings identify key requirements for the design of IML systems with test-time missingness, reflecting clinicians' preferences. Clinicians were generally open to using IML with missing values but emphasized the need for *clear guidelines*, *transparency in imputation*, and *communication of uncertainty.* Trust depended on *reliability*, and *intuitive explanations.* These findings are consistent with prior work showing AI's potential to support clinical decision-making. For example, van der Meijden et al. (2023) reports that 97% of clinicians were familiar with AI, and 86% believed it could aid their work. As emphasized by Tonekaboni et al. (2019) and Wiens et al. (2019), IML models must undergo rigorous validation for performance, interpretability, and legal compliance. Trust in IML grows when clinicians understand model behavior and see outputs align with their expertise (Kelly et al., 2019).

Future work should focus on developing IML methods that handle missing values transparently without relying heavily on imputation, instead emphasizing frequently observed features available at deployment time to better reflect clinical workflows. Promising directions include reinforcement learning to model the sequential nature of decision making, incorporating clinician intuition, and validating these methods across diverse healthcare systems to support broader adoption.

5.5 Paper E: How should we present history in interpretable models of clinical policies?

While Papers A–D focus on handling missingness in interpretable machine learning models, Paper E shifts the focus to learning clinical policies from patient time-series data using interpretable models. Modeling policies for sequential clinical decision making from observational data can help describe treatment patterns, standardize common practices, and evaluate alternative strategies. Across these tasks, interpretability of the policy model is essential. Building accurate models depends on how well the patient's state is captured, whether through sequence-based representations or carefully crafted summaries of their medical history. Although recent work (Deuschel et al., 2024; Pace, Chan & van der Schaar, 2022) has favored learned representations, the question of *how best to represent patient histories for interpretable policy modeling, and how much detail should such summaries retain remains open.* This work focuses on model fit and presents a systematic comparison of diverse approaches to summarizing patient history for interpretable modeling of clinical policies across four sequential decision-making tasks.



		$t=1$	$t=2$	$t=3$	$t=4$	$t=5$
Context X_t	Age	65	66	67	68	69
	CDAI	4.7	10.5	7.2	8.4	9.1
	Cancer	0	1	1	0	0
Action A_{t-1}	MTX	N/A	0	1	1	1
	TNF	N/A	1	0	0	1
	JAK	N/A	0	0	0	0

	$t=5$
Max Age	69
Max CDAI	10.5
Hx of Cancer	1
Hx of MTX	1
Hx of TNF	1
Hx of JAK	0

Figure 5.9: History truncation and history aggregation using the `max` operator applied to the history of a patient with rheumatoid arthritis. A rolling window of size three is used for the history truncation. The context X_t is a vector with three components, X_t^1, X_t^2, and X_t^3, representing the patient's age, clinical disease activity index (CDAI), and co-existence of cancer. The simplified action space consists of three therapies and their combinations: methotrexate (MTX), tumor necrosis factor (TNF) inhibitor, and Janus kinase (JAK) inhibitor.

The literature describes two common approaches to representing patient history for interpretable modeling of clinical policies: *(1) learned sequence representations* and *(2) hand-crafted summary features*, typically formed through *history truncation* or *history aggregation*. Sequence models such as recurrent neural networks (RNNs) can be used to learn compact summaries of patient histories, which serve as the state S_t. Alternatively, hand-crafted features derived from truncated or aggregated histories can be used to fit simpler, interpretable models such as linear or rule-based classifiers.

We specifically study how history summaries can be constructed and used in policy modeling. *History truncation* involves selecting a fixed-size window of the

Table 5.3: An overview of the models used in our experiments. Medical decisions (actions) $A_t \in \mathcal{A} = \{1, \ldots, K\}$, recorded at each stage $t = 1, \ldots, T$ of treatment.Truncated history, H_t

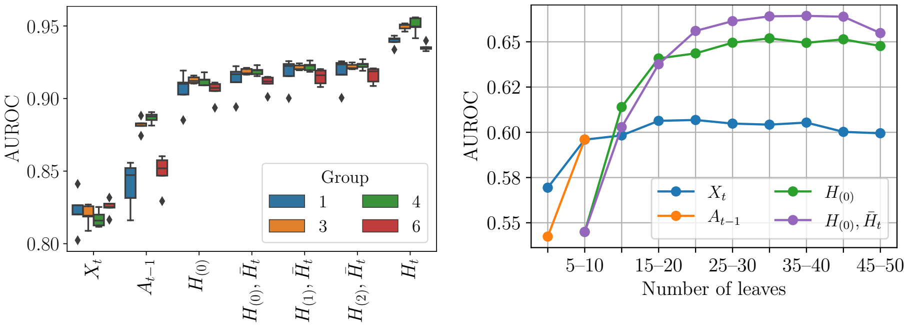
Model	Interpretable policy	Accepts $\|\mathcal{A}\| > 2$	Accepts H_t
Risk scores (RS)	✓	✗	✗
Logistic regression (LR)	✓	✓	✗
Decision tree (DT)	✓	✓	✗
Multilayer perceptron (MLP)	✗	✓	✗
Contextualized policy recovery (CPR)	✓	✗	✓
Prototypical sequence network (PSN)	✓	✓	✓
Recurrent decision tree (RDT)	✓	✓	✓
Recurrent neural network (RNN)	✗	✓	✓

most recent events, under the assumption that distant history has less impact on current decisions. *History aggregation*, in contrast, compresses historical information—such as diagnoses or treatments—into coarse summary features, disregarding temporal order. Figure 5.9 illustrates these representation settings.

To evaluate these strategies, we compare eight history representations across four clinical decision-making tasks: therapy selection for rheumatoid arthritis, MRI ordering for suspected Alzheimer's disease, and ICU management of sepsis and COPD exacerbations. We assess interpretable models trained on both learned and hand-crafted history representations, including risk scores (RS)(Ustun & Rudin, 2019), logistic regression(Feng et al., 2012; Spreeuwenberg et al., 2010), and decision trees (Banerjee et al., 2019; Chrimes et al., 2023), with multi-layer perceptrons (MLPs) serving as black-box baselines. For sequential decision-making tasks, we further include models designed to capture temporal dependencies: contextualized policy recovery (CPR)(Deuschel et al., 2024), prototypical sequence networks(Ming et al., 2019), recurrent decision trees (Pace, Chan & van der Schaar, 2022), and RNNs as non-interpretable benchmarks. An overview of all models is provided in Figure 5.3.

The study shows that interpretable models can achieve strong performance in clinical policy modeling—nearly matching black-box models—when patient history is represented with care. Specifically, combining current observations, the most recent treatment, and aggregated historical data provides sufficient context for accurate decision modeling across diverse tasks. However, simple representations using only the previous action (A_{t-1}) can be misleading: while they perform well on average, they fail in critical subgroups or early treatment stages where decisions are more variable. Figure 5.10(a) illustrates this clearly in the Sepsis task: patients with unstable conditions (based on NEWS2 score changes) require richer history representations for reliable predictions, as A_{t-1} alone underperforms significantly.

Figure 5.10(b) shows how the performance of decision trees varies with their complexity, measured by the number of leaves, for different state representations in RA. Since we could not control the number of leaves directly, we trained 500 different models for each state representation, using randomly selected hyperparameters. We then binned the models based on their complexity and



(a) AUROC across states and patient groups, identified based on the rate of change of the NEWS2 score, in Sepsis. PSN is used with $S_t = H_t$, LR with the others.

(b) Therapy switch prediction for RA. AUROC against the number of leaves for DT fit using different states. With $S_t = A_{t-1}$, the tree can have at most 5–10 leaves.

Figure 5.10: Experimental results based on RA and Sepsis data sets.

selected the best-performing model in each “complexity bucket” (e.g., 10–20 leaves) to present in the figure. We only performed this experiment for a single split of the data.

This work shows that simple summary features, especially recent treatments and aggregates, can match the performance of black-box models. While interpretable models generally performed well, limitations include potential unmeasured confounding, and a narrow set of history summaries. Future work could explore richer historical features and stage-aware policy models to reduce bias in use cases like policy evaluation.

Chapter 6

Concluding Remarks and Future Work

In this thesis, we study making predictions when missing values are present during both training and test time, and present several methods that maintain high predictive performance while ensuring interpretable model outputs. Established approaches often introduce bias through imputation or add complexity via missingness indicators. Moreover, current methods fall short of providing the interpretability needed for trust and accountability in high-stakes decision-making. As discussed in Chapter 4, these limitations motivate the need for alternative approaches. Our methods address this need by either leveraging missingness patterns directly or minimizing reliance on imputed values. In doing so, they enable more transparent and high-performing predictions. These contributions bridge theoretical foundations with practical deployment, supporting decision-making and advancing the field toward human-centered AI in real-world applications.

A key implication of our approaches, primarily described in Papers B and C, is that imputed values are not treated as equivalent to observed ones. Unlike most existing methods that optimize purely for predictive accuracy, e.g., by imputing missing values and proceeding as if they were real observations (Josse et al., 2024; van Buuren, 2018), our models are designed to express a preference for observed over imputed values. This distinction enables more efficient and informed use of available data, encouraging reliance on routinely collected, high-confidence features. Practically, this can help avoid the need for costly or invasive procedures, such as MRIs or biopsies, solely to fill in missing values. For instance, rather than imputing a missing MRI scan, the model may rely on routinely collected and lower-cost features, such as blood tests, vital signs, or patient history, that still contribute meaningful predictive information, even if they capture different aspects of the clinical picture. By prioritizing features typically available at test time, our methods align more closely with clinical workflows and deployment constraints. Unlike traditional approaches that rely on imputation or missingness indicators during feature selection, we support flexible variable selection tailored to real-world conditions.

6.1 Future Directions

As a future direction for the tree-based MA models described in Paper C, one approach to eliminate dependence on missing values at test time is to enforce missingness reliance, $\hat{\rho} = 0$, by adopting a fallback strategy: halt the decision process and return the label of the current node whenever a missing value is encountered. For Paper D, which focused on a network of trauma clinicians in France, extending the study to other healthcare systems in different countries would add significant value. It would allow us to assess the generalizability of our findings and compare how clinicians across systems handle missing values. Additionally, it would help us understand how models like MINTY or MA perform under varying clinical documentation practices. Such external validation also enables us to study how clinicians interact with interpretable machine learning systems in different settings. It offers insights into trust, usability, and workflow integration, and fosters communication and reasoning about missing values. These findings can inform the design of models that support–rather than disrupt–clinical decision-making, advancing robust, interpretable systems for real-world human–AI collaboration in healthcare.

An important challenge for models that reason over missing features is their vulnerability to distribution shifts, particularly in the medical domain (Sperrin et al., 2020). The interpretability offered by methods such as SPSM, MINTY, and MA learning is essential for detecting and adapting to such shifts. Future work could further investigate how changes in missingness patterns contribute to distribution shift and affect model reliability.

While Paper E investigates time-series data of patient histories using interpretable models, a promising direction is to extend MA-learning to temporal settings. As motivated in that paper, many health care problems are inherently sequential, where the timing and order of events, such as medication administration, lab tests, or symptoms onset, are critical for accurate predictions (Chakraborty & Moodie, 2013; Gottesman et al., 2019). The reasons for missingness are similar to those in non-sequential data, often due to missed appointments, delayed diagnostics, mislabeled samples, or sensor failures (Madden et al., 2016). These gaps are not only common but often informative (Groenwold, 2020), e.g., deteriorating patients may skip follow-ups or require emergency visits, making it crucial to design methods that leverage and remain robust to such patterns. An open question is whether MA learning principles can also benefit black-box models, such as recurrent neural networks or transformers. Although these models excel at capturing temporal dependencies, they typically rely on masking or imputation without modeling missingness mechanisms explicitly (Che et al., 2018). Incorporating MA-inspired objectives, such as auxiliary losses that penalize prediction variability across imputed or partially observed inputs or encourage shared representations across missingness patterns, could improve flexibility and interpretability.

Bibliography

Adebayo, J., Muelly, M., Liccardi, I., & Kim, B. (2020). Debugging tests for model explanations. *Advances in Neural Information Processing Systems*, *33*.

Afessa, B., Keegan, M. T., Gajic, O., Hubmayr, R. D., & Peters, S. G. (2005). The influence of missing components of the acute physiology score of apache iii on the measurement of icu performance. *Intensive care medicine*, *31*, 1537–1543.

Ahmad, M. A., Eckert, C., & Teredesai, A. (2018). Interpretable machine learning in healthcare. *Proceedings of the 2018 ACM International Conference on Bioinformatics, Computational Biology, and Health Informatics*, 559–560.

Antunes, P., Herskovic, V., Ochoa, S. F., & Pino, J. A. (2008). Structuring dimensions for collaborative systems evaluation. *ACM computing surveys (CSUR)*, *44*(2), 1–28.

Banerjee, M., Reynolds, E., Andersson, H. B., & Nallamothu, B. K. (2019). Tree-based analysis: A practical approach to create clinical decision-making tools. *Circulation: Cardiovascular Quality and Outcomes*, *12*(5), e004879.

Bénard, C., Biau, G., Da Veiga, S., & Scornet, E. (2021a). Interpretable random forests via rule extraction. *International Conference on Artificial Intelligence and Statistics*, 937–945.

Bénard, C., Biau, G., Da Veiga, S., & Scornet, E. (2021b). Sirus: Stable and interpretable rule set for classification. *Electronic Journal of Statistics*, *15*, 427–505.

Biran, O., & Cotton, C. (2017). Explanation and justification in machine learning: A survey. *IJCAI-17 workshop on explainable AI (XAI)*, *8*(1), 8–13.

Breiman, L. (2001). Random forests. *Machine learning*, *45*, 5–32.

Breiman, L. (2017). *Classification and regression trees*. Routledge.

Buck, S. F. (1960). A method of estimation of missing values in multivariate data suitable for use with an electronic computer. *Journal of the Royal Statistical Society: Series B (Methodological)*, *22*(2), 302–306.

Carpenter, J. R., Bartlett, J. W., Morris, T. P., Wood, A. M., Quartagno, M., & Kenward, M. G. (2023). *Multiple imputation and its application*. John Wiley & Sons.

Caruana, R., Lou, Y., Gehrke, J., Koch, P., Sturm, M., & Elhadad, N. (2015). Intelligible models for healthcare: Predicting pneumonia risk and hospital 30-day readmission. *Proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining*, 1721–1730.

Chakraborty, B., & Moodie, E. E. (2013). *Statistical methods for dynamic treatment regimes*. Springer.

Che, Z., Purushotham, S., Cho, K., Sontag, D., & Liu, Y. (2018). Recurrent neural networks for multivariate time series with missing values. *Scientific reports*, *8*(1), 1–12.

Chen, T., & Guestrin, C. (2016). Xgboost: A scalable tree boosting system. *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, 785–794.

Chen, Z., Tan, S., Chajewska, U., Rudin, C., & Caruana, R. (2023). Missing values and imputation in healthcare data: Can interpretable machine learning help? *Conference on Health, Inference, and Learning*, 86–99.

Chourib, I. (2025). Missing data handling: A comprehensive review, taxonomy, and comparative evaluation. *Journal of Computer and Communications*, *13*(6), 81–102.

Chrimes, D., et al. (2023). Using decision trees as an expert system for clinical decision support for covid-19. *Interactive Journal of Medical Research*, *12*(1), e42540.

Covert, I., Lundberg, S., & Lee, S.-I. (2021). Explaining by removing: A unified framework for model explanation. *Journal of Machine Learning Research*, *22*(209), 1–90.

Cowan, N. (2010). The magical mystery four: How is working memory capacity limited, and why? *Current directions in psychological science*, *19*(1), 51–57.

Dansson, H. V., Stempfle, L., Egilsdóttir, H., Schliep, A., Portelius, E., Blennow, K., Zetterberg, H., Johansson, F. D., & (ADNI), A. D. N. I. (2021). Predicting progression and cognitive decline in amyloid-positive patients with alzheimer's disease. *Alzheimer's Research & Therapy*, *13*, 1–16.

Dasu, T., & Johnson, T. (2003). *Exploratory data mining and data cleaning*. John Wiley & Sons.

de Goeij, M. C., Van Diepen, M., Jager, K. J., Tripepi, G., Zoccali, C., & Dekker, F. W. (2013). Multiple imputation: Dealing with missing data. *Nephrology Dialysis Transplantation*, *28*(10), 2415–2420.

Deuschel, J., Ellington, C., Luo, Y., Lengerich, B., Friederich, P., & Xing, E. P. (2024). Contextualized policy recovery: Modeling and interpreting medical decisions with adaptive imitation learning. *Proceedings of the 41st International Conference on Machine Learning*.

Doshi-Velez, F., & Kim, B. (2017). Towards a rigorous science of interpretable machine learning. *arXiv: Machine Learning*.

Ehrlinger, L., Grubinger, T., Varga, B., Pichler, M., Natschläger, T., & Zeindl, J. (2018). Treating missing data in industrial data analytics. *2018 Thirteenth International Conference on Digital Information Management (ICDIM)*, 148–155.

Emmanuel, T., Maupong, T., Mpoeleng, D., Semong, T., Mphago, B., & Tabona, O. (2021). A survey on missing data in machine learning. *Journal of Big data*, *8*(1), 140.

Feng, P., Zhou, X.-H., Zou, Q.-M., Fan, M.-Y., & Li, X.-S. (2012). Generalized propensity score for estimating the average treatment effect of multiple treatments. *Statistics in Medicine*, *31*(7), 681–697.

Friedman, J. H. (2001). Greedy function approximation: A gradient boosting machine. *Annals of statistics*, 1189–1232.

Friedman, J. H., & Popescu, B. E. (2008). Predictive learning via rule ensembles. *The annals of applied statistics*, 916–954.

Gottesman, O., Johansson, F., Komorowski, M., Faisal, A., Sontag, D., Doshi-Velez, F., & Celi, L. A. (2019). Guidelines for reinforcement learning in healthcare. *Nature Medicine*, *25*(1), 16–18.

Grinsztajn, L., Oyallon, E., & Varoquaux, G. (2022). Why do tree-based models still outperform deep learning on typical tabular data? *Proceedings of the 36th International Conference on Neural Information Processing Systems*.

Groenwold, R. H. (2020). Informative missingness in electronic health record systems: The curse of knowing. *Diagnostic and prognostic research*, *4*(1), 8.

Hastie, T., Tibshirani, R., & Friedman, J. H. (2009). *The elements of statistical learning: Data mining, inference, and prediction* (Vol. 2). Springer.

Ivarsson Orrelid, C., Rosberg, O., Weiner, S., Johansson, F. D., Gobom, J., Zetterberg, H., Mwai, N., & Stempfle, L. (2025). Applying machine learning to high-dimensional proteomics datasets for the identification of alzheimer's disease biomarkers. *Fluids and Barriers of the CNS*, *22*(1), 23.

Jamshidian, M., & Schott, J. R. (2007). Testing equality of covariance matrices when data are incomplete. *Computational statistics & data analysis*, *51*(9), 4227–4239.

Janssen, K. J., Donders, A. R. T., Harrell Jr, F. E., Vergouwe, Y., Chen, Q., Grobbee, D. E., & Moons, K. G. (2010). Missing covariate data in medical research: To impute is better than to ignore. *Journal of clinical epidemiology*, *63*(7), 721–727.

Janssen, K. J., Vergouwe, Y., Donders, A. R. T., Harrell Jr, F. E., Chen, Q., Grobbee, D. E., & Moons, K. G. (2009). Dealing with missing predictor values when applying clinical prediction models. *Clinical chemistry*, *55*(5), 994–1001.

Josse, J., Chen, J. M., Prost, N., Varoquaux, G., & Scornet, E. (2024). On the consistency of supervised learning with missing values. *Statistical Papers*, *65*(9), 5447–5479.

Josse, J., & Husson, F. (2012). Handling missing values in exploratory multivariate data analysis methods. *Journal de la Société Française de Statistique*, *153*(2), 79–99.

Josse, J., & Husson, F. (2016). Missmda: A package for handling missing values in multivariate data analysis. *Journal of statistical software*, *70*, 1–31.

Josse, J., Mayer, I., Tierney, N., & Vialaneix, N. (2025). Cran task view: Missing data [Accessed July 2025].

Josse, J., Prost, N., Scornet, E., & Varoquaux, G. (2019). On the consistency of supervised learning with missing values. *arXiv preprint arXiv:1902.06931*.

Kapelner, A., & Bleich, J. (2015). Prediction with missing data via bayesian additive regression trees. *Canadian Journal of Statistics*, *43*(2), 224–239.

Kaur, H., Conrad, M. R., Rule, D., Lampe, C., & Gilbert, E. (2024). Interpretability gone bad: The role of bounded rationality in how practitioners understand machine learning. *Proc. ACM Hum.-Comput. Interact.*, *8*(CSCW1).

Kelly, C. J., Karthikesalingam, A., Suleyman, M., Corrado, G., & King, D. (2019). Key challenges for delivering clinical impact with artificial intelligence. *BMC medicine*, *17*, 1–9.

Kim, B., Khanna, R., & Koyejo, O. O. (2016). Examples are not enough, learn to criticize! criticism for interpretability. *Advances in neural information processing systems*, *29*.

Knol, M. J., Janssen, K. J., Donders, A. R. T., Egberts, A. C., Heerdink, E. R., Grobbee, D. E., Moons, K. G., & Geerlings, M. I. (2010). Unpredictable bias when using the missing indicator method or complete case analysis for missing confounder values: An empirical example. *Journal of clinical epidemiology*, *63*(7), 728–736.

Koh, P. W., & Liang, P. (2017). Understanding black-box predictions via influence functions. *International conference on machine learning*, 1885–1894.

Le Morvan, M., Josse, J., Moreau, T., Scornet, E., & Varoquaux, G. (2020). Neumiss networks: Differentiable programming for supervised learning with missing values. *Advances in Neural Information Processing Systems*, *33*, 5980–5990.

Le Morvan, M., Josse, J., Scornet, E., & Varoquaux, G. (2021). What's a good imputation to predict with missing values? In M. Ranzato, A. Beygelzimer, Y. Dauphin, P. Liang & J. W. Vaughan (Eds.), *Advances in neural information processing systems* (pp. 11530–11540, Vol. 34). Curran Associates, Inc.

Le Morvan, M., Prost, N., Josse, J., Scornet, E., & Varoquaux, G. (2020, August). Linear predictor on linearly-generated data with missing values: Non consistency and solutions. In S. Chiappa & R. Calandra (Eds.), *Proceedings of the twenty third international conference on artificial intelligence and statistics* (pp. 3165–3174, Vol. 108). PMLR.

Letham, B., Rudin, C., McCormick, T. H., & Madigan, D. (2015). Interpretable classifiers using rules and bayesian analysis: Building a better stroke prediction model.

Lewis, R. J. (2000). An introduction to classification and regression tree (cart) analysis. *Annual meeting of the society for academic emergency medicine in San Francisco, California*, *14*.

Lipton, Z. C. (2018). The mythos of model interpretability: In machine learning, the concept of interpretability is both important and slippery. *Queue*, *16*(3), 31–57.

Little, R. J. (1988). A test of missing completely at random for multivariate data with missing values. *Journal of the American statistical Association*, *83*(404), 1198–1202.

Little, R. J. (1993). Pattern-mixture models for multivariate incomplete data. *Journal of the American Statistical Association*, *88*(421), 125–134.

Little, R. J., & Rubin, D. B. (2019). *Statistical analysis with missing data* (Vol. 793). John Wiley & Sons.

Liu, N., Kumara, S., & Reich, E. (2021). Gaining insights into patient satisfaction through interpretable machine learning. *IEEE Journal of Biomedical and Health Informatics*, *25*(6), 2215–2226.

Lundberg, S. M., Erion, G., & Lee, S.-I. (2020). From local explanations to global understanding with explainable ai for trees. *Nature Machine Intelligence*, *2*(1), 56–67.

Lundberg, S. M., & Lee, S.-I. (2017). A unified approach to interpreting model predictions. *Advances in neural information processing systems*, *30*.

Madden, J. M., Lakoma, M. D., Rusinak, D., Lu, C. Y., & Soumerai, S. B. (2016). Missing clinical and behavioral health data in a large electronic health record (ehr) system. *Journal of the American Medical Informatics Association*, *23*(6), 1143–1149.

Mannhardt, F., De Leoni, M., Reijers, H. A., & Van Der Aalst, W. M. (2016). Decision mining revisited-discovering overlapping rules. *International conference on advanced information systems engineering*, 377–392.

Margot, V., & Luta, G. (2021). A new method to compare the interpretability of rule-based algorithms. *AI*, *2*(4), 621–635.

Marston, L., Carpenter, J. R., Walters, K. R., Morris, R. W., Nazareth, I., & Petersen, I. (2010). Issues in multiple imputation of missing data for large general practice clinical databases. *Pharmacoepidemiology and drug safety*, *19*(6), 618–626.

Matsson, A., Stempfle, L., Rao, Y., Margolin, Z. R., Litman, H. J., & Johansson, F. D. (2025, December). How should we represent history in interpretable models of clinical policies? In S. Hegselmann, H. Zhou, E. Healey, T. Chang, C. Ellington, V. Mhasawade, S. Tonekaboni, P. Argaw & H. Zhang (Eds.), *Proceedings of the 4th machine learning for health symposium* (pp. 714–734, Vol. 259). PMLR.

Mattei, P.-A., & Frellsen, J. (2019). Miwae: Deep generative modelling and imputation of incomplete data sets. *International conference on machine learning*, 4413–4423.

Mazumder, R., Hastie, T., & Tibshirani, R. (2010). Spectral regularization algorithms for learning large incomplete matrices. *The Journal of Machine Learning Research*, *11*, 2287–2322.

McCullagh, P. (2019). *Generalized linear models*. Routledge.

McTavish, H., Donnelly, J., Seltzer, M., & Rudin, C. (2024). Interpretable generalized additive models for datasets with missing values.

Mealli, F., & Rubin, D. B. (2015). Clarifying missing at random and related definitions. *Statistical Science*, *30*(3), 257–269.

Mercaldo, S. F., & Blume, J. D. (2020). Missing data and prediction: the pattern submodel. *Biostatistics*, *21*(2), 236–252.

Miller, G. A. (1956). The magical number seven, plus or minus two: Some limits on our capacity for processing information. *Psychological review*, *63*(2), 81.

Miller, T. (2019). Explanation in artificial intelligence: Insights from the social sciences. *Artificial intelligence*, *267*, 1–38.

Ming, Y., Xu, P., Qu, H., & Ren, L. (2019). Interpretable and steerable sequence learning via prototypes. *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 903–913.

Mohan, K., & Pearl, J. (2021). Graphical models for processing missing data. In *Handbook of graphical models* (pp. 275–299). Chapman; Hall/CRC.

Mohan, K., Pearl, J., & Tian, J. (2013). Graphical models for inference with missing data. *Advances in Neural Information Processing Systems*, *26*.

Molenberghs, G., Fitzmaurice, G., Kenward, M. G., Tsiatis, A. A., & Verbeke, G. (2014). *Handbook of missing data methodology*. CRC Press.

Molnar, C. (2020). *Interpretable machine learning*. Lulu. com.

Molnar, C., & Freiesleben, T. (2024). *Supervised machine learning for science: How to stop worrying and love your black box*. Christoph Molnar.

Murdoch, W. J., Singh, C., Kumbier, K., Abbasi-Asl, R., & Yu, B. (2019). Definitions, methods, and applications in interpretable machine learning. *Proceedings of the National Academy of Sciences*, *116*(44), 22071–22080.

Nelder, J. A., & Wedderburn, R. W. (1972). Generalized linear models. *Journal of the Royal Statistical Society: Series A (General)*, *135*(3), 370–384.

Pace, A., Chan, A., & van der Schaar, M. (2022). POETREE: Interpretable policy learning with adaptive decision trees. *Proceedings of the 10th International Conference on Learning Representations*.

Pedersen, A. B., Mikkelsen, E. M., Cronin-Fenton, D., Kristensen, N. R., Pham, T. M., Pedersen, L., & Petersen, I. (2017). Missing data and multiple imputation in clinical epidemiological research. *Clinical epidemiology*, 157–166.

Poursabzi-Sangdeh, F., Goldstein, D. G., Hofman, J. M., Wortman Vaughan, J. W., & Wallach, H. (2021). Manipulating and measuring model interpretability. *Proceedings of the 2021 CHI conference on human factors in computing systems*, 1–52.

Ribeiro, M. T., Singh, S., & Guestrin, C. (2016). " why should i trust you?" explaining the predictions of any classifier. *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, 1135–1144.

Rubin, D. B. (1976). Inference and missing data. *Biometrika*, *63*(3), 581–592.

Rudin, C. (2019). Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nature machine intelligence*, *1*(5), 206–215.

Rudin, C., Chen, C., Chen, Z., Huang, H., Semenova, L., & Zhong, C. (2022). Interpretable machine learning: Fundamental principles and 10 grand challenges. *Statistic Surveys*, *16*, 1–85.

Rudin, C., Wang, C., & Coker, B. (2020). The Age of Secrecy and Unfairness in Recidivism Prediction. *Harvard Data Science Review*, *2*(1).

Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1986). Learning representations by back-propagating errors. *nature*, *323*(6088), 533–536.

Schafer, J. L. (1997). *Analysis of incomplete multivariate data*. CRC press.

Schafer, J. L., & Graham, J. W. (2002). Missing data: Our view of the state of the art. *Psychological methods*, *7*(2), 147.

Seaman, S., Galati, J., Jackson, D., & Carlin, J. (2013). What is meant by "missing at random"? *Statistical Science*, *28*(2), 257–268.

Shadbahr, T., Roberts, M., Stanczuk, J., Gilbey, J., Teare, P., Dittmer, S., Thorpe, M., Torné, R. V., Sala, E., Lió, P., et al. (2023). The impact of imputation quality on machine learning classifiers for datasets with missing values. *Communications Medicine*, *3*(1), 139.

Slack, D., Hilgard, S., Jia, E., Singh, S., & Lakkaraju, H. (2020). Fooling lime and shap: Adversarial attacks on post hoc explanation methods. *Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society*, 180–186.

Souillard-Mandar, W., Davis, R., Rudin, C., Au, R., Libon, D. J., Swenson, R., Price, C. C., Lamar, M., & Penney, D. L. (2016). Learning classification models of cognitive conditions from subtle behaviors in the digital clock drawing test. *Machine learning*, *102*, 393–441.

Souza, V. F., Cicalese, F., Laber, E., & Molinaro, M. (2022). Decision trees with short explainable rules. *Advances in neural information processing systems*, *35*, 12365–12379.

Sperrin, M., Martin, G. P., Sisk, R., & Peek, N. (2020). Missing data should be handled differently for prediction than for description or causal explanation. *Journal of Clinical Epidemiology*, *125*, 183–187.

Spreeuwenberg, M. D., Bartak, A., Croon, M. A., Hagenaars, J. A., Busschbach, J. J., Andrea, H., Twisk, J., & Stijnen, T. (2010). The multiple propensity score as control for bias in the comparison of more than two treatment arms: An introduction from a case study in mental health. *Medical Care*, *48*(2), 166–174.

Stekhoven, D. J., & Bühlmann, P. (2012). Missforest—non-parametric missing value imputation for mixed-type data. *Bioinformatics*, *28*(1), 112–118.

Stempfle, L., James, A., Josse, J., Gauss, T., & Johansson, F. D. (2024). Handling missing values in clinical machine learning: Insights from an expert study [Also presented at Findings of the 4th Machine Learning for Health Symposium (ML4H), 2024]. *arXiv preprint arXiv:2411.09591*.

Stempfle, L., & Johansson, F. D. (2024a). Learning replacement variables in interpretable rule-based models [a]. *ICML 3rd Workshop on Interpretable Machine Learning in Healthcare (IMLH)*.

Stempfle, L., & Johansson, F. D. (2024b). Minty: Rule-based models that minimize the need for imputing features with missing values [b]. *In-*

ternational Conference on Artificial Intelligence and Statistics, 964–972.

Stempfle, L., Matsson, A., Mwai, N., & Johansson, F. D. (2025). Prediction models that learn to avoid missing values. *To appear in Forty-second International Conference on Machine Learning, 267.*

Stempfle, L., Panahi, A., & Johansson, F. D. (2023). Sharing pattern submodels for prediction with missing values. *Proceedings of the AAAI Conference on Artificial Intelligence*, *37*(8), 9882–9890.

Stiglic, G., Kocbek, P., Fijacko, N., Zitnik, M., Verbert, K., & Cilar, L. (2020). Interpretability of machine learning-based prediction models in healthcare. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, *10*(5), e1379.

Struck, A. F., Tabaeizadeh, M., Schmitt, S. E., Ruiz, A. R., Swisher, C. B., Subramaniam, T., Hernandez, C., Kaleem, S., Haider, H. A., Cissé, A. F., et al. (2020). Assessment of the validity of the 2helps2b score for inpatient seizure risk prediction. *JAMA neurology*, *77*(4), 500–507.

Takada, M., Suzuki, T., & Fujisawa, H. (2020). Independently interpretable lasso for generalized linear models. *Neural computation*, *32*(6), 1168–1221.

Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, *58*(1), 267–288.

Tonekaboni, S., Joshi, S., McCradden, M. D., & Goldenberg, A. (2019). What clinicians want: Contextualizing explainable machine learning for clinical end use. *Machine learning for healthcare conference*, 359–380.

Troyanskaya, O., Cantor, M., Sherlock, G., Brown, P., Hastie, T., Tibshirani, R., Botstein, D., & Altman, R. B. (2001). Missing value estimation methods for dna microarrays. *Bioinformatics*, *17*(6), 520–525.

Twala, B. E., Jones, M., & Hand, D. J. (2008). Good methods for coping with missing data in decision trees. *Pattern Recognition Letters*, *29*(7), 950–956.

Unwin, A. (2020). Visualizing missing values: The visna plot. *Workshop on the Art of Learning with Missing Values (Artemiss), ICML.*

Ustun, B., & Rudin, C. (2016). Supersparse linear integer models for optimized medical scoring systems. *Machine Learning*, *102*, 349–391.

Ustun, B., & Rudin, C. (2019). Learning optimized risk scores. *Journal of Machine Learning Research*, *20*(150), 1–75.

Ustun, B., Traca, S., & Rudin, C. (2013). Supersparse linear integer models for predictive scoring systems. *AAAI (Late-Breaking Developments).*

Ustun, B., Westover, M. B., Rudin, C., & Bianchi, M. T. (2016). Clinical prediction models for sleep apnea: The importance of medical history over symptoms. *Journal of Clinical Sleep Medicine*, *12*(2), 161–168.

Valdiviezo, H. C., & Van Aelst, S. (2015). Tree-based prediction on incomplete data using imputation or surrogate decisions. *Information Sciences*, *311*, 163–181.

van Buuren, S. (2018). *Flexible imputation of missing data, second edition (2nd ed.)* hapman; Hall.

Van Buuren, S. (2007). Multiple imputation of discrete and continuous data by fully conditional specification. *Statistical methods in medical research*, *16*(3), 219–242.

Van Buuren, S., & Groothuis-Oudshoorn, K. (2011). Mice: Multivariate imputation by chained equations in r. *Journal of statistical software*, *45*, 1–67.

van der Meijden, S. L., de Hond, A. A., Thoral, P. J., Steyerberg, E. W., Kant, I. M., Cinà, G., & Arbous, M. S. (2023). Intensive care unit physicians' perspectives on artificial intelligence–based clinical decision support tools: Preimplementation survey study. *JMIR human factors*, *10*, e39114.

Van Ness, M., Bosschieter, T. M., Halpin-Gregorio, R., & Udell, M. (2023). The missing indicator method: From low to high dimensions. *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 5004–5015.

Vellido, A. (2020). The importance of interpretability and visualization in machine learning for applications in medicine and health care. *Neural computing and applications*, *32*(24), 18069–18083.

Wang, F., & Rudin, C. (2015). Falling rule lists. *Artificial intelligence and statistics*, 1013–1022.

Wei, D., Dash, S., Gao, T., & Gunluk, O. (2019). Generalized linear rule models. *International Conference on Machine Learning*, 6687–6696.

Weiner, M. W., Aisen, P. S., Jack Jr., C. R., Jagust, W. J., Trojanowski, J. Q., Shaw, L., Saykin, A. J., Morris, J. C., Cairns, N., Beckett, L. A., Toga, A., Green, R., Walter, S., Soares, H., Snyder, P., Siemers, E., Potter, W., Cole, P. E., Schmidt, M., & Initiative, A. D. N. (2010). The alzheimer's disease neuroimaging initiative: Progress report and future plans. *Alzheimer's & Dementia*, *6*(3), 202–211.e7.

Wells, B. J., Chagin, K. M., Nowacki, A. S., & Kattan, M. W. (2013). Strategies for handling missing data in electronic health record derived data. *Egems*, *1*(3).

Wiens, J., Saria, S., Sendak, M., Ghassemi, M., Liu, V. X., Doshi-Velez, F., Jung, K., Heller, K., Kale, D., Saeed, M., et al. (2019). Do no harm: A roadmap for responsible machine learning for health care. *Nature medicine*, *25*(9), 1337–1340.

Wysocki, O., Davies, J. K., Vigo, M., Armstrong, A. C., Landers, D., Lee, R., & Freitas, A. (2023). Assessing the communication gap between ai models and healthcare professionals: Explainability, utility and trust in ai-driven clinical decision-making. *Artificial Intelligence*, *316*, 103839.

Yoon, J., Jordon, J., & Schaar, M. (2018). Gain: Missing data imputation using generative adversarial nets. *International conference on machine learning*, 5689–5698.

Yu, H.-F., Rao, N., & Dhillon, I. S. (2016). Temporal regularized matrix factorization for high-dimensional time series prediction. *Advances in neural information processing systems*, *29*.

Zaffran, M., Dieuleveut, A., Josse, J., & Romano, Y. (2023). Conformal prediction with missing values. *Proceedings of the 40th International Conference on Machine Learning.*