UNIVERSITY OF CALIFORNIA
Los Angeles

Discovering Data-Driven Actionable Intelligence for Clinical Decision Support

A dissertation submitted in partial satisfaction
of the requirements for the degree
Doctor of Philosophy in Electrical and Computer Engineering

by

Ahmed M. Alaa H. H. Ibrahim

2019

ABSTRACT OF THE DISSERTATION

Discovering Data-Driven Actionable Intelligence
for Clinical Decision Support

by

Ahmed M. Alaa H. H. Ibrahim
Doctor of Philosophy in Electrical and Computer Engineering
University of California, Los Angeles, 2019
Professor Mihaela van der Schaar, Chair

The rapid digitization of healthcare has led to a proliferation of clinical data, manifesting through electronic health records, biorepositories, and disease registries. This dissertation addresses the question of how machine learning (ML) techniques can capitalize on these data resources to assist clinicians in predicting, preventing and treating illness. To this end, we develop a set of ML-based, data-driven models of patient outcomes that we envision to be embedded within systems of decision support deployed at different stages of patient care.

We focus on two broad setups for analyzing clinical data: (1) the *cross-sectional* setup wherein data is collected by observing many patients at a particular point of time, and (2) the *longitudinal* setup in which repeated observations of the same patient are collected over time. In both setups, we develop models that are: (a) capable of answering *counter-factual* questions, i.e., can predict outcomes under alternative treatment scenarios, (b) *interpretable* in the sense that clinicians can understand how the model predictions for individual patients are issued, and (c) *automated* in the sense that they adaptively tune their modeling choices for the dataset at hand, with little or no need for expert intervention. Models satisfying these *three* requirements would enable the realization of actionable, transparent and automated decision support systems that operate symbiotically within existing clinical workflows.

Our technical contributions are multi-faceted. In the cross-sectional data setup, we develop ML models that fulfill the aforementioned requirements (a)-(c) as follows. We start by developing a

comprehensive theoretical framework for causal inference, whereby we quantify the limits to how well ML models can recover the causal effects of counter-factual treatment decisions on individual patients using observational (retrospective) data, and we build ML models — based on *Gaussian processes* — that achieve these limits. Next, we develop a novel *symbolic meta-modeling* approach for interpreting the predictions of any ML-based prognostic model by converting the "black-box" model into an understandable symbolic equation that relates patients' features to their predicted outcomes. Finally, we develop a model selection approach based on *Bayesian optimization* that enables the automation of predictive and causal modeling. In the longitudinal data setup, we develop a novel deep probabilistic model for sequential clinical data that satisfies requirements (a)-(c) by capitalizing on the strengths of both state-space models and deep recurrent neural networks.

To demonstrate the utility of our models, we evaluate their performance on various real-world datasets for cohorts of breast cancer, cardiovascular disease and cystic fibrosis patients. We show that, compared to existing clinical scorers, our ML-based models can improve the accuracy of predicting individual-level prognoses, guide treatment decisions for individual patients, and provide insights into underlying disease mechanisms.

The dissertation of Ahmed M. Alaa H. H. Ibrahim is approved.

Greg Pottie

Yahya Rahmat-Samii

Patricia Ganz

Douglas Bell

Mihaela van der Schaar, Committee Chair

University of California, Los Angeles

2019

To my parents and my brothers.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGMENTS

I would not have been able to complete this dissertation without the help and support of my family, my adviser, my colleagues and friends. I am deeply grateful.

First, I wish to thank my adviser, Professor Mihaela van der Schaar, for her kindness and unyielding support. This dissertation would not have been possible without her research vision. Her new ideas, insights, opportunities and willingness to intrepidly explore multidisciplinary research terrains have had a profound impact on my development as a researcher. I am deeply grateful for her invaluable support and immense help. Thank you for everything, Mihaela. I also thank the other members of my dissertation committee, Professor Greg Pottie, Professor Yahya Rahmat-Samii, Professor Patricia Ganz, and Dr. Douglas Bell for their valuable perspectives and thoughtful feedback.

I would also like to thank all of my co-authors and collaborators at UCLA; Jinsung Yoon, Scott Hu, William Zame, Changhee Lee, Kartik Ahuja, William Hsu, Kyeong Moon and Martin Cadeiras. It has been a great privilege to be able to work with such talented people. I have also been fortunate to enjoy the collegiality of my labmates; Onur Atan, Trent Kyono, William Whoiles, Yangbo Song, Simpson Zhang, Jie Xu and Yuanzhang Xiao, with whom I had many insightful research discussions. I would also like to thank my friends at UCLA for their companionship and support. I especially want to thank Ahmed Hareedy, Mohammed Karmoose, Yahya Ezzeldin, Omar Hussien, Moustafa Alzantot, and Safa Cicek.

I have been so lucky to spend some time on the other side of the Atlantic as a visiting student at Oxford University. There, I was fortunate to work with my colleagues at Oxford University, Ioana Bica, James Jordon, Bryan Lim, and Michael Weisz, and my colleagues at Cambridge University, Yao Zhang, Zhaozhi Qian, Alexis Bellot, and Daniel Jarrett.

During my visit in the UK, I was blessed to be able to collaborate with many clinicians who provided me with the clinical data, guidance and feedback that made the clinical application of my work come to life. I would like to especially thank Dr. Jem Rashbass (National Director for Disease Registration at Public Health England) for granting me access to the UK breast cancer registry data, and Dr Janet Allen (Director of Strategic Innovation at the UK Cystic Fibrosis Trust) for enabling

my access to the UK Cystic Fibrosis registry. I would also like to thank my clinical collaborator at Oxford University, Professor Adrian Harris, my collaborators at Cambridge University, Professors Andres Floto, Fiona Gilbert, Emanuele Di Angelantonio, and Dr. James Rudd, and my collaborator at Queen Mary University of London, Deepti Gurdasani.

Finally, and most importantly, I wish to thank my parents, Hadeel and Alaa, and my brothers, Ali and Sherif for their love and support. No words can express what your encouragement and support have meant to me.

VITA

2011	Bachelor in Communications and Computer Engineering, Cairo University, Cairo, Egypt
2011–2014	Teaching Assistant, Communications Engineering Department, Cairo University, Cairo, Egypt
2014	Master of Science in Electronics and Communications Engineering, Cairo University, Cairo, Egypt
2014	Graduate Division Fellowship, Electrical Engineering, University of California, Los Angeles.
2014–2019	Research Assistant, University of California, Los Angeles.
2017–2018	Recognized PhD Student, University of Oxford, United Kingdom.

PUBLICATIONS

A. M. Alaa, T. Bolton, E. Di Angelantonio, J. H. F. Rudd, M. van der Schaar, "Cardiovascular Disease Risk Prediction using Automated Machine Learning: A Prospective Study of 423,604 UK Biobank Participants," *PloS One*, 2019.

A. M. Alaa, M. van der Schaar, "Demystifying Black-box Models with Symbolic Metamodels," *Neural Information Processing Systems* (NeurIPS), 2019.

A. M. Alaa, M. van der Schaar, "Attentive State-Space Modeling of Disease Progression," *Neural Information Processing Systems* (NeurIPS), 2019.

A. M. Alaa, M. van der Schaar, "Validating Causal Inference Models via Influence Functions," *International Conference on Machine Learning* (ICML), 2019.

I. Bica, **A. M. Alaa**, M. van der Schaar, "Estimating Counterfactual Treatment Outcomes over Time through Adversarially Balanced Representations," *NeurIPS Machine Learning for Health Workshop*, 2019.

I. Bica, **A. M. Alaa**, M. van der Schaar, "Time Series Deconfounder: Estimating Treatment Effects over Time in the Presence of Hidden Confounders," *NeurIPS Machine Learning for Health Workshop*, 2019.

C. Lee, W. R. Zame, **A. M. Alaa**, M. van der Schaar, "Temporal Quilting for Survival Analysis," *International Conference on Artificial Intelligence and Statistics* (AISTATS), 2019.

A. M. Alaa, M. van der Schaar, "Prognostication and Risk Factors for Cystic Fibrosis via Automated Machine Learning," *Scientific Reports*, 2018.

A. M. Alaa, M. van der Schaar, "Bayesian Nonparametric Causal Inference: Information Rates & Learning Algorithms," *IEEE Journal of Selected Topics in Signal Processing*, 2018.

J. Yoon, W. R. Zame, A. Banerjee, M. Cadeiras, **A. M. Alaa**, M. van der Schaar, "Personalized survival predictions via Trees of Predictors: An application to cardiac transplantation," *PloS One*, 2018.

B. Lim, **A. M. Alaa**, M. van der Schaar, "Forecasting Treatment Responses Over Time Using Recurrent Marginal Structural Networks," *Neural Information Processing Systems* (NeurIPS), 2018.

A. M. Alaa, M. van der Schaar, "AutoPrognosis: Automated Clinical Prognostic Modeling via Bayesian Optimization with Structured Kernel Learning," *International Conference on Machine Learning* (ICML), 2018.

A. M. Alaa, M. van der Schaar, "Limits of Estimating Heterogeneous Treatment Effects: Guidelines for Practical Algorithm Design," *International Conference on Machine Learning* (ICML), 2018.

A. M. Alaa, T. Daniels, R. Floto, M. van der Schaar, "Machine Learning-Based Predictions of Prognosis in Cystic Fibrosis," *Pediatric Pulmonology*, 2018. **(Abstract)**

A. M. Alaa, T. Bolton, E. Di Angelantonio, J. Rudd, M. van der Schaar, "Cardiovascular Disease Risk Prediction Using Machine Learning: A Prospective Cohort Study of 423,604 Participants," *Circulation*, 2018. **(Abstract)**

A. Banerjee, J. Yoon, W. R. Zame, M. Cadeiras, **A. M. Alaa**, M. van der Schaar, "Personalized risk prediction for wait-list and post-transplant mortality in cardiac transplantation: machine learning using predictive clusters," *European Heart Journal*, 2017. **(Abstract)**

A. M. Alaa, M. van der Schaar, "A Hidden Absorbing Semi-Markov Model for Informatively Censored Temporal Data," *Journal of Machine Learning Research*, 2017.

A. M. Alaa, J. Yoon, S. Hu, and M. van der Schaar, "Personalized Risk Scoring for Critical Care Prognosis using Mixtures of Gaussian Processes," *IEEE Transactions on Biomedical Engineering*, 2017.

A. M. Alaa, M. van der Schaar, "Deep Multi-task Gaussian Processes for Survival Analysis with Competing Risks," *Neural Information Processing Systems* (NeurIPS), 2017. **(Selected for a spotlight presentation)**

A. M. Alaa, M. van der Schaar, "Bayesian Inference of Individualized Treatment Effects using Multi-task Gaussian Processes," *Neural Information Processing Systems* (NeurIPS), 2017.

A. M. Alaa, M. Weisz, M. van der Schaar, "Deep Counterfactual Networks with Propensity-Dropout," *ICML Workshop on Principled Approaches to Deep Learning*, 2017.

A. M. Alaa, M. van der Schaar, "Learning from Clinical Judgments: Semi-Markov-Modulated

Marked Hawkes Processes for Risk Prognosis," *International Conference on Machine Learning* (ICML), 2017.

A. M. Alaa, M. van der Schaar, "Individualized Risk Prognosis for Critical Care Patients: A Multi-task Gaussian Process Model," *Big Data in Medicine: Tools, Transformation and Translation, Cambridge*, 2017.

A. Banerjee, J. Yoon, W. R. Zame, M. Cadeiras, **A. M. Alaa**, M. van der Schaar, "Personalized Risk Prediction using Predictive Pursuit Machine Learning: A Pilot Study in Cardiac Transplantation," *European Society of Cardiology Congress*, 2017. **(Selected as Best Poster in Advanced Heart Failure)**

A. M. Alaa, K. Ahuja, and M. van der Schaar, "A Micro-foundation of Social Capital in Evolving Social Networks," *IEEE Transactions on Network Science and Engineering*, 2017.

A. M. Alaa, M. van der Schaar, "Balancing Suspense and Surprise: Timely Decision Making with Endogenous Information Acquisition," *Neural Information Processing Systems* (NeurIPS), 2016.

A. M. Alaa, M. van der Schaar, "A Semi-Markov Switching Linear Gaussian Model for Censored Physiological Data," *NeurIPS workshop on Machine Learning for Health*, 2016.

A. M. Alaa, K. H. Moon, W. Hsu and M. van der Schaar, "ConfidentCare: A Clinical Decision Support System for Personalized Breast Cancer Screening," *IEEE Transactions on Multimedia — Special Issue on Multimedia-based Healthcare*, 2016.

A. M. Alaa, M. van der Schaar, "ForecastICU: A Prognostic Decision Support System for Timely Prediction of Intensive Care Unit Admission," *International Conference on Machine Learning* (ICML), 2016.

A. M. Alaa, J. Yoon, M. van der Schaar, "Personalized Risk Scoring for Critical Care Patients using Mixtures of Gaussian Process Experts," *ICML Workshop on Computational Frameworks for Personalization*, 2016.

A. M. Alaa, K. Ahuja, M. van der Schaar, " Self-organizing Networks of Information Gathering Cognitive Agents," *IEEE Transactions on Cognitive Communications and Networking - Inaugural issue (invited paper)*, 2015.

CHAPTER 1

Introduction

Current advances in health information technology — including digital patient records and data management tools, wearable devices, efficient methods for genomic sequencing — are expected to drastically increase the amount of data collected for individual patients through electronic health records (EHR), biorepositories, and disease registries. The proliferation of health data is evident by the dramatic increase in the rate of adoption of EHR technologies in healthcare facilities all over the developed world; in 2015, 84% of hospitals in the US adopted an EHR system, which represents a 9-fold increase since 2008 [1].

The availability of large-scale data resources that keep track of patients' features and health outcomes paves the way for more *individualized* approaches to patient care, whereby examples and experiences encoded in data for previous patients are used to unravel disease phenotypic diversity. To achieve this, data by itself is not sufficient — we need *models* that learn from this data how prognoses would vary among future patients based on their individual traits. In this dissertation, we use machine learning (ML) to develop such models — we envision our models as being embedded within systems of decision support deployed at different stages of care to assist clinicians in predicting, preventing and treating illness.

In this Chapter, we lay out our vision for applying ML to healthcare data, and summarize the contributions presented in each of the subsequent chapters. Section 1.1 provides an overview of the type of problems and clinical setups that we address throughout the dissertation, and coins the notion of a "typical ML modeling pipeline" — the basic modeling steps and requirements that are shared among all of the problems under consideration. In Section 1.2, we flesh out the research vision laid out in Section 1.1, and specify the technical contributions of each chapter in the light of this vision.

1.1 Machine Learning for Individualized Medicine

Throughout this dissertation, we address the two main setups for ML-based modeling of clinical data: (1) the *cross-sectional* setup in which data is collected by observing many patients at a particular point of time [2], and (2) the *longitudinal* setup in which repeated observations of the same patient are collected over time [3]. In Chapters 2, 3 and 4, we develop techniques that cover various aspects of ML modeling in the cross-sectional setup, whereas in Chapter 5, we tackle the longitudinal setup by developing a comprehensive model for disease trajectories. In Chapters 6, 7 and 8, we delve deeper into the clinical application of the ML models developed in earlier chapters by applying these models to data from large-scale cohorts of breast cancer, cardiovascular disease and cystic fibrosis patients.

1.1.1 Machine Learning Modeling Pipelines

Both the cross-sectional and longitudinal setups share a set of modeling stages that we call "the ML modeling pipeline". A high-level abstraction of the typical ML modeling pipeline is illustrated in Figure 1.1. In what follows, we describe each stage of the pipeline and its significance in both the cross-sectional and longitudinal setups, then in Section 1.1.2 we provide a brief overview on how the dissertation is organized around the different stages of the pipeline.

Stage 1: Modeling Choices

The first stage of the pipeline is concerned with making modeling choices. Modeling choices comprise the broad, preset assumptions on the nature of the model used to fit the data — for instance, in the cross-sectional setup where we might seek to fit a regression model to predict patient outcomes based on their variables, potential modeling choices would include: whether the model should be a simple linear model, or a complex non-linear one, and whether interactions between patient variables should be accounted for [4]. In the longitudinal setup, we might choose to fit a "memoryless" model that does not take temporal correlations between data samples into account, or a model that does [5]. The first stage of the pipeline is crucial because it sets an upper bound on how well the ML model can accurately capture the data.

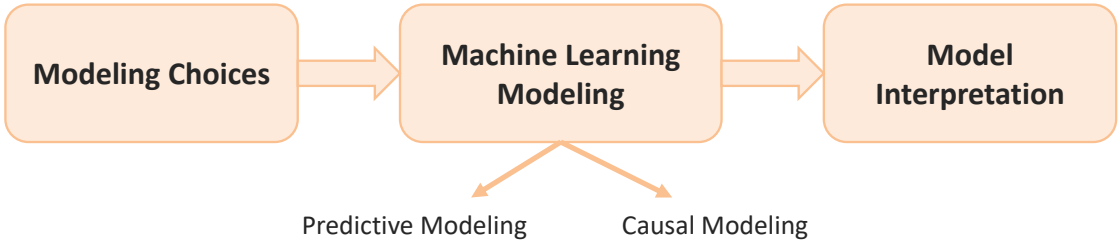


Figure 1.1: Illustration for the typical machine learning modeling pipeline.

Because traditional epidemiological research had a relatively few number of possible models at its disposal, the first stage of the modeling pipeline was typically overlooked. That is, most standard cross-sectional studies resort to either a Cox regression or a logistic regression model, and further complexity is induced in these models by manually adding non-linear effects and interaction terms that are exogenous to the model itself [6]. However, when considering an ML-based approach, the space of possible modeling choices is virtually unbounded. ML modeling choices do not only involve a choice of the basic model structure (e.g., Random forests, neural networks, etc), but also a choice of the hyper-parameters of these basic model structures (e.g., number of trees in a random forest, number of layers in a neural network, etc.). Naïve or arbitrary modeling decisions can seriously hinder the accuracy of an ML model, because different ML model structures and hyper-parameters can lead to drastically different predictive accuracy.

Central to our proposed ML models is the idea of *automation*. In Chapters 4 and 5, we develop methods for making modeling choices in a completely data-driven fashion for both the corss-sectional and longitudinal setups, without the need for manual tuning or expert intervention. Through these automated methods, the ML system is able to craft the model structure by itself so that it best fits the dataset at hand, thereby guarding against naïve modeling decisions that may bottleneck the model's predictive accuracy.

Stage 2: Machine Learning Modeling

At the core of the modeling pipeline is the actual ML model being used to make predictions. For simple supervised prediction tasks, such as predicting a patient's risk of cardiovascular disease or

diabetes based on their age and lifestyle-related variables, one can simply use standard off-the-shelf regression or classification models [7]. However, many clinical questions cannot be simply reduced to a straightforward prediction problem. As we discuss in detail in Chapter 2, in many cases we would be interested in answering questions such as "*What is the effect of a given treatment on an individual patient?*", "*What is the best treatment option for the patient at hand?*", or "*Would this patient have better outcomes had they received a different medication?*" For this type of questions, the answer requires inferring the causal effects of interventions, which in turn requires inferring the patient outcomes in counter-factual scenarios that are not observed in the data [8]. Thus, our core ML models must be able to carry out causal inference tasks and not just predictive inference ones.

The ML modeling stage should account only for a broad range of clinical questions, but also for different data formats. In the cross-sectional setup, data is simply a static array of variables characterizing the patient state at a given point of time, whereas in the longitudinal setup, data is (irregularly) collected for every patient over time, and each patient may have a different number of observations.

Stage 3: Model Interpretation

Once the appropriate modeling choices have been made (stage 1), and a model has been trained using the available data (stage 2), we have a functioning ML pipeline in the pragmatic sense — i.e., we have a model that makes the predictions that we are interested in. However, in almost all clinical setups, this is not enough. An accurate but inscrutable ML model may fail to gain patient and clinician trust. In fact, the conspicuous reluctance of many clinical researchers and epidemiologists to use ML models is often attributed to these models' "black-box" nature, which hinders their transparency and interpretability [9].

Because decision support systems based on ML models will be used to inform critical decision-making, clinicians and patients must be able to understand what these models have learned from data and how they makes their predictions. Various regulatory committees have even listed the transparency and intelligibility of prognostic models as a requirement for their deployment [10]. The final stage of the ML pipeline thus comprises an interpretation method that enables the users

of the ML model to understand its predictions. This stage is inextricable from the preceding stages — the appropriate kind of model interpretation depends on the ML model being used and format of the data used to train this model.

1.1.2 Outline of the Dissertation

The rest of this dissertation is organized around the ML pipeline in Figure 1.1. For both the cross-sectional and longitudinal setups, we develop models and algorithms that belong in different stages of the ML pipeline. The dissertation is divided into two parts: the first part (Chapters 2-5) focuses on our technical contributions with respect to developing ML tailored to healthcare applications, and the second part (Chapters 6-8) applies these methods to real-world clinical data. In what follows, we provide a sneak peek into each of the upcoming chapters, and explain how it relates to the high-level vision in Figure 1.1.

1.1.2.1 Models for Cross-sectional Data

Chapters 2, 3 and 4 deal with the 3 stages of the ML pipeline in the cross-sectional setup. In this setup, we examine the relationship between a health outcome Y (e.g., prevalence of a disease, survival outcomes, etc) and patient features X by taking a static "snapshot of a population" at a single point of time. (Note that our notion of a "cross-sectional setup" corresponds to what is known in the epidemiological literature as *observational studies*, which covers cohort, cross-sectional, and case-control studies.) Chapter 2 starts off with the core component of the pipeline (stage 2), where we develop a comprehensive framework for ML-based models for causal effect estimation. Chapter 3 proceeds by providing a novel symbolic approach for interpreting the predictions of any black-box ML model (stage 3). In Chapter 4, we conclude the ML pipeline for cross-sectional data by developing an algorithm for automating the predictive and causal modeling choices.

1.1.2.2 Models for Longitudinal Data

In Chapter 5, we study the longitudinal setup. In this setup, we are presented sequential data of the form $X_1, \ldots, X_t$ collected for patients who were followed up over an extended period of time.

When dealing with longitudinal data, our goal is typically to capture disease trajectories in order to predict patient prognoses in a dynamic fashion and understand the underlying disease mechanisms and dynamics. Unlike in the cross-sectional setup where, in Chapter 5 we do not analyze each stage of the pipeline separately, but rather develop a single comprehensive model for sequential data that executes all stages of the pipeline jointly.

1.1.2.3 Clinical Application

In Chapters 6, 7, and 8, we present a summary of the clinical studies that we conducted based on the ML models developed in earlier chapters. In these studies, we applied our models to cross-sectional data from large-scale cohorts of breast cancer, cardiovascular disease and cystic fibrosis patients, and longitudinal data from the UK cystic fibrosis registry.

1.2 Summary of Technical Contributions

In what follows, we present a brief summary of the technical contributions of each of the upcoming chapters with respect to existing literature.

Chapter 2 Contributions

In Chapter 2, we consider the problem of using ML to estimate the *causal effect* of a treatment on individual patients on the basis of retrospective, observational data (causal modeling in stage 2 of the pipeline). This problem differs fundamentally from supervised learning since we never observe the treatment effects in the observational data — we only observe the outcomes of a patient with or without the treatment, but never both. Despite a variety of recently proposed algorithmic solutions to this problem, a principled guideline for building estimators of treatment effects using machine learning algorithms is still lacking. In this chapter, we provide such guidelines by characterizing the fundamental limits of estimating heterogeneous treatment effects, establishing conditions under which these limits can be achieved, and building a practical algorithm for estimating treatment effects based on Gaussian processes.

Chapter 3 Contributions

In this Chapter, we tackle stage 3 of the pipeline: understanding the predictions of a general ML model. To this end, we introduce the *symbolic metamodeling* framework — a general methodology for interpreting predictions by converting "black-box" models into "white-box" functions that are understandable to human subjects. A symbolic metamodel is a model of a model, i.e., a surrogate model of a trained (machine learning) model expressed through a succinct symbolic expression that comprises familiar mathematical functions and can be subjected to symbolic manipulation. We parameterize metamodels using Meijer G-functions — a class of complex-valued contour integrals that depend on real-valued parameters, and whose solutions reduce to familiar algebraic, analytic and closed-form functions for different parameter settings. This parameterization enables efficient optimization of metamodels via gradient descent, and allows discovering the functional forms learned by a model with minimal a priori assumptions. We show that symbolic metamodeling provides a generalized framework for model interpretation many common forms of model explanation can be analytically derived from a symbolic metamodel.

Chapter 4 Contributions

This Chapter addresses stage 1 of the pipeline. We developed an algorithm for automating the design of predictive and causal modeling tailored for clinical prognosis. Our algorithm optimizes ensembles of model configurations efficiently using a novel batched Bayesian optimization (BO) algorithm that learns a low-dimensional decomposition of the models' high-dimensional hyperparameter space in concurrence with the BO procedure. This is achieved by modeling the models' performances as a black-box function with a Gaussian process prior, and modeling the "similarities" between the pipelines' baseline algorithms via a sparse additive kernel with a Dirichlet prior. For causal models, we propose an influence function-based approach to estimate their accuracy.

Chapter 5 Contributions

Chapter 5 focuses on the longitudinal setup, where we develop a sequential model for predicting patient outcomes and understanding disease dynamics. Existing models provide the patient

with pragmatic (supervised) predictions of risk, but do not provide the clinician with intelligible (unsupervised) representations of disease pathology. In this Chapter, we develop the *attentive state-space model*, a deep probabilistic model that learns accurate and interpretable structured representations for disease trajectories. Unlike Markovian state-space models, in which state dynamics are memoryless, our model uses an attention mechanism to create "memoryful" dynamics, whereby attention weights determine the dependence of future disease states on past medical history. To learn the model parameters from medical records, we develop an inference algorithm that jointly learns a compiled inference network and the model parameters, leveraging the attentive representation to construct a variational approximation of the posterior state distribution.

Part I

Machine Learning for Individualized Medicine

CHAPTER 2

Estimating Treatment Effects from Observational Data

We start off with the core component of the ML pipeline: ML modeling. There is already a wide range of well-established, off-the-shelf predictive models, hence we focus on *causal modeling*. The problem of estimating heterogeneous (individualized) causal effects of a treatment from observational data is central in public health and drug development [11]. The increasing availability of observational data in these domains has encouraged the development of various machine learning algorithms tailored for inferring treatment effects using observational data (e.g. [12–15]). Due to the peculiarity of the treatment effect estimation problem, these algorithms address various modeling aspects that are foreign to standard supervised learning setups; such aspects include ways to handle *sample selection bias* [16], and ways to model *treated* and *untreated* data points. Despite a variety of recent algorithmic approaches, principled guidelines for model design are lacking.

In this Chapter, we identify guiding principles for designing practical treatment effect estimation algorithms in the context of Bayesian nonparametric inference, and propose one an algorithm that follows these guidelines. We set these guidelines by characterizing the fundamental limits of estimating treatment effects, and studying the impact of various common modeling choices on the achievability of those limits. In what follows, we provide a brief technical background for the treatment effect estimation problem, along with a summary of our contributions.

2.1 Background and Summary of Contributions

Our analysis hinges on the Rubin-Neyman potential outcomes model [17]. That is, we consider an observational dataset with a population of subjects, where each subject i is endowed with a d-dimensional feature $X_i \in \mathcal{X}$. We assume that $\mathcal{X} = [0,1]^d$, but most of our results hold for

general compact metric spaces (bounded, closed sets in $\mathbb{R}^d$). A treatment assignment indicator $W_i \in \{0,1\}$ is associated with subject i; $W_i = 1$ if the treatment under study was applied to subject i, and $W_i = 0$ otherwise. Subject i's responses with and without the treatment (the potential outcomes) are denoted as $Y_i^{(1)}$ and $Y_i^{(0)}$, respectively. Treatments are assigned to subjects according to an underlying policy that depends on the subjects' features, i.e. $W_i \not\perp\!\!\!\perp X_i$. This dependence is quantified via the conditional distribution $p(x) = \mathbb{P}(W_i = 1 | X_i = x)$, also known as the *propensity score* of subject i [18]. The response $Y_i^{(W_i)}$ is the "factual outcome" which we observe in the data, whereas $Y_i^{(1-W_i)}$ is the unrealized "counterfactual outcome" [19]. An observational dataset $\mathcal{D}_n$ comprises n samples of the form:

$$\mathcal{D}_n = \{X_i, W_i, Y_i^{(W_i)}\}_{i=1}^n \tag{2.1}$$

The causal effect of the treatment on subject i with a feature $X_i = x$ is characterized through the *conditional average treatment effect* (CATE) function $T(x)$, which is defined as the expected difference between the two potential outcomes [17], i.e.

$$T(x) = \mathbb{E}\big[\, Y_i^{(1)} - Y_i^{(0)} \,|\, X_i = x \,\big] \tag{2.2}$$

Our goal is to identify a set of guiding principles for building estimators of the CATE $T(x)$ using samples from $\mathcal{D}_n$. Throughout this Chapter, we will assume that the density $d\mathbb{P}(X_i, W_i, Y_i^{(0)}, Y_i^{(1)})$ supports the assumptions of *unconfoundedness* and *overlap*, which are necessary for causal identifiability and consistency. Unconfoundedness requires that $(Y_i^{(0)}, Y_i^{(1)}) \perp\!\!\!\perp W_i \,|\, X_i$, whereas overlap requires that $0 < p(x) < 1$ [18]. **Selection bias** occurs in $\mathcal{D}_n$ since the distribution of the treated/control subjects does not match that of the overall population.

In order to come up with principled guidelines for building estimators of $T(x)$, we characterize the fundamental (information-theoretic) limits of estimating the CATE using samples from $\mathcal{D}_n$, and identify the modeling choices that would allow achieving those limits. To this end, in **Section 2.3** we tackle the following question: **what are the fundamental limits of CATE estimation?** We answer this question by deriving the *optimal minimax rate* for estimating $T(x)$ using $\mathcal{D}_n$. Interestingly, it turns out that the optimal rate **does not** depend on **selection bias**, but rather on the **smoothness** and **sparsity** of the more "complex" of the functions $\mathbb{E}[\, Y_i^{(0)} \,|\, X_i = x \,]$ and

$\mathbb{E}[\, Y_i^{(1)} \,|\, X_i = x \,]$. We focus our analysis on Bayesian nonparametric methods, since they have the appealing properties of being robust to misspecification and are accessible for theoretical analysis.

Our analysis reveals that the relative importance of the different modeling aspects vary with the sample size. In particular, in the **large-sample regime**, selection bias does not pose a serious problem, and the model's performance would be mainly determined by its **structure**, i.e. the way the outcomes $Y_i^{(0)}$ and $Y_i^{(1)}$ are modeled, and the impact of that on variable selection and hyperparameter tuning. On the contrary, selection bias can seriously harm a model's generalization performance in **small-sample regimes**. A good model should then be carefully designed so that it operates well in both regimes by possessing the right **model structure** that would allow learning at a fast rate, and the right **model selection** (hyperparameter optimization) scheme that would account for selection bias.

In Section 2.5, we build a practical CATE estimation algorithm guided by the results of the analyses in Section 2.3. We model the outcomes $Y_i^{(0)}$ and $Y_i^{(1)}$ using a Gaussian process with a *non-stationary* kernel that captures the different relevant variables and different levels of smoothness of the functions $\mathbb{E}[\, Y_i^{(0)} \,|\, X_i = x \,]$ and $\mathbb{E}[\, Y_i^{(1)} \,|\, X_i = x \,]$. We prove that this model structure can achieve the optimal rate of CATE estimation when tuned with the right hyperparameters. We also propose a *doubly-robust* hyperparameter optimization scheme that accounts for selection bias in small-sample regimes, without hindering the model's minimax-optimality in the large sample limit. We show that our algorithm outperforms state-of-the-art methods using a well-known semi-synthetic simulation setup.

2.2 Related Work

Very few works have attempted to characterize the limits of CATE estimation, or study the impact of different modeling choices on the CATE estimation performance in a principled manner. [20] characterized the asymptotic "information rates" for different CATE estimators, but provided no clear guidelines on practical model design or an analysis of the impact of sample selection bias. The study in [21] was rather empirical in nature, comparing the performance of different regression structures for the potential outcomes while ignoring selection bias. A similar study, but focusing

only on random forest models, was conducted in [22].

Most of the previous works have been algorithmic in nature, focusing mainly on devising algorithms that correct for selection bias (e.g. [12, 13, 23, 24]). Some of these works cast the selection bias problem as a problem of *covariate shift* [25], and use techniques from *representation learning* to learn feature maps that balance the biased data (e.g. [12, 14, 23]). However, those works report much bigger improvements in CATE estimation when changing their model structure (e.g. architecture of a neural network), as compared to the gains attained by only accounting for bias (see the comparisons between the TARnet and BNN models in [14]). Similar observations are reported in [15, 26], where the selection of the model structure seemed to influence the achieved CATE estimation performance even when selection bias is not accounted for. However, none of these works offer a discussion on whether selection bias is actually the main challenge in CATE estimation, or whether the outcomes' model structure may have a bigger influence on performance.

In contrast to the works above, in this Chapter we do not attempt to develop a model by presupposing that particular modeling aspects are of greater importance than others, but rather provides a framework for understanding the limits on the achievable performance, and how different modeling aspects influence a model's chance of achieving those limits. We use our analyses to both reflect on the modeling choices made in the works above, and also devise a novel, principled CATE estimation algorithms that achieves the fundamental performance limits.

2.3 Estimating CATE: Problem Setup

2.3.1 Potential Outcomes and Propensity Score

We consider the following *random design* regression model for the potential outcomes:

$$Y_i^{(w)} = f_w(X_i) + \varepsilon_{i,w},\ w \in \{0,1\}, \tag{2.3}$$

where $\varepsilon_{i,w} \sim \mathcal{N}(0, \sigma_w^2)$ is a Gaussian noise variable. It follows from (2.2) that the CATE is $T(x) = f_1(x) - f_0(x)$. The *response surfaces* $f_1(x)$ and $f_0(x)$ correspond to the subjects' responses with and without the treatment. We assume that $f_w(.) : \mathcal{X} \to \mathbb{R}$, $w \in \{0,1\}$, is a totally bounded function that lives in a space of "smooth" or "regular" functions, with an unknown smoothness

parameter α_w. We use Hölder balls for concreteness, although our results extend to other function spaces. A function $f_w(.)$ lies in the Hölder ball H^{α_w}, with a Hölder exponent $\alpha_w > 0$, if and only if it is bounded in sup-norm by a constant $C > 0$, all its partial derivatives up to order $\lfloor \alpha_w \rfloor$ exist, and all its partial derivatives of order $\lfloor \alpha_w \rfloor$ are Lipschitz with exponent $(\alpha_w - \lfloor \alpha_w \rfloor)$ and constant C. The Hölder exponents quantify the complexities of f_0 and f_1, and hence the hardness of estimating $T(x)$ would depend on α_0 and α_1.

2.3.2 Bayesian Nonparametric Inference

Nonparametric inference is immune to misspecification of the outcomes' and propensity models [27], and hence we focus on Bayesian nonparametric methods for inferring $T(.)$ on the basis of $\mathcal{D}_n$. Bayesian inference entails specifying a prior distribution $\mathit{\Pi}$ over $f_1(.)$ and $f_0(.)$, i.e.

$$f_0, f_1 \sim \mathit{\Pi}(\bar{\varphi}_{\beta_0}, \bar{\varphi}_{\beta_1}), \tag{2.4}$$

where $\bar{\varphi}_{\beta_w} = \{\varphi^k_{\beta_w}\}_{k=1}^{\infty}, w \in \{0, 1\}$, are complete orthonormal bases (indexed by a parameter $\beta_w > 0$) with respect to Lebesgue measure in $\mathcal{X}$, $f_w = \sum_k \bar{f}^k_w \cdot \varphi^k_{\beta_w}$, and $\bar{f}^k_w = \langle f_w, \varphi^k_{\beta_w} \rangle$. Thus, for given bases $\bar{\varphi}_{\beta_0}$ and $\bar{\varphi}_{\beta_1}$, $\mathit{\Pi}$ places a probability distribution on the projections $\{\bar{f}^k_w\}_k$. Potential choices for the basis $\bar{\varphi}_{\beta_w}$ that would give rise to implementable Bayesian inference algorithms include regular wavelet basis [28], radial basis for a reproducing kernel Hilbert space (RKHS) [29], etc. In general, β_w would determine the smoothness of the function space spanned by $\bar{\varphi}_{\beta_w}$.

2.3.3 Towards Principled CATE Estimation

To evaluate the predictive accuracy of the Bayesian inference procedure, we analyze the "frequentist" loss of point estimators $\hat{T}(x)$ induced by the Bayesian posterior $d\mathit{\Pi}_n(T(x) \,|\, \mathcal{D}_n)$, assuming that $\mathcal{D}_n$ is generated based on fixed, *true* response surfaces $f_1(x)$ and $f_0(x)$. (This type of analysis is sometimes referred to as the "Frequentist-Bayes" analysis [30].) In particular, we quantify the performance of a point estimator $\hat{T}(x) = \delta(d\mathit{\Pi}_n(T(x) \,|\, \mathcal{D}_n))$ by its squared-$L^2(\mathbb{P})$ error, which was dubbed the *precision of estimating heterogeneous effects* **(PEHE)** in [31], and is formally

defined as:

$$\boldsymbol{\psi}(\hat{T}) \triangleq \mathbb{E} \parallel \hat{T} - T \parallel^2_{L^2(\mathbb{P})}, \tag{2.5}$$

where $L^2(\mathbb{P})$ is the L^2 norm with respect to $\mathbb{P}(X)$, i.e. $\|f(x)\|^2_{L^2(\mathbb{P})} = \int f^2(x)\, d\mathbb{P}(X = x)$.

The "fundamental problem of causal inference" is that for every subject i in $\mathcal{D}_n$, we only observe the **factual** outcome $Y_i^{(W_i)}$, whereas the **counterfactual** $Y_i^{(1-W_i)}$ remains unknown, which renders empirical evaluation of the PEHE in (2.5) impossible. Moreover, $\mathcal{D}_n$ would generally exhibit sample **selection bias** [16], because the treatment assignment mechanism (decided by $p(x)$) creates a discrepancy between the feature distributions of the treated/control population and the overall population. Thus, standard **supervised learning** approaches based on empirical risk minimization cannot be used to learn a generalizable model for the CATE from samples in $\mathcal{D}_n$. This gives rise to the following fundamental modeling questions that are peculiar to CATE estimation:

- **[Q1]:** How should the treatment assignment W_i be incorporated into the learning model?
- **[Q2]:** How should selection bias be handled?

Adequate answers to **[Q1]** and **[Q2]** would provide guidelines for selecting the prior $\mathit{\Pi}(\bar{\varphi}_{\beta_0}, \bar{\varphi}_{\beta_1})$. Addressing the modeling questions above requires a profound understanding of the **fundamental limits** of CATE estimation, in addition to an understanding of the impact of different modeling choices on the **achievability** of such limits. The next Sections provide principled answers to **[Q1]** and **[Q2]** by addressing the following, more fundamental questions:

- **Section 2.4: What are the *limits* on the performance achieved by *any* CATE estimator?**
- **Section 2.5: How can we build *practical algorithms* that can achieve these limits?**

2.4 Fundamental Limits of CATE Estimation

In this Section, we establish an information-theoretic limit on the performance of *any* CATE estimator. In what follows, we use the standard Bachmann-Landau order notation, and write $a \vee b = \max\{a, b\}$, $a \wedge b = \min\{a, b\}$. The notation $a \lesssim b$ means that $a \leq Cb$ for a universal constant C, and $\asymp$ denotes asymptotic equivalence.

2.4.1 Optimal Minimax Rates

The "hardness" of a nonparametric estimation problem is typically characterized by its *minimax* risk [32], i.e. the minimum worst case risk achieved by *any* estimator when the estimand is known to live in a given function space [33]. In the following Theorem, we establish the optimal minimax rate for the PEHE risk in terms of the complexity of the response surfaces f_0 and f_1.

Theorem 1. *Suppose that* $\mathcal{X} = [0,1]^d$*, and that* f_w *depends on a subset of* d_w *features with* $d_w \leq \min\{n, d\}$ *for* $w \in \{0,1\}$*. If* $f_0 \in H^{\alpha_0}$ *and* $f_1 \in H^{\alpha_1}$*, then the optimal minimax rate is:*

$$\inf_{\hat{T}} \sup_{f_0, f_1} \boldsymbol{\psi}(\hat{T}) \asymp \underbrace{n^{-\left(1+\frac{1}{2}\left(\frac{d_0}{\alpha_0} \vee \frac{d_1}{\alpha_1}\right)\right)^{-1}}}_{\textbf{CATE estimation}} \vee \underbrace{\log\left(\frac{d^{d_0+d_1}}{d_0^{d_0}\, d_1^{d_1}}\right)^{\frac{1}{n}}}_{\textbf{Variable selection}}.$$

The above holds for any $p(.) \in H^{\alpha_p}$*,* $\alpha_p > 0$*.* $\square$

In Theorem 1, the supremum is taken over α_w-Hölder balls ($w \in \{0,1\}$), whereas the infimum is taken over all possible Bayesian estimators. The minimax rate in Theorem 1 corresponds to the **fastest rate** by which **any** (Bayesian) estimator $\hat{T}(.)$ can approximate the CATE function $T(.)$. The proof of Theorem 1 uses information-theoretic techniques based on Fano's method to derive algorithm-independent estimation rates [34]. In the following set of remarks, we revisit **[Q1]** and **[Q2]** in the light of the results of Theorem 1.

How can Theorem 1 help us address [Q1] and [Q2]?

▷ **Remark 1 (Smoothness and sparsity)**

Theorem 1 says that estimating CATE is as hard as nonparametric regression for functions with additive sparsity [33, 35]. The minimax rate in Theorem 1 decomposes into a term reflecting the complexity of CATE estimation under correct variable selection for f_0 and f_1, and a term reflecting the complexity of variable selection. Variable selection complexity remains small as long as $\log(d) = \Theta(n^{\zeta})$, for some $\zeta \in (0,1)$, and approaches the parametric rates as $\zeta \to 0$. The minimax rate will generally be dominated by the complexity of CATE estimation, and will

approach the parametric rates only for very smooth response surfaces with small number of relevant dimensions, i.e. $\frac{d_0}{\alpha_0} \vee \frac{d_1}{\alpha_1} \to 0$.

The main takeaway from Theorem 1 is that the CATE learning rate is determined by the more "complex" of the surfaces f_0 and f_1, where complexity is quantified by the sparsity-to-smoothness ratio d_w/α_w for $w \in \{0, 1\}$. Thus, a model would achieve the optimal CATE learning rate only if it selects the correct relevant variables for f_0 and f_1, and tunes its "hyperparameters" (i.e. smoothness of the prior) to cope with a complexity of $\frac{d_0}{\alpha_0} \vee \frac{d_1}{\alpha_1}$. When $\frac{d_0}{\alpha_0}$ and $\frac{d_1}{\alpha_1}$ are very different (e.g. f_0 and f_1 have different relevant features), rate-optimal estimation is possible only if the model incorporates such differences in $\mathit{\Pi}(\bar{\varphi}_{\beta_0}, \bar{\varphi}_{\beta_1})$.

The discussion above **provides a concrete answer to [Q1]:** the treatment assignment variable w should be incorporated into the model in such a way that it **encodes the different relevant dimensions and smoothness levels of f_0 and f_1 in the bases $\bar{\varphi}_{\beta_0}$ and $\bar{\varphi}_{\beta_1}$**. (The simplest way to achieve this is to use two separate models for f_0 and f_1.) This is not fulfilled by many of the previous models that built a single regression function of the from $f : \mathcal{X} \times \{0, 1\} \to \mathbb{R}$, and estimated the CATE as $\hat{T}(x) = f(x, 1) - f(x, 0)$ [23, 31, 36]. This is because such models enforced the smoothness of the prior along all features to be the same for $w = 0$ and $w = 1$.

▷ **Remark 2 (Selection bias)**

Theorem 1 gives a rather surprising answer to **[Q2]**: the **optimal learning rate** is **oblivious to selection bias**. Such a finding is consistent with previous results on nonparametric kernel density estimation under selection bias [37], and parametric Bayesian inference under *covariate shift* [38, 39]. It shows that many of the recent works have missed the target; the works in [14, 15, 23] cast the problem of CATE estimation as one of **covariate shift** that results from selection bias. However, Theorem 1 says that selection bias is not a problem when we have a sufficiently large amount of data. This is because selection bias is inherently a misspecification problem, and hence its impact on nonparametric inference is washed away in large-sample regimes.

Remarks 1 and 2 posit an explanation for various recurrent (empirical) findings reported in previous literature. For instance, [40] found that separate modeling of f_0 and f_1 via Bayesian addi-

tive regression trees (BART) outperforms the well-known single-surface BART model developed in [31]. Similar findings were reported for models based on Gaussian processes [15], and models based on deep neural networks [14]. All such findings can be explained in the light of Remark 1. On the other hand, Remark 2 may provide an explanation as to why the "TARnet" model in [14], which models f_0 and f_1 using separate neural networks and does not account for selection bias, outperformed the "BNN" model in [23], which regularizes for selection bias but fits a single-output network for f_0 and f_1.

2.4.2 Backing off from "Asymptopia"

Theorem 1 shows that selection bias does not hinder the optimal minimax rates, and that it is only the structural properties of the prior $\Pi(\bar{\varphi}_{\beta_0}, \bar{\varphi}_{\beta_1})$ that determine a model's rate of learning. But does the achieved learning rate suffice as a sole criterion for addressing the modeling questions **[Q1]** and **[Q2]**? The answer is "yes" only if $\mathcal{D}_n$ comes from a large observational dataset, in which case the learning rate suffices as a descriptor for the large-sample performance. However, if $\mathcal{D}_n$ is small, which is typical in post-hoc analyses of clinical trials [11], then one should make the design choices that would optimize the small-sample performance. In order to give a complete picture of the performance in large and small-sample regimes, we derive the following bound on the PEHE:

$$\boldsymbol{\psi}(\hat{T}) \leq \bar{C} \cdot \exp(D_2(Q_0 \,\|\, Q)) \cdot \|f_0 - \hat{f}_0\|^2_{L^2(\mathbb{P}_0)} + \bar{C} \cdot \exp(\underbrace{D_2(Q_1 \,\|\, Q)}_{\substack{\textbf{Reyni} \\ \textbf{Divergence}}}) \cdot \underbrace{\|f_1 - \hat{f}_1\|^2_{L^2(\mathbb{P}_1)}}_{\substack{\textbf{Supervised} \\ \textbf{learning loss}}}, \quad (2.6)$$

for some $\bar{C} > 0$, where $L^2(\mathbb{P}_w)$, for $w \in \{0, 1\}$, is the L^2 norm with respect to $d\mathbb{P}(X = x \,|\, W = w)$, $Q = d\mathbb{P}(X = x)$, $Q_w = d\mathbb{P}(X = x \,|\, W = w)$, and $D_m(p \,\|\, q)$ is the m^{th} order Réyni divergence. The bound in (2.6) holds for all $n > 0$, and is tight (refer to the Appendix); it shows that the PEHE is a weighted linear combination of the mean squared losses for the two underlying supervised problems of learning f_0 and f_1 with **no covariate shift**, where the weights are determined by the extent of the mismatch between the distributions of the treated and control populations, quantified by the Réyni divergence measure. If $\mathcal{D}_n$ is a dataset obtained from a randomized controlled trial ($Q = Q_0 = Q_1$), then we have $D_2(Q_0 \,\|\, Q) = D_2(Q_1 \,\|\, Q) = 0$, and the bound boils down to a sum of two supervised learning losses, i.e. $\boldsymbol{\psi}(\hat{T}) \leq \bar{C} \cdot \|f_0 - \hat{f}_0\|^2_{L^2(\mathbb{P})} + \bar{C} \cdot \|f_1 - \hat{f}_1\|^2_{L^2(\mathbb{P})}$.

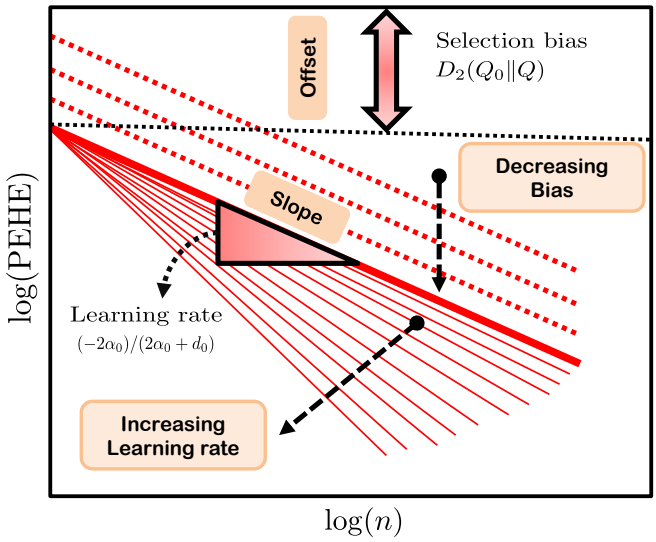


Figure 2.1: The PEHE in (2.7) plotted on a log-log scale.

Since the minimax rate for standard nonparametric regression is $\|f_w - \hat{f}_w\|_2^2 \asymp C_w \cdot n^{\frac{-2\alpha_w}{2\alpha_w+d_w}}$ [32], when $d_0/\alpha_0 >> d_1/\alpha_1$, the first-order Taylor approximation for the logarithm of the PEHE in (2.6) is given by:

$$\log(\boldsymbol{\psi}(\hat{T})) \approx \underbrace{D_2(Q_0\|Q)}_{\substack{\textbf{Selection}\\ \textbf{bias}}} + \underbrace{\log(C_0)}_{\substack{\textbf{Bias}\\ \textbf{correction}}} - \underbrace{\frac{2\alpha_0}{2\alpha_0+d_0}}_{\textbf{Learning rate}} \log(n) + O\left(n^{\frac{-2\alpha_1}{2\alpha_1+d_1}+\frac{2\alpha_0}{2\alpha_0+d_0}}\right). \tag{2.7}$$

That is, when viewed on a log-log scale, the behavior of the PEHE versus the number of samples can be described as follows. log(PEHE) is a linear function of $\log(n)$. Selection bias adds a constant offset to log(PEHE), but does not affect its slope, which harms the performance only in the small-sample regime. In the large-sample regime, the slope of log(PEHE), which depends solely on the smoothness and sparsity of the response surfaces, dominates the performance, and selection bias becomes less of a problem. Figure 2.1 depicts the PEHE in (2.7) on a log-log scale.

2.5 CATE Estimation using Non-Stationary Gaussian Process Regression

In this Section, we build on the analyses conducted in Section 2.4 to design a practical algorithm for CATE estimation.

2.5.1 Non-Stationary Gaussian Process Priors

We specify the prior $\Pi(\bar{\varphi}_{\beta_0}, \bar{\varphi}_{\beta_1})$ as a Gaussian process (GP) over $g : \mathcal{X} \times \{0,1\} \to \mathbb{R}$, with a kernel $\boldsymbol{K}_\beta$, and a hyperparameter set β as follows:

$$g \sim \mathcal{GP}\left(0, \boldsymbol{K}_\beta(z, z')\right), \tag{2.8}$$

where $z = (x, w) \in \mathcal{X} \times \{0,1\}$, and $f_w(x) = g(x, w)$. The kernel $\boldsymbol{K}_\beta$ specifies the bases $\bar{\varphi}_{\beta_0}$ and $\bar{\varphi}_{\beta_1}$ through its induced canonical feature map $\boldsymbol{K}_\beta(., z)$ [41, 42]. As pointed out in **Remark 1**, the treatment assignment variable w should encode the different relevant dimensions and smoothness levels of f_0 and f_1. Thus, we model $\boldsymbol{K}_\beta$ as a *non-stationary* kernel that depends on w as follows:

$$\begin{aligned}
\boldsymbol{K}_\beta(z, z') &= \boldsymbol{\Gamma}(w, w') \cdot \boldsymbol{k}_\beta^T(x, x'), \\
\boldsymbol{k}_\beta(x, x') &= \left[k_{\beta_0}(x, x'), k_{\beta_1}(x, x'), k_{\beta_0}(x, x') + k_{\beta_1}(x, x')\right], \\
\boldsymbol{\Gamma}(w, w') &= \left[\Gamma_0(w, w'), \Gamma_1(w, w'), 1 - \Gamma_0(w, w') - \Gamma_1(w, w')\right],
\end{aligned}$$

where $\Gamma_0(w, w') = (1 - w)(1 - w')$, $\Gamma_1(w, w') = w \cdot w'$, and $k_{\beta_w}(x, x')$ is a Matérn kernel with a length-scale parameter β_w, for $w \in \{0,1\}$. The kernel defined above ensures that any covariance matrix induced by points in $\mathcal{X} \times \{0,1\}$ is positive definite. Variable selection is implemented by using the *automatic relevance determination* version of the Matérn kernel [41]. The non-stationarity of $\boldsymbol{K}_\beta$ allows setting **different** length-scales and relevant variables for the marginal priors on f_0 and f_1 while sharing data between the two surfaces, i.e.

$$\begin{aligned}
\boldsymbol{K}_\beta((x, w), (x', w)) &= k_{\beta_w}(x, x'), \;\; w \in \{0,1\}, \\
\boldsymbol{K}_\beta((x, w), (x', w')) &= k_{\beta_0}(x, x') + k_{\beta_1}(x, x'), \; w \neq w'.
\end{aligned} \tag{2.9}$$

That is, all draws from the prior give Matérn sample paths with different smoothness levels (β_0 and β_1) for f_0 and f_1, respectively, and the correlations between the paths are captured via the kernel mixture $k_{\beta_0}(x, x') + k_{\beta_1}(x, x')$. Note that draws from a Matérn prior with length-scale β are almost surely $\bar{\beta}$-Hölder for all $\bar{\beta} \leq \beta$ [43]. Thus, $\mathcal{GP}(0, \boldsymbol{K}_\beta)$ specifies a β_w-Hölder ball as an a priori regularity class for response surface f_w, $w \in \{0,1\}$.

In the following Theorem, we show that point estimators induced by the prior $\mathcal{GP}(0, \boldsymbol{K}_\beta)$ can achieve the optimal minimax rate in Theorem 1.

Theorem 2. *Suppose that the d_w relevant features for f_w are known a priori for $w \in \{0,1\}$. If $f_0 \in H^{\alpha_0}$, $f_1 \in H^{\alpha_1}$, $\Pi = \mathcal{GP}(0, \boldsymbol{K}_\beta)$, and $\hat{T} = \mathbb{E}_\Pi[\, T \,|\, \mathcal{D}_n \,]$, then we have that*

$$\boldsymbol{\psi}(\hat{T}) \lesssim n^{-\frac{2(\alpha_0 \wedge \beta_0)}{2\beta_0 + d_0}} \vee n^{-\frac{2(\alpha_1 \wedge \beta_1)}{2\beta_1 + d_1}}$$

whenever $\min\{\alpha_0, \alpha_1, \beta_0, \beta_1\} \geq d/2$. $\square$

Note that posterior consistency holds for all combinations of $(\alpha_0, \alpha_1, \beta_0, \beta_1)$ since the support of the Matérn prior is the space of bounded continuous functions[1]. The bound in Theorem 2 can be shown to be tight using the results in [45]. Theorem 2 says that the posterior induced by the prior $\mathcal{GP}(0, \boldsymbol{K}_\beta)$ contracts around the true CATE function at the optimal rate given in Theorem 1 provided that the following **matching condition** is met:

$$\begin{aligned} \beta_v \;\; &= \alpha_v \\ \alpha_v \frac{d_{1-v}}{d_v} \;\; &\leq \beta_{1-v} \leq \alpha_{1-v} + \frac{\alpha_{1-v} \cdot d_v}{2\alpha_v} - \frac{d_{1-v}}{2}, \end{aligned} \tag{2.10}$$

where $v = 1$ if $d_1/\alpha_1 > d_0/\alpha_0$, and $v = 0$ otherwise. The condition in (2.10) implies that achieving the optimal rate (steepest slope in Figure 2.1) via the non-stationary GP prior in Section 2.5.1 is only a matter of hyperparameter tuning: the smoothness of the prior needs to match the smoothness of the "more complex" of the two response surfaces. Note that Theorem 2 implies that we do not need to handle selection bias in order to achieve the optimal rate, which is consistent with the earlier discussion in **Remark 2**.

2.5.2 Doubly-Robust Hyperparameters

Theorem 2 says that the optimal minimax rate for CATE estimation can be achieved by satisfying the smoothness matching condition in (2.10). However, in practice, the smoothness levels of the true response functions are unknown and need to be learned from the data. Moreover, since selection bias is impactful in small-sample regimes, ignoring it may lead to a poor generalization performance when the size of $\mathcal{D}_n$ is small. In this Section, we propose a hyperparameter

[1] This is because the RKHS associated with the prior lies dense in the space of bounded continuous functions [29, 44].

optimization algorithm that accounts for selection bias while ensuring minimax-optimality in the large-sample limit.

Previous works tend to adjust for selection bias "mechanically" using variants of importance sampling approaches based on inverse-propensity-weighting (IPW) [25, 38], and kernel mean matching [46], or by learning a "balanced representation" of treated and control populations [12]. We do not attempt to explicitly adjust for selection bias using ad-hoc approaches, and rather seek the "informationally optimal" estimator of the PEHE. That is, we seek the **most efficient** (unbiased) estimator $\hat{\psi}^*(\hat{T})$ of $\psi(\hat{T})$, which satisfies an analog of the Cramér-Rao bound (information-inequality) in parametric estimation, i.e. $\text{Var}[\hat{\psi}^*(\hat{T})] \leq \text{Var}[\hat{\psi}(\hat{T})]$, for any estimator $\hat{\psi}(\hat{T})$.

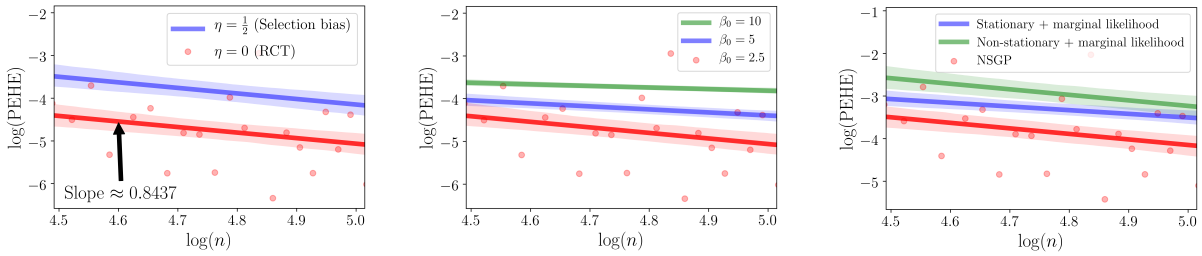
Classical Cramér-Rao bounds do not apply to estimators of the form $\hat{\psi}^*(\hat{T})$, since such estimators are functionals of nonparametric objects. There are, however, analogous information inequalities for nonparametric estimation, including Bhattacharyya's variance bound [47], and its generalization due to Bickel [48]. We proceed by realizing that the PEHE $\psi(\hat{T})$ is simply a functional that belongs to the *doubly-robust* class of functionals analyzed by Robins in [49]. Thus, one can construct the "most" efficient estimator of $\psi(\hat{T})$ using the most *efficient influence function* of $\psi(\hat{T})$ as follows [49, 50]:

$$\hat{\psi}^*(\hat{T}) = \sum_{i=1}^{n} \left(\frac{Y_i^{(W_i)} - (W_i - p(X_i)) \cdot \hat{T}(X_i)}{p(X_i) \cdot (1 - p(X_i))} \right)^2 .$$

The derivation of the estimator above can be found in Theorem 9 in [50] and Section 5 in [49]. When the propensity function $p(.)$ is known, this estimator approximate the PEHE at its optimal minimax rate. We estimate $p(.)$ via standard kernel density estimation methods. It can be easily shown using the results in [51] that when using the estimator above to tune the GP hyperparameters via cross-validation, then the learned length-scale parameters will satisfy the matching condition for minimax optimality.

2.6 Experiments

In this Section, we check the validity of our analyses using a synthetic simulation setup (Subsection 2.6.1), and then evaluate the performance of our proposed model using data from a real-world



(a) Impact of selection bias. (b) Impact of over-smoothed priors. (c) Performance of different models.

Figure 2.2: Scatter-plots and linear fits for the PEHE of NSGP on a log-log scale in different simulation setups (RCT: randomized controlled trial).

clinical trial with simulated potential outcomes (Subsection 2.6.2). We will use the acronym **NSGP** to refer to the non-stationary GP model proposed in Section 2.5.

2.6.1 Learning Brownian Response Surfaces

2.6.1.1 Synthetic Model

Let $\mathcal{X} = [0, 1]$, and define a κ-fold integrated Brownian motion B_κ, $\kappa \in \mathbb{N}_+$, on $\mathcal{X}$ as follows:

$$B_\kappa(x) = \int_0^x \int_0^{x_\kappa} \cdots \int_0^{x_2} B_0(x_1)\, dx_1\, dx_2 \cdots dx_{x_\kappa},$$

where $B_0(.)$ is a standard Brownian motion (Wiener process). Sample paths of B_0 are almost surely Hölder regular with exponent $\frac{1}{2}$ [52]. Since $B_0(x)$ is almost surely non-differentiable everywhere in $\mathcal{X}$, then sample paths of $B_\kappa(x)$ are Hölder with exponent $\kappa + \frac{1}{2}$, i.e. $B_\kappa \in H^{\kappa+\frac{1}{2}}$ with probability 1. Therefore, when the true response surfaces are κ-fold integrated Brownian paths, the optimality and achievability results in Theorems 1 and 2 should hold. To this end, we simulate the true response surfaces $f_0 \in H^{\alpha_0}$ and $f_1 \in H^{\alpha_1}$ as $\boldsymbol{f_0} \sim \boldsymbol{B_{\alpha_0 - \frac{1}{2}}}$, and $\boldsymbol{f_1} \sim \boldsymbol{B_{\alpha_1 - \frac{1}{2}}}$, where we set $\boldsymbol{\alpha_0 = 2.5}$ and $\boldsymbol{\alpha_1 = 5.5}$. The propensity score is modeled as a parametrized logistic function $\boldsymbol{p(x \,|\, \eta) = (1 + e^{-\eta\,(x - \frac{1}{2})})^{-1}}$, where $\eta \in \mathbb{R}$ is a parameter that determines the severity of selection bias. For a pair of fixed Brownian paths f_0 and f_1, synthetic observational samples $(\boldsymbol{X_i, W_i, Y_i^{(W_i)}})_i$ are generated as follows: $\boldsymbol{X_i \sim \text{Uniform}[0, 1]}$, $\boldsymbol{W_i \sim \text{Bernoulli}(p(x \,|\, \eta))}$, and $\boldsymbol{Y_i^{(W_i)} \sim f_{W_i} + \mathcal{N}(0, \sigma^2)}$, where $\boldsymbol{\sigma^2 = 0.1}$.

[Q1]	[Q2]	Model	In-sample $\sqrt{\text{PEHE}}$	Out-of-sample $\sqrt{\text{PEHE}}$	[Q1]	[Q2]	Model	In-sample $\sqrt{\text{PEHE}}$	Out-of-sample $\sqrt{\text{PEHE}}$
✓	✓	**NSGP**	0.51 ± 0.013	0.64 ± 0.030	✓		T-XGBoost	1.46 ± 0.081	1.98 ± 0.152
		SGP	0.95 ± 0.021	1.21 ± 0.052			S-XGBoost	2.97 ± 0.211	3.04 ± 0.216
✓	✓	CMGP	0.61 ± 0.011	0.76 ± 0.012	✓		T-AdaBoost	2.40 ± 0.177	2.79 ± 0.212
✓		TARNet	0.88 ± 0.021	0.95 ± 0.025			S-AdaBoost	4.53 ± 0.317	4.56 ± 0.312
	✓	BNN	2.21 ± 0.115	2.15 ± 0.125	✓		T-OLS	1.85 ± 0.107	1.94 ± 0.122
✓	✓	CFR Wass.	0.71 ± 0.018	0.76 ± 0.032			S-OLS	5.06 ± 0.357	5.05 ± 0.352
✓	✓	CFR MMD	0.73 ± 0.021	0.78 ± 0.022	✓		T-DNN	3.36 ± 0.137	3.46 ± 0.142
✓		T-Random Forest	1.41 ± 0.071	2.21 ± 0.162			S-DNN	3.56 ± 0.217	3.64 ± 0.212
		S-Random Forest	2.72 ± 0.241	2.91 ± 0.252		✓	MARS	1.66 ± 0.106	1.74 ± 0.112
	✓	Causal Forest	2.41 ± 0.141	2.82 ± 0.181			k-NN	2.69 ± 0.177	4.06 ± 0.212
		BART	2.00 ± 0.141	2.22 ± 0.151		✓	PSM	4.92 ± 0.312	4.92 ± 0.312
	✓	BCF	1.31 ± 0.061	1.71 ± 0.102		✓	TMLE	5.27 ± 0.357	5.27 ± 0.352

Table 2.1: Simulation results for the IHDP dataset. The values reported correspond to the average PEHE ($\pm$ 95% confidence intervals).

2.6.1.2 Experiments and Results

Using the setup in Section 2.6.1.1, we conducted the following Monte Carlo simulations to verify our theoretical findings and highlight the merits of our NSGP model.

• ***Verifying Theorems 1 and 2*:** In order to check the validity of the results of Theorems 1 and 2, we use a NSGP Matérn prior $\mathcal{GP}(0, \boldsymbol{K}_\beta)$, with length-scale parameters β_0 and β_1 that are matched exactly with the regularities of the Brownian paths f_0 and f_1 (i.e., $\beta_0 = 2.5$ and $\beta_1 = 5.5$). According to Theorem 1, the optimal rate for estimating the CATE $T = f_1 - f_0$ is $n^{\frac{-5}{6}}$, and from Theorem 2, the NSGP with $\beta_0 = 2.5$ and $\beta_1 = 5.5$ should achieve that rate.

Figure 2.2 provides a scatter-plot for the PEHE achieved by the NSGP with respect to the number of samples on a log-log scale for different settings of η. We fit a linear regression model that describes the PEHE behavior in the log-log scale. We found the slope of the linear fit to be

0.8437, which is very close[2] to the slope of $\frac{5}{6} \approx 0.833$ predicted by Theorem 1. Moreover, by changing the magnitude of η from 0 to $\frac{1}{2}$, the PEHE curve did not exhibit any significant change in its slope, and was only moved upwards by a constant offset. On the contrary, Figure 2.2 shows the PEHE behavior when the NSGP prior is over-smoothed ($\beta_0 > \alpha_0$) for $\eta = 0$: as predicted by Theorem 2, learning becomes sluggish (slopes become less steep) as β_0 increase since the matching condition in (2.10) does not hold any more.

• ***NSGPs do not leave any money on the table:*** In this experiment, we show that the different components of the NSGP model allow it to perform well in small and large sample regimes. We set a strong selection bias of $\eta = \frac{1}{2}$ and compare the $\log(\text{PEHE})$ characteristic of NSGP with a model that uses the same non-stationary kernel as NSGP, and another model that uses a standard stationary kernel, but both models are tuned using marginal likelihood maximization. As we can see in Figure 2.2, the model with the non-stationary kernel achieves the same learning rate as NSGP, but exhibits a large offset as it does not account for selection bias, whereas the stationary model fails to learn the smoothness of the rougher Brownian motion since it assigns the same length-scale to both surfaces, and hence it over-smooths the prior, achieving a suboptimal rate.

2.6.2 The Infant Health and Development Program

We evaluated the performance of the NSGP model presented in Section 2.5.1 using the standard semi-synthetic experimental setup designed by Hill in [31]. We report a state-of-the-art result in this setup, and draw connections between our experimental results and our analyses.

2.6.2.1 Data and Benchmarks

The Infant Health and Development Program (IHDP) is an interventional program intended to enhance the health of premature infants [31]. [31] extracted features and treatment assignments from a real-world clinical trial, and introduced selection bias to the data artificially by removing a subset of the patients. The potential outcomes are simulated according to the standard non-linear "Response Surface B" setting in [31]. The dataset comprised 747 subjects, with 25 features for

[2]The minor discrepancy is a result of the residual error in the linear regression fit.

each subject. Our experimental setup is identical to [14, 15, 23, 31]: we run 1000 experiments in which we compute the in-sample and out-of-sample $\sqrt{\text{PEHE}}$ (with 80/20 training/testing splits), and report average results in Table 2.1.

We compared the performance of NSGP with a total of 23 CATE estimation benchmarks. We considered: tree-based algorithms (BART [31], Causal forests [13], Bayesian causal forests [40]), methods based on deep learning (CFR Wass., CFR MMD, BNN, TARnet [14]), multivariate additive regression splines (MARS) [36], Gaussian processes (CMGP) [15], nearest neighbor matching (k-NN), propensity score matching (PSM), and targeted maximum likelihood (TMLE) [53]. We also composed a number of T-learners and S-learners as in [21], using a variety of baseline machine learning algorithms (DNN stands for deep networks and OLS stands for linear regression).

2.6.2.2 Results and Conclusions

As we can see in Table 2.1, the proposed NSGP model significantly outperforms **all** competing benchmarks. The combined benefit of the two components of an NSGP (non-stationary kernel and doubly-robust hyperparameters) is highlighted by comparing its performance to a vanilla SGP (stationary GP) with marginal likelihood maximization. The gain with respect to such a model is a 2-fold improvement in the PEHE.

Because the IHDP dataset has a "moderate" sample size, both selection bias and learning rate seem to impact the performance. Thus, our method took advantage of having addressed modeling questions **[Q1]** and **[Q2]** appropriately by being both "rate-optimal" and "bias-aware".

The check marks in columns **[Q1]** and **[Q2]** designate methods that address modeling questions **[Q1]** and **[Q2]** "appropriately" in the light of the analysis presented in Section 2.3. Methods with **[Q1]** checked use a regression structure with "outcome-specific" hyperparameters, and methods with **[Q2]** checked adjust for selection bias. A general observation is that the structure of the regression model seem to matter much more than the strategy for handling selection bias. This is evident from the fact that the TARnet model (does not handle bias but models outcomes separately) significantly outperforms BNN (handles bias but uses a single-surface model [14]), and that all T-learners (models 2 separate response surfaces) outperformed their S-shaped counterparts (models

a single surface). For parametric models, such as OLS, the issue of selecting the right regression structure is even more crucial.

To sum up, the results in Table 2.1 imply that selecting the right regression structure is crucial for rate-optimality in sufficiently large dataset, whereas handling selection bias provides an extra bonus. In Table 2.1, methods that address both **[Q1]** and **[Q2]** (NSGP, CMGP, and CFR. Wass and MMD) displayed a superior performance.

CHAPTER 3

Symbolic Approaches to Prognostic Model Interpretability

The ability to interpret the predictions of a (predictive or causal) machine learning model brings about patient and clinician trust, and supports understanding of the underlying disease being modeled [54–56]. In many clinical settings, interpretability can be a crucial requirement for the deployment of machine learning, since a model's predictions would inform critical decision-making. Model explanations can also be central in other domains, such as natural sciences [57, 58], where the primary utility of a model is to help understand an underlying phenomenon, rather than merely making predictions about it. Unfortunately, most state-of-the-art models — such as ensemble models, kernel methods, and neural networks — are perceived as being complex "black-boxes", the predictions of which are too hard to be interpreted by human subjects [54, 59–69].

In this Chapter, we approach the problem of model interpretation by introducing the *symbolic metamodeling* framework for expressing black-box models in terms of transparent mathematical equations that can be easily understood and analyzed by human subjects (Section 3.1). The proposed metamodeling procedure takes as an input a (trained) model — represented by a black-box function $f(\boldsymbol{x})$ that maps a feature $\boldsymbol{x}$ to a prediction y — and retrieves a *symbolic metamodel* $g(\boldsymbol{x})$, which is meant to be an interpretable mathematical abstraction of $f(\boldsymbol{x})$. The metamodel $g(\boldsymbol{x})$ is a tractable symbolic expression comprising a finite number of familiar functions (e.g., polynomial, analytic, algebraic, or closed-form expressions) that are combined via elementary arithmetic operations (i.e., addition and multiplication), which makes it easily understood by inspection, and can be analytically manipulated via symbolic computation engines such as `Mathematica` [70], `Wolfram alpha` [71], or `Sympy` [72]. Our approach is appropriate for models with small to moderate number of features, where the physical interpretation of these features are of primary interest.

A high-level illustration of the proposed metamodeling approach is shown in Figure 3.1. In

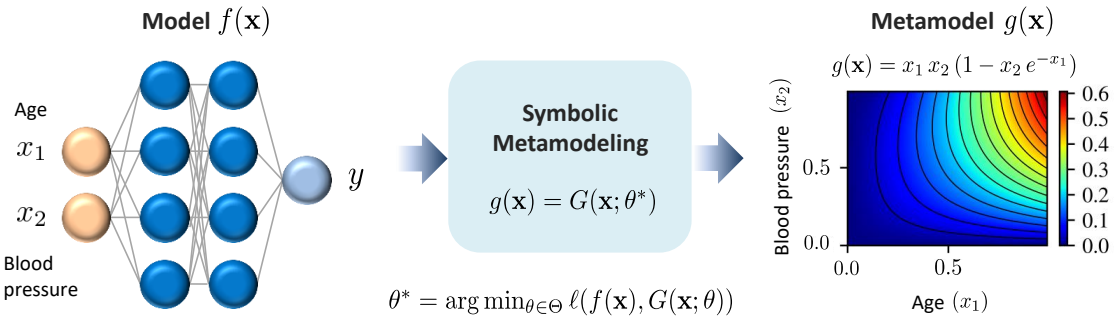


Figure 3.1: Pictorial depiction of the symbolic metamodeling framework. Here, the model $f(\boldsymbol{x})$ is a deep neural network (left), and the metamodel $g(\boldsymbol{x})$ is a closed-form expression $x_1\, x_2\, (1 - x_2\, \exp(-x_1))$ (right).

this Figure, we consider an example of using a neural network to predict the risk of cardiovascular disease based on a (normalized) feature vector $\mathbf{x} = (x_1, x_2)$, where x_1 is a person's age and x_2 is their blood pressure. For a clinician using this model in their daily practice or in the context of an epidemiological study, the model $f(\mathbf{x})$ is completely obscure — it is hard to explain or draw insights into the model's predictions, even with a background knowledge of neural networks. On the other hand, the metamodel $g(\mathbf{x}) = x_1\, x_2\, (1 - x_2 \exp(-x_1))$ is a fully transparent abstraction of the neural network model, from which one can derive explanations for the model's predictions through simple analytic manipulation, without the need to know anything about the model structure and its inner workings[1]. Having such an explicit (simulatable) equation for predicting risks is already required by various clinical guidelines to ensure the transparency of prognostic models [10].

In order to find the symbolic metamodel $g(\mathbf{x})$ that best approximates the original model $f(\mathbf{x})$, we need to search a space of mathematical expressions and find the expression that minimizes a "metamodeling loss" $\ell(g(\mathbf{x}), f(\mathbf{x}))$. But how can we construct a space of symbolic expressions without predetermining its functional from? In other words, how do we know that the metamodel $g(\mathbf{x}) = x_1\, x_2\, (1 - x_2 \exp(-x_1))$ in Figure 3.1 takes on an exponential form and not, say, a trigonometric or a polynomial functional form?

[1]Note that here we are concerned with explaining the predictions of a trained model, i.e., its *response surface*. Other works, such as [73], focus on explaining the model's *loss surface* in order to understand how it learns.

To answer this question, we introduce a novel parameterized representation of symbolic expressions (Section 3.2), $G(\mathbf{x};\theta)$, which reduces to most familiar functional forms — e.g., arithmetic, polynomial, algebraic, closed-form, and analytic expressions, in addition to special functions, such as Bessel functions and Hypergeometric functions — for different settings of a real-valued parameter θ. The representation $G(\mathbf{x};\theta)$ is based on Meijer G-functions [74–76], a class of contour integrals used in the mathematics community to find closed-form solutions for otherwise intractable integrals. The proposed Meijer G-function parameterization enables minimizing the metamodeling loss efficiently via gradient descent — this is a major departure from existing approaches to *symbolic regression*, which use genetic programming to select among symbolic expressions that comprise a small number of predetermined functional forms [77–79].

Existing methods for model interpretation focus on crafting explanation models that support only one "mode" of model interpretation. For instance, methods such as DeepLIFT [61] and LIME [69], can explain the predictions of a model in terms of the contributions of individual features to the prediction, but cannot tell us whether the model is nonlinear, or whether statistical interactions between features exist. Other methods such as GA^{2}M [62] and NIT [66], focus exclusively on uncovering the statistical interactions captured by the model, which may not be the most relevant mode of explanation in many application domains. Moreover, none of the existing methods can uncover the functional forms by which a model captures nonlinearities in the data — such type of interpretation is important in applications such as applied physics and material sciences, since researchers in these fields focus on distilling an analytic law that describes how the model fits experimental data [57, 58].

Our perspective on model interpretation departs from previous works in that, a symbolic metamodel $g(\mathbf{x})$ is not hardwired to provide any specific type of explanation, but is rather designed to provide a full mathematical description of the original model $f(\mathbf{x})$. In this sense, symbolic metamodeling should be understood as a tabula rasa upon which different forms of explanations can be derived — as we will show in Section 3.3, most forms of model explanation covered in previous literature can be arrived at through simple analytic manipulation of a symbolic metamodel.

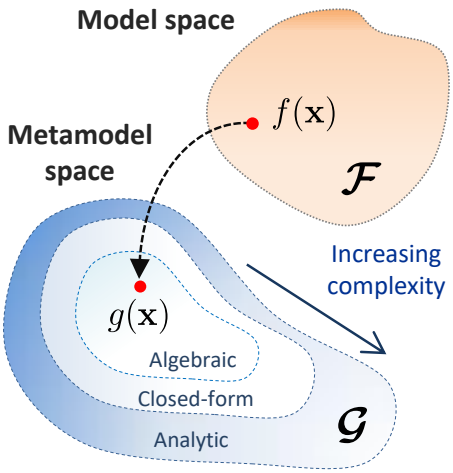


Figure 3.2: The metamodeling problem.

3.1 Symbolic Metamodeling

Let $f : \mathcal{X} \to \mathcal{Y}$ be a machine learning model trained to predict a target outcome $y \in \mathcal{Y}$ on the basis of a d-dimensional feature instance $\mathbf{x} = (x_1, \ldots, x_d) \in \mathcal{X}$. We assume that $f(.)$ is a *black-box* model to which we only have query access, i.e., we can evaluate the model's output $y = f(\mathbf{x})$ for any given feature instance $\mathbf{x}$, but we do not know the model's internal structure. Without loss of generality, we assume that the feature space $\mathcal{X}$ is the unit hypercube, i.e., $\mathcal{X} = [0, 1]^d$.

The metamodeling problem. A *symbolic metamodel* $g \in \mathcal{G}$ is a "model of the model" f that approximates $f(\mathbf{x})$ for all $\mathbf{x} \in \mathcal{X}$, where $\mathcal{G}$ is a class of succinct mathematical expressions that are understandable to users and can be analytically manipulated. Typically, $\mathcal{G}$ would be set as the class of all arithmetic, polynomial, algebraic, closed-form, or analytic expressions. Choice of $\mathcal{G}$ will depend on the desired complexity of the metamodel. In most medical applications, we might opt to restrict $\mathcal{G}$ to algebraic expressions. Given $\mathcal{G}$, the metamodeling problem consists in finding the function g in $\mathcal{G}$ that bests approximates the model f.

Figure 3.2 shows a pictorial depiction of the metamodeling problem as a mapping from the *modeling space* $\mathcal{F}$ — i.e., the function class that the model f inhabits[2] — to the interpretable

[2]For instance, for an L-layer neural network, $\mathcal{F}$ is the space of compositions of L nested activation functions. For a random forest with L trees, $\mathcal{F}$ is the space of summations of L piece-wise functions.

metamodeling space $\mathcal{G}$. Metamodling is only relevant when $\mathcal{F}$ spans functions that are considered uninterpretable to users. For models that are deemed interpretable, such as linear regression, $\mathcal{F}$ will already coincide with $\mathcal{G}$, because the linear model is already an algebraic expression (and a first-order polynomial). In this case, the best metamodel for f is the model f itself, i.e., $g = f$.

Formally, metamodeling can be formulated through the following optimization problem:

$$g^* = \arg\min_{g \in \mathcal{G}} \ell(g, f), \quad \ell(g, f) = \| f - g \|_2^2 = \int_{\mathcal{X}} (g(\mathbf{x}) - f(\mathbf{x}))^2 \, d\mathbf{x}, \tag{3.1}$$

where $\ell(.)$ is the *metamodeling loss*, which we set to be the mean squared error (MSE) between f and g. In the following Section, we will focus on solving the optimization problem in (3.1).

3.2 Metamodeling via Meijer G-functions

In order to solve the optimization problem in (3.1), we need to induce some structure into the metamodeling space $\mathcal{G}$. This is obviously very challenging since $\mathcal{G}$ encompasses infinitely many possible mathematical expressions with very diverse functional forms. For instance, consider the exemplary metamodel in Figure 3.1, where $g(\mathbf{x}) = x_1 \, x_2 \, (1 - x_2 \exp(-x_1))$. If $\mathcal{G}$ is set to be the space of all closed-form expressions, then it would include all polynomial, hyperbolic, trigonometric, logarithmic functions, rational and irrational exponents, and any combination thereof [80, 81]. Expressions such as $g'(\mathbf{x}) = (x_1^2 + x_2^2)$ and $g''(\mathbf{x}) = \sin(x_1) \cdot \cos(x_2)$ are both valid metamodels, i.e., $g', g'' \in \mathcal{G}$, yet they each have functional forms that are very different from g. Thus, we need to parameterize $\mathcal{G}$ in such a way that it encodes all such functional forms, and enables an efficient solution to (3.1).

To this end, we envision a parameterized metamodel $g(\mathbf{x}) = G(\mathbf{x}; \theta), \ \theta \in \Theta$, where $\Theta = \mathbb{R}^M$ is a parameter space that fully specifies the metamodeling space $\mathcal{G}$, i.e., $\mathcal{G} = \{G(.; \theta) : \theta \in \Theta\}$. Such parameterization should let $G(\mathbf{x}; \theta)$ reduce to different functions for different settings of θ — for the aforementioned example, we should have $G(\mathbf{x}; \theta') = (x_1^2 + x_2^2)$ and $G(\mathbf{x}; \theta'') = \sin(x_1) \cdot \cos(x_2)$ for some $\theta', \theta'' \in \Theta$. Given the parameterization $G(\mathbf{x}; \theta)$, the problem in (3.1) reduces to

$$g^*(\mathbf{x}) = G(\mathbf{x}; \theta^*), \ \text{ where } \theta^* = \arg\min_{\theta \in \Theta} \ \ell(G(\mathbf{x}; \theta), f(\mathbf{x})). \tag{3.2}$$

Thus, if we have a parameterized symbolic expression $G(\mathbf{x};\theta)$, then metamodeling boils down to a straightforward parameter optimization problem. We construct $G(\mathbf{x};\theta)$ in Section 3.2.1.

3.2.1 Parameterizing symbolic metamodels with Meijer G-functions

We propose a parameterization of $G(\mathbf{x};\theta)$ that includes two steps. The first step involves decomposing $G(\mathbf{x};\theta)$ into a combination of univariate functions. The second step involves modeling these univariate functions through a very general class of special functions that includes most known familiar functions as particular cases. Both steps are explained in detail in what follows.

Step 1: Decomposing the metamodel. We breakdown the *multivariate* function $g(\mathbf{x})$ into simpler, *univariate* functions. From the *Kolmogorov superposition theorem* [82], we know that every multivariate continuous function $g(\mathbf{x})$ can be written as a finite composition of univariate continuous functions and the addition operation as follows[3]:

$$g(\mathbf{x}) = g(x_1,\ldots,x_n) = \sum_{i=0}^{r} g_i^{out}\left(\sum_{j=1}^{d} g_{ij}^{in}(x_j)\right), \tag{3.3}$$

where g_i^{in} and g_{ij}^{out} are continuous univariate *basis functions*, and $r \in \mathbb{N}_+$. The exact decomposition in (3.3) always exists for $r = 2d$, and for some basis functions $g_i^{out} : \mathbb{R} \to \mathbb{R}$, and $g_{ij}^{in} : [0,1] \to \mathbb{R}$ [88]. When $r = 1$, (3.3) reduces to the generalized additive model [89]. While we proceed our analysis with the general formula in (3.3), in our implementation we set $r = 1$, g^{out} as the identify function, and include extra functions g_{ij}^{in} of the interactions $\{x_i\, x_j\}_{i,j}$ to account for the complexity of $g(\mathbf{x})$.

Step 2: Meijer G-functions as basis functions. Based on the decomposition in (3.3), we can now parameterize metamodels in terms of their univariate bases, i.e., $G(\mathbf{x};\theta) = G(\mathbf{x};\{g_i^{out}\}_i,\{g_{ij}^{in}\}_{i,j})$, where every selection of a different set of bases would lead to a different corresponding metamodel. However, in order to fully specify the parameterization $G(\mathbf{x};\theta)$, we still need to parameterize the basis functions themselves in terms of real-valued parameters that we can practically optimize, while ensuring that the corresponding parameter space spans a wide range of symbolic expressions.

[3]The Kolmogorov decomposition in (3.3) is a universal function approximator [83]. In fact, (3.3) can be thought of as a 2-layer neural network with generalized activation functions [83–87].

To specify $G(\mathbf{x};\theta)$, we model the basis functions in (3.3) as instances of a Meijer G-function — a *univariate* special function given by the following line integral in the complex plane s [74,75]:

$$G_{p,q}^{m,n}\left(\begin{smallmatrix}a_1,\ldots,a_p\\ b_1,\ldots,b_q\end{smallmatrix}\middle|x\right) = \frac{1}{2\pi i}\int_{\mathcal{L}} \frac{\prod_{j=1}^{m}\Gamma(b_j-s)\prod_{j=1}^{n}\Gamma(1-a_j+s)}{\prod_{j=m+1}^{q}\Gamma(1-b_j+s)\prod_{j=n+1}^{p}\Gamma(a_j+s)}\, x^s\, ds, \tag{3.4}$$

where $\Gamma(.)$ is the Gamma function and $\mathcal{L}$ is the integration path in the complex plane. The contour integral in (3.4) is known as Mellin-Barnes representation [76]. An instance of a Meijer G-function is specified by the real-valued parameters $\mathbf{a}_p = (a_1,\ldots,a_p)$, $\mathbf{b}_q = (b_1,\ldots,b_q)$, and indexes n and m, which define the *poles* and *zeros* of the integrand in (3.4) on the complex plane[4]. In the rest of this Chapter, we refer to Meijer G-functions as G functions for brevity.

For each setting of $\mathbf{a}_p$ and $\mathbf{b}_q$, the integrand in (3.4) is configured with different poles and zeros, and the resulting integral converges to a different function of x. A powerful feature of the G function is that it encompasses most familiar functions as special cases [76] — for different settings of $\mathbf{a}_p$ and $\mathbf{b}_q$, it reduces to almost all known elementary, algebraic, analytic, closed-form and special functions. Examples for special values of the poles and zeros for which the G function reduces to familiar functions are shown in Table 3.1. Perturbing the poles and zeros around their values in Table 3.1 gives rise to variants of these functional forms, e.g., $x\log(x)$, $\sin(x)$, x^2e^{-x}, etc. Tables of equivalence between G functions and familiar functions can be found in [90], or computed using programs such as `Mathematica` [70] and `Sympy` [72].

By using G functions as univariate basis functions (g_i^{in} and g_{ij}^{out}) for the decomposition in (3.1), we arrive at the following parameterization for $G(\mathbf{x};\theta)$:

$$G(\mathbf{x};\theta) = \sum_{i=0}^{r} G_{p,q}^{m,n}\left(\theta_i^{out}\,\middle|\,\sum_{j=1}^{d} G_{p,q}^{m,n}\left(\theta_{ij}^{in}\,|\,x_j\right)\right), \tag{3.5}$$

where $\theta = (\theta^{out},\theta^{in})$, $\theta^{out} = (\theta_0^{out},\ldots,\theta_r^{out})$ and $\theta^{in} = \{(\theta_{i1}^{in},\ldots,\theta_{id}^{in})\}_i$ are the G function parameters. Here, we use $G_{p,q}^{m,n}(\theta\,|\,x) = G_{p,q}^{m,n}(\mathbf{a}_p,\mathbf{b}_q\,|\,x)$, $\theta = (\mathbf{a}_p,\mathbf{b}_q)$, as a shortened notation for the G function for convenience. The indexes (m,n,p,q,r) are viewed as hyperparameters of the metamodel.

[4]Since $\Gamma(x) = (x-1)!$, the zeros of factors $\Gamma(b_j-s)$ and $\Gamma(1-a_j+s)$ are (b_j-k) and $(1-a_j-k), k\in\mathbb{N}_0$, respectively, whereas the poles of $\Gamma(1-b_j+s)$ and $\Gamma(a_j+s)$ are $(-a_j-k)$ and $(1-b_j-k), k\in\mathbb{N}_0$.

G-function	Equivalent form
$G^{0,1}_{3,1}\left(\begin{smallmatrix}2,2,2\\1\end{smallmatrix}\middle\vert x\right)$	x
$G^{1,0}_{0,1}\left(\begin{smallmatrix}-\\0\end{smallmatrix}\middle\vert x\right)$	e^{-x}
$G^{1,2}_{2,2}\left(\begin{smallmatrix}1,1\\1,0\end{smallmatrix}\middle\vert x\right)$	$\log(1+x)$
$G^{1,0}_{0,2}\left(\begin{smallmatrix}-\\0,\frac{1}{2}\end{smallmatrix}\middle\vert \frac{x^2}{4}\right)$	$\frac{1}{\sqrt{\pi}}\cos(x)$
$G^{1,2}_{2,2}\left(\begin{smallmatrix}\frac{1}{2},1\\\frac{1}{2},0\end{smallmatrix}\middle\vert x\right)$	$2\arctan(x)$

Table 3.1: Representation of familiar elementary functions in terms of the G function.

To demonstrate how the parameterization $G(\mathbf{x};\theta)$ in (3.5) captures symbolic expressions, we revisit the stylized example in Figure 3.1. Recall that in Figure 3.1, we had a neural network model with two features, x_1 and x_2, and a metamodel $g(\mathbf{x}) = x_1\, x_2\, (1 - x_2\, e^{-x_1})$. In what follows, we show how the metamodel $g(\mathbf{x})$ can be arrived at from the parameterization $G(\mathbf{x};\theta)$.

Figure 3.3 shows a schematic illustration for the parameterization $G(\mathbf{x};\theta)$ in (3.5) — with $r = 2$ — put in the format of a "computation graph". Each box in this graph corresponds to one of the basis functions $\{g_i^{in}\}_i$ and $\{g_{ij}^{out}\}_{i,j}$, and inside each box, we show the corresponding instance of G function that is needed to give rise to the symbolic expression $g(\mathbf{x}) = x_1\, x_2\, (1 - x_2 e^{-x_1})$. To tune the poles and zeros of each of the 6 G functions in Figure 3.3 to the correct values, we need to solve the optimization problem in (3.2). In Section 3.2.2, we show that this can be done efficiently via gradient descent.

3.2.2 Optimizing symbolic metamodels via gradient descent

Another advantage of the parameterization in (3.5) is that the gradients of the G function with respect to its parameters can be approximated in analytic form as follows [76]:

$$\begin{aligned}
\frac{d}{da_k} G^{m,n}_{p,q}\left(\begin{smallmatrix}\mathbf{a}_p\\\mathbf{b}_q\end{smallmatrix}\middle\vert x\right) &\approx x^{a_k-1}\cdot G^{m,n+1}_{p+1,q}\left(\begin{smallmatrix}-1,a_1-1,\ldots,a_n-1,a_{n+1}-1,\ldots,a_p-1\\b_1,\ldots,b_m,b_{m+1},\ldots,b_q\end{smallmatrix}\middle\vert x\right),\ 1\le k\le p,\\
\frac{d}{db_k} G^{m,n}_{p,q}\left(\begin{smallmatrix}\mathbf{a}_p\\\mathbf{b}_q\end{smallmatrix}\middle\vert x\right) &\approx x^{1-b_k}\cdot G^{m,n}_{p,q+1}\left(\begin{smallmatrix}a_1,\ldots,a_n,a_{n+1},\ldots,a_p\\b_1-1,\ldots,b_m-1,0,b_{m+1}-1,\ldots,b_q-1\end{smallmatrix}\middle\vert x\right)\ 1\le k\le q.
\end{aligned} \tag{3.6}$$

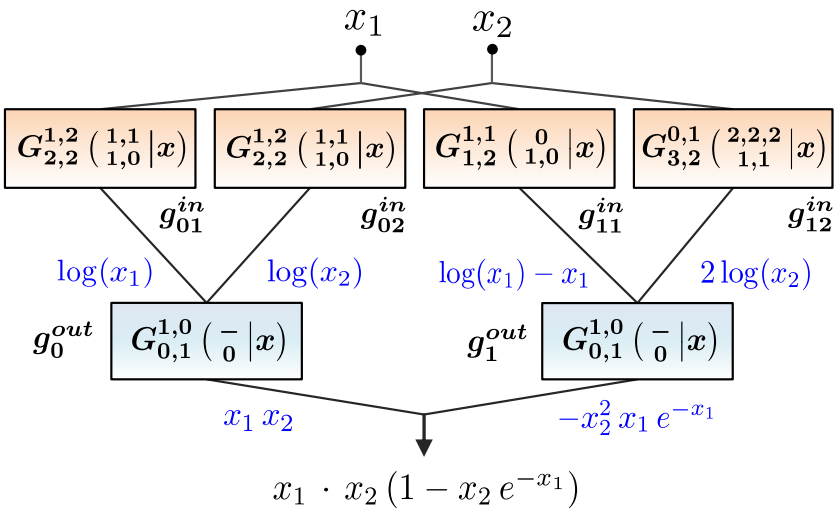


Figure 3.3: Schematic for the metamodel in Figure 3.1.

From (3.6), we see that the approximate gradient of a G function is also a G function, and hence the optimization problem in (3.2) can be solved efficiently via standard gradient descent algorithms.

The solution to the metamodel optimization problem in (3.2) must be confined to a predefined space of expressions $\mathcal{G}$. In particular, we consider the following classes of expressions:

Polynomial expressions $\subset$ **Algebraic expressions** $\subset$ **Closed-form expressions** $\subset$ **Analytic expressions**,

where the different classes of mathematical expressions correspond to different levels of metamodel complexity, with polynomial metamodels being the least complex (See Figure 3.2).

Algorithm 1 summarizes all the steps involved in solving the metamodel optimization problem. The algorithm starts by drawing n feature points uniformly at random from the feature space $[0, 1]^d$ — these feature points are used to evaluate the predictions of both the model and the metamodel in order to estimate the metamodeling loss in (3.1). Gradient descent is then executed using the gradient estimates in (3.6) until convergence. (Any variant of gradient descent can be used.) We then check if every basis function in the resulting metamodel $g(\mathbf{x})$ lies in $\mathcal{G}$. If $g(\mathbf{x}) \notin \mathcal{G}$, we search for an approximate version of the metamodel $\tilde{g}(\mathbf{x}) \approx g(\mathbf{x})$, such that $\tilde{g}(\mathbf{x}) \in \mathcal{G}$.

Algorithm 1 Symbolic Metamodeling

■ **Input:** Model $f(\mathbf{x})$, hyperparameters (m, n, p, q, r)

■ **Output:** Metamodel $g(\mathbf{x}) \in \mathcal{G}$

• $X_i \sim \text{Unif}([0,1]^d)$, $i = \{1, \ldots, n\}$.

• **Repeat until convergence:**

$$\theta^{k+1} := \theta^k - \gamma \nabla_\theta \textstyle\sum_i \ell(G(X_i;\theta), f(X_i))\big|_{\theta=\theta_k}$$

• $g(\mathbf{x}) \leftarrow G(X_i; \theta^k)$

• **If** $g(\mathbf{x}) \notin \mathcal{G}$:

$\tilde{g}(\mathbf{x}) = G(\mathbf{x}; \bar{\theta})$, $G(\mathbf{x}; \bar{\theta}) \in \mathcal{G}$, $\|\bar{\theta} - \theta^k\| < \delta$, or

$\tilde{g}(\mathbf{x}) = \text{Chebyshev}(g(\mathbf{x}))$

3.3 Related Work: Symbolic Metamodels as Gateways to Interpretation

The strand of literature most relevant to our work is the work on *symbolic regression* [77–79]. This is a regression model that searches a space of mathematical expressions using *genetic programming*. The main difference between this method and ours is that symbolic regression requires predefining the functional forms to be searched over, hence the number of its parameters increases with the number of functions that it can fit. On the contrary, our Meijer G-function parameterization enables recovering infinitely many functional forms through a fixed-dimensional parameter space, and allows optimizing metamodels via gradient descent. We compare our method with symbolic regression in Section 3.4.

Symbolic metamodeling as a unifying framework for interpretation. We now demonstrate how symbolic metamodeling can serve as a *gateway* to the different forms of model explanation covrered in the literature. To vivify this view, we go through common types of model explanation, and show that given $g(\mathbf{x})$ we can recover these explanations via analytic manipulation of $g(\mathbf{x})$.

The most common form of model explanation involves computing importance scores of each feature dimension in $\mathbf{x}$ on the prediction of a given instance. Examples for methods that provide this type of explanation include SHAP [54], INVASE [59], DeepLIFT [61], L2X [68], LIME [63,69], GAM [89], and Saliency maps [91]. Each of these methods follows one of two approaches. The first approach, adopted by saliency maps, use the *gradients* of the model output with respect to the input as a measure of feature importance. The second approach, followed by LIME, DeepLIFT, GAM and SHAP, uses *local additive approximations* to explicitly quantify the additive contribution of each feature.

Symbolic metamodeling enables a *unified* framework for (instancewise) feature importance scoring that encapsulates the two main approaches in the literature. To show how this is possible, consider the following Taylor expansion of the metamodel $g(\mathbf{x})$ around a feature point $\mathbf{x}_0$:

$$g(\mathbf{x}) = g(\mathbf{x}_0) \;+\; (\mathbf{x}-\mathbf{x}_0)\cdot\nabla_{\mathbf{x}}\, g(\mathbf{x}_0) \;+\; (\mathbf{x}-\mathbf{x}_0)\cdot\boldsymbol{H}(\mathbf{x})\cdot(\mathbf{x}-\mathbf{x}_0) \;+\; \ldots, \tag{3.7}$$

where $\boldsymbol{H}(\mathbf{x}) = [\partial^2 g/\partial x_i \partial x_j]_{i,j}$ is the Hessian matrix. Now consider — for simplicity of exposition — a second-order approximation of (3.7) with a two-dimensional feature space $\mathbf{x} = (x_1, x_2)$, i.e.,

$$\begin{aligned} g(\mathbf{x}) \approx\; g(\mathbf{x}_0) \;&+\; (x_1 - x_{0,1})\cdot g_{x_1}(\mathbf{x}_0) - x_{0,2}\cdot x_1 \cdot g_{x_1x_2}(\mathbf{x}_0) + \tfrac{1}{2}\,(x_1 - x_{0,1})^2\, g_{x_1x_1}(\mathbf{x}_0) \\ &+\; (x_2 - x_{0,2})\cdot g_{x_2}(\mathbf{x}_0) - x_{0,1}\cdot x_2 \cdot g_{x_1x_2}(\mathbf{x}_0) + \tfrac{1}{2}\,(x_2 - x_{0,2})^2\, g_{x_2x_2}(\mathbf{x}_0) \\ &+\; x_1 \cdot x_2 \cdot g_{x_1x_2}(\mathbf{x}_0), \end{aligned} \tag{3.8}$$

where $g_x = \nabla_x\, g$ and $\mathbf{x}_0 = (x_{0,1}, x_{0,2})$. In (3.8), the term in blue (first line) reflects the importance of feature x_1, the term in red (second line) reflects the importance of feature x_2, whereas the last term (third line) is the interaction between the two features. The first two terms are what generalized additive models, such as GAM and SHAP, compute. LIME is a special case of (3.8) that corresponds to a first-order Taylor approximation. Similar to saliency methods, the feature contributions in (3.8) are computed using the gradients of the model with respect to the input, but (3.8) is more general as it involves higher order gradients to capture the feature contributions more accurately. All the gradients in (3.8) can be computed efficiently since the exact gradient of the G function with respect to its input can be represented analytically in terms of another G function.

Statistical interactions between features are another form of model interpretation that has been recently addressed in [62, 66]. As we have seen in (3.8), feature interactions can be analytically derived from a symbolic metamodel. The series in (3.8) resembles the structure of the pairwise interaction model GA^{2}M in [62] and the NIT disentanglement method in [66]. Unlike both methods, a symbolic metamodel can analytically quantify the strength of higher-order (beyond pairwise) interactions with no extra algorithmic complexity. Moreover, unlike the NIT model in [66], which is tailored to neural networks, a symbolic metamodel can quantify the interactions in any machine learning model (3.7).

Table 3.2: Comparison between SM and SR.

	$\boldsymbol{f_1(x) = e^{-3x}}$	$\boldsymbol{f_2(x) = \frac{x}{(x+1)^2}}$	$\boldsymbol{f_3(x) = \sin(x)}$	$\boldsymbol{f_4(x) = J_0(10\sqrt{x})}$
SM^p	$-x^3 + \frac{5}{2}(x^2 - x) + 1$	$\frac{x^3}{3} - \frac{4x^2}{5} + \frac{2x}{3}$	$\frac{-1}{4}\,x^2 + x$	$-7\,(x^2 - x) - 1.4$
	R^2: 0.995	R^2: 0.985	R^2 : 0.999	R^2 : -4.75
SM^c	$x^{4\times10^{-6}} e^{-2.99x}$	$x\,(x+1)^{-2}$	$1.4\,x^{1.12}$	$I_{0.0003}\left(10\,e^{\frac{j\pi}{2}}\,\sqrt{x}\right)$
	R^2: 0.999	R^2: 0.999	R^2 : 0.999	R^2 : 0.999
SR	$x^2 - 1.9x + 0.9$	$\frac{0.7x}{x^2+0.9x+0.75}$	$-0.17\,x^2 + x + 0.016$	$-x\,(x - 0.773)$
	R^2: 0.970	R^2 0.981	R^2 : 0.998	R^2 : 0.116

3.4 Experiments

Building on the discussions in Section 3.3, we demonstrate the use cases of symbolic metamodeling through experiments on synthetic and real data. In all experiments, we used `Sympy` [72] (a symbolic computation library in Python) to carry out computations involving Meijer G-functions.

3.4.1 Learning Uni-variate Symbolic Expressions

We start off with four synthetic experiments with the aim of evaluating the richness of symbolic expressions discovered by our metamodeling algorithm. In each experiment, we apply Algorithm 1

(Section 3.2.2) on a ground-truth univariate function $f(x)$ to fit a metamodel $g(x) \approx f(x)$, and compare the resulting mathematical expression for $g(x)$ with that obtained by Symbolic regression [77] implemented using `gplearn` library [92].

In Table 3.2, we compare symbolic metamodeling (SM) and symbolic regression (SR) in terms of the expressions they discover and their R^2 coefficient with respect to the true functions. We consider four functions: an exponential e^{-3x}, a rational $x/(x+1^2)$, a sinusoid $\sin(x)$ and a *Bessel function* of the first kind $J_0(10\sqrt{x})$. We consider two versions of SM: SM^p for which $\mathcal{G}$ = Polynomial expressions, and SM^c for which $\mathcal{G}$ = Closed-form expressions. As we can see, SM is generally more accurate and more expressive than SR. For $f_1(x)$, $f_2(x)$ and $f_4(x)$, SM managed to figure out the functional forms of the true functions ($J_0(x) = I_0(e^{\frac{j\pi}{2}}x)$, where $I_0(x)$ is the *Bessel function of the second kind*. For $f_3(x)$, SM^c recovered a parsimonious approximation $g_3(x)$ since $\sin(x) \approx x$ for $x \in [0, 1]$. Moreover, SM^p managed to retrieve more accurate polynomial expressions than SR.

3.4.2 Instance-wise feature importance

Now we evaluate the ability of symbolic metamodels to explain predictions in terms of instance-wise feature importance (Section 3.3). To this end, we replicate the experiments in [68] with the following synthetic data sets: *XOR*, *Nonlinear additive features*, and *Feature switching*. (See Section 4.1 in [68] or Appendix B for a detailed description of the data sets.) Each data set has a 10-dimensional feature space and 1000 data samples.

For each of the three data sets above, we fit a 2-layer neural network $f(\mathbf{x})$ (with 200 hidden units) to predict the labels based on the 10 features, and then fit a symbolic metamodel $g(\mathbf{x})$ for the trained network $f(\mathbf{x})$ using the algorithm in Section 3.2.2. Instancewise feature importance is derived using the (first-order) Taylor approximation in (3.8). Since the underlying true features are known for each sample, we use the median feature importance ranking of each algorithm as a measure of the accuracy of its feature ranks as in [68]. Lower median ranks correspond to more accurate algorithms.

In Figure 3.4, we compare the performance of metamodeling (SM) with DeepLIFT, SHAP,

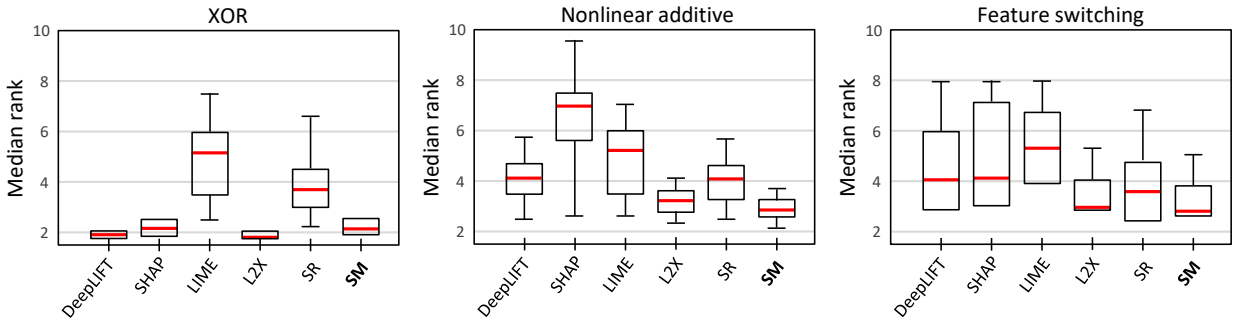


Figure 3.4: Box-plots for the median ranks of features by their estimated importance per sample over the 1000 samples of each data set. The red line is the median. Lower median ranks are better.

LIME, and L2X. We also use the Taylor approximation in (3.8) to derive feature importance scores from a symbolic regression (SR) model as an additional benchmark. For all data sets, SM performs competitively compared to L2X, which is optimized specifically to estimate instancewise feature importance. Unlike LIME and SHAP, SM captures the strengths of feature interactions, and consequently it provides more modes of explanation even in the instances where it does not outperform the additive methods in terms of feature ranking. Moreover, because SM recovers more accurate symbolic expressions than SR, it provides a more accurate feature ranking as a result.

CHAPTER 4

Automated Prognostic Modeling

Given the abundance of ML-based predictive and causal models, which model should we use for the dataset at hand? Despite a variety of ML-based modeling options at our disposal, there is, however, a concerning gap between the potential and actual utilization of ML in prognostic research; the reason being that clinicians with no expertise in data science find it hard to manually design and tune ML pipelines [93]. To fill this gap, we developed AUTOPROGNOSIS, an automated ML (AutoML) framework tailored for clinical prognostic modeling that encapsulates the modules presented in the previous Chapters, and automates their design choices. AUTOPROGNOSIS takes as an input data from a patient cohort, and uses such data to automatically configure ML *pipelines*. Every ML pipeline comprises all stages of prognostic modeling: missing data imputation, feature preprocessing, prediction, and calibration. An overview of the system is provided in Figure 4.1.

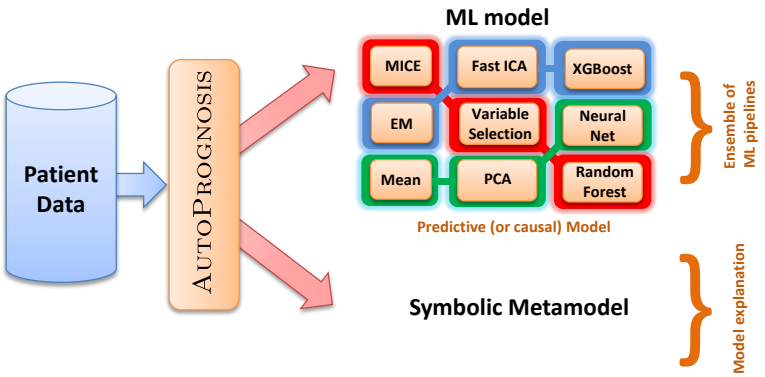


Figure 4.1: High-level illustration for AUTOPROGNOSIS.

The core component of AUTOPROGNOSIS is an algorithm for configuring ML pipelines using Bayesian optimization (BO) [94]. Our BO algorithm models the pipelines' performances as a black-box function, the input to which is a "pipeline configuration", i.e. a selection of algorithms

and hyperparameter settings, and the output of which is the performance (predictive accuracy) achieved by such a configuration. We implement BO with a *Gaussian process* (GP) prior on the black-box function. To deal with the high-dimensionality of the pipeline configuration space, we capitalize on the fact that for a given dataset, **the performance of one ML algorithm may not be correlated with that of another algorithm**. For instance, it may be the case that the observed empirical performance of *logistic regression* on a given dataset does not tell us much information about how a *neural network* would perform on the same dataset. In such a case, both algorithms should not share the same GP prior, but should rather be modeled independently. Our BO **learns** such a decomposition of algorithms from data in order to break down the high-dimensional optimization problem into a set of lower-dimensional sub-problems. We model the decomposition of algorithms via an additive kernel with a Dirichlet prior on its structure, and learn the decomposition from data in concurrence with the BO iterations. We also propose a batched (parallelized) version of the BO procedure, along with a computationally efficient algorithm for maximizing the BO acquisition function.

In addition to the causal models we proposed in Chapter 2, numerous machine learning-based models for causal inference were developed in the past few years, capitalizing on ideas from representation learning [95], multi-task learning [96] and adversarial training [24]. The literature on machine learning-based causal inference is constantly growing, with various related workshops and competitions being held every year [97]. Automating the selection of ML causal models is tricky since it is impossible to know what the counterfactual outcome would have been had patients received an alternative treatment. Since causal effects are determined by both factual and counterfactual outcomes, ground-truth effects can never be measured in an observational study, and hence empirical validation of causal modeling choices is anything but straightforward [98].

To address this issue, we use influence functions — a key technique in robust statistics and efficiency theory [49, 99] — to develop a model validation procedure that estimates the performance of causal inference methods applied to a given observational dataset **without the need to access counterfactual data**, in order to enable the AUTOPROGNOSIS system to select among causal inference models. To the best of our knowledge, ours is the first validation procedure for models of individualized causal effects. Our procedure can be easily extended to other under-explored

problems involving unlabeled data, such as semi-supervised learning [100].

4.1 Overview of Related Literature

4.1.1 Automated ML and Bayesian Optimization

To the best of our knowledge, none of the existing AutoML frameworks, such as AUTO-WEKA [101], AUTO-SKLEARN [102], and TPOT [103] use principled GP-based BO to configure ML pipelines. All of the existing frameworks model the sparsity of the pipelines' hyperparameter space via frequentist tree-based structures. Both AUTO-WEKA and AUTO-SKLEARN use BO, but through tree-based heuristics, such as random forest models and tree Parzen estimators, whereas TPOT uses a tree-based genetic programming algorithm. Previous works have refrained from using principled GP-based BO because of its statistical and computational complexity in high-dimensional hyperparameter spaces. Our algorithm makes principled, high-dimensional GP-based BO possible by **learning** a sparse additive kernel decomposition for the GP prior. This approach confers many advantages as it captures the uncertainty about the sparsity structure of the GP prior, and allows for principled approaches for (Bayesian) meta-learning and ensemble construction that are organically connected to the BO procedure.

Various previous works have addressed the problem of high-dimensional GP-based BO. [104] identifies a low-dimensional effective subspace for the black-box function via random embedding. However, in the AutoML setup, this approach cannot incorporate our prior knowledge about dependencies between the different hyperparameters (we know the sets of hyperparameters that are "activated" upon selecting an algorithm [105]). This prior knowledge was captured by the *Arc-kernel* proposed in [106], and similarly in [107], where a BO algorithm for domains with tree-structured dependencies was proposed. Unfortunately, both methods require full prior knowledge of the dependencies between the hyperparameters, and hence cannot be used when jointly configuring hyperparameters across multiple algorithms, since the correlations of the performances of different algorithms are not known a priori. [108] proposed a naïve approach that defines an independent GP for every set of hyperparameters that belong to the same algorithm. Since it does

not share any information between the different algorithms, this approach would require trying all combinations of algorithms in a pipeline exhaustively. (In our system, there are 4,800 possible pipelines.) Our model solves the problems above via a **data-driven** kernel decomposition, through which only relevant groups of hyperparameters share a common GP prior, thereby balancing the trade-off between "information sharing" among hyperparameters and statistical efficiency.

4.1.2 Causal Model Validation

Researchers developing new methods for causal inference validate their models using synthetic data-generating distributions that encode pre-specified causal effects — e.g., [13, 31, 36]. However, such synthetic distributions bear very little resemblance to real-world data, and hence are not informative of what methods would actually work best on a given real-world observational study [109]. Because no single model will be superior on all observational studies [97], model selection must be guided by a data-driven validation procedure.

While the literature is rich with causal inference models, it falls short of rigorous methods for validating those models on real-world data. Applied researchers currently rely on simple heuristics to predict a model's performance on a given dataset [110–112], but such heuristics do not provide any theoretical guarantees, and can fail badly in certain scenarios [113].

Despite their popularity in statistics, influence functions are seldom used in machine learning. Recently in [73], influence functions were used for interpreting black-box models by tracing the impact of data points on a model's predictions. Our usage of influence functions differs from [73] in that we use them to construct efficient estimators of a model's loss and not to explain the inner workings of a learning algorithm. In that sense, our work is more connected to the literature on plug-in estimation and nonparametric efficiency theory [49, 114–116].

4.2 AUTOPROGNOSIS: A System for Automated Prognostic Modeling

We start with the predictive modeling setup, where our goal is to automate the design of predictive models for the dataset at hand. Consider a dataset $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^{n}$ for a cohort of n patients, with

Pipeline Stage	Algorithms			
□ Data Imputation	□ missForest (2)	□ Median (0)	□ Most-frequent (0)	□ Mean (0)
	□ Matrix completion (2)	□ MICE (1)	□ None (0)	□ EM (1)
♣ Feature process.	♣ Feature agglo. (4)	♣ Kernel PCA (5)	♣ Polynomial (3)	♣ Fast ICA (4)
	♣ R. kitchen sinks (2)	♣ Nystroem (5)	♣ Linear SVM (3)	♣ Select Rates (3)
	♣ PCA (2)	♣ None (0)		
• Prediction	• Bernoulli NB (2)	• AdaBoost (4)	• Decision Tree (4)	• Grad. Boost. (6)
	• Gaussian NB (0)	• XGBoost (5)	• Extr. R. Trees (5)	• Light GBM (5)
	• Multinomial NB (2)	• R. Forest (5)	• Neural Net. (5)	• Log. Reg. (0)
	• Ridge Class. (1)	• Bagging (4)	• k-NN (1)	• Surv. Forest (5)
	• LDA (4)	• L. SVM (4)	• GP (3)	• Cox Reg. (0)
⋆ Calibration	⋆ Sigmoid (0)	⋆ Isotonic (0)	⋆ None (0)	

Table 4.1: List of algorithms included in every stage of the pipeline. Numbers in brackets correspond to the number of hyperparameters.

x_i being patient i's features, and y_i being the patient's clinical endpoint. AUTOPROGNOSIS takes $\mathcal{D}$ as an input, and outputs an automatically configured *prognostic model* which predicts the patients' risks. This Section provides an overview of the components of AUTOPROGNOSIS.

The core component of AUTOPROGNOSIS is an algorithm that automatically configures ML pipelines, where every pipeline comprises algorithms for missing data imputation (□), feature preprocessing (♣), prediction (•), and calibration (★). Table 4.1 lists the baseline algorithms adopted by the system in all the stages of a pipeline. The imputation and calibration stages are particularly important for clinical prognostic modeling [117], and are not supported in existing AutoML frameworks. The total number of hyperparameters in AUTOPROGNOSIS is 106, which is less than those of AUTO-WEKA (786) and AUTO-SKLEARN (110). The pipeline configuration algorithm uses **Bayesian optimization** to estimate the performance of different pipeline configurations in a **scalable** fashion by learning a **structured kernel decomposition** that identifies algorithms with *correlated* performance. Details of the Bayesian optimization algorithm are provided in Sections 4.3.

4.3 Pipeline Configuration via Structured Bayesian Optimization

Let $(\mathcal{A}_d, \mathcal{A}_f, \mathcal{A}_p, \mathcal{A}_c)$ be the sets of all missing data imputation, feature processing, prediction, and calibration algorithms in AUTOPROGNOSIS (Table 4.1), respectively. A **pipeline** P is a tuple:

$$P = (A_d, A_f, A_p, A_c)$$

where $A_v \in \mathcal{A}_v$, $\forall v \in \{d, f, p, c\}$. The space of all pipelines is given by $\mathcal{P} = \mathcal{A}_d \times \mathcal{A}_f \times \mathcal{A}_p \times \mathcal{A}_c$. Thus, a pipeline is a selection of algorithms from the elements of Table 4.1. An exemplary pipeline can be specified as follows: $P = \{$**MICE**, **PCA**, **Random Forest**, **Sigmoid**$\}$. The total number of pipelines in AUTOPROGNOSIS is $|\mathcal{P}| = 4,800$.

The specification of a **pipeline configuration** is completed by determining the hyperparameters of its constituting algorithms. The space of hyperparameter configurations for a pipeline is $\Theta = \Theta_d \times \Theta_f \times \Theta_p \times \Theta_c$, where $\Theta_v = \cup_a \Theta_v^a$, for $v \in \{d, f, p, c\}$, with Θ_v^a being the space of hyperparameters associated with the a^{th} algorithm in $\mathcal{A}_v$. Thus, a pipeline configuration $P_\theta \in \mathcal{P}_\Theta$ is a selection of algorithms $P \in \mathcal{P}$, and hyperparameter settings $\theta \in \Theta$; $\mathcal{P}_\Theta$ is the space of all possible pipeline configurations.

4.3.1 The Pipeline Selection & Configuration Problem

The main goal of AUTOPROGNOSIS is to identify the best pipeline configuration $P^*_{\theta^*} \in \mathcal{P}_\Theta$ for a given patient cohort $\mathcal{D}$ via J-fold cross-validation as follows:

$$P^*_{\theta^*} \in \arg\max_{P_\theta \in \mathcal{P}_\Theta} \frac{1}{J} \sum_{i=1}^{J} \mathcal{L}(P_\theta; \mathcal{D}^{(i)}_{\text{train}}, \mathcal{D}^{(i)}_{\text{valid}}), \tag{4.1}$$

where $\mathcal{L}$ is a given accuracy metric (AUC-ROC, c-index, etc), $\mathcal{D}^{(i)}_{\text{train}}$ and $\mathcal{D}^{(i)}_{\text{valid}}$ are training and validation splits of $\mathcal{D}$ in the i^{th} fold. The optimization problem in (4.1) is dubbed the *Pipeline Selection and Configuration Problem* (PSCP). The PSCP can be thought of as a generalization for the *combined algorithm selection and hyperparameter optimization* (CASH) problem in [101,102], which maximizes an objective with respect to selections of single algorithms from the set $\mathcal{A}_p$, rather than selections of full-fledged pipelines from $\mathcal{P}_\Theta$.

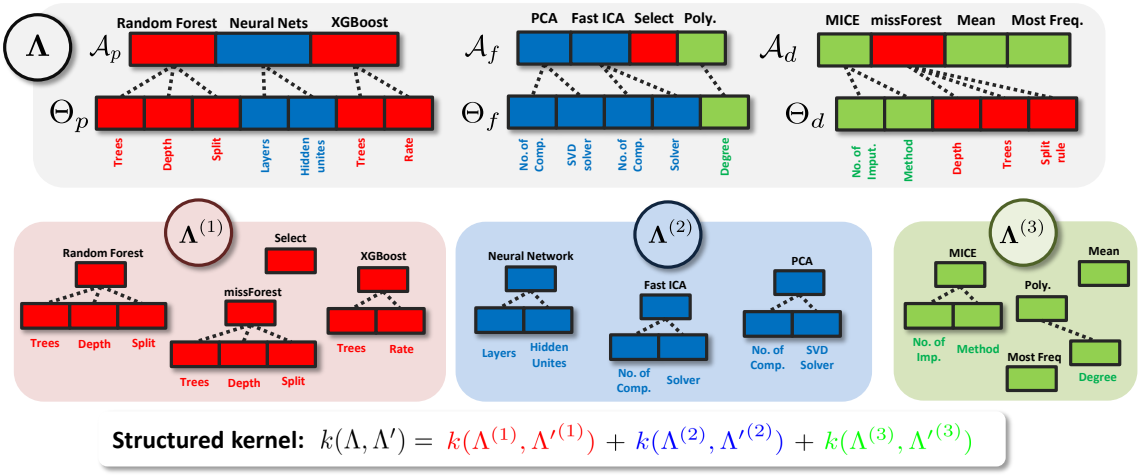


Figure 4.2: Illustration for a exemplary subspace decomposition $\{\mathbf{\Lambda}^{(m)}\}_{m=1}^{3}$.

4.3.2 Solving the PSCP via Bayesian Optimization

The objective in (4.1) has no analytic form, and hence we treat the PSCP as a *black-box* optimization problem. In particular, we assume that $\frac{1}{J}\sum_{i=1}^{J}\mathcal{L}(P_\theta;\mathcal{D}_{\text{train}}^{(i)},\mathcal{D}_{\text{valid}}^{(i)})$ is a noisy version of a black-box function $f:\mathbf{\Lambda}\to\mathbb{R}$, were $\mathbf{\Lambda}=\Theta\times\mathcal{P}$, and use BO to search for the pipeline configuration $P_{\theta^*}^*$ that maximizes the black-box function $f(.)$ [94]. The BO algorithm specifies a Gaussian process (GP) prior on $f(.)$ as follows:

$$f\sim\mathcal{GP}(\mu(\Lambda),k(\Lambda,\Lambda')), \tag{4.2}$$

where $\mu(\Lambda)$ is the *mean function*, encoding the expected performance of different pipeline, and $k(\Lambda,\Lambda')$ is the *covariance kernel* [41], encoding the similarity between the different pipelines.

4.3.3 Bayesian Optimization via Structured Kernels

The function f is defined over the D-dimensional space $\mathbf{\Lambda}$, where $D=\dim(\mathbf{\Lambda})$ is given by

$$D=\dim(\mathcal{P})+\sum_{v\in\{d,f,p,c\}}\sum_{a\in\mathcal{A}_v}\dim(\Theta_v^a). \tag{4.3}$$

In AUTOPROGNOSIS, the domain $\mathbf{\Lambda}$ is high-dimensional, with $D=106$. (The dimensionality of $\mathbf{\Lambda}$ can be calculated by summing up the number of pipeline stages and the number of hyperparameters in Table 4.1.) High-dimensionality renders standard GP-based BO infeasible as both the sample

complexity of nonparametric estimation and the computational complexity of maximizing the acquisition function are exponential in D [118, 119]. For this reason, existing AutoML frameworks have refrained from using GP priors, and relied instead on scalable tree-based heuristics [101, 102]. Despite its superior performance, recent empirical findings have shown that plain-vanilla GP-based BO is feasible only for problems with $D \leq 10$ [104]. Thus, the deployment of GP-based BO has been limited to hyperparameter optimization for single, pre-defined ML models via tools such as `Google`'s `Visier` and `HyperTune` [120]. AUTOPROGNOSIS overcomes this challenge by leveraging the structure of the PSCP problem as we show in what follows.

4.3.3.1 The Structure of the PSCP Problem

The key idea of our BO algorithm is that for a given dataset, **the performance of a given group of algorithms may not be *informative* of the performance of another group of algorithms**. Since the kernel $k(\Lambda, \Lambda')$ encodes the correlations between the performances of the different pipeline configurations, the underlying "informativeness" structure that relates the different hyperparameters can be expressed via the following **sparse additive kernel decomposition**:

$$k(\Lambda, \Lambda') = \sum_{m=1}^{M} k_m(\Lambda^{(m)}, \Lambda'^{(m)}), \tag{4.4}$$

where $\Lambda^{(m)} \in \mathbf{\Lambda}^{(m)}, \forall m \in \{1, \ldots, M\}$, with $\{\mathbf{\Lambda}^{(m)}\}_m$ being a set of *disjoint* subspaces of $\mathbf{\Lambda}$. (That is, $\cup_m \Lambda^{(m)} = \mathbf{\Lambda}$, and $\Lambda^{(m)} \cap \Lambda^{(m')} = \emptyset$.) The subspaces are assigned mutually exclusive subsets of the dimensions of $\mathbf{\Lambda}$, so that $\sum_m \dim(\Lambda^{(m)}) = D$. The structure of the kernel in (4.4) is **unknown** a priori, and needs to be **learned from data**. The kernel decomposition breaks down f as follows:

$$f(\Lambda) = \sum_{m=1}^{M} f_m(\Lambda^{(m)}). \tag{4.5}$$

The additively sparse structure in (4.4) gives rise to a statistically efficient BO procedure. That is, if f is γ-smooth, then our additive kernels reduce **sample complexity** from $O(n^{\frac{-\gamma}{2\gamma+D}})$ to $O(n^{\frac{-\gamma}{2\gamma+D_m}})$, where D_m is the maximum number of dimensions in any subspace [33, 35]. (Similar improvements hold for the cumulative regret [119].)

Each subspace $\mathbf{\Lambda}^{(m)} \subset \mathbf{\Lambda}$ contains the hyperparameters of algorithms with correlated performances, whereas algorithms residing in two different subspaces $\mathbf{\Lambda}^{(m)}$ and $\mathbf{\Lambda}^{(m')}$ have uncorrelated

performances. Since a hyperparameter in Θ is only *relevant* to $f(.)$ when the corresponding algorithm in $\mathcal{P}$ is selected [121], then the decomposition $\{\mathbf{\Lambda}^{(m)}\}_m$ must ensure that all the hyperparameters of the same algorithm are bundled together in the same subspace. This a priori knowledge about the "conditional relevance" of the dimensions of $\mathbf{\Lambda}$ makes it easier to learn the kernel decomposition from data. Figure 4.2 provides an illustration for an exemplary subspace decomposition for the hyperparameters of a set of prediction, feature processing and imputation algorithms. Since the structured kernel in (4.4) is not fully specified a priori, we propose an algorithm to learn it from the data in the next Section.

4.3.3.2 Structured Kernel Learning

AUTOPROGNOSIS uses a Bayesian approach to learn the subspace decomposition $\{\mathbf{\Lambda}^{(m)}\}_m$ in concurrence with the BO procedure, where the following Dirichlet-Multinomial prior is placed on the structured kernel [122]:

$$\alpha \sim \text{Dirichlet}(M, \gamma), \;\; z_{v,a} \sim \text{Multi}(\alpha), \tag{4.6}$$

$\forall a \in \mathcal{A}_v, v \in \{d, f, p, c\}$, where $\gamma = \{\gamma_m\}_m$ is the parameter of a Dirichlet prior, $\alpha = \{\alpha_m\}_m$ are the Multinomial mixing proportions, and $z_{v,a}$ is an indicator variable that determines the subspace to which the a^{th} algorithm in $\mathcal{A}_v$ belongs. The kernel decomposition in (4.4) is learned by updating the posterior distribution of $\{\mathbf{\Lambda}^{(m)}\}_m$ in every iteration of the BO procedure. The posterior distribution over the variables $\{z_{v,a}\}_{v,a}$ and α is given by:

$$\mathbb{P}(z, \alpha \,|\, \mathcal{H}_t, \gamma) \propto \mathbb{P}(\mathcal{H}_t \,|\, z)\, \mathbb{P}(z \,|\alpha)\, \mathbb{P}(\alpha, \gamma), \tag{4.7}$$

where $z = \{z_{v,a} : \forall a \in \mathcal{A}_v, \forall v \in \{d, f, p, c\}\}$, and $\mathcal{H}_t$ is the history of evaluations of the black-box function up to iteration t. Since the variables $\{z_{v,a}\}_{v,a}$ are sufficient statistics for the subspace decomposition, the posterior over $\{\mathbf{\Lambda}^{(m)}\}_m$ is fully specified by (4.6) marginalized over α, which can be evaluated using Gibbs sampling as follows:

$$\mathbb{P}(z_{v,a} = m \,|\, z/\{z_{v,a}\}, \mathcal{H}_t) \propto \mathbb{P}(\mathcal{H}_t \,|\, z)\, (|\mathcal{A}_v^{(m)}| + \gamma_m),$$

where $\mathbb{P}(\mathcal{H}_t \,|\, z)$ is the GP likelihood under the kernel induced by z. The Gibbs sampler is imple-

mented via the Gumble-Max trick [123] as follows:

$$\omega_m \overset{\text{i.i.d}}{\sim} \text{Gumbel}(0,1),\ m \in \{1,\ldots,M\}, \tag{4.8}$$
$$z_{v,a} \sim \arg\max_m \mathbb{P}(\mathcal{H}_t \,|\, z, z_{v,a}=m)(|\mathcal{A}_v^{(m)}| + \gamma_m) + \omega_m.$$

4.3.3.3 Exploration via Diverse Batch Selection

The BO procedure solves the PSCP problem by exploring the performances of a sequence of pipelines $\{P_{\theta^1}^1, P_{\theta^2}^2, \ldots\}$ until it (hopefully) converges to the optimal pipeline $P_{\theta^*}^*$. In every iteration t, BO picks a pipeline to evaluate using an *acquisition function* $A(P_\theta; \mathcal{H}_t)$ that balances between *exploration* and *exploitation*. AUTOPROGNOSIS deploys a 2-step batched (parallelized) exploration scheme that picks B pipelines for evaluation at every iteration t as follows:

- **Step 1:** Select the frequentist kernel decomposition $\{\hat{\mathbf{\Lambda}}^{(m)}\}_m$ that maximizes $\mathbb{P}(z \,|\, \mathcal{H}_t)$.
- **Step 2:** Select the B pipelines $\{P_\theta^b\}_{b=1}^B$ with the highest values for the acquisition function $\{A(P_\theta^b; \mathcal{H}_t)\}_{b=1}^B$, such that each pipeline P_θ^b, $b \in \{1,\ldots,B\}$, involves a **distinct prediction** algorithm from a **distinct subspace** in $\{\hat{\mathbf{\Lambda}}^{(m)}\}_m$.

We use the well-known *Upper Confidence Bound* (UCB) as acquisition function [94]. The decomposition in (4.4) offers an **exponential speed up** in the overall **computational complexity** of Step 2 since the UCB acquisition function is maximized separately for every (low-dimensional) component f_m; this reduces the number of computations from to $\mathcal{O}(n^{-D})$ to $\mathcal{O}(n^{-D_m})$. The batched implementation is advantageous since sequential evaluations of $f(.)$ are time consuming as it involves training the selected ML algorithms.

Step 2 in the algorithm above encourages exploration as follows. In every iteration t, we select a "diverse" batch of pipelines for which every pipeline is representative of a **distinct** subspace in $\{\hat{\mathbf{\Lambda}}^{(m)}\}_m$. The batch selection scheme above encourages diverse exploration without the need for sampling pipelines via a determinantal point process with an exponential complexity as in [122, 124, 125]. We also devise an efficient **backward induction** algorithm that exploits the structure of a pipeline to maximize the acquisition function efficiently.

4.4 Validating Causal Models

In the previous sections, we developed a BO procedure for conducting predictive model selection. However, for causal models, the empirical performance measure $\mathcal{L}$ cannot be straightforwardly evaluated. In this Section, we develop an efficient estimator for the empirical performance of a causal inference model based on observational data.

4.4.1 Notation and Definitions

4.4.1.1 Causal Inference from Observational Data

Recall from Chapter 2 the standard *potential outcomes* framework for modeling causal effects in observational and experimental studies [17, 126]. In this framework, a "subject" is associated with a feature $X \in \mathcal{X}$, a treatment assignment indicator $W \in \{0, 1\}$, and an outcome $Y \in \mathbb{R}$. The outcome variable Y takes on the value of either of the two "potential outcomes" $Y^{(0)}$ and $Y^{(1)}$, where $Y = Y^{(1)}$ if the subject received the treatment ($W = 1$), and $Y = Y^{(0)}$ otherwise, i.e., $Y = W\,Y^{(1)} + (1 - W)\,Y^{(0)}$. The causal effect of the treatment on the subject is thus given by $Y^{(1)} - Y^{(0)}$.

■ **Observational data.** In a typical observational study, we are given n samples of the tuple $Z = (X, W, Y)$ drawn from a probability distribution with a parameter θ, i.e.,

$$Z_1, \ldots, Z_n \sim \mathbb{P}_\theta\,, \tag{4.9}$$

where $\mathbb{P}_\theta$ belongs to the family $\mathcal{P} = \{\mathbb{P}_{\theta'} : \theta' \in \Theta\}$, and Θ is the parameter space. We break down the parameter θ into a collection of *nuisance* parameters $\theta = \{\mu_0, \mu_1, \pi, \eta\}$, where μ_0 and μ_1 are the conditional potential outcomes, i.e.,

$$\mu_w(x) = \mathbb{E}_\theta\big[\, Y^{(w)} \,|\, X = x \,\big],\ w \in \{0, 1\}, \tag{4.10}$$

and π is the treatment assignment mechanism, i.e.

$$\pi(x) = \mathbb{P}_\theta(W = 1 \,|\, X = x\,), \tag{4.11}$$

whereas $\eta(x) = \mathbb{P}_\theta(X = x)$. To ensure the generality of our analysis, we assume that $\mathcal{P}$ is a *non-parametric* family of distributions. That is, Θ is an infinite-dimensional parameter space, with the

nuisance parameters $\{\mu_0, \mu_1, \pi, \eta\}$ being specified only through mild smoothness conditions.

▪ **The causal inference task.** The goal of causal inference is to use the samples $\{Z_i\}_{i=1}^n$ in order to infer the causal effect of the treatment on individual subjects based on their features, i.e., the estimand is a function $T : \mathcal{X} \to \mathbb{R}$, where

$$T(x) = \mathbb{E}_\theta\big[\, Y^{(1)} - Y^{(0)} \,|\, X = x \,\big]. \tag{4.12}$$

The function $T(x)$ in (4.12) is commonly known as the conditional average treatment effect (CATE)[1]. Its importance resides in the fact that it can guide *individualized* decision-making policies (e.g., patient-specific treatment plans or personalized advertising policies [19]). For this reason, the CATE function is the estimand of interest for almost all modern machine learning-based causal inference methods (e.g., [13, 24, 95, 96]).

▪ **Accuracy of causal inference.** A causal inference model $\mathcal{M}$ maps a dataset $\{Z_i\}_{i=1}^n$ to an estimate $\widehat{T}(.)$ of the CATE. The accuracy of a model is typically characterized by the squared-L^2 loss incurred by its estimate, i.e.,

$$\boldsymbol{\ell}_\theta(\widehat{T}) \triangleq \big\| \widehat{T}(X) - T(X) \big\|_\theta^2, \tag{4.13}$$

where $\|f(X)\|_\theta^2 = \mathbb{E}_\theta[f^2(X)]$. The performance evaluation metric in (4.13) was dubbed the *precision of estimating heterogeneous effects* (PEHE) in [31] — it quantifies the ability of a model to capture the heterogeneity of the causal effects of a treatment among individuals in a population.

4.4.1.2 Model Validation & Selection

We now consider a set of candidate causal inference models $\overrightarrow{\mathcal{M}} = \{\mathcal{M}_1, \ldots, \mathcal{M}_M\}$. These may include, for example, different machine learning methods (e.g., Causal Gaussian processes, GANITE, causal forests, etc.), different hyperparameter settings of one method, etc. Our goal is to select the best model $\mathcal{M}^* \in \overrightarrow{\mathcal{M}}$ that incurs the minimum PEHE for a given dataset.

[1] To ensure the identification of the CATE, we assume that $\mathbb{P}_\theta$ satisfies the standard "unconfoundedness" and "overlap" conditions in [17, 127].

■ **Beyond cross-validation.** Evidently, reliable model selection requires a model validation procedure that estimates the PEHE accuracy of each model in $\overrightarrow{\mathcal{M}}$. Unlike standard supervised learning in which all data points are definitely "labeled", in the causal inference setting we do not have access to the ground-truth causal effect $Y^{(1)} - Y^{(0)}$. This is because in an observational dataset, we can only observe the factual outcome $Y^{(W)}$, but not the counterfactual $Y^{(1-W)}$. This renders the empirical measure of PEHE, i.e., $\frac{1}{n}\sum_{i=1}^{n}(\widehat{T}(X_i) - (Y_i^{(1)} - Y_i^{(0)}))^2$, incalculable from the samples $\{Z_i = (X_i, W_i, Y_i)\}_{i=1}^{n}$, and hence standard cross-validation techniques cannot be used to evaluate the performance of a given causal inference model[2].

4.5 Causal Model Validation via Influence Functions

How can we test the PEHE performance of a causal inference model without observing a test label $Y^{(1)} - Y^{(0)}$*?* To answer this question, we develop a consistent and efficient validation procedure that estimates the PEHE of any causal inference model via a statistic that **does not depend on the counterfactual data** $\boldsymbol{Y^{(1-W)}}$. Using this procedure, practitioners can evaluate, compare and select causal inference models as envisioned in Section 4.4.1.

Our validation procedure adopts a *plug-in* estimation principle [128], whereby the true (unobserved) causal effect T is replaced with an estimate $\widetilde{T}$. The key idea of our procedure is that — since PEHE is a *functional* of distributions spanned by Θ — if we know a model's PEHE under a proximal plug-in distribution $\mathbb{P}_{\tilde{\theta}} \approx \mathbb{P}_{\theta}$, then we can approximate its true PEHE under $\mathbb{P}_{\theta}$ using a (generalized) Taylor expansion. In such an expansion, the *influence functions* of the PEHE functional are analogous to derivatives of a function in standard calculus.

A high-level description of our procedure is given below.

1. **Step 1: Plug-in estimation**

    - Fit a *plug-in* model $\tilde{\theta} = \{\tilde{\mu}_0, \tilde{\mu}_1, \tilde{\pi}, \tilde{\eta}\}$.

[2] In Appendix B, we analyze a number of naïve alternatives to cross-validation that were used in previous works to tune the hyperparameter of causal inference models [14, 38], etc.). We show that all such alternatives provide either inconsistent or inefficient estimates of the PEHE.

- Compute a plug-in estimate $\boldsymbol{\ell}_{\tilde{\theta}}$ of the PEHE.

2. **Step 2: Unplugged validation**

   Use the influence functions of $\boldsymbol{\ell}_{\tilde{\theta}}$ to predict $\boldsymbol{\ell}_{\theta}$.

In what follows, we provide a detailed explanation of the two-step procedure above.

4.5.1 Step 1: Plug-in Estimation

In Step 1, we obtain an initial guess of a model's PEHE by evaluating the PEHE functional at an estimated parameter $\tilde{\theta}$ instead of the true parameter θ, i.e.,

$$\boldsymbol{\ell}_{\tilde{\theta}}(\widehat{T}) = \left\| \widehat{T}(X) - \widetilde{T}(X) \right\|^2_{\tilde{\theta}}, \tag{4.14}$$

where $\widehat{T}$ is the CATE estimate of the model $\mathcal{M}$ being validated, $\tilde{\theta} = \{\tilde{\mu}_0, \tilde{\mu}_1, \tilde{\pi}, \tilde{\eta}\}$ is a *plug-in model* that is obtained from the observational data, and $\widetilde{T}(x) = \tilde{\mu}_1(x) - \tilde{\mu}_0(x)$.

The plug-in model $\tilde{\theta} = \{\tilde{\mu}_0, \tilde{\mu}_1, \tilde{\pi}, \tilde{\eta}\}$ is estimated from the observational data $\{Z_i\}_{i=1}^n$ as follows:

- $\tilde{\mu}_w$, $w \in \{0, 1\}$, is obtained by fitting a supervised model to the sub-dataset $\{(X_i, Y_i) \,|\, W_i = w\}$.
- $\tilde{\pi}$ is obtained using a supervised classification model fit to the sub-dataset $\{(X_i, W_i)\}_{i=1}^n$.

The feature distribution component of $\tilde{\theta}$, $\tilde{\eta}(x)$, can be obtained by estimating the density of X using the feature samples $\{X_i\}_{i=1}^n$. Once we have obtained $\tilde{\theta}$, the plug-in PEHE estimate in (4.14) can be easily evaluated.

The plug-in approach in (4.14) solves the problem of the inaccessibility of the label $Y^{(1)} - Y^{(0)}$ by "synthesizing" such label through the plug-in model $\tilde{\theta}$, and testing a model's inferences against the synthesized CATE function $\widetilde{T}$. This idea is the basis for recent model selection schemes, such as Synth-Validation [110] and Plasmode simulations [129], which propose similar plug-in approaches for validating causal inference models.

■ **Plug-in estimation is not sufficient.** The plug-in estimate in (4.14) exhibits a model-dependent

plug-in bias $\boldsymbol{\ell}_\theta - \boldsymbol{\ell}_{\tilde{\theta}}$ that makes it of little use for model selection. This is because $\boldsymbol{\ell}_{\tilde{\theta}}(\widehat{T})$ measures how well $\widehat{T}$ approximates the synthesized causal effect $\widetilde{T}$ and not the true effect T. Thus, comparing plug-in PEHE estimates of different models can reveal their true comparative performances only if the plug-in bias is small[3], i.e., $\|\widetilde{T} - T\|^2_\theta \approx 0$. However, if $\|\widetilde{T} - T\|^2_\theta$ is large, then plug-in PEHEs tell us nothing about how different models compare on the true distribution $\mathbb{P}_\theta$.

4.5.2 Step 2: Unplugged Validation

How can we get the plug-in bias "unplugged"? We begin by noting that the plug-in PEHE and the true PEHE are two evaluations of the same functional at $\tilde{\theta}$ and θ, respectively. Therefore, we can write $\boldsymbol{\ell}_\theta$ in terms of $\boldsymbol{\ell}_{\tilde{\theta}}$ via a *von Mises* expansion as follows [130]:

$$\boldsymbol{\ell}_\theta(\widehat{T}) = \boldsymbol{\ell}_{\tilde{\theta}}(\widehat{T}) + \sum_{k=1}^{\infty} \int \frac{\dot{\boldsymbol{\ell}}^{(k)}_{\tilde{\theta}}(\boldsymbol{z}; \widehat{T})}{k!}\, d(\mathbb{P}_\theta - \mathbb{P}_{\tilde{\theta}})^{\otimes k}, \tag{4.15}$$

where $\dot{\boldsymbol{\ell}}^{(k)}_\theta(\boldsymbol{z}; \widehat{T}) = \dot{\boldsymbol{\ell}}^{(k)}_\theta(z_1, \ldots, z_k; \widehat{T})$ is a k^{th} order influence function of the PEHE functional at θ (indexed by $\widehat{T}$), with z being a realization of the variable Z in (4.9), and $\mathbb{P}^{\otimes k}_\theta$ is the k-fold product measure of $\mathbb{P}_\theta$.

■ **Influence functions.** The von Mises expansion generalizes Taylor expansion to functionals — it recovers the PEHE at θ based solely on its (higher order) influence functions at $\tilde{\theta}$. In this sense, the influence functions of functionals are analogous to the derivatives of (analytic) functions. Influence functions may not be unique: any set of unbiased k-input functions — i.e., $\mathbb{E}_\theta[\dot{\boldsymbol{\ell}}^{(k)}_\theta(\boldsymbol{Z}; \widehat{T})] = 0$ — that satisfy (4.15) are valid influence functions. We discuss how to calculate the influence functions of $\boldsymbol{\ell}_{\tilde{\theta}}$ in the next section.

An influence function $\dot{\boldsymbol{\ell}}^{(k)}_{\tilde{\theta}}(z_1, \ldots, z_k; \widehat{T})$ can be interpreted as a "measure of the dependence of $\boldsymbol{\ell}_{\tilde{\theta}}$ on the value of k data points in the observational data", i.e., its value reflects the sensitivity of the plug-in PEHE estimate to perturbations in the data. Marginalizing out the data $(z_1, \ldots, z_k)$ with respect to $d(\mathbb{P}_\theta - \mathbb{P}_{\tilde{\theta}})$ results in a functional derivative of $\boldsymbol{\ell}_{\tilde{\theta}}$ in the "direction" $(\mathbb{P}_\theta - \mathbb{P}_{\tilde{\theta}})$ [115].

[3]Paradoxically enough, if $\widetilde{T}$ is a perfect estimate of T (i.e., $\|\widetilde{T} - T\|^2_\theta = 0$), then the model selection task itself becomes obsolete, because the plug-in model would already be better than any model being evaluated. With the plug-in approach, however, we can never know how accurate $\widetilde{T}$ is, since $\boldsymbol{\ell}_{\tilde{\theta}}(\widetilde{T}) = 0$ by definition.

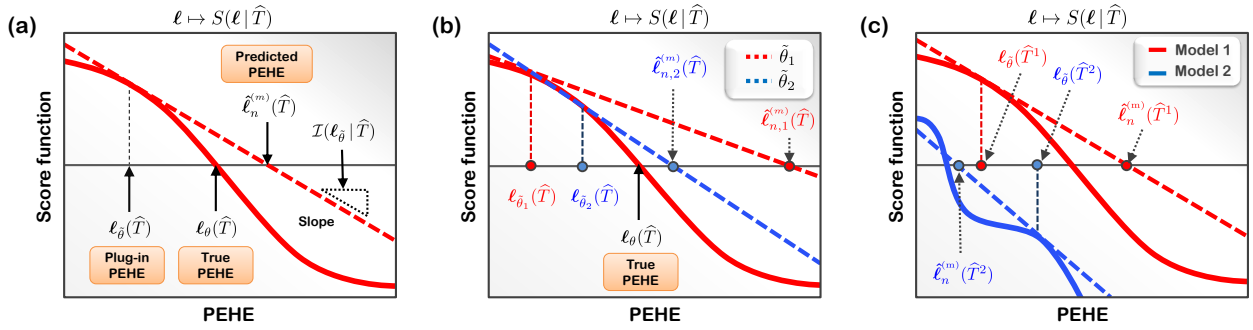


Figure 4.3: Panels (a)-(c) depict exemplary MLE estimating equations for the PEHE as explained in Section 4.5.2. The x-axis corresponds to PEHE values (ℓ), and the y-axis corresponds to the score function $S(\ell \,|\, \widehat{T})$. The true PEHE $\ell_\theta(\widehat{T})$ solves the estimating equation $S(\ell \,|\, \widehat{T}) = 0$. Solid lines (——) correspond to $S(\ell \,|\, \widehat{T})$, whereas dashed lines (- - - -) depict the derivative of the score at the plug-in PEHE. **(a)** The unplugged validation step is analogous to the first iteration of Fisher scoring via Newton-Raphson method. The predicted PEHE is obtained by correcting for the plug-in bias, which is inversely proportional to the Fisher information metric $\mathcal{I}(\ell \,|\, \widehat{T})$. **(b)** Comparison between two plug-in estimates $\tilde{\theta}_1$ and $\tilde{\theta}_2$ for a score function $S(\ell \,|\, \widehat{T})$ (——). The better plug-in estimate conveys more (Fisher) information on the true PEHE, i.e., slope of (- - - -) is steeper than that of (- - - -), and hence it provides a better PEHE prediction. **(c)** Selecting between two models $\widehat{T}^1$ and $\widehat{T}^2$ with score functions $S(\ell \,|\, \widehat{T}^1)$ and $S_\theta(\ell \,|\, \widehat{T}^2)$, respectively. While $\widehat{T}^1$ has a smaller plug-in PEHE than $\widehat{T}^2$, predicted PEHEs flip after correcting for plug-in bias.

The expansion in (4.15) represents the plug-in bias $\ell_\theta - \ell_{\tilde{\theta}}$ in terms of functional derivatives of $\ell_{\tilde{\theta}}$. To see how the bias is captured, consider the first-order von Mises expansion, i.e.,

$$\ell_\theta(\widehat{T}) \approx \ell_{\tilde{\theta}}(\widehat{T}) + \int \dot{\ell}^{(1)}_{\tilde{\theta}}(z; \widehat{T})\, d(\mathbb{P}_\theta - \mathbb{P}_{\tilde{\theta}}). \tag{4.16}$$

Thus, the plug-in bias will be large if the functional derivative of $\ell_{\tilde{\theta}}$ is large, i.e., the PEHE estimate is sensitive to changes in the plug-in model $\tilde{\theta}$. This derivative will be large if many data points have large influence, and for each such data point, the plug-in distribution is not a good representative of the true distribution, i.e., $d(\mathbb{P}_\theta - \mathbb{P}_{\tilde{\theta}})$ is large.

▪ **Dispensing with the counterfactuals.** Note that the expansion in (4.15) quantifies the plug-

in bias in terms of fixed functions of "factual" observations $Z = (X, W, Y^{(W)})$ only. Thus, the true PEHE can be estimated without knowledge of the counterfactual outcome $Y^{(1-W)}$ by calculating the sample average of the first m terms of (4.15) as follows:

$$\hat{\boldsymbol{\ell}}_n^{(m)}(\widehat{T}) = \boldsymbol{\ell}_{\tilde{\theta}}(\widehat{T}) + \sum_{k=1}^{m} \frac{1}{k!} \mathbb{U}_n \left[\dot{\boldsymbol{\ell}}_{\tilde{\theta}}^{(k)}(\boldsymbol{Z}; \widehat{T}) \right], \tag{4.17}$$

where $\mathbb{U}_n$ is the empirical U-statistic, i.e., the sample average of a multi-input function. (4.17) follows directly from (4.15) by capitalizing on the unbiasedness of influence functions.

4.5.3 Relation to Maximum Likelihood Estimation

In Section 4.5.2, we used functional calculus to construct a mathematical approximation of a model's performance that does not depend on counterfactual data. But is this approximation also a *statistically efficient* estimate?

Recall that in (parametric) maximum likelihood estimation (MLE), a parameter estimate θ^* is obtained by solving the estimating equation $S(\theta) = 0$, where $S(\theta)$ is the *score* function — i.e., the derivative of the log-likelihood. For estimating equations that cannot be solved analytically, the classical *Fisher scoring* procedure [131] is used to obtain a numerical solution for MLE.

Our two-step validation procedure[4] is equivalent to finding the MLE of a model's PEHE using the classical Fisher scoring procedure. To illustrate this equivalence, we capture the structural resemblance between the two procedures in Figure 4.3 as well as the tabulated comparison below.

Estimating equation	**Fisher scoring**
(Parametric MLE) $S(\theta^*) = 0$	$\hat{\theta} \approx \theta_0 + \ \mathcal{I}^{-1}(\theta_0)\, S(\theta_0)$
(Our procedure) $S(\boldsymbol{\ell}^* \mid \widehat{T}) = 0$	$\boldsymbol{\ell}_\theta(\widehat{T}) \approx \boldsymbol{\ell}_{\tilde{\theta}}(\widehat{T}) + \ \mathbb{E}_\theta[\dot{\boldsymbol{\ell}}_{\tilde{\theta}}^{(1)}(\boldsymbol{z}; \widehat{T})]$

[4]Here we consider a first-order von Mises approximation.

Fisher scoring implements the Newton-Raphson numerical method to solve $S(\theta) = 0$. It utilizes the Taylor approximation of $S(\theta)$ around an initial θ_0 to formulate an iterative equation $\hat{\theta}_{k+1} = \theta^k + \mathcal{I}^{-1}(\theta_k)\, S(\theta_k)$ — where $\mathcal{I}(\theta)$ is the *Fisher information* — that eventually converges to θ^*. From the tabulated comparison above, we can see that our procedure is analogous to the first Newton-Raphson iteration of Fisher scoring. That is, plug-in estimation is similar to finding an initial estimate θ_0, and the unplugged validation step is similar to updating the initial estimate.

This analogy suggests that our procedure is statistically sound. Similar to cross-validation in supervised learning [51], our procedure is a de facto MLE algorithm that computes the "most likely PEHE of a model given observational data". As shown in Figure 4.3-(a), it does so by searching for the root of the score $S(\boldsymbol{\ell} \,|\, \widehat{T})$ via a one-shot Newton-Raphson procedure.

The juxtaposition of our procedure and Fisher scoring — in the tabulated comparison above — suggests an operational definition for Fisher information $\mathcal{I}(\boldsymbol{\ell} \,|\, \widehat{T})$ as the ratio between the score function and influence function. (This relation also holds in parametric models [132].) The expression of the plug-in bias in terms of the Fisher metric provides an information-geometric interpretation of our validation procedure. That is, the Fisher information content of the plug-in model $\tilde{\theta}$ determines how much the final PEHE estimate will deviate from the initial plug-in estimate (see Figures 4.3-(b) and 4.3-(c) for depictions).

4.5.4 Consistency and Efficiency

In the following Theorem, we establish the conditions under which our validation procedure is statistically efficient.

Theorem 1. *Let μ_0, μ_1, and π be bounded Hölder functions with Hölder exponents α_0, α_1 and β, respectively, and $X \in [0,1]^d$. If (i) $\widehat{T}$ and $\tilde{\theta}$ are fit using a separate sample than that used to compute $\hat{\boldsymbol{\ell}}_n^{(m)}(\widehat{T})$, and (ii) $\widetilde{T}$ is a minimax optimal estimate of T, then we have that:*

$$\hat{\boldsymbol{\ell}}_n^{(m)}(\widehat{T}) - \boldsymbol{\ell}_\theta(\widehat{T}) = O_{\mathbb{P}}\left(\frac{1}{\sqrt{n}} \vee n^{\frac{-(\alpha_0 \wedge \alpha_1)(m+1)}{2(\alpha_0 \wedge \alpha_1)+d}}\right).$$

If $m \geq \lceil \frac{d}{2(\alpha_0 \wedge \alpha_1)} \rceil$*, then the following is satisfied:*

$$\textbf{(Consistency)} \quad \sqrt{n}\,(\hat{\boldsymbol{\ell}}_n^{(m)}(\widehat{T}) - \boldsymbol{\ell}_\theta(\widehat{T})) \xrightarrow{d} \mathcal{N}(0, \sigma^2),$$

$$\textbf{(Efficiency)} \quad \text{Var}[\hat{\boldsymbol{\ell}}_n^{(m)}(\widehat{T})] \leq \text{Var}[\hat{\boldsymbol{\ell}}'(\widehat{T})],$$

for some constant $\sigma > 0$*, and any estimator* $\hat{\boldsymbol{\ell}}'(\widehat{T})$*.* □

This result gives a cut-off value on the minimum number of influence terms m needed for the PEHE estimator $\hat{\boldsymbol{\ell}}_n^{(m)}(\widehat{T})$ to be statistically efficient. This cut-off value depends on the dimensionality and smoothness of the CATE function.

Theorem 1 also says that the plug-in model $\tilde{\theta}$ needs to be good enough for our procedure to work, i.e., $\widetilde{T}$ must be a minimax optimal approximation of T. This is a viable requirement: it is satisfied by models such as Gaussian processes and regression trees [96].

Finally, Theorem 1 also says that our procedure yields the minimum variance estimator of a model's PEHE. This can be understood in the light of the analogy with MLE (Section 4.5.2): since influence functions are proportional to Fisher information, the PEHE estimate in (4.17) satisfies the Cramér-Rao lower bound on estimation variance.

4.6 Calculating Influence Functions

Recall that influence functions operationalize the derivatives of $\boldsymbol{\ell}_\theta(.)$ with respect to distributions induced by θ. But since $\mathbb{P}_\theta$ is nonparametric — i.e., θ is infinite-dimensional — how can we compute such derivatives?

A common approach for computing the influence functions of a functional of a nonparametric family $\mathcal{P}$ is to define a smooth parametric submodel of $\mathcal{P}$, and then differentiate the functional with respect to the submodel's (scalar) parameter [27,116]. A parametric submodel $\mathcal{P}_\varepsilon = \{\mathbb{P}_\varepsilon : \varepsilon \in \mathbb{R}\} \subset \mathcal{P}$ is a subset of models in $\mathcal{P}$ that coincides with $\mathbb{P}_\theta$ at $\varepsilon = 0$. In this Chapter, we choose to work with the following parametric submodel: $d\mathbb{P}_\varepsilon(z) = (1 + \varepsilon h(z))\, d\mathbb{P}_\theta(z)$, for a bounded function $h(z)$.

Given the submodel $\mathbb{P}_\varepsilon$, it can be shown (by manipulating the von Mises series in (4.17)) that

the first order influence function satisfies the following condition:

$$\frac{\partial\, \boldsymbol{\ell}_{\varepsilon}(\widehat{T})}{\partial \varepsilon}\Big|_{\varepsilon=0} = \mathbb{E}_{\theta}[\, \dot{\boldsymbol{\ell}}_{\theta}^{(1)}(z;\widehat{T}) \cdot S_{\varepsilon}(z)|_{\varepsilon=0}\,], \tag{4.18}$$

where $S_{\varepsilon}(z) = \partial \log(d\mathbb{P}_{\varepsilon}(z))/\partial\varepsilon$ is the score function of the parametric submodel, and $\boldsymbol{\ell}_{\varepsilon}$ is the PEHE functional evaluated at $\mathbb{P}_{\varepsilon}$. In the next Theorem, we derive a closed-form expression for $\dot{\boldsymbol{\ell}}_{\theta}^{(1)}(z;\widehat{T})$.

Theorem 2. *The first order influence function of the PEHE $\boldsymbol{\ell}_{\theta}(\widehat{T})$ is unique, and is given by:*

$$\dot{\boldsymbol{\ell}}_{\theta}^{(1)}(Z;\widehat{T}) = (1-B)\, T^2(X) + B\, Y\, (T(X) - \widehat{T}(X)) - \\ A\, (T(X) - \widehat{T}(X))^2 + \widehat{T}^2(X) - \boldsymbol{\ell}_{\theta}(\widehat{T}),$$

where $A = (W - \pi(X))$, and $B = 2W\,(W - \pi(X)) \cdot C^{-1}$
for $C = \pi(X)(1-\pi(X))$. □

This result implies that the influence functions of $\boldsymbol{\ell}_{\theta}(\widehat{T})$ do not depend on $\eta(x)$. Thus, the plug-in model $\tilde{\theta}$ does not need to be generative. This is a great relief since estimating (high-dimensional) feature distributions can be daunting.

4.7 Experiments

As envisioned in the beginning of this Chapter, practitioners can use our validation procedure to select the best causal inference method for a given dataset. Unlike pervasive "expert-driven" modeling practices [133], this *automated* and data-driven approach to model selection enables confident deployment of (black-box) machine learning-based methods, and safeguards against naïve modeling choices.

In this Section, we demonstrate the practical significance of influence function-based validation by assessing its utility in model selection. In particular, we assemble a pool of models — comprising all methods published recently in ICML, NeurIPS and ICLR — and use our validation procedure to predict the best performing model on 77 benchmark datasets from a recent causal inference competition.

4.7.1 Experimental Setup

■ **Influence function-based validation.** We implement a stratified P-fold version of our validation procedure as follows. First, we randomly split the training data into P mutually exclusive subsets, with $\mathcal{Z}_p$ being the set of indexes of data points in the p^{th} subset, and $\mathcal{Z}_{-p}$ its complement. In the p^{th} fold, the model being evaluated is trained on the data in $\mathcal{Z}_{-p}$, and issues a CATE estimate $\widehat{T}_{-p}$. For validation, we execute our two-step procedure as follows:

Step 1: Plug-in estimation

Using the dataset indexed by $\mathcal{Z}_{-p}$, we fit the plug-in model $\tilde{\theta}_{-p} = \{\tilde{\mu}_{-p,0}, \tilde{\mu}_{-p,1}, \tilde{\pi}_{-p}\}$ as explained in Section 4.4.1.1. We use two XGBoost regression models for $\tilde{\mu}_{-p,0}$ and $\tilde{\mu}_{-p,1}$, and then calculate $\widetilde{T}_{-p} = \tilde{\mu}_{-p,1} - \tilde{\mu}_{-p,0}$. For $\tilde{\pi}_{-p}$, we use an XGBoost classifier. Our choice of XGBoost is motivated by its minimax optimality [134], which is required by Theorem 1.

Step 2: Unplugged validation

Given $\tilde{\theta}_{-p}$, we estimate the model's PEHE on the held-out sample $\mathcal{Z}_p$ using the estimator in (4.17) with $m = 1$, i.e.,

$$\hat{\boldsymbol{\ell}}_p^{(1)} = \sum_{i \in \mathcal{Z}_p} \left[(\widehat{T}_{-p}(X_i) - \widetilde{T}_{-p}(X_i))^2 + \dot{\boldsymbol{\ell}}^{(1)}_{\tilde{\theta}_{-p}}(Z_i; \widehat{T}_{-p}) \right],$$

where $\dot{\boldsymbol{\ell}}^{(1)}_{\theta}(.)$ is given by Theorem 2. (Here, the first order U-statistic $\mathbb{U}_1$ in (4.17) reduces to a sample average.) The final PEHE estimate is given by the average PEHE estimates over the P validation folds, i.e., $\hat{\boldsymbol{\ell}}_n^{(1)} = n^{-1} \sum_p \hat{\boldsymbol{\ell}}_p^{(1)}$.

■ **Automated causal inference.** Using our validation procedure, we implement the AutoPrognosis BO procedure, and then pick the model with smallest $\hat{\boldsymbol{\ell}}_n^{(1)}$. Our candidate models include all methods published in ICML, NeurIPS and ICLR conferences from 2016 to 2018. This comprises a pool of 8 models, with modeling approaches ranging from Gaussian processes to generative adversarial networks. In addition, we included two other key models developed in the statistics

Method name	Reference	% Winner
BNN★	Johansson et al. (2016)	3 %
CMGP‡	Alaa et al. (2017)	12 %
TARNet★	Shalit et al. (2017)	8 %
CFR Wass.★	Shalit et al. (2017)	12 %
CFR MMD★	Shalit et al. (2017)	9 %
NSGP★	Alaa et al. (2018)	17 %
GAN-ITE◇	Yoon et al. (2018)	7 %
SITE‡	Yao et al. (2018)	7 %
BART	Hill (2011)	15 %
Causal Forest	Wager et al. (2017)	10 %
Factual	—	53 %
IPTW	—	54 %
Plug-in	—	65 %
AutoPrognosis	—	72 %
Random	—	10 %
Supervised	—	84 %

Table 4.2: Comparison of baselines over all datasets.

community (BART and causal forests). All candidate models are presented in Table 4.2.

■ **Data description.** We conducted extensive experiments on benchmark datasets released by the "Atlantic Causal Inference Competition" [135], a data analysis competition that compared models of treatment effects. The competition involved 77 semi-synthetic datasets: all datasets shared the same data on features X, but each dataset had its own simulated outcomes and assignments (W, Y). Features were extracted from a real-world observational study, whereas outcomes and assignments were simulated via data generating processes that were carefully designed to mimic real data. Each

of the 77 datasets had a unique data generating process encoding varying properties (e.g., levels of treatment effect heterogeneity, dimensionality of the relevant feature space, etc.) Detailed explanation of the data generating processes was published by the organizers of the competition in [97].

The feature data shared by all datasets was extracted from the Collaborative Perinatal Project [136], a study conducted on a cohort of pregnant women to identify causes of infants' developmental disorders. The treatment was a child's birth weight ($W = 1$ if weight $< 2.5\ kg$), and outcome was the child's IQ after a given follow-up period. The study contained 4,802 data points with 55 features (5 are binary, 27 are count data, and 23 are continuous).

▪ **Performance evaluation.** We applied automated causal inference on 10 realizations of the simulated outcomes for each of the 77 datasets, i.e., a total of 770 replications. (Those realizations were generated by the competition organizers and are publicly accessible [135].) For each realization, we divide the data into 80/20 train/test splits, and use training data to predict the PEHE of the 10 candidate models via 5-fold influence function-based validation. Then, we select the model with smallest estimated PEHE, and evaluate its PEHE on the out-of-sample testing data.

▪ **Baselines.** We compare influence function-based validation with 3 heuristics commonly used in the epidemiological and statistical literature [113]:

Baseline	PEHE estimator
Factual validation	$\hat{\ell}_n(\widehat{T}) = \frac{1}{n}\sum_i(\hat{\mu}_{W_i}(X_i) - Y_i^{(W_i)})^2$
IPTW validation	$\hat{\ell}_n(\widehat{T}) = \frac{1}{n}\sum_i \frac{(\hat{\mu}_{W_i}(X_i) - Y_i^{(W_i)})^2}{(1-2W_i)(1-W_i-\tilde{\pi}(X_i))}$
Plug-in validation	$\hat{\ell}_n(\widehat{T}) = \frac{1}{n}\sum_i(\widehat{T}(X_i) - \widetilde{T}(X_i))^2$

Factual validation evaluates the error in the potential outcomes (μ_0, μ_1) using factual samples only. Inverse propensity weighted (IPTW) validation is similar, but weights each sample with its (estimated) "propensity score" $\tilde{\pi}(x)$ to obtain unbiased estimates [112]. Plug-in validation is identical

to Step 1 of our procedure: it obtains a plug-in PEHE estimate [110, 111]. To ensure fair comparisons, we model $\widetilde{T}$ and $\widetilde{\pi}$ in the heuristics above using XGBoost models similar to the ones used in Step 1 of our procedure.

■ **Results.** Table 4.2 summarizes the fraction of datasets for which each baseline comes out as winner across all datasets[5]. As we can see, our influence function-based (IF-based) approach that automatically picks the best model for every dataset outperforms any single model applied repeatedly to all datasets. This is because the 77 datasets encode different data generating processes, and hence no single model is expected to be a good fit for all datasets. The gains achieved by automation are substantial — the PEHE of the automated approach was (on average) 47% smaller than that of the best performing single model.

It comes as no surprise that our procedure outperforms the factual, IPWT and plug-in validation heuristics. This is because, as we have shown in Theorem 1, the IF-based approach is the most efficient estimator of PEHE. We also compare our validation procedure with the "supervised" cross-validation procedure that is allowd to observe the counterfactual data in the training set. As we can see, despite having access to less information, our IF-based approach comes closer to the supervised approach (as compared to the competing validation methods).

[5]The magnitudes of causal effects were not consistent across datasets, hence PEHE values were in different numerical ranges.

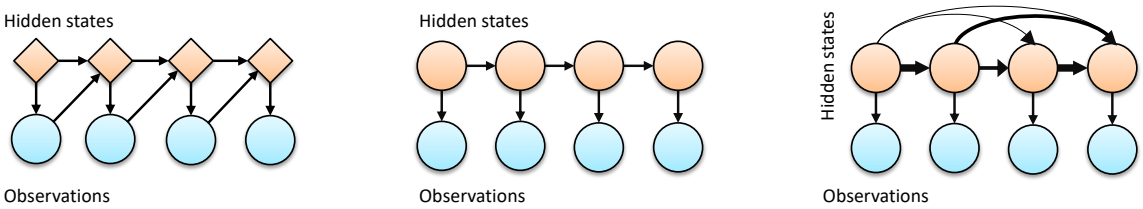
CHAPTER 5

Deep Probabilistic Modeling of Longitudinal Data

Chronic diseases — such as cardiovascular disease, cancer and diabetes — progress slowly throughout a patient's lifetime, causing increasing burden to the patients, their carers, and the healthcare delivery system [137]. The advent of modern electronic health records (EHR) provides an opportunity for building models of disease progression that can *predict* individual-level disease trajectories, and distill *intelligible* and *actionable* representations of disease dynamics [138]. Models that are both highly accurate and capable of extracting knowledge from data are important for informing practice guidelines and identifying the patients' needs and interactions with health services [139–141].

In this Chapter, we develop a deep probabilistic model of disease progression that capitalizes on both the interpretable structured representations of probabilistic models and the predictive strength of deep learning methods. Unlike the previous Chapters, here we address the longitudinal data setup where follow-up data are collected for the same patient over time. Our model uses a state-space representation to segment a patient's disease trajectory into "stages" of progression that manifest through clinical observations. But unlike conventional state-space models, which are predominantly Markovian, our model uses recurrent neural networks (RNN) to capture more complex state dynamics. The proposed model learns hidden disease states from observational data in an unsupervised fashion, and hence it is suitable for EHR data where a patient's record is seldom annotated with "labels" indicating their true health state [142].

Our model uses an *attention* mechanism to capture state dynamics, hence we call it an *attentive state-space* model. The attention mechanism observes the patient's clinical history, and maps it to attention weights that determine how much influence previous disease states have on future state transitions. In that sense, attention weights generated for an individual patient explain the causative



(a) RNN (b) HMM (c) Attentive state space

Figure 5.1: **Sequential data models.** (a) Graphical model for an RNN. ◊ denotes a deterministic representation, (b) Graphical model for an HMM. ◯ denotes probabilistic states, (c) Graphical depiction of an attentive state space model. With a slight abuse of graphical model notation, thickness of arrows reflect attention weights.

and associative relationships between the hidden disease states and the past clinical events for that patient. Because the attention mechanism can be made arbitrarily complex, our model can capture complex dynamics while maintaining its structural interpretability. We implement this dynamic attention mechanism via a sequence-to-sequence RNN architecture [143].

Because our model is non-Markovian, inference of posterior disease states is intractable and cannot be conducted using standard forward-backward routines (e.g., [144–146]). To address this issue, we devise a structured inference network trained to predict posterior state distributions by mimicking the attentive structure of our model. The inference network shares attention weights with the generative model, and uses those weights to create summary statistics needed for posterior inference. We jointly train the inference and model networks using stochastic gradient descent.

To demonstrate the practical significance of the attentive state-space model, we use it to model the progression trajectories of breast cancer using data from the UK Cystic Fibrosis registry. Our experiments show that attentive state-space models can extract clinically meaningful representations of disease progression while maintaining superior predictive accuracy for future outcomes.

5.1 Related Literature

Various predictive models based on RNNs have been recently developed for healthcare settings — e.g., "Doctor AI" [147], "L2D" [148], and "Disease-Atlas" [149]. Unfortunately, RNNs are of a

"black-box" nature since their hidden states do not correspond to clinically meaningful variables (Figure 5.1a). Thus, all the aforementioned methods do not provide an intelligible model of disease progression, but are rather limited to predicting a target outcome.

There have been various attempts to create interpretable RNN-based predictive models using attention. The models in [150–152] use a reverse-time attention mechanism to learn visit-level attention weights that explain the predictions of an RNN. The main difference between the way attention is used in our model and the way it is used in models like "RETAIN" [150] is that our model applies attention to the latent *state space*, whereas RETAIN applies attention to the observable *sample space*. Hence, attention gives different types of explanations in the two models. In our model, attention interprets the hidden disease dynamics, hence it provides an explanation for the mechanisms underlying disease progression. On the contrary, RETAIN uses attention to measure feature importance, hence it can only explain discriminative predictions, but not the underlying generative disease dynamics.

Almost all existing models of disease progression are based on variants of the HMM model [144, 153, 154]. Disease dynamics in such models are very easily interpretable as they can be perfectly summarized through a single matrix of probabilities that describes transition rates among disease states. Markovian dynamics also simplify inference because the model likelihood factorizes in a way that makes efficient forward and backward message passing possible. However, memoryless Markov models assume that a patient's current state d-separates her future trajectory from her clinical history (Figure 5.1b). This renders HMM-based models incapable of properly explaining the heterogeneity in the patients' progression trajectories, which often results from their varying clinical histories or the chronologies (timing and order) of their clinical events [141]. This limitation is crucial in complex chronic diseases that are accompanied with multiple morbidities. Our model addresses this limitation by creating memoryful state transitions that depend on the patient's entire clinical history (Figure 5.1c).

Most existing works on deep probabilistic models have focused on developing structured inference algorithms for deep Markov models and their variants [145, 155–158]. All such models use neural networks to model the transition and emission distributions, but are limited to Markovian dynamics. Other works develop stochastic versions of RNNs for the sake of generative modeling;

examples include variational RNNs [159], SRNN [160], and STORN [161]. These models augment stochastic layers to an RNN in order to enrich its output distribution. However, transition and emission distributions in such models cannot be decoupled, and hence their latent states do not map to clinically meaningful identification of disease states. To the best of our knowledge, ours is the first deep probabilistic model that provides clinically meaningful latent representations, with non-Markovian state dynamics that can be made arbitrarily complex while remaining interpretable.

5.2 Attentive State-Space Models

We start off by describing the structure of EHR data in Section 5.2.1, and then we develop the attentive state-space representation of disease progression in Section 5.2.2.

5.2.1 Structure of the EHR Data

A patient's EHR record, denoted as $\vec{\boldsymbol{x}}_T$, is a collection of sequential follow-up data gathered during repeated hospital visits. We represent a given patient's record as

$$\vec{\boldsymbol{x}}_T = \{x_t\}_{t=1}^{T}, \tag{5.1}$$

where x_t is the follow-up data collected during the t^{th} hospital visit, and T is the total number of visits. The follow-up data $x_t \in \mathcal{X}$ is a multi-dimensional vector that comprises information on biomarkers and clinical events, such as treatments and ICD-10 diagnosis codes [138].

5.2.2 Attentive State-Space Representation

We model the progression trajectory of the target disease via a state-space representation. That is, at each time step t, the patient's health is characterized by a state $z_t \in \mathcal{Z}$ which manifests through the follow-up data x_t. The state space is the (discrete) set of all possible stages of disease progression $\mathcal{Z} = \{1, \ldots, K\}$. In general, progression stages correspond to distinct disease phenotypes. For instance, chronic kidney disease progresses through 5 stages (Stage I to Stage IV), each of which corresponds to a different level of renal dysfunction [162]. We assume that $\{z_t\}_t$ is *hidden*, i.e., the true health state of a patient is not observed, and should be learned in an unsupervised fashion.

We model the joint distribution of states and observations via the following factorization:

$$p_\theta(\vec{\boldsymbol{x}}_T, \vec{\boldsymbol{z}}_T) = \prod_{t=1}^{T} \underbrace{p_\theta(x_t \mid z_t)}_{\textbf{Emission}} \underbrace{p_\theta(z_t \mid \vec{\boldsymbol{x}}_{t-1}, \vec{\boldsymbol{z}}_{t-1})}_{\textbf{Transition}}, \tag{5.2}$$

where $\vec{\boldsymbol{z}}_t = \{z_1, \ldots, z_t\}$, $1 \leq t \leq T$, and θ is the set of parameters of our model.

Attentive state transitions. What makes the model in (5.2) differ from standard state-space models? The main difference is that the transition probability in (5.2) assumes that the patient's health state at time t depends on their entire history $(\vec{\boldsymbol{x}}_{t-1}, \vec{\boldsymbol{z}}_{t-1})$. This is a major departure from the standard Markovian assumption, which posits that $p_\theta(z_t \mid \vec{\boldsymbol{x}}_{t-1}, \vec{\boldsymbol{z}}_{t-1}) = p_\theta(z_t \mid z_{t-1})$, i.e., future states depend only on current state. Most existing disease models are Markovian (e.g., [144, 153]).

To capture non-Markovian dynamics, we model the state transition distribution as follows:

$$p_\theta(z_t \mid \vec{\boldsymbol{x}}_{t-1}, \vec{\boldsymbol{z}}_{t-1}) = p_\theta(z_t \mid \vec{\boldsymbol{\alpha}}_t, \vec{\boldsymbol{z}}_{t-1}), \tag{5.3}$$

where $\vec{\boldsymbol{\alpha}}_t = \{\alpha_1^t, \ldots, \alpha_{t-1}^t\}$, $\alpha_i^t \in [0, 1], \forall i$, $\sum_i \alpha_i^t = 1$, is a set of *attention weights* that act as sufficient statistics of future states. The attention weights admit to a simple interpretation: they determine the influences of past state realizations on future state transitions via the linear dynamic

$$p_\theta(z_t \mid \vec{\boldsymbol{\alpha}}_t, \vec{\boldsymbol{z}}_{t-1}) = \sum_{t'=1}^{t-1} \alpha_{t'}^{t-1}\, \boldsymbol{P}(z_{t'}, z_t),\ \forall t \geq 1, \tag{5.4}$$

where $\boldsymbol{P}$ is a baseline state transition matrix, i.e., $\boldsymbol{P} = p_{ij} \in [0, 1]$, $\sum_j p_{ij} = 1$, and the initial state distribution is $\boldsymbol{\pi} = [p_1, \ldots, p_K]$. The attention weights $\vec{\boldsymbol{\alpha}}_t$ assigned to all previous state realizations at time t are generated using the patient's *context* $\vec{\boldsymbol{x}}_t$ via an attention mechanism $\boldsymbol{A}$ as follows:

$$\vec{\boldsymbol{\alpha}}_t = A_t(\vec{\boldsymbol{x}}_t). \tag{5.5}$$

where $\boldsymbol{A}$ is a deterministic algorithm that generates a sequence of functions $\{A_t\}_t$, $A_t : \mathcal{X}^t \to [0, 1]^t$. We specify our choice of the attention mechanism in the next Section.

Emission distribution. The follow-up data $x_t = (x_t^c, x_t^b)$ comprises both a continuous component x_t^c (e.g., biomarkers and test results) and a binary component x_t^b (e.g., clinical events and

Model	Attention mechanism
HMM [144]	$\alpha^t_{t-1} = 1, \alpha^t_j = 0,\ j \leq t-2.$
Order-m HMM [163]	$\alpha^t_j = \mathbf{1}_{\{m \leq j \leq t-1\}},\ j \leq t-2.$
Variable-order HMM [164]	$\alpha^t_j \in \{0, \bar{n}^{-1}\}, \bar{n} = \sum_i \mathbf{1}_{\{\alpha^t_i \geq \gamma\}}.$

Table 5.1: Reduction of attentive state-space models to standard models.

ICD-10 codes). To capture both components, we model the emission distribution in (5.2) through the following factors $p_\theta(x_t \mid z_t) = p_\theta(x^b_t \mid x^c_t, z_t) \cdot p_\theta(x^c_t \mid z_t)$, where

$$p_\theta(x^c_t \mid z_t) = \mathcal{N}(\mu_{z_t}, \Sigma_{z_t}), \quad p_\theta(x^b_t \mid x^c_t, z_t) = \text{Bernoulli}(\text{Logistic}(x^c_t, \Lambda_{z_t})). \tag{5.6}$$

The model in (5.6) specifies a state-specific distribution for binary (Bernoulli) and continuous (Gaussian) variables, with state-specific emission distribution parameters $(\mu_{z_t}, \Sigma_{z_t}, \Lambda_{z_t})$. This, an attentive state-space model is completely specified through $\theta = (\boldsymbol{\pi}, \boldsymbol{P}, \boldsymbol{A}, \boldsymbol{\mu}, \boldsymbol{\Sigma}, \boldsymbol{\Lambda})$.

Generality of the attentive representation. For particular choices of the attention mechanism in (5.5), our model reduces to various classical models of sequential data as shown in Table 5.1. The generality of the attentive state representation is a powerful feature because it implies that by learning the attention functions $\{A_t\}_t$, we are effectively testing the structural assumptions of various commonly-used time series models in a data-driven fashion.

5.2.3 Sequence-to-sequence Attention Mechanism

To complete the specification of our model, we now specify the attention mechanism $\boldsymbol{A}$ in (5.5). Recall that $\boldsymbol{A}$ is a sequence of deterministic functions that map a patient's context $\vec{\boldsymbol{x}}_t$ to a set of attention weights $\vec{\boldsymbol{\alpha}}_t$ at each time step. Since our model must output an entire sequence of attention weights every time step, we implement $\boldsymbol{A}$ via a sequence-to-sequence (Seq2Seq) model [143].

Our Seq2Seq model uses LSTM encoder-decoder architecture as shown in Figure 5.2. For each time step t, the patient context $\vec{\boldsymbol{x}}_t$ is fed to the LSTM encoder, and the final state of the encoder,

$\boldsymbol{h}_t$, is viewed as a fixed-size representation of the patient's context, and is passed together with the last output O to the decoder side.

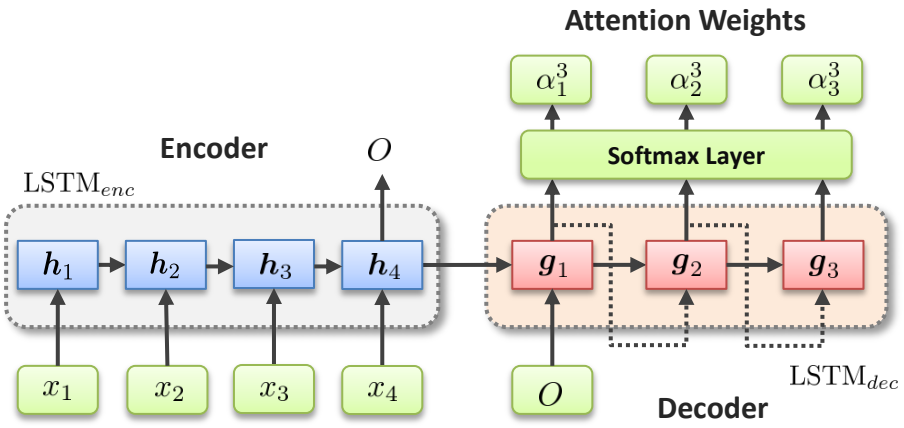


Figure 5.2: Seq2Seq architecture for the attention mechanism $\boldsymbol{A}$.

In the decoding phase, the last state of the encoding LSTM is used as an initial state of the decoding LSTM, and O is used as its first input. Then, the decoding LSTM iteratively uses its output at one time step as its input for the next step. After $t-1$ decoding iterations, we collect the $t-1$ (normalized) attention weights via a Softmax output layer.

The main difference between our architecture and other Seq2Seq models — often used in language translation tasks [143, 165] — is that in our case, we learn an entire sequence of attention weights for each of the T data vectors in $\vec{\boldsymbol{x}}_T$. We achieve this by running $t-1$ decoding iterations to collect $t-1$ outputs for every single encoding step. Moreover, in our setup attention sequence is the target sequence being learned. This should not be confused with other Seq2Seq schemes with attention, where attention is used as an intermediate representation within the decoder [166].

5.2.4 Why Attentive State Space Modeling?

Most existing models of disease progression are based on Hidden Markov models [144, 153, 154, 167]. However, the Markovian dynamic is oversimplified: in reality, a patient transition to a given state depends not only on her current stage, but also on her individual history of past clinical events [137]. In this sense, a Markov models is of a "one-size-fits-all" nature — under a Markov model, all patients at the same stage of progression would have the same expected future trajectory,

irrespective of their potentially different individual clinical histories. Because Markov models explain away individual-level variations in progression trajectories, their interpretable nature should be thought of as a bug and not a feature, i.e., a Markov model is easily interpretable only because it *does not explain much*, it only encodes our own prior assumptions about disease dynamics.

Attentive state space models overcome the shortcomings of Markov models by using attention weights to create non-stationary, variable-order generalization of Markovian transitions, whereby the dynamics of each patient changes over time based on her *individual* clinical context. An attentive state model can learn dynamics that are as complex as those of an RNN, but through the factorization in (5.2), it ensures that its hidden states correspond to meaningful disease states.

5.3 Attentive Variational Inference

Learning the model parameter θ and inferring a patient's health state in real-time requires computing the posterior $p_\theta(\vec{\boldsymbol{z}}_t \,|\, \vec{\boldsymbol{x}}_t)$. However, the non-Markovian nature of our model renders posterior computation intractable. In this Section, we develop a variational learning algorithm that jointly learns the model parameter θ and a structured inference network that approximates the posterior $p_\theta(\vec{\boldsymbol{z}}_t \,|\, \vec{\boldsymbol{x}}_t)$. We show that the attentive representation proposed is useful not only for improving predictions and extracting clinical knowledge, but also can help improve structured inference.

5.3.1 Variational Lower Bound

In variational learning, we maximize an evidence lower bound (ELBO) for the data likelihood, i.e.,

$$\log p_\theta(\vec{\boldsymbol{x}}_T) \geq \mathbb{E}_{q_\phi}\left[\log p_\theta(\vec{\boldsymbol{x}}_T, \vec{\boldsymbol{z}}_T) - \log q_\phi(\vec{\boldsymbol{z}}_T \,|\, \vec{\boldsymbol{x}}_T)\right],$$

where $q_\phi(\vec{\boldsymbol{z}}_T \,|\, \vec{\boldsymbol{x}}_T)$ is a variational distribution that approximates the posterior $p_\theta(\vec{\boldsymbol{z}}_T \,|\, \vec{\boldsymbol{x}}_T)$. We model the variational distribution $q_\phi(\vec{\boldsymbol{z}}_T \,|\, \vec{\boldsymbol{x}}_T)$ using an *inference network* that is trained jointly with the model through the following optimization problem [168, 169]:

$$\theta^*, \phi^* \;= \arg\max_{\theta,\phi} \mathbb{E}_{q_\phi}\left[\log p_\theta(\vec{\boldsymbol{x}}_T, \vec{\boldsymbol{z}}_T) - \log q_\phi(\vec{\boldsymbol{z}}_T \,|\, \vec{\boldsymbol{x}}_T)\right]. \tag{5.7}$$

By estimating θ and ϕ from the EHR data, we recover the generative model $p_\theta(\vec{\boldsymbol{x}}_T, \vec{\boldsymbol{z}}_T)$, through which we can extract clinical knowledge, and the inference network $q_\phi(\vec{\boldsymbol{z}}_T \mid \vec{\boldsymbol{x}}_T)$, through which we can use to infer the health trajectory of the patient at hand.

5.3.2 Attentive Inference Network

We construct the inference network $q_\phi(\vec{\boldsymbol{z}}_T \mid \vec{\boldsymbol{x}}_T)$ so that it mimics the structure of the true posterior [155]. Recall that the posterior factorizes as follows:

$$p_\theta(\vec{\boldsymbol{z}}_T \mid \vec{\boldsymbol{x}}_T) = p_\theta(z_1 \mid \vec{\boldsymbol{x}}_T) \prod_{t=2}^{T} p_\theta(z_t \mid \vec{\boldsymbol{\alpha}}_{t-1}, \vec{\boldsymbol{z}}_{t-1}, \vec{\boldsymbol{x}}_{t:T}).$$

Consequently, we impose a similar factorization on the inference network, i.e.,

$$q_\phi(\vec{\boldsymbol{z}}_T \mid \vec{\boldsymbol{x}}_T) = q_\phi(z_1 \mid \vec{\boldsymbol{x}}_T) \prod_{t=2}^{T} q_\phi(z_t \mid \vec{\boldsymbol{\alpha}}_{t-1}, \vec{\boldsymbol{z}}_{t-1}, \vec{\boldsymbol{x}}_{t:T}). \tag{5.8}$$

To capture the factorization in (5.8), we use the architecture in Figure 5.3 to construct an inference network that mimics the attentive structure of the generative model. In this architecture, a "combiner function" $\boldsymbol{C}(.)$ is fed with all the sufficient statistics of a state z_t, and outputs its posterior distribution. The combiner uses the attention weights created by $\boldsymbol{A}$ to condense summary statistics of z_t.

As dictated by (5.8), the parent nodes of z_t are the attention weights $\vec{\boldsymbol{\alpha}}_t$, the previous states $\vec{\boldsymbol{z}}_{t-1}$ and the future observations $\vec{\boldsymbol{x}}_{t:T}$. The inference network encodes these sufficient statistics as follows. The attention weights $\vec{\boldsymbol{\alpha}}_t$ are shared with the attention network in Figure 5.2. The future observations $\vec{\boldsymbol{x}}_{t:T}$ are summarized at time t via a backward LSTM that reads $\vec{\boldsymbol{x}}_T$ in a reversed order as shown in Figure 5.3. Finally, the previous states $\vec{\boldsymbol{z}}_{t-1}$ are sampled from the combiner functions at previous time steps as described below.

Posterior sampling. In order to sample posterior state trajectories from the inference network, we iterate over the combiner function $\boldsymbol{C}(.)$ for $t \in \{1, \ldots, T\}$ as follows:

$$\begin{aligned} \tilde{\boldsymbol{p}}_t &= C(\boldsymbol{h}_q^t, \vec{\boldsymbol{\alpha}}_t, (\tilde{z}_1, \ldots, \tilde{z}_{t-1}) \mid \boldsymbol{\pi}, \boldsymbol{P}), \\ \tilde{z}_t &\sim \text{Multinomial}(\tilde{\boldsymbol{p}}_t), \end{aligned} \tag{5.9}$$

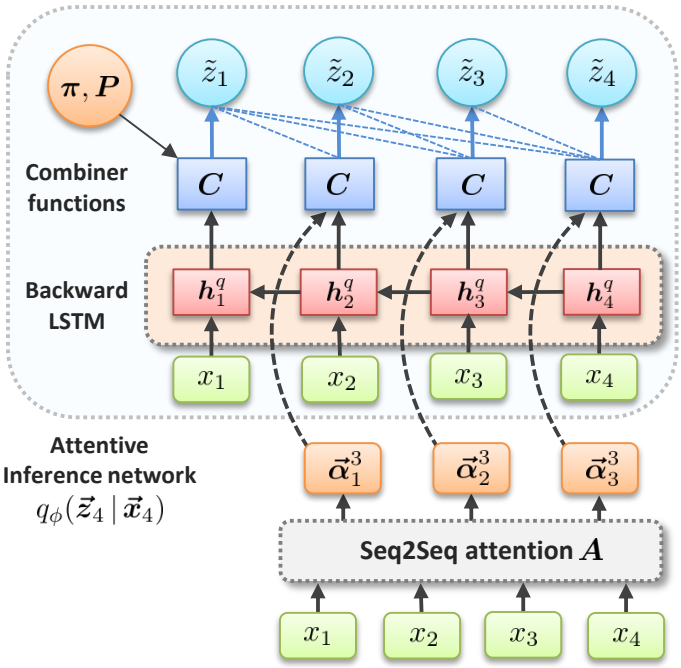


Figure 5.3: Attentive inference network.

where $\tilde{\boldsymbol{p}}_t = (\tilde{p}_1^t, .., .\tilde{p}_K^t), \sum_k \tilde{p}_k^t = 1$, is the posterior state distribution estimated by the inference network at time t, and $\boldsymbol{h}_q^t$ is the t^{th} state of the backward LSTM in Figure 5.3, which summarizes the information in $\vec{\boldsymbol{x}}_{t:T}$. As we can see in (5.9), at each time step t, the combiner function takes as an input *all* the previous states $(\tilde{z}_1, \ldots, \tilde{z}_{t-1})$ sampled by earlier executions of the combiner function. The dashed blue lines in Figure 3 depict the passage of older state samples to later executions of the combiner function.

The combiner function estimates the posterior $\tilde{\boldsymbol{p}}_t$ by emulating the state transition model in (5.5), i.e.,

$$\begin{aligned}
\tilde{p}_{k,forward}^t &= \textstyle\sum_{t'=1}^{t-1} \alpha_{t'}^t \boldsymbol{P}(\tilde{z}_{t'}, k),\ k \in \{1, \ldots, K\}, \\
\tilde{\boldsymbol{h}}_q^t &= [\boldsymbol{h}_q^t, \tilde{p}_{1,forward}^t \ldots, \tilde{p}_{K,forward}^t], \\
\tilde{\boldsymbol{p}}_t &= \text{Softmax}(\boldsymbol{W}_q^\top \tilde{\boldsymbol{h}}_q^t + b_q).
\end{aligned} \tag{5.10}$$

As shown in (5.10), the combiner emulates the generative model to compute an estimate of the "filtering" distribution $\tilde{p}_{k,forward}^t \approx p_\theta(z_t \mid \vec{\boldsymbol{x}}_t)$, i.e., it attends to previously sampled states with proportions determined by the attention weights. Then, to augment information from the future

observations $\vec{\boldsymbol{x}}_{t:T}$, it concatenates the filtering distribution with the backward LSTM state and estimates the posterior through a Softmax output layer.

5.3.3 Learning with Stochastic Gradient Descent

In order to simultaneously learn the parameters of the generative model and inference network, we use stochastic gradient descent to solve (5.10) as follows:

1. Sample $(\tilde{z}_1^{(i)}, \ldots, \tilde{z}_T^{(i)}) \sim q_\phi(\vec{\boldsymbol{z}}_T \,|\, \vec{\boldsymbol{x}}_T)$, $i = 1, \ldots, N$.
2. Estimate ELBO $\hat{\mathcal{L}} = \frac{1}{N} \sum_i \ell_{\theta,\phi}(\vec{\boldsymbol{x}}_T, \tilde{z}_1^{(i)}, \ldots, \tilde{z}_T^{(i)})$.
3. Estimate the gradients $\nabla_\theta \hat{\mathcal{L}}$ and $\nabla_\phi \hat{\mathcal{L}}$.
4. Update ϕ and θ.

In Step 2, the term $\ell_{\theta,\phi}(.)$ denotes the objective function in (5.10). We estimate the gradients in Step 3 via stochastic backpropagation [170]. In Step 4, we use ADAM [171] to update the parameters of the attention mechanism (Figure 5.2) and the inference network (Figure 5.3). The emission parameters are updated straightforwardly by their maximum likelihood estimates.

Rao-Blackwellization via attention. As we have seen, our attentive inference network architecture enables sharing parameters between the generative model and the inference model, which would definitely accelerate learning. Another key advantage of the attentive structure $q_\phi(z_t \,|\, \vec{\boldsymbol{x}}_T)$ is that it acts as a Rao-Blackwellization of the conventional structured inference network which conditions on *all* observation (i.e., $q_\phi(z_t \,|\, \vec{\boldsymbol{x}}_T)$ [146, 155, 156]). Because attention weights (together with $\vec{\boldsymbol{z}}_{t-1}$) and $\vec{\boldsymbol{x}}_{t:T}$)) act as sufficient statistics for state transitions, our inference networks guides the posterior to focus only on the pieces of information that matter. Rao-Blackwellization helps reduce the variance of gradient estimates (Step 3 in the learning algorithm above), and hence accelerate learning [172].

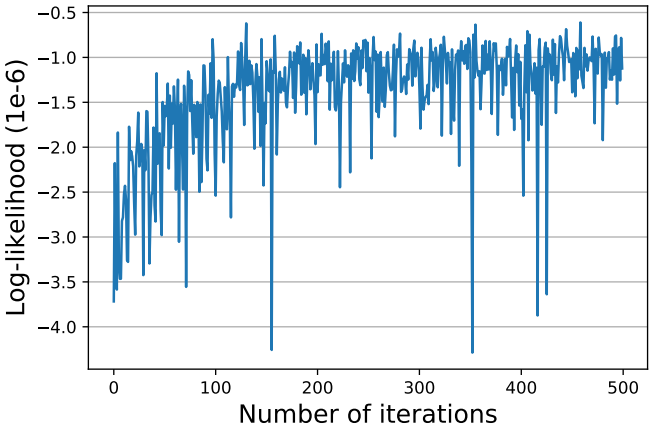


Figure 5.4: LL vs. training epochs.

5.4 Experiments

In this Section, we use our attentive state-space framework to model cystic fibrosis (CF) progression trajectories. CF is a life-shortening disease that causes lung dysfunction, and is the most common genetic disease in Caucasian populations [173]. Experimental details are listed hereunder.

Implementation. We implemented our model using `Tensorflow`. The LSTM cells in both the attention network (Figure 5.2) and the inference network (Figure 5.3) had 2 hidden layers of size 100. The model and inference networks were trained using ADAM with a learning rate of 5×10^{-4}, and a mini-batch size of 100. The same hyperparameters' setting was used for all baseline models involving RNNs. All results reported in this Section where obtained via 5-fold cross-validation.

Data description. We used data from a cohort of patients enrolled in the UK CF registry, a database held by the UK CF trust[1]. The dataset records annual follow-ups for 10,263 patients over the period from 2008 and 2015, with a total of 60,218 hospital visits. Each patient is associated with 90 variables, including information on 36 possible treatments, diagnoses for 31 possible comorbidities and 16 possible infections, in addition to biomarkers and demographic information. The FEV1 biomarker is the main measure of severity in CF patients [174].

[1] https://www.cysticfibrosis.org.uk/the-work-we-do/uk-cf-registry/

Training. In Figure 5.4, we show the model's log-likelihood (LL) versus the number of training epochs. As we can see, the more training iterations we apply, the better the model likelihood gets: it jumped from -4×10^{-6} in the initial iterations to -8×10^{-5} after training was completed. The best value of the log-likelihood is 0, which is achieved when the inference network $q_\phi(z_t \mid \vec{\boldsymbol{x}}_T)$ coincides with the true model $p_\theta(z_t \mid \vec{\boldsymbol{x}}_T)$, and the observed data likelihood given the model is 1. Attentive inference is accurate because it utilizes the minimally sufficient set if past information, which reduces the variance in gradient estimates (Section 5.3.3).

Use cases. We assess our model with respect to the two use cases it was designed for: (1) extracting clinical knowledge on disease progression mechanisms from the data, and (2) predicting a patient's health trajectory over time. We assess each use case in Sections 5.4.1 and 5.4.2.

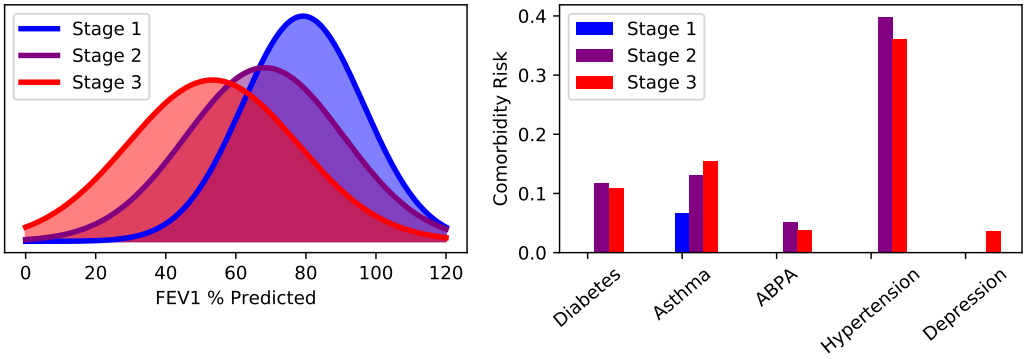


Figure 5.5: Distribution of observations in each progression stage.

5.4.1 Understanding CF Progression Mechanisms

Population-level phenotyping. Our model learned a representation of $K = 3$ CF progression stages (Stages 1, 2 and 3) in an unsupervised fashion, i.e., each stage is a realization of the hidden state z_t. As we show in what follows, each learned progression stage corresponded to a clinically

Model	Diabetes AUC-ROC	ABPA AUC-ROC	Depression AUC-ROC	Pancreatitus AUC-ROC	P. Aeruginosa AUC-ROC
Attentive SS	**0.709 ± 0.02**	**0.787 ± 0.01**	**0.751 ± 0.03**	**0.696 ± 0.04**	**0.680 ± 0.01**
HMM	0.625 ± 0.02	0.686 ± 0.03	0.667 ± 0.08	0.625 ± 0.04	0.610 ± 0.02
RNN	0.634 ± 0.03	0.727 ± 0.10	0.575 ± 0.01	0.590 ± 0.06	0.654 ± 0.01
LSTM	0.675 ± 0.03	0.740 ± 0.07	0.609 ± 0.12	0.578 ± 0.05	0.671 ± 0.01
RETAIN	0.610 ± 0.06	0.718 ± 0.05	0.580 ± 0.09	0.600 ± 0.08	0.676 ± 0.02

Table 5.2: Performance of the different competing models for the 5 prognostic tasks.

distinguishable phenotype of disease activity. The learned baseline transition matrix was

$$\boldsymbol{P} = \begin{bmatrix} 0.85 & 0.10 & 0.05 \\ 0.13 & 0.72 & 0.15 \\ 0.24 & 0.10 & 0.66 \end{bmatrix}.$$

The FEV1 biomarker is currently used by clinicians as a proximal measure of a patient's health in order to guide clinical and therapeutic decisions [175]. In order to check that the learned progression stages correspond to different levels of disease severity, we plot the estimated mean of the emission distribution for the FEV1 biomarker in Stages 1, 2 and 3 in Figure 5.5 (left). As we can see from Figure 5.5 (left), the mean values of the FEV1 biomarker in each stage were 79%, 68% and 53%, respectively. These values matched with the cutoff values on FEV1 used in current guidelines for referring critically-ill patients to a lung transplant [175]. Thus, the learned progression stages can be translated into actionable information for clinical decision-making.

The progression stages learned by our model represented clinically distinguishable phenotypes with respect to multiple clinical variables. To illustrate these phenotypes, in Figure 5.5 (right) we plot the risks of various comorbities (Diabetes, asthma, ABPA, hypertension and depression) for patients in the 3 CF progression stages learned by the model. (Those risks were obtained directly from the learned emission distribution corresponding to the binary component x_t^b of the clinical observation x_t.) As we can see, the incidences of those comorbidities and infections increase significantly in the more severe progression Stages 2 and 3 as compared to Stage 1.

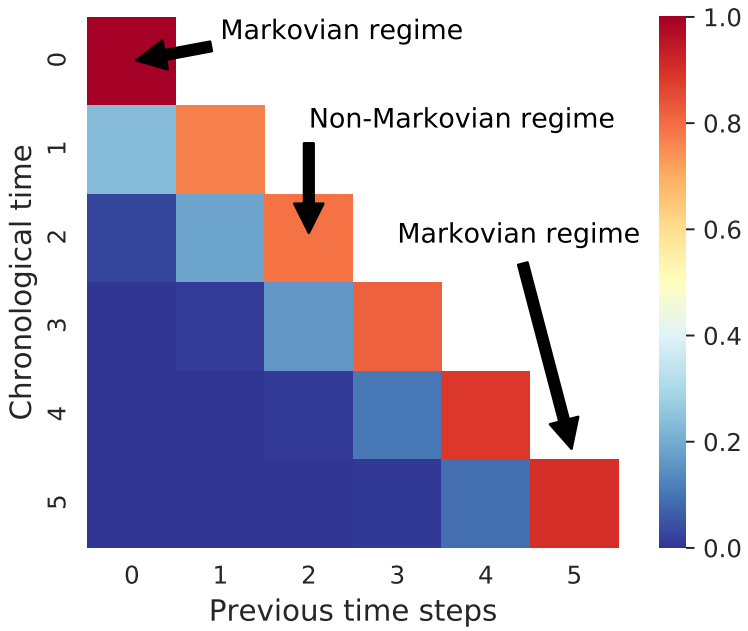


Figure 5.6: Average attention weights over time.

Individualized contextual diagnosis. Population level modeling of disease stages can be already obtained with simple HMM models, but our model captures more complex dynamics that are specific to individuals, and can be made non-Markovian and non-stationary depending on the patient's context. To demonstrate the complex and non-stationary nature of the learned state dynamics, we plot the average attention weights assigned to the patients' previous state realizations in every "chronological" time step of a patient trajectory. The average attention weights per time step is plotted in Figure 5.6.

As we can see, a patient's state trajectory behaves in a quasi-Markovian fashion (only current state takes all the weight) only on its edges. That is, at the first time step and the last time step, the only thing that matters for prediction is the patient's current state. This is because in the first time step, the patient has no history, whereas in the final step, the patient is already in the most severe state and hence her current health deterioration depends overrides all past clinical events. Memory becomes important only in intermediate Stages — this is because patients in Stages 2 and 3 are more likely to have been diagnosed with more comorbidities in the past.

5.4.2 Predicting Prognosis

As we have seen in Section 5.4.1, our model is capable of extracting clinical intelligence from data, but does this compromise its predictive ability? To test the predictive ability of attentive state-space models, we sequentially predict the 1-year risk of 4 comorbidities (ABPA, diabetes, depression and pancreatitus), and 1 lung infections (Pseudomonas Aeruginosa) that are common in the CF population. We use the area under receiver operating characteristic curve (AUC-ROC) for performance evaluation. We report average AUC-ROC with 95% confidence intervals. We compare our model with 4 baselines: a vanilla RNN and an LSTM trained for sequence prediction, a state-of-the-art predictive model for healthcare data known as RETAIN [150, 152], and an HMM.

As we can see in Table 5.2, our model did not incur any performance loss when compared to models trained and tailored for the given prediction task (RNN, LSTM and RETAIN), and was in fact more accurate on all of the 5 tasks. The source of the predictive power in attentive state-space models comes from the usage of LSTM networks to model state dynamics in a low-dimensional space that summarizes the 90 variables associated with each patient. While HMMs can also learn interpretable representations of disease progression, they displayed modest predictive performance because of their oversimplified Markovian dynamics. Because attentive state-space models are capable of combining the interpretational benefits of probabilistic models and the predictive strength of deep learning, we envision them being used for large-scale disease phenotyping and clinical decision-making.

Part II

Application to Clinical Data

CHAPTER 6

Predicting Deterioration of Lung Function in Cystic Fibrosis

6.1 Background

Cystic fibrosis (CF) is an autosomal recessive disease caused by the presence of mutations in both alleles at the cystic fibrosis transmembrane conductance regulator (CFTR) gene, and is the most common genetic disease in Caucasian populations [176, 177]. Impaired CFTR functionality gives rise to different forms of lung dysfunction, all of which eventually lead to progressive respiratory failure [173, 178]. Despite recent therapeutic progress that significantly improved CF prognosis [179], only half of the current CF population are expected to live to over 40 years old [180]. Lung transplantation (LT) is recommended for patients with end-stage respiratory failure as a means to improved life expectancy [181–183]. Unfortunately, there are more LT candidates than available lung donors [181], and in addition, the LT procedure is accompanied by serious risks of subsequent post-transplant complications [184]. An effective LT referral policy should ensure an efficient allocation of the scarce donor lungs by precisely identifying high-risk patients as candidates for transplant, without overwhelming the LT waiting list with low-risk patients for whom a LT might be an unnecessary exposure to the risk of post-transplant complications [185].

Current consensus guidelines, such as those recommended by the International Society for Heart and Lung Transplantation (ISHLT) [186], consider referring a patient for LT evaluation when the forced expiratory volume (FEV_1) drops below 30% of its predicted nominal value. This guideline, which is widely followed in clinical practice [187, 188], is based mainly on the seminal study by Kerem *et. al* [189], which identified FEV_1 as the main predictor of mortality in CF patients using survival data from a cohort of Canadian CF patients (patients eligible 1977-1989). While the FEV_1 biomarker has been repeatedly confirmed to be a strong predictor of mortality in CF

patients [184, 190, 191], recent studies have shown that the survival behavior of CF patients with FEV_1 <30% exhibits substantial heterogeneity [192], and that the improvements in CF prognosis over the past years have changed the epidemiology and demography of CF populations [193, 194], which may have consequently altered the relevant CF risk factors (A striking example of a significant change in the demography of the CF population is the sharp decline in pediatric mortality in recent years [193].) However, none of the existing prognostic models that combine multiple risk factors [195–198] have been able to demonstrate a significant improvement in mortality prediction compared to the FEV_1 criterion in terms of the positive predictive value, which is a proximal measure for the rate of premature LT referral (low-risk patients referred to a transplant) [184].

The goal of this Chapter is to develop a CF prognostic model that can guide clinical decision-making by precisely selecting high-risk patients for LT referral. We use the automated ML algorithm developed in Chapter 4 (AutoPrognosis) to accomplish this goal. In particular, we apply AutoPrognosis to discover an accurate, data-driven prognostic model on the basis of a contemporary cohort from the UK CF registry; a database that includes 99% of the CF population in the UK [199–201].

6.2 Data and Experimental Setup

Experiments were conducted using retrospective longitudinal data from the UK cystic fibrosis Registry; a database sponsored and hosted by the UK cystic fibrosis Trust [199]. The registry comprises a list of annual follow-up variables for individual CF patients that includes demographics, genetic mutations, airway colonization and microbiological infections, comorbidities and complications, transplantation, hospitalization, spirometry and therapeutic management. We used AutoPrognosis to automatically construct a prognostic model for predicting 3-year mortality (a realistic waiting time in a lung transplantation waiting list [184]) based on the follow-up variables at baseline.

All experiments were conducted using data for a baseline cohort comprising patients' follow-up variables collected in 2012: this was the most recent cohort for which 3-year mortality data was available. A total of 115 variables were associated with every patient, all of which were fed into AutoPrognosis in order to encourage an agnostic, data-driven approach for discovering risk

Variable	Alive & no LT n = 3,682 (%)	Death/LT n = 382 (%)	*p*-value	Variable	Alive & no LT n = 3,682 (%)	Death/LT n = 382 (%)	*p*-value
Gender (% male)	2,027 (55.0)	192 (50.2)	0.075	*Pancreatic*			
Age (years)§	27.6 (12)	29.2 (14)	<0.001	Cirrhosis	86 (2.3)	24 (6.3)	<0.001
Height (cm)§	168.0 (14)	166.0 (15)	<0.001	Liver Disease	578 (15.7)	81 (21.2)	0.007
Weight (kg)§	63.1 (17)	54.8 (15)	<0.001	Pancreatitis	57 (1.5)	3 (0.8)	0.368
BMI (kg/m^2)§	22.3 (4)	20.1 (4)	<0.001	Liver Enzymes	521 (14.1)	98 (25.7)	<0.001
CFTR genotype				Gall Bladder	20 (0.5)	3 (0.8)	0.472
Homozygous	1,784 (48.4)	208 (54.4)	<0.001	GI Bleed (variceal)	3 (0.1)	3 (0.8)	0.013
Heterozygous	1,240 (33.7)	92 (24.0)	<0.001	*Gastrointestinal*			
ΔF508	3,189 (86.6)	325 (85.0)	0.388	GERD	747 (20.3)	100 (26.2)	0.008
G551D	224 (6.0)	15 (3.9)	0.108	GI Bleed (no variceal)	4 (0.1)	1 (0.3)	0.390
Class I	169 (4.6)	23 (6.0)	0.205	Intestinal Obstruction	303 (8.2)	33 (8.6)	0.770
Class II	3,207 (87.1)	326 (85.3)	0.338	*Musculoskeletal*			
Class III	3,281 (89.1)	330 (86.3)	0.123	Arthropathy	338 (9.2)	52 (13.6)	0.008
Class IV	184 (5.0)	4 (1.0)	<0.001	Bone Fracture	39 (1.1)	6 (1.6)	0.310
Class V	130 (3.5)	8 (2.0)	0.179	Osteopenia	710 (19.3)	126 (33.0)	<0.001
Class VI	3,189 (86.6)	325 (85.0)	0.388	*Other*			

Table 6.1: Baseline characteristics of patients in the UK CF Registry on December 31st 2012. (§Continuous variables: median (inter-quartile range).)

Variable	Alive & no LT n = 3,682 (%)	Death/LT n = 382 (%)	*p*-value	Variable	Alive & no LT n = 3,682 (%)	Death/LT n = 382 (%)	*p*-value
Spirometry[§]				Cancer	8 (0.2)	5 (1.3)	0.005
FEV_1 (L)	2.34 (1.4)	0.99 (0.6)	<0.001	Diabetes	906 (24.6)	199 (52.1)	<0.001
$FEV_1\%$	67.8 (35)	29.6 (19)	<0.001	CFRD	1,096 (29.8)	223 (58.4)	<0.001
Lung Infections				Hypertension	121 (3.3)	23 (6.0)	0.012
B. Cepacia	176 (4.8)	35 (9.2)	0.001	Atypical Mycobacteria	127 (3.4)	17 (4.5)	0.308
P. Aeruginosa	2,190 (59.5)	295 (77.2)	<0.001	Hearing Loss	82 (2.2)	26 (6.8)	<0.001
MRSA	154 (4.2)	17 (4.5)	0.789	Depression	257 (7.0)	59 (15.4)	<0.001
Aspergillus	478 (13.0)	70 (18.3)	0.006	**Inhaled Antibiotics**	2,194 (59.6)	280 (73.3)	<0.001
NTM	186 (5.1)	20 (5.2)	0.902	**Muco-active Therapies**			
H. Influenza	191 (5.2)	10 (2.6)	0.025	DNase	2,057 (55.9)	297 (77.7)	<0.001
E. Coli	17 (0.5)	2 (0.5)	0.698	Hypertonic Saline	859 (23.3)	109 (28.5)	0.027
K. Pneumoniae	10 (0.3)	3 (0.8)	0.116	**Promixin**	765 (20.8)	71 (18.6)	0.352
Gram-negative	14 (0.4)	4 (1.0)	0.082	**Tobramycin**	110 (3.0)	28 (7.3)	<0.001
ALCA	97 (2.6)	25 (6.5)	<0.001	**iBuprofen**	8 (0.2)	2 (0.5)	0.241
Staph. Aureus	1,175 (31.9)	64 (16.8)	<0.001	**Oral Corticosteroids**	347 (9.4)	122 (31.9)	<0.001
Comorbidities				**Non-IV Hospitalization**	312 (8.5)	62 (16.2)	<0.001
Respiratory				**Non-invasive Ventilation**	161 (4.4)	82 (21.5)	<0.001
ABPA	432 (11.7)	71 (18.6)	<0.001	**Oxygen Therapy**	279 (7.6)	205 (53.7)	<0.001
Nasal Polyps	123 (3.3)	4 (1.0)	0.012	Continuous	13 (0.4)	75 (19.6)	<0.001
Asthma	578 (15.7)	58 (15.2)	0.825	Nocturnal	42 (1.1)	48 (12.6)	<0.001
Sinus Disease	486 (13.2)	41 (10.7)	0.200	Exacerbation	100 (2.7)	46 (12.0)	<0.001
Hemoptysis	48 (1.3)	11 (2.9)	0.022	Pro re nata	37 (1.0)	29 (7.6)	<0.001

Table 6.2: Baseline characteristics of patients in the UK CF Registry on December 31st 2012. ([§]Continuous variables: median (inter-quartile range).)

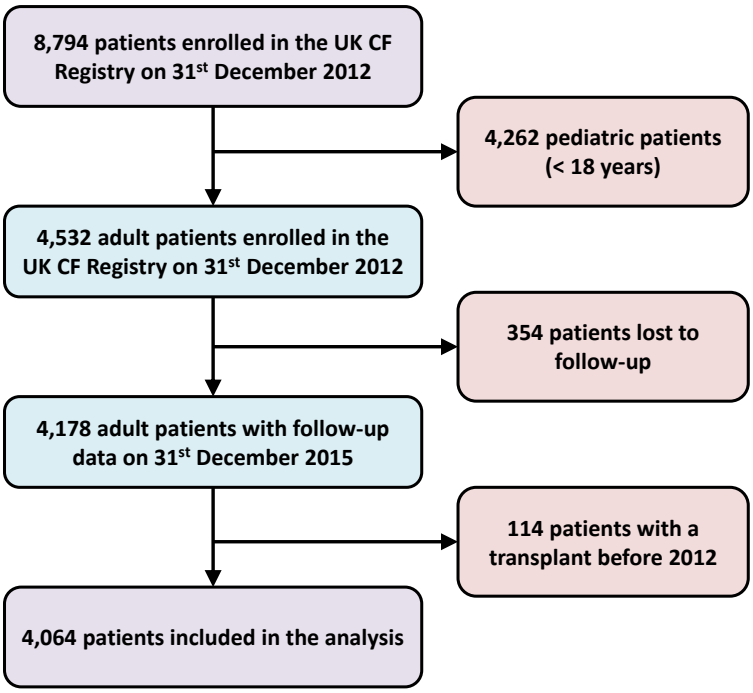


Figure 6.1: Patient selection and data assembly process.

factors. Since transplantation decisions are mostly relevant for adults (93.75% of transplantation operations recorded in the registry were performed in adults), we excluded pediatric patients, and included only patients who were more than 18 years old. (Deaths in children with CF are now very rare in developed countries [193, 202].) Outcomes are defined as death or lung transplantation within 3 years of the baseline data collection date. Patients who were lost to follow-up or have already undergone a transplant before 2012 were excluded. Figure 6.1 depicts a flow chart of the data assembly process involved in our analysis. Of the 4,532 patients who were aged 18 years or older in 2012, a total of 114 patients underwent a lung transplant before their 2012 annual review, and a total of 354 patients were lost to follow-up. Of the remaining 4,064 patients, 382 patients (9.4%) experienced an adverse outcome within a 3-year period.

Of the 382 patients who experienced an adverse outcome, 266 died without receiving a transplant, 104 underwent a successful transplant, and 12 patients received a transplant but died within the 3-year horizon. The characteristics of the patients in the baseline cohort are provided in Ta-

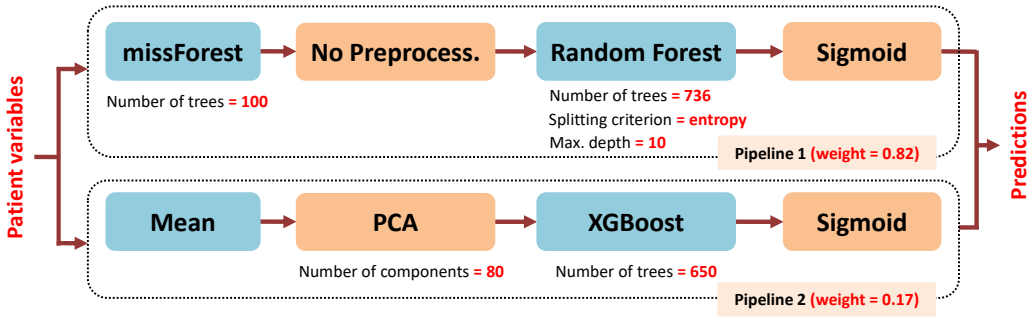


Figure 6.2: Schematic depiction for the in-sample model ft obtained by AutoPrognosis.

bles 6.1 and 6.2. The study population was stratified into two subgroups based on the endpoint outcomes and the characteristics of the two subgroups were compared using Fisher's exact test for discrete (and categorical) variables, and Mann-Whitney U test for continuous variables.

The number of CFTR mutations (in either alleles) whose frequencies in the cohort exceeded 1% was 66, with the most frequent five mutations being ΔF508, G551D, R117H, G542X, and 621+1G→T. Previous studies on CF genetics have classified CFTR mutations into 6 different categories according to the mechanism by which they obstruct the synthesis and traffic of CFTR [177]. We used the CFTR genetic classification in order to cluster the (high-dimensional) genotype information. In particular, we converted the genotype information of every patient into a vector of 9 binary features which encodes the following information: whether the CFTR mutation is homozygous, whether any of the two alleles carries a ΔF508 or a G551D mutation, and the class to which the mutation carried by the patient belongs.

6.3 Training and Validation of AutoPrognosis

All evaluations of diagnostic accuracy were obtained via 10-fold stratified cross-validation in order to assess the generalization performance, where a held-out sample was used to evaluate the performance of the model learned by AutoPrognosis in every fold using a mutually exclusive training sample. In every cross-validation fold, AutoPrognosis conducts up to 200 iterations of the Bayesian optimization procedure presented in Chapter 4, where in every iteration it explores a new pipeline and tunes its hyper-parameters. AutoPrognosis builds an ensemble of all the pipelines that it ex-

plored in which every pipeline is given a weight that is proportional to its empirical performance. The in-sample model fit obtained by AutoPrognosis is depicted in Figure 6.2. The model combines two pipelines: the first uses *missForest* imputation [203] and a *random forest* classifier (with 736 trees) with no feature processing, whereas the second pipeline uses simple mean imputation, a PCA transformation with 80 components followed by an XGBoost classifier with 650 trees. Both pipelines used sigmoid regression for calibration. The in-sample area under receiver operating characteristic curve was 0.9714, and the model was well-calibrated, with a Brier score of 0.0543.

6.4 Results

6.4.1 Systematic Review of Existing Risk Scores

We compared the diagnostic accuracy of AutoPrognosis with state-of-the-art prognostic models that were developed for predicting short-term CF outcomes. In order to identify and select the competing prognostic models, we searched PubMed for studies published in the last 10 years (in all languages) with the terms "(cystic fibrosis) and survival and (prognostic or predictive model)". We filtered the relevant studies by their clinical end-points, focusing only on studies that defined the composite end-point of death and lung transplantation in a time horizon of less than 5 years. We identified 3 contemporary studies that developed and validated prognostic models using multicenter or registry data [197, 202, 204, 205]. In the first study, Buzzetti *et al.* [197] developed a parsimonious multivariate logistic regression model for predicting 5-year outcomes for CF patients using 4 variables, and demonstrated that it outperforms the model developed by Liou *et al.* [196] using retrospective data from 9 Italian CF centers. McCarthy *et al.* [204] developed a predictive model, dubbed "CF-ABLE", for predicting 4-year CF outcomes using 4 variables, and validated their model using data for 370 patients enrolled in the Irish CF registry data. Dimitrov *et al.* [205] proposed a modified version of the CF-ABLE score, dubbed "CF-ABLE-UK", which they (externally) validated through the UK CF registry data, reporting a c-statistic of 0.80 (95% CI: 0.79−0.83). More recently, Nkam *et al.* [202] developed a multivariate logistic regression model for predicting 3-year CF outcomes using 8 risk factors. The model was internally validated through the French CF registry, reporting a c-statistic of 0.91 (95% CI: 0.89−0.92). We compared

the diagnostic accuracy of AutoPrognosis with these 3 models as they considered similar clinical end-points and were validated on contemporary retrospective cohorts.

All of the studies mentioned above explored the usage of only a few risk factors in model development. To the best of our knowledge, ours is the first study to investigate an agnostic, machine learning-based approach for discovering risk factors for CF using a representative cohort that covers the entire CF population in the UK. In order to assess the clinical utility of AutoPrognosis, we also compared its diagnostic accuracy with the simple FEV_1-based prediction rule proposed by Kerem *et al.* [189], where a LT referral criterion that selects CF patients with an $FEV_1\%$ of less than 30% predicted was recommended. This simple prediction rule continues to be the main criterion for LT referral in current clinical practice guidelines [187, 188, 206].

6.4.2 Diagnostic Accuracy Evaluation

The main objective of CF prognostic models is to inform LT referral decisions [181, 184, 188, 207]. Since donor lungs are scarce [181, 182, 185], the clinical utility of a prognostic model should be quantified in terms of the model's ability to (precisely) identify patients who are truly at risk and hence should be allocated in a LT waiting list. Many of the previously developed models have been validated only through goodness-of-fit measures [195, 198], which reveal little information about the models' actual clinical utility. The area under receiver operating characteristic (AUC-ROC) curve has been used to quantify the discriminative power of the models developed by Nkam *et al.* [202], McCarthy *et al.* [204] and Buzzetti *et al.* [197]. AUC-ROC is nevertheless a misleading quantifier for the usefulness of a CF prognostic model as it is insensitive to the prevalence of poor outcomes in the population, and assumes that positive and negative predictions are equally important [208]. Since most patients would not need a LT at the 3-year horizon (the prevalence of poor outcomes is as low as 9.4%), a model's AUC-ROC evaluation can be deceptively high, only reflecting a large number of "easy" and "non-actionable" true negative predictions, without reflecting the actual precision of the LT referral decisions guided by the model. The inappropriateness of AUC-ROC as a sole measure of diagnostic accuracy in the context of LT referral for CF patients was highlighted by Mayer-Hamblett *et al.* [184], where it was shown that models with seemingly high

AUC-ROC can still have modest predictive values (refer to Table 3 therein). A detailed technical analysis of the shortcomings of the AUC-ROC in imbalanced datasets was recently conducted by Saito *et al.* [209].

In order to ensure a comprehensive assessment for the clinical usefulness of AutoPrognosis, we evaluated the *positive predictive values* (PPV) and *negative predictive values* (NPV) for all predictive models under consideration, in addition to the standard AUC-ROC metrics. (PPV is also known as the *precision* metric.) The PPV reflects the fraction of patients who are truly at risk among those identified by the model as high risk patients. A model's PPV characteristic best represents its clinical usefulness as it reflects the precision in the associated LT referral decisions [184]. That is, at a fixed sensitivity, models with higher PPV would lead to fewer patients who are not at risk being enrolled in a transplant waiting list, resulting in a more effective lung allocation scheme with fewer premature referrals.

In Table 6.3, we compare the performance of AutoPrognosis with the competing models in terms of various diagnostic accuracy metrics that capture the models' sensitivity, specificity and predictive values. In particular, we evaluate the models' AUC-ROC, Youden's J statistic, area under precision-recall curve (AUC-PR), average precision and the F_1 score. The AUC-ROC and Youden's J statistic characterize the models' sensitivity and specificity; the J statistic, also known as the *"informedness"*, characterizes the probability of an "informed decision", and is computed by searching for the optimal cutoff point on the ROC curve that maximizes the sum of sensitivity and specificity [210, 211]. As discussed earlier, the clinical usefulness of a model is better represented via its PPV characteristics, and hence we evaluate the models' AUC-PR, average precision and F_1 scores. The three metrics characterize the models' precision (PPV) and recall (sensitivity): the AUC-PR is an estimate for the area under the precision-recall curve using the trapezoidal rule [209, 212], whereas the average precision is a weighted mean of precisions achieved at each threshold on the (non-interpolated) precision-recall curve, where the weights are set to be the increase in recall across the different thresholds [213]. We chose to report both the AUC-PR and the average precision since the trapezoidal rule used to estimate the AUC-PR can provide overly optimistic estimates for the precision-recall performance; both AUC-PR and average precision provide numerically close estimates for well-behaved precision-recall curves [214]. The F_1 score

Prognostic model	**AUC-ROC**	**Youden's J statistic**	**AUC-PR**	**Average Precision**	**F_1 score**
AutoPrognosis	0.89 ± 0.01	0.67 ± 0.02	0.58 ± 0.04	0.59 ± 0.04	0.60 ± 0.03
Nkam *et al.* [202], 2017	0.86 ± 0.01	0.58 ± 0.03	0.50 ± 0.03	0.48 ± 0.03	0.52 ± 0.02
Buzzetti *et al.* [197], 2012	0.83 ± 0.01	0.54 ± 0.03	0.42 ± 0.02	0.44 ± 0.03	0.49 ± 0.02
CF-ABLE-UK [205] (2015)	0.77 ± 0.01	0.48 ± 0.05	0.28 ± 0.04	0.20 ± 0.02	0.34 ± 0.02
$FEV_1\%$ predicted criterion [189]	0.70 ± 0.01	0.41 ± 0.02	0.50 ± 0.02	0.27 ± 0.02	0.47 ± 0.01
SVM	0.84 ± 0.03	0.60 ± 0.05	0.50 ± 0.09	0.51 ± 0.09	0.52 ± 0.07
Gradient Boosting	0.87 ± 0.02	0.63 ± 0.01	0.55 ± 0.03	0.55 ± 0.04	0.56 ± 0.01
Bagging	0.83 ± 0.03	0.58 ± 0.05	0.51 ± 0.04	0.47 ± 0.04	0.52 ± 0.03
TPOT	0.84 ± 0.01	0.56 ± 0.03	0.51 ± 0.02	0.49 ± 0.02	0.51 ± 0.02

Table 6.3: Comparison of various diagnostic accuracy metrics (with 95% CI) for the prognostic models under consideration.

is the harmonic mean of the model's precision and recall; in Table 6.3 we compute each model's F_1 score at the cutoff point determined by its Youden's J statistic.

AutoPrognosis outperformed the competing models with respect to all diagnostic metrics under consideration. We found the model developed by Nkam *et al.* [202] to be the most competitive clinical model with respect to all metrics. All the results in Table 6.3 are statistically significant: 95% confidence intervals and p-values were obtained via 10-fold stratified cross-validation. All prognostic models performed markedly better than the simple criterion based on the FEV_1 biomarker. AutoPrognosis displayed a satisfactory discriminative power, with an AUC-ROC of 0.89 (95% CI: 0.88−0.90) and a J statistic of 0.67 (95% CI: 0.65−0.69), outperforming the most competitive clinical model which achieves an AUC-ROC of 0.86 (95% CI: 0.85−0.87, p-value < 0.001) and a J statistic of 0.58 (95% CI: 0.55−0.61, p-value < 0.001). More importantly, AutoPrognosis displayed an even more significant gain with respect to the precision-recall performance metrics. In particular, it achieved an AUC-PR (Random guessing achieves an AUC-PR that is as low as 0.09.) of 0.58 (95% CI: 0.54−0.62), an average precision of 0.59 (95% CI: 0.55−0.63) and an F_1 score of 0.60 (95% CI: 0.57−0.63), whereas the most competitive clinical model achieved an AUC-PR of 0.50 (95% CI: 0.47−0.53, p-value < 0.001), an average precision of 0.48 (95% CI: 0.45−0.51, p-value < 0.001) and an F_1 score of 0.52 (95% CI: 0.50−0.54, p-value < 0.001).

We observe that the competing clinical models, albeit satisfying high AUC-ROC figures, are providing marginal (or no) gains with respect to the precision-recall metrics (The big gap between the AUC-PR and average precision values for the FEV_1-based criterion reported in Table 6.3 resulted from the fact that this criterion creates a binary statistic with limited number of operating points, while the average precision is computed using the non-interpolated precision-recall curve.) For instance, the CF-ABLE-UK score achieves a better AUC-ROC compared to the FEV_1-based criterion, but performs rather poorly in terms of the precision-recall measures since it additively combines the FEV_1 predictors and many of the variables correlated with it, and hence it double-counts the risk factors for a large number of patients. (As we will show later, the CF-ABLE-UK score also ignores Oxygen therapy intake, which is an important variable for precise identification of low-FEV_1 patients at risk.) The models developed by Nkam *et al.* and Buzzetti *et al.* achieve impressively high gains in AUC-ROC, but only modest gains in the AUC-PR and F_1 scores, im-

	Cutoff	PPV (95% CI) (%)	NPV (95% CI) (%)	Sens (95% CI) (%)	Spec (95% CI) (%)	Accuracy (%)	F_1 score
FEV_1% predicted	< 20	66 (62,70)	92 (91,93)	13 (9,17)	99 (98,100)	92 (91,93)	21 (19,23)
	$\underline{< 30}$	$\underline{48\ (44,52)}$	95 (94,96)	$\underline{46\ (42,50)}$	95 (94,96)	91 (90,92)	47 (45,49)
	< 40	29 (27,31)	96 (95,97)	62 (60,64)	86 (84,88)	84 (83,85)	40 (38,42)
	< 50	21 (19,23)	97 (96,98)	73 (71,75)	75 (73,77)	75 (74,76)	33 (31,35)
Nkam *et al.*, 2017	> 6.5	75 (64,86)	92 (91,93)	13 (11,15)	99 (98,100)	92 (91,93)	22 (19,25)
	$\underline{\geq 4}$	$\underline{56\ (52,60)}$	95 (94,96)	$\underline{46\ (44,48)}$	96 (95,97)	92 (91,93)	50 (49,51)
	> 2.5	42 (37,47)	96 (95,97)	61 (60,62)	91 (90,92)	88 (87,89)	49 (45,53)
	> 2	31 (27,35)	97 (96,98)	73 (72,74)	83 (79,87)	82 (78,86)	43 (39,47)
AutoPrognosis	> 0.50	88 (79,97)	92 (91,93)	13 (12,14)	99 (98,100)	92 (91,93)	23 (22,24)
	$\underline{\geq 0.33}$	$\underline{65\ (61,69)}$	95 (94,96)	$\underline{46\ (45,47)}$	97 (96,98)	93 (92,94)	53 (51,55)
	> 0.15	49 (43,55)	96 (95,97)	62 (61,63)	93 (92,94)	90 (89,91)	54 (50,58)
	> 0.10	36 (32,40)	97 (96,98)	74 (73,75)	87 (86,88)	86 (84,88)	48 (45,51)

Table 6.4: Comparison of the diagnostic accuracy for the prognostic models under consideration at different cutoff points.

plying a limited clinical significance. Contrarily, AutoPrognosis was able to provide not only a high AUC-ROC figure, but also a significant improvement in the precision-recall metrics.

6.4.3 Assessing the Clinical Utility of AutoPrognosis

Practical deployment of a prognostic model in clinical decision-making would entail converting the model's (continuous) outputs into binary decisions on whether a patient might be an appropriate candidate for transplant referral [184]. This can be achieved by setting a cutoff point on the model output (which corresponds to the patient's risk), beyond which the patient is recommended for a transplant. In order to examine the potential impact of the prognostic models under study on clinical decision-making, we evaluated the diagnostic accuracy of AutoPrognosis, the best performing clinical model, and the FEV_1-based criterion, at various cutoff points for transplant referral. The results are summarized in Table 6.4.

In order to ensure a sensible comparison, sensitivity was fixed for all models at four levels (0.13, 0.46, 0.62, and 0.73); these are the four levels of sensitivity achieved by the FEV_1 criterion at the cutoff thresholds 20%, 30%, 40% and 50%, respectively. The results in Table 6.4 show that at each cutoff threshold, the model learned via AutoPrognosis outperforms both the FEV_1 criterion and the best performing competing model in terms of PPV, specificity, accuracy, and F_1 scores. Of particular interest is the cutoff point of FEV_1 <30 % (underlined in Table 6.4), which represents the main transplant referral criterion adopted in current clinical practices. The transplant referral policy achieving the same sensitivity as that achieved by the FEV_1 <30 % criterion places a threshold of 0.33 on the output of AutoPrognosis. At this operating point, AutoPrognosis yields a PPV of 65%, which is significantly higher than that achieved by the FEV_1 criterion (48%), and that achieved by the model developed by Nkam *et al.* [202] (56%). That is, by adopting the model learned by AutoPrognosis for LT referral, we expect that the fraction of patients populating the lung transplant waiting list who are truly at risk would rise from 48% to 65%. In other words, in a waiting list of 100 patients, our model would replace 17 patients who were unnecessarily referred to a transplant with 17 other patients who truly needed one.

The clinical utility of AutoPrognosis is not limited to transplant referral; the predictions prompted

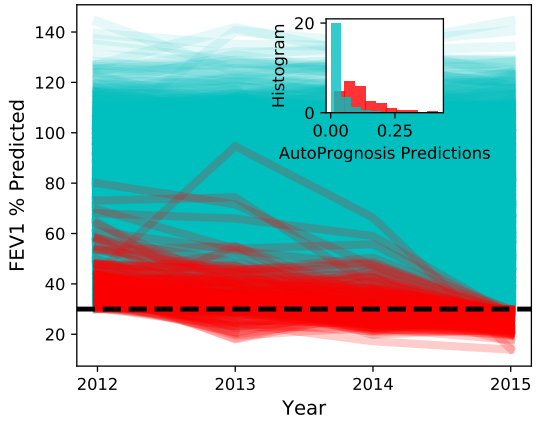


Figure 6.3: FEV_1 trajectories.

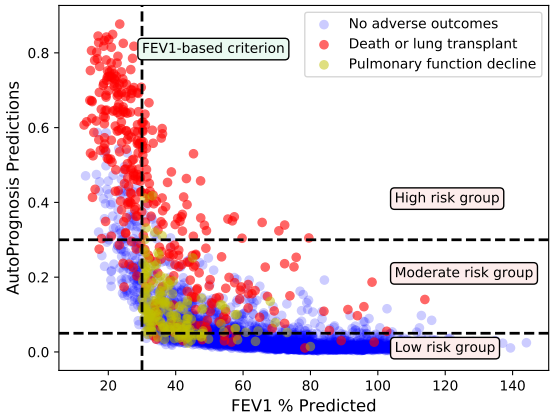


Figure 6.4: Predicted risk groups.

by AutoPrognosis serve as granular risk scores that can quantify the severity of future outcomes and hence can be used for treatment planning, follow-up scheduling, or estimating the time at which a transplant would be needed in the future. For instance, decisions on whether a CF patient carrying a G551D mutation should start taking the (expensive) ivacaftor or lumacaftor drugs can be guided by the predictions of our model [215, 216]. Patients with risk predictions that do not exceed the LT referral threshold are not equally healthy; higher risk scores are still indicative of higher levels of CF severity. The results in Tables 6.3 and 6.4 quantify the models' ability to distinguish patients with and without poor (binary) outcomes (death or LT), but do not show how well the different models are able to predict less severe outcomes. To this end, we sought to classify the predictions of AutoPrognosis into low, moderate and high risk categories, and test the model's ability to predict intermediate poor outcomes. We chose *pulmonary function decline* within a 3-year period as the intermediate poor outcome; we define pulmonary decline as the event when a patient has an $FEV_1\%$ predicted less than 30% in the year 2015 (but did not undergo a lung transplant) when her $FEV_1\%$ predicted was greater than 30% in 2012.

The FEV_1 trajectories for all patients enrolled in the UK CF registry in 2012 are visualized in Figure 6.3; FEV_1 trajectories corresponding to pulmonary decline events are highlighted in red. The trajectories in Figure 6.3 belong only to patients who had $FEV_1 > 30\%$ in 2012 and did not die or undergo a transplant in 2015. A total of 4.4% of those patients experienced pulmonary function decline in 2015. The inset plot in Figure 6.4 shows a histogram for the predictions of AutoPrognosis stratified by the occurrence of a pulmonary decline; we can visually see that AutoPrognosis is

able to discriminate patients with and without the intermediate poor outcome. A two-sample t-test rejects the hypothesis that the average predictions for AutoPrognosis for patients with and without pulmonary decline are equal (p-value < 0.0001). The average predicted risk for patients without pulmonary decline was 0.046, whereas for those with pulmonary decline, the average predicted risk was 0.116. In order to assess the ability of our model to predict the pulmonary decline events, we redefined the poor outcomes as being death, lung transplant or pulmonary decline in a 3-year period. The in-sample average precision and AUC-PR of the predictive model learned by AutoPrognosis were 0.66 (95% CI: 0.63−0.69) and 0.65 (95% CI: 0.63−0.69), respectively, whereas those achieved by the model developed by Nkam *et. al* were 0.51 (95% CI: 0.48−0.54) and 0.48 (95% CI: 0.45−0.51). (95% confidence intervals were obtained via bootstrapping.) This demonstrates that AutoPrognosis is more precise than the existing models in predicting intermediate poor outcomes.

Predicated on the results above, we classified the CF population into three risk groups, with low, moderate and high risk, based on the risk predictions of AutoPrognosis. (In what follows, we converted the outputs of AutoPrognosis, which are real numbers between 0 and 1, into percentages.) The risk groups are defined as follows: the low risk group is associated with risk predictions in the range (0-5%), whereas the moderate risk group is associated with risk predictions in the range (5-30%), and finally, the high risk group is associated with risk predictions that exceed 30%. Figure 6.4 is a scatter plot for the CF patient outcomes in 2015 (red colored dots correspond to deaths or transplants, yellow dots correspond to pulmonary decline events, and blue dots correspond to patients with no adverse outcomes). The outcomes are plotted against the predictions issued by AutoPrognosis (y-axis), and every individual patient's FEV_1 measure in 2012 (x-axis). As we can see, the FEV_1 criterion can only provide a low-precision classification of patients with and without the poor outcome, whereas AutoPrognosis provides a more precise risk stratification for the CF population in which most patients with intermediate poor outcomes (pulmonary decline) reside in the moderate risk group, and patient allocation to the high risk group exhibits lower false alarm rates (refer to Table 6.4). Clinicians can use the risk predictions and risk strata learned by AutoPrognosis as actionable information that guide clinical decisions. For instance, patients in the high risk group would be immediately referred to a transplant, patients in the moderate risk

group would be recommended a drug with potential consideration for a transplant in the future, and patients in the low risk group should routinely pursue their next annual review.

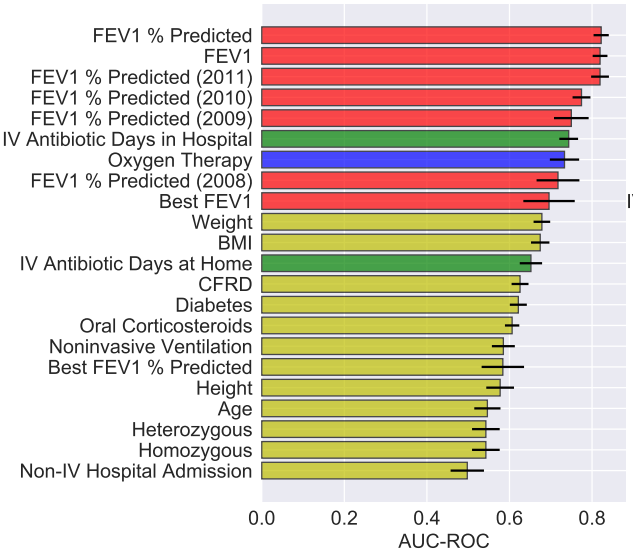


Figure 6.5: AUC-ROC of individual variables.

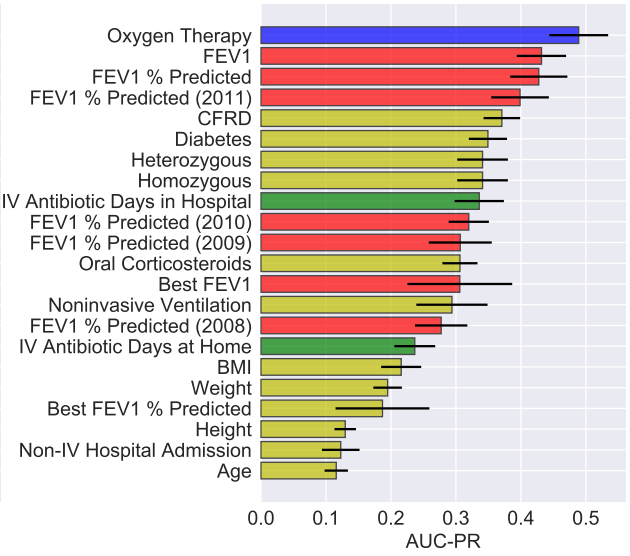


Figure 6.6: AUC-PR of individual variables.

6.4.4 Variable Importance Analysis

We sought to understand how the different patient variables contribute to the predictions issued by AutoPrognosis. Previous studies have identified a wide range of CF risk factors including $FEV_1\%$ predicted [173,185,198,202,204], female gender [173,198], BMI [204,205], Pseudomonas Aeruginosa infection [198], Burkholderia cepacia colonization [202], hospitalization [202], CF-related diabetes [173,217], non-invasive ventilation [202], and ΔF508 homozygous mutation [198]. Since AutoPrognosis was trained in order to provide precise predictions, we focus not only on identifying variables that are most predictive of the outcomes in the sense of AUC-ROC maximization, but also on understanding which variables AutoPrognosis exploited in order to improve the precision (i.e. PPV) of the learned model (refer to Tables 6.3 and 6.4). These variables can then be considered when updating the current consensus guidelines on LT referral and waiting list priority allocation [186].

We evaluated the predictive power of each individual variable by providing AutoPrognosis with one variable at a time, and assessing the diagnostic accuracy of the model that it constructs using

only that variable. We evaluated the AUC-ROC and the AUC-PR metrics (using 10-fold stratified cross-validation) in order to get a full picture of each variable's predictive power with respect to sensitivity, specificity, precision and recall. The most predictive 22 variables with respect to both the AUC-ROC and the AUC-PR metrics are illustrated in Figures 6.5 and 6.6. In both figures, the bars associated with the variables correspond to the AUC-ROC/AUC-PR performance achieved by AutoPrognosis using only this variable. The black error bars correspond to the 95% confidence intervals. Since CF patients may encounter pulmonary disorders manifesting in either increased *airway resistance* or impaired *gas exchange* [218], we labeled the patients' variables in Figures 6.5 and 6.6 based on the aspect of lung function that they reflect. Variables that describe lung function in terms of airway resistance (e.g. FEV_1, $FEV_1\%$ predicted, FEV_1 trajectory, etc) are represented through red bars. Variables that describe lung function in terms of gas exchange (e.g. Oxygenation) are represented through blue bars. Variables that represent pulmonary disorders resulting from bacterial infections are represented through green bars. All other variables had their corresponding bars colored in yellow.

Figure 6.5 shows that the spirometric (FEV_1) biomarkers, including the FEV_1 measurements collected 3 years prior to 2012, display the best AUC-ROC performance. Interestingly, we found that the history of FEV_1 measurements (e.g. the $FEV_1\%$ predicted 1 year before baseline) is as predictive as the FEV_1 measurements at baseline. Variables reflecting pulmonary disorders resulting from bacterial infections (intravenous antibiotic courses in hospital [219]) were the second most predictive in terms of the AUC-ROC performance. The most predictive complications were found to be diabetes and CF-related diabetes. Apart from intravenous antibiotics, the most predictive treatment-related variable was usage of oral corticosteroids. Genetic variables and microbiological infections were found to have a poor predictive power when used solely for predictions, though intravenous antibiotic courses can be thought of as proxies for microbiological infections.

Figure 6.6 shows that the importance ranking for the patients' variables changes significantly when using precision (i.e. AUC-PR) as a measure of the variables' predictive power. Most remarkably, reception of Oxygen therapy turns out to be the variable with the highest AUC-PR. Hence, precise risk assessment and transplant referral decisions need to consider, in addition to the spirometric biomarkers, other biomarkers that reflect disorders in gas exchange, such as the partial

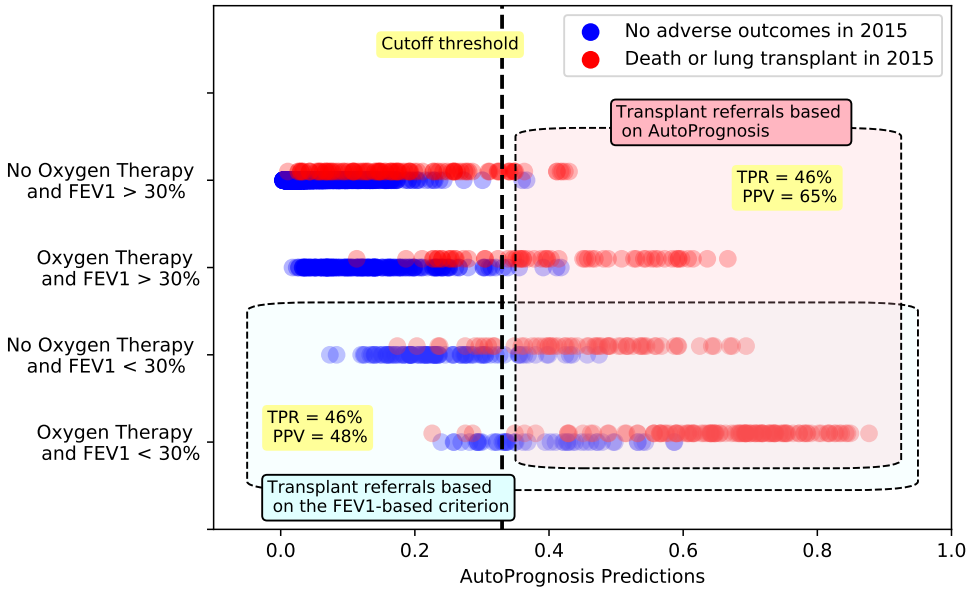


Figure 6.7: Depiction for transplant referral policies based on AutoPrognosis and the FEV_1 criterion for different patient subgroups.

pressure of carbon dioxide in arterial blood ($PaCO_2$) and Oxygen saturation by pulse oximetry (SpO_2) [220]. Prevalence of respiratory failures that are usually treated via Oxygenation, such as hypoxemia and hypercapnia [191, 218, 220, 221], should be considered as decisive criteria for LT referral even when airway obstruction is not severe (i.e. FEV_1 >30%). AutoPrognosis was able to learn a prediction rule that carefully combines spirometric and gas exchange variables in order to come up with a precise lung transplant referral criterion that accurately disentangles patients who are truly at risk from those who do not need a lung in the near future (refer to Tables 6.3 and 6.4). Our results indicate that looking at the right accuracy metric that reflects the true clinical utility (in this case the precision-recall curve) is important not only for tuning and comparing predictive models, but also for discovering risk factors that are relevant for clinical decision-making.

Figures 6.7 illustrates how LT referral policies based on AutoPrognosis handle patient subgroups stratified by spirometric and Oxygenation variables. In this Figure, we look at 4 subgroups: patients with $FEV_1 < 30\%$ who received Oxygen therapy, patients with $FEV_1 < 30\%$ who did not receive Oxygen therapy, patients receiving Oxygen therapy but had $FEV_1 \geq 30\%$, and patients who were neither Oxygenated nor had their FEV_1 drop below the 30% threshold. The subgroup memberships are labeled on the y-axis; every patient is represented as a dot in a scatter plot, with

the x-axis quantifying the risk estimate of AutoPrognosis for every individual patient. Patients with adverse outcomes are represented via red dots, whereas those with no adverse outcomes are depicted as blue dots. As we can see in Figure 6.7, the simple FEV_1 criterion would refer the two subgroups with poor spirometric biomarkers ($\text{FEV}_1 < 30\%$) to a transplant; this leads to a referral list with many blue dots (this is depicted via the dotted box that groups all patients with $\text{FEV}_1 < 30\%$ in Figure 6.7), and consequently a high false positive rate that leads to a PPV of 48%. Contrarily, AutoPrognosis orders the risks of the 4 subgroups by accounting for both Oxygenation and spirometry; this results in a more precise list of referrals at any given cutoff threshold (as can be seen in the dotted box that groups all patients with risk cutoff of 0.33, where the majority of the dots in the box are red). AutoPrognosis achieves precision by assigning a high risk to Oxygenated patients, even if their spirometric biomarkers are not severe. At a fixed TPR of 46%, this leads to some patients with $\text{FEV}_1 < 30\%$ but good clinical outcomes being replaced with Oxygenated patients with $\text{FEV}_1 > 30\%$ who experienced adverse outcomes, which raises the PPV to 65%.

6.5 Discussion and Conclusions

In this Chapter, we applied the AutoPrognosis framework, developed earlier in Chapter 4, to the problem of predicting short-term survival of cystic fibrosis patients using data from the UK CF registry. AutoPrognosis was capable of learning an ensemble of machine learning models (including the well-known random forest and XGBoost algorithms) that outperformed existing risk scores developed in the clinical literature, mainstream practice guidelines, and naïve implementation of vanilla machine learning models. We demonstrated the clinical utility of the prognostic model learned by AutoPrognosis by examining its potential impact on lung transplant referral decisions. Our analysis showed that the model learned by AutoPrognosis achieves significant gains in terms of a wide variety of diagnostic accuracy metrics. Most notably, AutoPrognosis achieves significant gains in terms of the positive predictive values, which implies a remarkable improvement in terms of the precision of lung transplant referral decisions. AutoPrognosis' interpreter module revealed that the model is able to achieve such gains because it recognizes the importance of variables that reflect disorders in pulmonary gas exchange (such as Oxygenation), and learns their interactions

with spirometric biomarkers reflecting airway obstruction (such as FEV_1). This gave rise to a precise survival prediction rule which disentangles patients who are truly at risk from those who do not necessarily need a transplant in the short term.

Although our study provided empirical evidence for the clinical usefulness of applying automated machine learning in prognostication, it has some limitations. First, the prognostic model learned by AutoPrognosis needs to be externally validated in order to ensure that our findings generalize to other CF populations. Second, the net clinical utility of our model needs to be evaluated by considering post-transplant survival data, through which we can identify high-risk patients for whom a transplant is indeed beneficial. Finally, we had no access for data on patients who went through a transplant evaluation process or were enrolled in wait list but did not get a transplant within the 3-year analysis horizon, which rendered direct comparisons with the actually realized clinical policy impossible.

CHAPTER 7

Cardiovascular Disease Risk Prediction

7.1 Background

Globally, cardiovascular disease (CVD) remains the leading cause of morbidity and mortality [222]. Current guidelines for primary prevention of CVD emphasize the need to identify asymptomatic patients who may benefit from preventive action (e.g., initiation of statin therapy [223]) based on their predicted risk [224–227]. Different guidelines recommend different algorithms for risk prediction. For example, the 2010 American College of Cardiology/American Heart Association (ACC/AHA) guideline [228] recommended use of Framingham Risk Score [225], whereas the 2016 European guidelines recommended use of the Systematic Coronary Risk Evaluation (SCORE) algorithm [229]. In the UK, the current National Institute for Health and Care Excellence (NICE) guidelines recommend use of the QRISK2 score to guide the initiation of lipid lowering therapies [230, 231].

Existing CVD risk scores are typically developed using multivariate regression models that combine information on a limited number of well-established risk factors, and generally assume that all such factors are related to the CVD outcomes in a linear fashion, with limited or no interactions between the different factors. Because of their restrictive modeling assumptions and limited number of predictors, existing algorithms generally exhibit modest predictive performance [232], especially for certain sub-populations such as individuals with diabetes [233–236] or rheumatoid arthritis [224]. Data-driven techniques based on machine learning (ML) can improve the performance of risk predictions by exploiting large data repositories to agnostically identify novel risk predictors and more complex interactions between them. However, only a few studies have investigated the potential advantages of using ML approaches for CVD risk prediction, focusing only

on a limited number of ML methods [237, 238] or a limited number of risk predictors [7].

In this Chapter, we explore the potential value of using ML approaches to derive risk prediction models for CVD. We analyzed data on 423,604 participants without CVD at baseline in UK Biobank, a large prospective cohort study in which participants were recruited from 22 centers throughout the UK. Similar to the previous Chapter, we used the AutoPrognosis framework — developed in Chapter 4 — to derive an ML-based CVD risk prediction model and evaluated its predictive performances in the overall population and clinically relevant sub-populations.

7.2 Data and Experimental Setup

7.2.1 Study Design and Participants

Participants were enrolled in the UK Biobank from 22 assessment centers across England, Wales, and Scotland, during the period spanning from 2006 to 2010 [239]. We extracted a cohort of participants who were 40 years of age or older and had no known history of CVD at baseline. That is, patients with previous history of coronary heart disease, other heart disease, stroke, transient ischaemic attack, peripheral arterial disease, or cardiovascular surgery were excluded from the analysis. The total number of participants who met the inclusion criteria was 423,604. The last available date of participant follow-up was Feb 17, 2016. UK Biobank obtained approval from the North West Multi-centre Research Ethics Committee (MREC), and the Community Health Index Advisory Group (CHIAG). All participants provided written informed consent prior to enrollment in the study. The UK Biobank protocol is available online [240].

The UK Biobank dataset keeps track of a large number of variables for each participant, but most of those variables are missing for most patients. In order to include the maximum possible number of (informative) variables in our analysis, we included all variables that are missing for less than 50% of patients with CVD outcomes. This corresponded to a rate of missingness of 85% for the entire population of participants. Our rationale for assessing the missingness rate among patients with CVD is that missingness itself maybe informative (i.e., the chance of a variable being missing may depend on the outcome). By excluding all variables that were missing for

more than 85% of the participants, a total of 473 variables were included in our analysis. We categorized all variables in the UK Biobank into 9 categories: health and medical history, lifestyle and environment, blood assays, physical activity, family history, physical measures, psychosocial factors, dietary and nutritional information, and sociodemographics [241].

7.2.2 Outcome

The primary outcome was the first fatal or non-fatal CVD event. A CVD event was defined as the assignment of any of the ICD-10 diagnosis codes F01 (vascular dementia), I20-I25 (coronary/ischaemic heart diseases), I50 (heart failure events, including acute and chronic systolic heart failures), and I60-I69 (cerebrovascular diseases), or any of the ICD-9 codes 410-414 (ischemic heart disease), 430-434, and 436-438 (cerebrovascular disease). Follow-up data was obtained from the hospital episode statistics (a data warehouse containing records of all patients admitted to NHS hospitals), and the equivalent datasets in Scotland and Wales [242].

7.2.3 Characteristics of the Study Population

A total of 423,604 participants had sufficient information for inclusion in this analysis. Overall, the mean (SD) age of participants at baseline was 56.4 (8.1) years, and 188,577 participants (44.5%) were male. Over a median follow-up of 7 years (5th-95th percentile: 5.7-8.4 years; 3 million person-years at risk), there were 6,703 CVD cases. The mean age of CVD cases was 60.5 years (60.2 years for men and 61.1 years for women). Because the minimum follow-up period for all participants was 5 years, we evaluated the accuracy of the different models in predicting the 5-year risk of CVD. At a 5-year horizon, the total number of CVD cases was 4,801.

7.2.4 Models Tested

Framingham Risk Score

At the time of conducting this study, the UK Biobank had not yet released data on the participants' total cholesterol, HDL cholesterol and LDL cholesterol, which are used as predictors

in various established algorithms, such as Framingham score [225], ACC/AHA [243], QRISK2 [230], and SCORE [226]. The Framingham score, however, provides an incarnation of its underlying model based on nonlaboratory predictors, which replaces lipids with Body Mass Index (BMI) [225]. Since BMI is currently collected for 99.38% of the UK Biobank participants, we compared our model with the BMI version of the Framingham score. We used the published predicting equations (beta-coefficients and survival functions) of the BMI-based Framingham model developed in [225]. (Framingham risk calculator and model coefficients are publicly available in: https://www.framinghamheartstudy.org.)

The Framingham score is based on 7 core risk factors: gender, age, systolic blood pressure, treatment for hypertension, smoking status, history of diabetes, and BMI. All of those variables were complete for the participants in the extracted cohort, with the exception of systolic blood pressure (missing for 6.8% of the participants), and BMI (missing for 0.62% of the participants). We used the *MissForest* non-parametric data imputation algorithm [203] to recover the missing values. Using the MissForest algorithm, we sampled 5 imputed datasets and averaged the model predictions for each participant on the 5 datasets (this is known in the literature as Rubin's rules [203]). The number of imputed datasets was selected via cross-validation.

Cox Proportional Hazards Model

We evaluated the performance of two Cox Proportional Hazards (PH) models derived from the analysis cohort: a model that only uses the traditional 7 risk factors used by the Framingham score, and a model that uses all of the 473 variables in the UK Biobank. To fit the Cox PH models, we imputed the missing data using the MissForest imputation algorithm (with 5 imputations). The Cox PH model that uses the traditional 7 risk factors used by Framingham score can be thought of as a variant of Framingham score calibrated to the UK population (the Framingham score was originally derived for a US population). For the Cox PH model that uses all of the 473 predictors, we applied variable selection using the LASSO method [244]. (Variable selection was applied since fitting the Cox model with all variables resulted in an inferior performance due to the numerical collapse of the Cox model solvers in high dimensions.) To apply variable selection, we fit a

LASSO regression model (a linear model penalized with the L1 norm) to predict the (binary) CVD outcomes. The fitted model gives a sparse solution whereby many of the estimated coefficients are zero. We select all the variables with non-zero coefficients in the fitted LASSO model and feed those variables into a Cox model fitted on the same batch of data. We optimize the LASSO model regularization parameter via cross-validation.

7.3 Model Development using AutoPrognosis

Model Training

To train the AutoPrognosis model, we conduct 200 iterations of the Bayesian optimization procedure in presented in Chapter 4, where in each iteration the algorithm explores a new ML pipeline and tunes its hyper-parameters. Cross-validation was used in every iteration to evaluate the performance of the pipeline under evaluation. The (in-sample) model learned by AutoPrognosis combined 200 weighted ML pipelines, the strongest of which comprised the MissForest data imputation algorithm, no feature processing steps, an *XGBoost* ensemble classifier (with 200 estimators) [245], and sigmoid regression for calibration.

Variable Ranking

In order to identify the relative importance of the 473 variables used to build our model, we use a *post-hoc* approach to rank the contribution of the different variables in the predictions issued by the model. The ranking is obtained by fitting a random forest model with the participants' variables as the inputs, and the predictions of our model as the outputs, and then assigning variable importance scores to the different variables using the standard permutation method in [246]. Using the permutation method, we assess the mean decrease in classification accuracy for every variable after permuting that variable over all trees. The resulting variable importance scores reflect the impact each variable has on the predictions issued by AutoPrognosis. We used the random forest algorithm for post-hoc variable ranking because it is a nonparametric algorithm that can recognize complex patterns of variable interaction while enabling principled evaluation of variable importance [246].

Model	AUC-ROC	Absolute AUC-ROC Change
Framingham score	0.724 ± 0.004	Baseline model
Cox PH Model (7 core variables)	0.734 ± 0.005	+ 1.0%
Cox PH Model (all variables)	0.758 ± 0.005	+ 3.4%
AutoPrognosis (7 core variables)	0.744 ± 0.005	+ 2.0%
AutoPrognosis (369 non-lab. variables)	0.761 ± 0.005	+ 3.7%
AutoPrognosis (104 lab. variables)	0.735 ± 0.008	+ 1.1%
AutoPrognosis (all variables)	0.774 ± 0.005	+ 5.0%

Table 7.1: Performance of different CVD risk prediction models.

Other variable ranking methods based on associative classifiers (such as the one proposed in [247]) entail a computational complexity that is exponential in the number of variables, and hence are not suitable for our study as it involves more than 400 variables.

To disentangle the "modeling gain" achieved by utilizing ML-based techniques from the "information gain" achieved by just using more variables, we created a simpler version of AutoPrognosis that only uses the same 7 core risk factors (age, gender, systolic blood pressure, smoking status, treatment of hypertension, history of diabetes, and BMI) used by the existing prediction algorithms. In addition, we created another version of the AutoPrognosis model that uses only non-laboratory variables in UK Biobank.

Statistical Analysis

In order to avoid over-fitting, we evaluated the prediction accuracy of all models under consideration via 10-fold stratified cross-validation using area under the receiver operating characteristic curve (AUC-ROC). In every cross-validation fold, a training sample (381,244 participants) was used to derive the Cox PH models, standard ML models, and our model (AutoPrognosis), and then a held-out sample (42,360 participants) was used for performance evaluation. We report the mean AUC-ROC and the 95% confidence intervals (Wilson score intervals) for all models. The calibration performance of our model was evaluated via the Brier score.

7.4 Results

Prediction Accuracy

Comparison of Prediction Models

The prediction accuracy of the different models under consideration evaluated at a 5-year horizon is shown in Table 7.1. We used the Framingham score as a baseline model for performance evaluation (AUC-ROC: 0.724, 95% CI: 0.720-0.728). Both the Cox PH model with the 7 conventional risk factors (AUC-ROC: 0.734, 95% CI: 0.729-0.739), and the Cox PH model with all variables (AUC-ROC: 0.758, 95% CI: 0.753-0.763) achieved an improvement in the AUC-ROC compared to the baseline model ($p < 0.001$). The improvement achieved by the Cox PH model that uses the same predictors used by the Framingham score is due in part to the fact that the Cox PH model is directly derived from the analysis cohort, whereas the Framingham score coefficients were derived from a different population.

Most of the variables in the UK Biobank are non-laboratory variables collected through an automated touchscreen questionnaire about lifestyle, clinical history and nutritional habits. We evaluated the accuracy of AutoPrognosis once when it is trained with 369 variables corresponding to the participants' self-reported information (questionnaires) only, and once when it is trained with 104 variables obtained from blood assays, diagnostic tests, and physiological measurements. As we can see in Table 7.1, AutoPrognosis with only questionnaire-related variables still achieves a significant improvement over the baseline Framingham score (AUC-ROC: 0.752, 95% CI: 0.747-0.757, $p < 0.001$), and is superior to the model that only uses laboratory-based variables.

Classification Analysis

In order to better assess the clinical significance of our results, we compared the AutoPrognosis model with the traditional Framingham score in predicting 7.5% CVD risk (threshold for initiating lipid-lowering therapies recommended by the NICE guidelines [231]). At this operating point, the Framingham baseline model predicted 2,989 CVD cases correctly from 4,801 total cases, resulting in a sensitivity of 62.2% and PPV of 1.5%. Our AutoPrognosis model correctly predicted 3,357

out of the 4,801 CVD cases, resulting in a sensitivity of 69.9% and PPV of 2.6%. This corresponds to 368 net increase in the number of CVD patients who would benefit from receiving a preventive treatment in a timely manner when utilizing the predictions of our model.

Variable Importance

Table 7.2 lists the 20 most important variables ranked according to their contribution to the predictions of the AutoPrognosis model (along with their importance scores). Variables related to physical activity (usual walking pace) and information on blood measurements appeared to be more important for the predictions of AutoPrognosis than traditional risk factors included in most existing scoring systems. For women, a remarkable predictor of CVD risk was the measured "ankle spacing width". This may be linked to symptoms of poor circulation, such as swollen legs, which is predictive of future CVD events [248]. We also found that usage of hormone-replacement therapy (HRT) was on the list of top predictors of CVD risk for women. For men, blood measurements such as haematocrit percentage and haemoglobin concentration, and variables such as urinary sodium concentration were among the most important risk factors.

Prediction Accuracy in Individuals with History of Diabetes

Among the 423,604 participants included in our cohort, a total of 17,908 participants (4.22%) had a known history of diabetes (either Type 1 or Type 2) at baseline. In Table 7.3, we show the AUC-ROC performance of AutoPrognosis and the baseline Framingham score when validated separately on the diabetic and non-diabetic populations. As we can see, the baseline Framingham score was less accurate in the diabetic population (AUC-ROC: 0.578, 95% CI: 0.560-0.596) compared to its achieved accuracy for the overall population (AUC-ROC: 0.724, 95% CI: 0.720-0.728, $p < 0.001$). On the contrary, AutoPrognosis maintained high predictive accuracy for the diabetic population (AUC-ROC: 0.713, 95% CI: 0.703-0.723).

We note that the list of important variables in the diabetic subgroup is substantially different from that of the overall population. One major difference is that for diabetic patients, microalbuminuria appeared to be strongly linked to an elevated CVD risk. In the overall population (423,604

participants), the average measure of microalbumin in urine was 27.8 mg/L for participants with no CVD events, and 52.2 mg/L for participants with CVD events. In the diabetic population (17,908 participants), participants with no CVD events had an average microalbumin in urine of 61.0 mg/L, whereas for those with a CVD event, the average microalbumin in urine was 128.76 mg/L. (Information on microalbumin in urine was available for 30% of the patients in the overall population, and 50% of patients in the diabetic population.)

Predictive Ability of Individual Variables in UK Biobank

In order to evaluate the individual predictive ability of the UK Biobank variables, we exhaustively fitted simple versions of our AutoPrognosis model for each of the 473 variables. For each such model, we use one distinct variable as an input and evaluate the resulting AUC-ROC. Because most variables are correlated with age and gender, we use the age variable as a second predictor for all models, and fit separate models for men and women. The AUC-ROC values of the resulting models are depicted in the scatter-plot in Fig 7.1.

As shown in Fig 7.1, variables related to smoking habits or exposure to tobacco smoke displayed the highest predictive ability. Self-reported health rating was predictive for both genders, but more predictive for women. Existence of long-standing illness was strongly predictive of CVD events for women, and less predictive for men. Variables extracted from the electrocardiogram (ECG) records possessed stronger predictive ability for men.

7.5 Discussion and Conclusions

In this large prospective cohort study, we developed a ML model based on the AutoPrognosis framework for predicting CVD events in asymptomatic individuals. The model was built using data for more than 400,000 UK Biobank participants, with over 450 variables for each participant. Our study conveys several key messages. First, AutoPrognosis significantly improved the accuracy of CVD risk prediction compared to well-established scoring systems based on conventional risk factors and currently recommended by primary prevention guidelines (Framingham score).

Second, AutoPrognosis was able to agnostically discover new predictors of CVD risk. Among the discovered predictors were non-laboratory variables that can be collected relatively easily via questionnaires, such as the individuals' self-reported health ratings and usual walking pace. Third, AutoPrognosis uncovered complex interaction effects between different characteristics of an individual, which led to recognition of risk predictors that are specific to certain sub-populations for whom existing guidelines were providing unreliable predictions.

When can ML help in prognostic modeling?

The abundance of a large number of informative variables in the UK Biobank (473 variables) guarantees an "information gain" that can be achieved by any data-driven model, including the standard Cox PH model, compared to the existing prediction algorithms that use only a limited number of conventional risk factors (e.g., Framingham score). The results in Table 7.1 show that, in addition to the information gain, AutoPrognosis also attained a "modeling gain" that allowed it to outperform the standard Cox PH model that uses all of the 473 variables. In general, the modeling gain achieved by AutoPrognosis would result from its ability to select among different models with various levels of complexity and numerical robustness in a completely data-driven fashion, without committing to any presupposition about the superiority of any given model. In our experiments, the Cox PH supplied with all of the 473 variables (without variable selection) provided a noticeably poor performance (i.e., an average AUC-ROC of 0.6). This is because the numerical solvers of the Cox PH model collapse when the data dimensionality is very large — this is why a variable selection pre-processing step was essential for fitting the Cox PH model. This implies that, even if the true underlying data model is perfectly linear, fitting standard linear models such as Cox PH or linear regression may not be sufficient for harnessing the information gain, since such models are not numerical robust in high-dimensional settings. AutoPrognosis solves this problem by selecting more robust models that better fit the high-dimensional data — in our experiments, these where tree-based models such as XGBoost and random forests. This observation shows that information gain and modeling gain are inherently entangled: to harness the information gain, we need to consider a more complex modeling space.

While the information gain appeared to be more significant than the modeling gain in our experiments, we note that even when provided with the same 7 core risk factors used by the Framingham score, AutoPrognosis was still able to offer a statistically significant AUC-ROC gain compared to the Framingham score and a Cox PH model that uses the same 7 variables. This shows that the modeling gain is not necessarily limited to settings where many predictors are available and numerical robustness, but is rather achievable whenever a small number of predictors display complex interactions.

Risk prediction with non-laboratory variables

Individuals in developed countries tend to seek out health information through online resources and web-based risk calculators [249]. In developing countries, where 80% of all world-wide CVD deaths occur [250], there are limited resources for risk assessment strategies that require laboratory testing [250, 251]. The results in Table 7.1 show that AutoPrognosis could potentially provide reliable risk predictions by using information from non-laboratory variables about the participants' lifestyle and medical history. The most predictive non-laboratory variables included in our model were ages, gender, smoking status, usual walking pace, self-reported overall health rating, previous diagnoses of high blood pressure, income, Townsend index and parents' ages at death. Inclusion of such variables in web-based risk calculators can help provide reasonably accurate risk predictions when obtaining laboratory variables is not viable.

One remarkable finding in Table 7.1 (and Fig 7.1) is that apart from the well-established age and gender risk factors, two other non-laboratory variables were found to be very predictive of the CVD outcomes; those are the "self-reported health rating", and the "usual walking pace". (Both variables were also found to be predictive of the overall mortality risk in a recent study on the UK Biobank [241].) Neither of the two variables is included in any of the existing risk prediction tools. Walking pace was equally predictive for men and women, but the self-reported health rating was more predictive for women and less for men. This may be explained by either gender-specific reporting bias or true clinical differences. Therefore, prediction tools that would include subjective non-laboratory variables, such as the self-reported health rating, should be carefully designed in

such a way that self-reporting bias is reduced.

Risk predictors specific to diabetic patients

Unlike the Framingham score, AutoPrognosis was able to maintain high predictive accuracy for participants diagnosed with diabetes at baseline. This suggests that the AutoPrognosis model has learned diabetes-specific risk factors that were not previously captured by the existing prediction algorithms. By investigating the risk factor ranking within the diabetic subgroup (Table 7.3), we found that urinary microalbumin (measured in mg/L) is a very strong marker for increased CVD risk among individuals with diabetes. The dismissal of urinary microalbumin in existing risk scoring systems may explain their poor prognostic performance when validated in cohorts of diabetic patients [233,234]. Our results indicate that predictions based on AutoPrognosis can provide better guidance for CVD preventive care in diabetic patients.

It is worth mentioning that the microalbumin in urine measures were available for only 125,406 participants in the overall cohort (29.6%). In a standard prognostic study, such a variable may get omitted from the analysis because of its high missingness rate. AutoPrognosis automatically recognized that this variable is relevant for diabetic patients, and hence did not omit it in its feature processing stage.

Limitations

The main limitation of our study is the absence of the cholesterol biomarkers (total cholesterol, HDL cholesterol and LDL cholesterol) from the latest release of the UK Biobank data repository, which hindered direct comparisons with the QRISK2 scores currently recommended by the NICE guidelines. Furthermore, other blood-based biomarkers have been reported to be associated with CVD risk, but were also not yet released in the UK Biobank data repository, such as triglycerides [252], measures of glycemia [253], markers of inflammation [254], and and natriuretic peptides [255]. Inclusion of such predictors could improve the predictive accuracy of all models tested in this study, and could also alter the risk predictors' ranking in Table 7.2, but is unlikely to change our conclusions on the usefulness of ML modeling in CVD risk prediction.

Variable (Men)	Score	Variable (Women)	Score
Age*	0.346	**Age***	0.370
Smoking*	0.101	**Smoking***	0.099
Usual walking pace	0.052	Usual walking pace	0.057
Systolic blood pressure*	0.040	Ankle spacing width	0.035
Microalbumin in urine	0.032	Self-reported health rating	0.030
High blood pressure	0.030	**Systolic blood pressure***	0.026
Red blood cell distribution width	0.025	High blood pressure	0.024
Self-reported health rating	0.019	Red blood cell distribution width	0.023
Haematocrit percentage	0.014	Microalbumin in urine	0.017
Father age at death	0.014	Father age at death	0.017
BMI*	0.013	White blood cell count	0.011
Diastolic blood pressure	0.012	Number of Treatments	0.011
White blood cell count	0.012	Mean reticulocyte volume	0.008
Impedance of arm (left)	0.009	Leg predicted mass (right)	0.006
Haemoglobin concentration	0.007	Neutrophill count	0.006
Neutrophill count	0.005	Basal metabolic rate	0.005
Number of Treatments	0.004	Hormone-replac. therapy usage	0.005
Mean reticulocyte volume	0.004	Blood clot in the leg	0.004
Urinary sodium concentration	0.004	Forced expiratory volume	0.004
Monocyte count	0.004	Duration of fitness test	0.004

Table 7.2: Variable ranking by their contribution to the predictions of AutoPrognosis.

Model	AUC-ROC (No diabetes)	AUC-ROC (Diabetes)
Framingham score	0.724 ± 0.004	0.578 ± 0.018
AutoPrognosis	0.774 ± 0.005	0.713 ± 0.010

Table 7.3: Performance of AutoPrognosis in the diabetic patient subgroup.

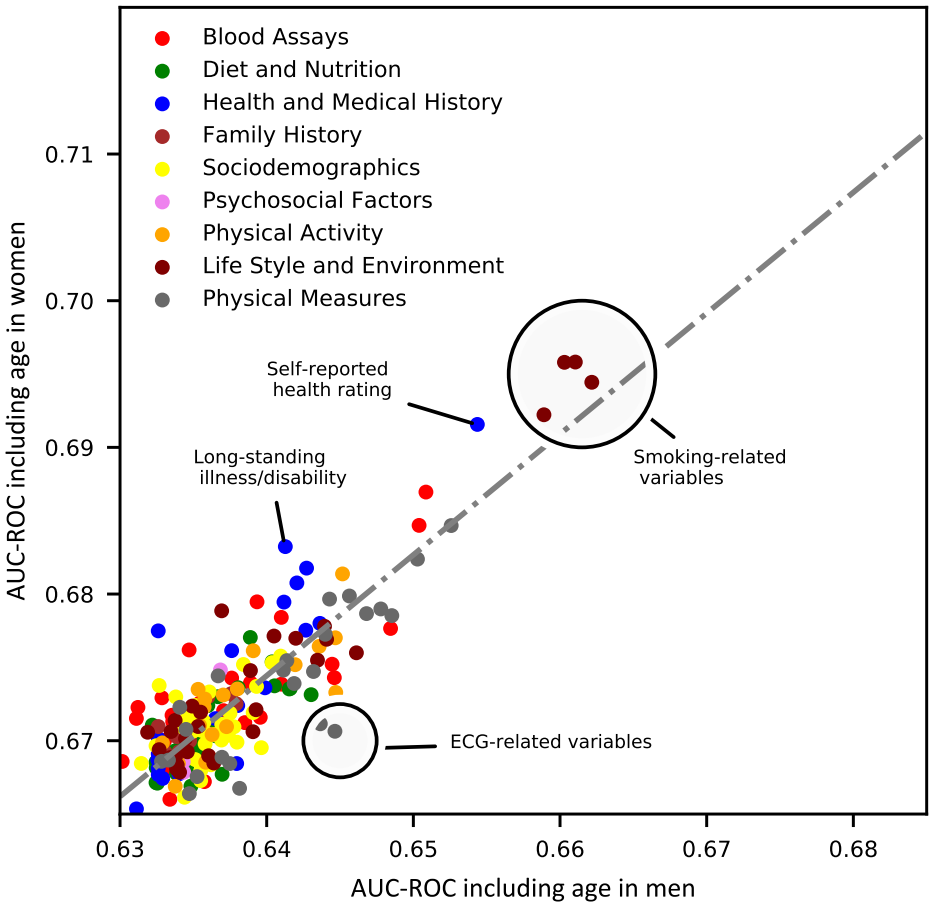


Figure 7.1: **Predictive ability of the UK Biobank variables for men and women.** Each point represents a variable in the UK Biobank ordered by the ability to predict CVD events for men and women. Predictions based solely on age achieved an AUC-ROC of 0.632 ± 0.003 for men and 0.665 ± 0.002 for women. We report the AUC-ROC from models trained with individual variables in addition to age, and only display variables that achieved a statistically significant improvement in AUC-ROC compared to predictions based on age only. Each color represents a different variable category. Variables deviating from the (dotted gray) regression line have an AUC-ROC that differs between men and women more than expected in view of the overall association between the two genders, suggesting a stronger relative importance in one gender group.

CHAPTER 8

Breast Cancer Prognostication and Treatment Benefit Prediction

8.1 Background

Breast cancer is the most common cancer among women globally, with incidence rates varying from 19.3 per 100,000 women in Eastern Africa to 89.7 per 100,000 women in Western Europe. [256, 257] While prognosis of early-stage breast cancer has improved substantially since the introduction of adjuvant endocrine and chemotherapies, [258] these treatments need to be used judiciously, with careful balancing of risks and benefits, particularly in patient subgroups where their utility is as yet unclear. Accurate prediction of survival rates following breast-conversing surgery can guide individualized therapeutic decisions by applying the relative risk reduction of a given adjuvant therapy to the predicted risk of an individual patient, thereby estimating the net survival benefit for that patient. [259, 260]

Over the years, various breast cancer prognostication models have been developed to enable tailored post-surgical therapeutic decisions by predicting the survival profiles of individual patients on the basis of their clinicopathological features. Of these, Adjuvant! Online [261] and PREDICT [262, 263] have been the most commonly used worldwide. [264] Adjuvant! Online has been previously used to predict the expected benefits of specific adjuvant therapies in early breast cancer but is currently not in use. PREDICT v2.1 [263] (https://predict.nhs.uk) is currently recommended by the UK NICE guidelines, and was endorsed by the American Joint Committee on Cancer (AJCC). [10] In the period spanning from 2011 to 2019, PREDICT was accessed through more than 1 million sessions from more than 100 cities all over the world (https://breast.predict.nhs.uk/statistics.html).

However, despite its widespread use, PREDICT v2.1 has been shown to under-perform in specific subgroups of patients, including older patients, patients with tumours over 50mm, small ER-positive tumours, or larger ER negative tumours. [265] Over or under-estimation of the survival rates within specific patient subgroups could lead to under or over-treatment, thereby, negatively impacting patient outcomes. [266–269] We hypothesize that the limitations of existing tools arise from: (1) the lack of flexibility in the underlying Cox regression method predominantly used to develop prognostic models, [261, 263] and (2) the derivation of models using outdated and relatively modest-sized cohorts where certain subgroups of patients may not be sufficiently represented. Machine learning (ML) technologies that can readily infer complex patterns from data, accoutered with big data resources provide the opportunity to address the aforementioned limitations. [270, 271]

In this Chapter, we apply the AutoPrognosis framework (presented in Chapter 4) to develop *Adjutorium*; a breast cancer prognostication tool that predicts patient outcomes and treatment benefits in order to guide personalized therapeutic decisions. We develop and validate Adjutorium using data for nearly 1 million women in two large-scale cohorts that are representative of the UK and US populations.

8.2 Data and Experimental Setup

8.2.1 Study Participants

Patient data for the study were obtained from two cohorts: the UK National Cancer Registration and Analysis Service (NCRAS, n=620,249), and the US Surveillance, Epidemiology and End Results program [272] (SEER, n=588,735). NCRAS is the population-based cancer registry for England, hosted and maintained by Public Health England. The SEER program at the National Cancer Institute collects data on cancer diagnoses, treatment and survival for approximately 30% of the US population. The two databases combined hold data for over 1.2 million cases diagnosed between 2000 and 2016. To develop our model, we considered standard prognostic factors included in models in current clinical use, [263, 273, 274] including age at diagnosis, mode of detection

(screening/symptomatic), estrogen receptor (ER) status, human epidermal growth factor receptor 2 (HER2) status, number of lymph nodes involved, tumour size and histological tumour grade.

We included patients who were diagnosed after January 1^{st} 2005, and were aged 30 to 90 years at diagnosis. Specific age data were not available on patients less than 30 years of age in NCRAS; hence, these were excluded. Furthermore, we excluded patients with missing data on more than 4 variables ($< 10\%$ of all participants), and a small number of patients who were outliers for tumour size (> 90 mm tumour), and number of positive lymph nodes (> 50). A total of 395,862 and 571,635 patients met the inclusion criteria in NCRAS and SEER, respectively (Figure 8.1).

8.2.2 Outcomes

The primary outcome of interest for prognostication was survival from all-cause mortality with and without adjuvant therapies at 3, 5 and 10 years after surgery for breast cancer. All-cause mortality was further subdivided into breast cancer related mortality, which was assessed as a secondary outcome, and mortality due to other causes. Breast cancer related mortality was defined as ICD-10 code C.50 listed on the death certificate as a cause of death, whereas mortality due to other causes was defined as any other ICD-10 code.

8.2.3 Missing Data Imputation

A limitation of existing models has been their dependence on complete case analysis, and lack of flexibility to incorporate missing variables. Our analysis suggested that missingness was highly informative [275]; (log-rank test for difference in 5-year survival between patients with complete data and one or more missing variable, $p < 0.001$). In this context, including only patients with complete data is likely to affect model generalisability. Therefore, in the interest of generalisability, we opted to impute any missing data using data available on other variables. For all study cohorts, we imputed missing data using using the model-based multiple chained equations [276] (MICE) method. We tested the robustness of the model to missing data in sensitivity analyses, as discussed subsequently.

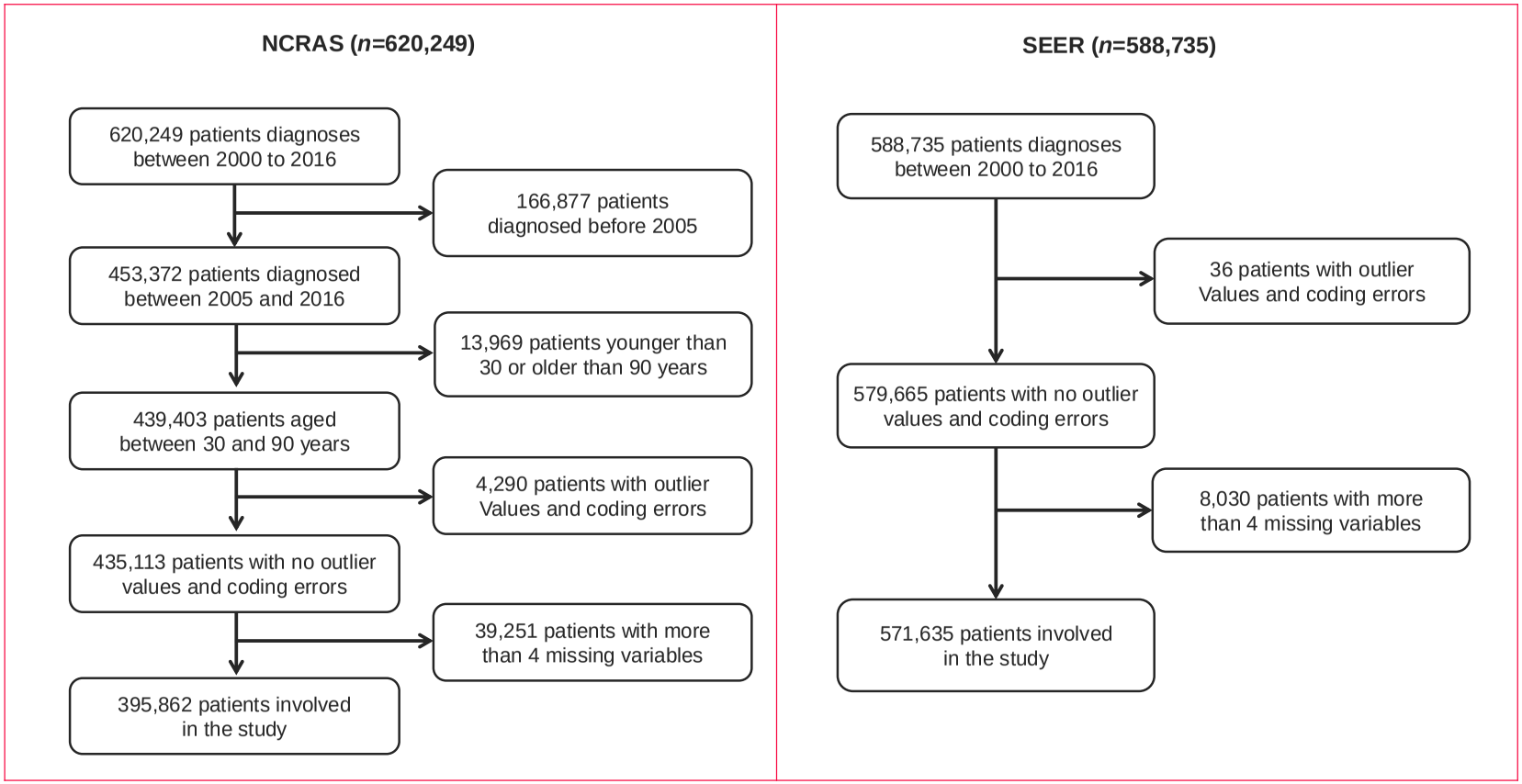


Figure 8.1: Flow charts for the sample selection and patient inclusion process.

8.3 Model Development using AutoPrognosis

AutoPrognosis was used to automatically construct an optimized prognostic model fit to the dataset at hand by tuning the parameters of an ensemble of 20 state-of-the-art machine learning models (such as gradient boosting and deep neural networks). The overall Adjutorium model was constructed by fitting 10 binary classification models (optimized via AutoPrognosis) to predict outcomes at 10 distinct knots (time horizons spanning from 1 to 10 years from baseline, with 1-year increments). Survival curves were created by smoothing the predictions at the 10 knots by fitting the discrete predictions to a Weibull survival function. We used the symbolic metamodeling methodolgy presented in Chapter 3 to convert the trained ensemble model into an understandable mathematical equation that links patient variables to predicted outcomes. An illustrative schematic of AutoPrognosis is provided in Figure 8.2.

8.4 Statistical Analysis

Comparison with Cox Proportional Hazards Model

A standard Cox proportional hazards (PH) model fit on the same data as Adjutorium was also assessed for comparison. Consistent with previous methods, [263] we applied two separate models, with different baseline hazards for ER positive and ER negative cancer. We included an age squared term to allow for non-linear effects of baseline age at diagnosis on breast cancer mortality.

Model Training, Internal and External Validation

Patient samples from the NCRAS database were randomly split into two mutually exclusive cohorts: a training cohort of 316,690 patients used for model derivation, and an internal validation cohort of 79,172 patients used to evaluate model accuracy. The entire SEER cohort (571,635 patients) was reserved for external validation. We trained Adjutorium using the NCRAS training data to predict breast cancer and all-cause mortality without adjuvant therapies by adjusting survival times for treatment effects, to create a counterfactual "untreated" survival cohort. The

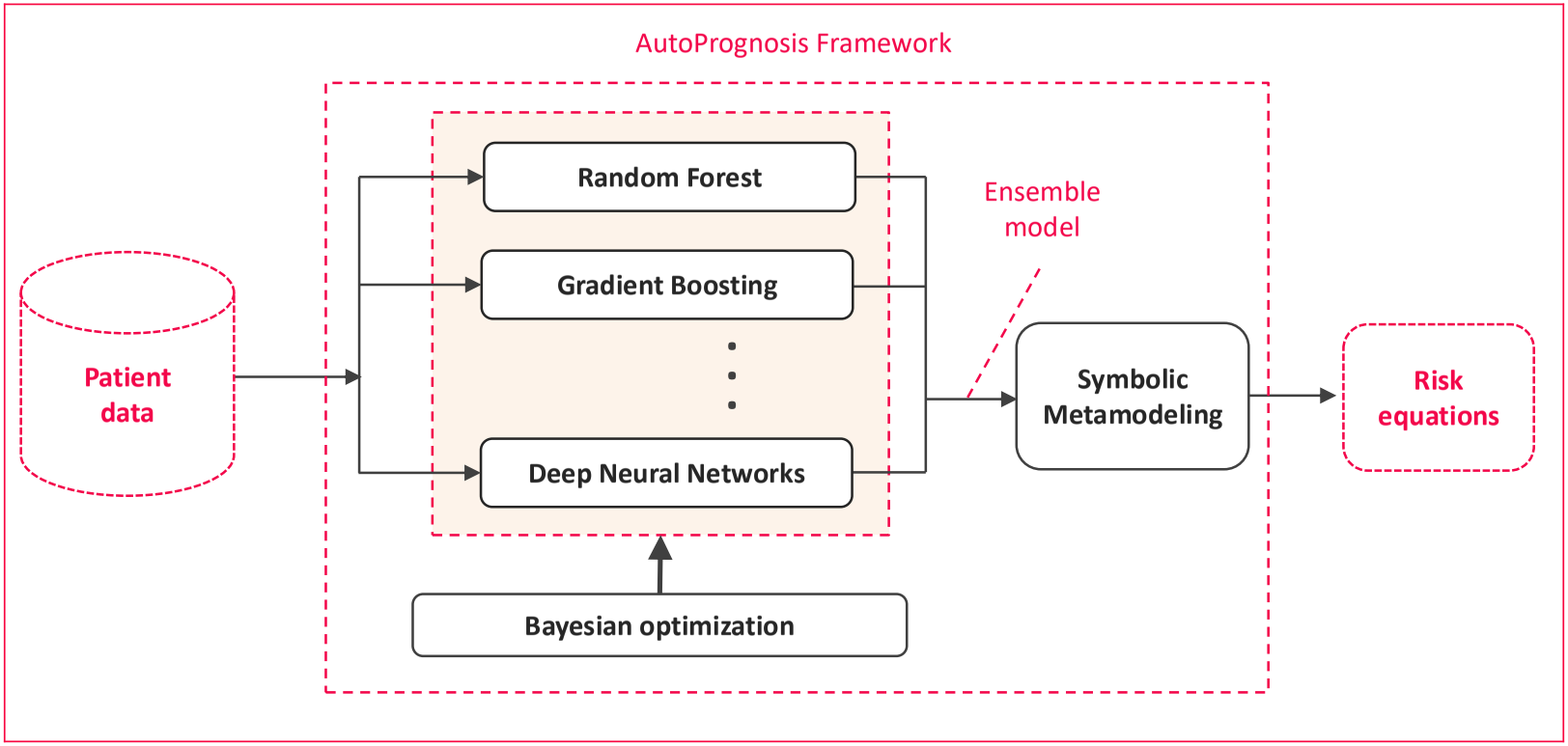


Figure 8.2: **Schematic depiction for the AutoPrognosis framework.** Given patient data, AutoPrognosis uses a Bayesian optimization algorithm to search for the optimal parameters of a collection of machine learning models and the optimal weight assigned to each model in an ensemble. (Here, we depict random forests, gradient boosting and neural network models as exemplary elements of the ensemble.) After fitting the ensemble model, a symbolic regression algorithm is used to convert the fitted model into a mathematical equation that maps patient variables to predicted risk.

estimated survival time in the absence of treatments was calculated as:

$$S_{bc}^{T=0} = S_{bc}^{T=1} \times HR,$$

where S_{bc} represents the uncensored survival time for each individual, T is the indicator for treatment, and HR is the hazard ratio associated with a specific treatment based on the EBCTCG meta-analysis. [277] This is consistent with previous approaches used to create adjusted counterfactual survival times in cross-over trials. [278] The same procedure was applied to the Cox PH model.

We conducted internal and external validation of Adjutorium within the NCRAS validation cohort (n=79,172) and the SEER cohort (n=571,635), respectively. We validated predicted outcomes in the original unadjusted cohort, incorporating treatment effects for patients that had received a given therapy. Using this approach allowed us to evaluate the composite predictive accuracy as well as treatment effects. As breast cancer mortality and mortality from other causes are competing causes, overall survival probability from all causes was calculated as follows:

$$P_{all}(t) = P_{bc}(t) \times P_{nbc}(t).$$

Here, $P_{all}(t)$, $P_{bc}(t)$ and $P_{nbc}(t)$ represent overall survival, survival from breast cancer, and survival from other non-breast cancer related causes at time horizon t, respectively. For individuals on adjuvant therapy, $P_{bc}(t)$ was calculated as a function of survival without treatment $P_{bc}^{T=0}(t)$ (as predicted by the trained model), and the effect of treatment, as follows:

$$P_{bc}^{T=1}(t) = \left(P_{bc}^{T=0}(t)\right)^{HR}.$$

Performance Evaluation

Discriminative Accuracy. We compared the discriminative accuracy of Adjutorium in predicting all-cause and breast cancer-specific mortality at 3, 5 and 10 years from baseline relative to PREDICT v2.1, [263] the Nottingham Prognostic Index (NPI), [279] and the in-house Cox PH model fitted to the NCRAS training cohort. We assessed the discriminative accuracy of Adjutorium using the time-dependent area under receiver operating characteristic curve [280] (AUC-ROC), Harrells

concordance index [281] (C-index), and Unos C-index. [282] For all evaluations, 95% confidence intervals were obtained using bootstrapped re-sampling.

Calibration. We evaluated the calibration curves of Adjutorium by comparing predicted risk of mortality with observed risk in the cohort at the time horizons of interest. For each time horizon, we divided the risk ranges predicted by Adjutorium into 10 quantiles, and within each quantile, we estimated the observed risk in the corresponding patient samples using a Kaplan-Meier estimator. [283] Calibration curves were evaluated by plotting the predicted risks by Adjutorium on the x-axis, and plotting the corresponding observed risk on the y-axis.

Sensitivity analyses. In order to examine the robustness of Adjutorium to missingness, we validated its performance separately on individuals with complete data and those with at least one missing variable. Moreover, in order to assess the robustness of Adjutorium to time-cohort effects, due to changes in patient management and survival over time, we also compared the discriminative accuracy of our model with that of PREDICT v2.1 in subsets of patients diagnosed within 1-year windows spanning from 2005 to 2016.

Subgroup analyses. We validated Adjutorium within specific patient subgroups stratified by age, ER status, HER2 status, tumour size and tumour grade. We specifically assessed the performance of Adjutorium relative to PREDICTv2.1 in patients aged more than 65 years, patients with larger tumours, and patients with negative ER status. Error counts in each subgroup were obtained through decision thresholds that maximize the Youden J-statistic for each model. To assess the prognostic value of each variable, we also evaluated the predictive ability of each individual variable within each subgroup by re-fitting the machine learning model with one variable at a time.

8.5 Results

Adjutorium Model Development. A high-level illustration for the machine learning model generated by AutoPrognosis when fitted to the development cohort (n=316,690) is provided in Figure

8.3. The overall model comprised an ensemble of four binary classification models [284]: random forest, neural network, gradient boosting, and AdaBoost. The prediction issued by Adjutorium is a weighted combination of the predictions of the four members of the ensemble in Figure 8.3.

The risk equation that maps patient variables to breast-cancer-related and non-breast-cancer-related survival curves (i.e., $P_{bc}(t)$ and $P_{nbc}(t)$) are visualized in Figure 8.4. For a given patient, the breast-cancer-related survival probability is given by $P_{bc}(t) = 1/(1+\exp(-\lambda_{bc}(t)))$, where t is the time horizon at which the survival probability is evaluated. The term $\lambda_{bc}(t)$ can be interpreted as the *odds ratio* for survival at time t, and is decomposed as follows:

$$\lambda_{bc}(t) = \underbrace{\bar{\lambda}_{bc}(t)}_{\text{Population-level}} + \underbrace{\bar{\lambda}_{bc}^{G,ER}(t)}_{\text{Grade-ER-specific}} ,$$

where the first term $\bar{\lambda}_{bc}(t)$ is shared among all patients in the population, and includes the non-linear effects of the age and number of lymph nodes variables, in addition to interaction terms between age, mode of detection, tumour size and number of lymph nodes (Figure 8.4). The second term $\bar{\lambda}_{bc}^{G,ER}(t)$ includes linear contributions of all prognostic variables, with coefficients that are specific to every possible combination of tumour grade and ER status. The risk equations in Figure 8.4 demonstrate that our machine learning approach identified new interactions that were not incorporated in previous models [263], namely the interactions between tumour grade and all other prognostic factors. The risk equation for $P_{nbc}(t)$ is similar to that of $P_{bc}(t)$.

Discriminative Accuracy. Adjutorium uniformly outperformed PREDICT v2.1, NPI, and the conventional Cox PH model in predicting all-cause and breast cancer-specific mortality, both when validated internally within NCRAS (Table 8.5), and externally within the SEER cohort (Table 8.6). The improvements were achieved with respect to all discriminative accuracy metrics and all time horizons under study.

In internal validation, Adjutorium predicted 10-year all-cause mortality with an AUC-ROC accuracy of 0.813 (95% CI: 0.811-0.815), compared with 0.771 (95% CI: 0.769-0.773) by PREDICT v2.1, 0.687 (95% CI: 0.685-0.689) by NPI, and 0.773 (95% CI: 0.769-0.777) by the Cox PH model. Similar performance gains were achieved over the other time horizons, and with respect to the C-index statistic (Table 8.5). The improvements in accuracy achieved by Adjutorium were even more significant in predicting breast cancer-specific mortality, with an AUC-ROC of 0.824 (95%

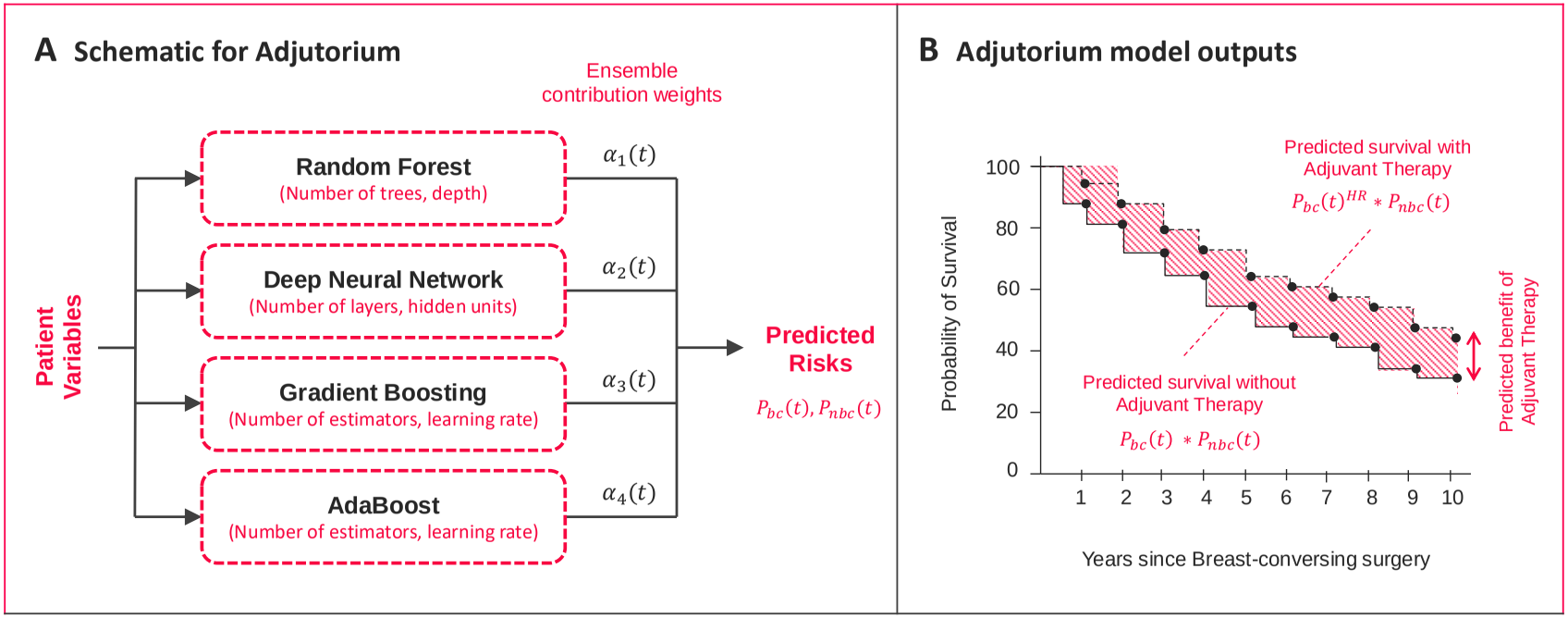


Figure 8.3: **Illustration for the machine learning model underlying Adjutorium.** Panel A displays the model learned by the AutoPrognosis framework. The overall model comprises an ensemble of four basic machine learning models: random forest, neural network, gradient boosting, and AdaBoost. The prediction issued by Adjutorium is a weighted combination of the predictions issued by each of the four members of the ensemble. Each model in the ensemble has a set of parameters (listed between brackets in Panel A), and an assigned weight $\alpha(t)$ determining its contribution in the final risk prediction. Both the model parameters and its weight change depending on the prediction horizon t. The predicted survival curve for an exemplary patient (with and without adjuvant therapy) is shown in Panel B. Here, each prediction horizon (1 to 10 years since diagnosis, with 1-year steps) corresponds to a knot in the survival curve, and each knot is associated with a distinct set of model parameters and contribution weights in the ensemble in Panel A.

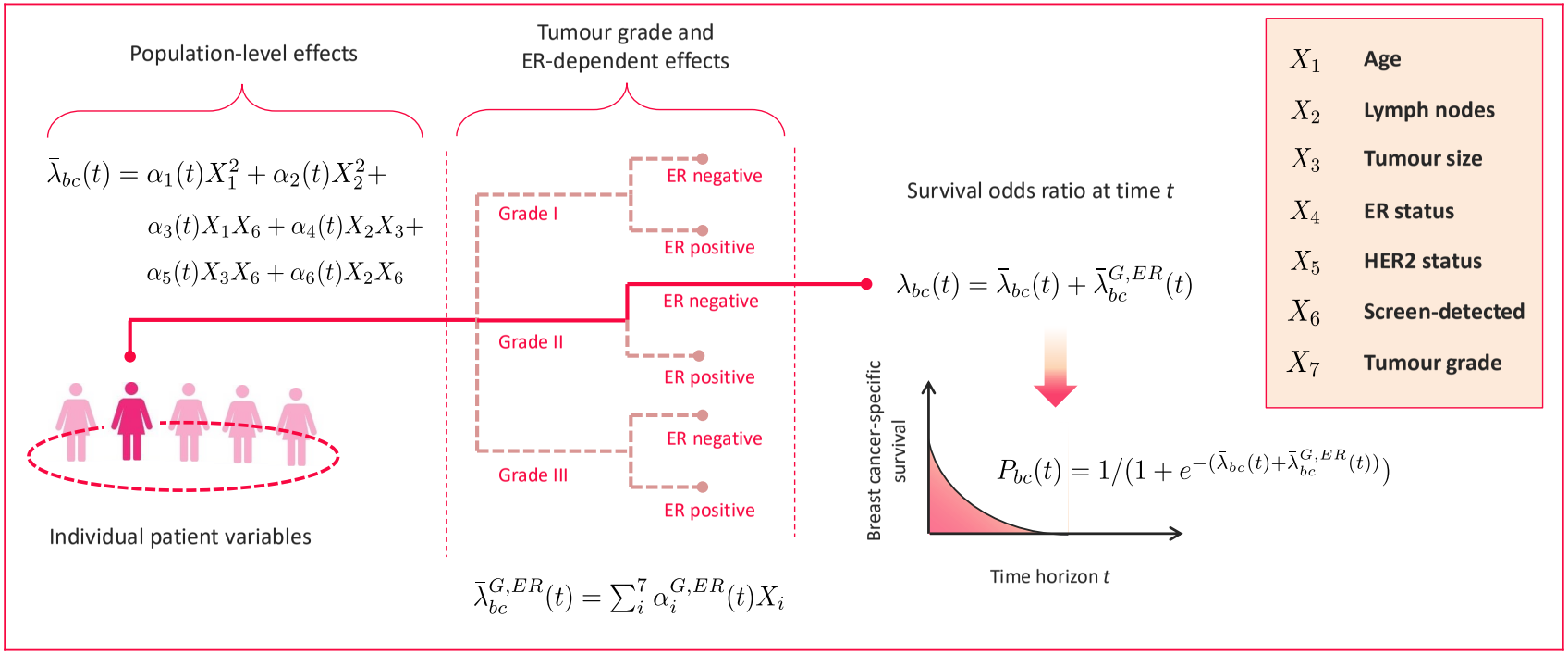


Figure 8.4: **Risk equations underlying Adjutorium.** Given the individual-level variables of a patient, the risk equation evaluates a survival curve corresponding to the probability of survival at future time horizons. The odds ratio for survival at time t is decomposed into two components: (1) a population-level term that models non-linear effects of age and number of lymph nodes, in addition to interactions between different variables using six coefficients that are fixed for all patients, and (2) a tumour grade and ER-specific term that evaluates the linear effects of all prognostic factors with coefficients that are specific to every group of patients with the same grade and ER status. Here we show an exemplary patient with ER negative cancer and tumour grade 2 and. The risk equation above is an abstraction for the predictions issued by the machine learning model in Figure 8.3 that ensure the model's interpretability and transparency.

Time Horizon	Metric (95% CI)	All-cause Mortality: Adjutorium	All-cause Mortality: Cox PH	All-cause Mortality: PREDICT	All-cause Mortality: NPI	Breast cancer-specific Mortality: Adjutorium	Breast cancer-specific Mortality: Cox PH	Breast cancer-specific Mortality: PREDICT	Breast cancer-specific Mortality: NPI
3 years	H. C-index	0.781 (0.780–0.782)	0.756 (0.753–0.759)	0.748 (0.747–0.749)	0.707 (0.705–0.709)	0.808 (0.807–0.809)	0.772 (0.770–0.774)	0.740 (0.739–0.741)	0.759 (0.757–0.761)
	U. C-index	0.753 (0.751–0.755)	0.733 (0.730–0.736)	0.706 (0.704–0.708)	0.660 (0.658–0.662)	0.763 (0.761–0.765)	0.730 (0.726–0.734)	0.702 (0.700–0.704)	0.730 (0.728–0.732)
	AUC-ROC	0.819 (0.817–0.821)	0.796 (0.793–0.799)	0.785 (0.783–0.787)	0.745 (0.743–0.747)	0.847 (0.845–0.849)	0.816 (0.812–0.820)	0.765 (0.763–0.767)	0.785 (0.783–0.787)
5 years	H. C-index	0.787 (0.786–0.788)	0.757 (0.755–0.759)	0.758 (0.757–0.759)	0.707 (0.705–0.709)	0.807 (0.806–0.808)	0.775 (0.772–0.778)	0.748 (0.747–0.749)	0.760 (0.758–0.762)
	U. C-index	0.756 (0.754–0.758)	0.734 (0.731–0.737)	0.718 (0.716–0.720)	0.660 (0.658–0.662)	0.766 (0.764–0.768)	0.736 (0.732–0.740)	0.708 (0.706–0.710)	0.730 (0.728–0.732)
	AUC-ROC	0.814 (0.812–0.816)	0.775 (0.772–0.778)	0.773 (0.771–0.775)	0.729 (0.727–0.731)	0.834 (0.832–0.836)	0.795 (0.792–0.798)	0.753 (0.751–0.755)	0.775 (0.773–0.777)
10 years	H. C-index	0.771 (0.817–0.821)	0.758 (0.756–0.760)	0.771 (0.770–0.772)	0.707 (0.705–0.709)	0.791 (0.790–0.792)	0.777 (0.774–0.780)	0.750 (0.748–0.752)	0.759 (0.757–0.761)
	U. C-index	0.747 (0.745–0.749)	0.737 (0.733–0.741)	0.734 (0.732–0.736)	0.657 (0.655–0.659)	0.755 (0.753–0.757)	0.736 (0.731–0.741)	0.714 (0.712–0.716)	0.730 (0.728–0.732)
	AUC-ROC	0.813 (0.811–0.815)	0.773 (0.769–0.777)	0.771 (0.769–0.773)	0.687 (0.685–0.689)	0.824 (0.821–0.827)	0.784 (0.780–0.788)	0.729 (0.726–0.732)	0.756 (0.754–0.758)

* CI denotes Confidence Interval. H. C-index and U. C-index denote the Harrell and Uno concordance indexes, respectively.

Figure 8.5: Discriminative Accuracy with Respect to the Primary and Secondary Outcomes in the Internal Validation Cohort (NCRAS, n=79,172).*

Time Horizon	Metric (95% CI)	All-cause Mortality				Breast cancer-specific Mortality			
		Adjutorium	Cox PH	PREDICT	NPI	Adjutorium	Cox PH	PREDICT	NPI
3 years	H. C-index	0.751 (0.749–0.753)	0.747 (0.745–0.749)	0.735 (0.733–0.737)	0.666 (0.664–0.668)	0.796 (0.794–0.798)	0.763 (0.760–0.766)	0.763 (0.761–0.765)	0.774 (0.771–0.777)
	U. C-index	0.741 (0.737–0.745)	0.733 (0.731–0.735)	0.697 (0.693–0.701)	0.631 (0.629–0.633)	0.754 (0.748–0.760)	0.726 (0.720–0.732)	0.722 (0.717–0.727)	0.758 (0.755–0.761)
	AUC-ROC	0.770 (0.768–0.772)	0.772 (0.768–0.776)	0.761 (0.759–0.763)	0.703 (0.700–0.706)	0.823 (0.820–0.826)	0.792 (0.788–0.796)	0.783 (0.780–0.786)	0.795 (0.792–0.798)
5 years	H. C-index	0.757 (0.755–0.759)	0.744 (0.742–0.746)	0.743 (0.741–0.745)	0.667 (0.665–0.669)	0.794 (0.792–0.796)	0.768 (0.764–0.772)	0.765 (0.763–0.767)	0.776 (0.774–0.778)
	U. C-index	0.735 (0.730–0.740)	0.731 (0.724–0.738)	0.708 (0.705–0.711)	0.629 (0.627–0.631)	0.760 (0.755–0.765)	0.722 (0.714–0.730)	0.735 (0.730–0.740)	0.761 (0.759–0.763)
	AUC-ROC	0.777 (0.775–0.779)	0.763 (0.759–0.767)	0.755 (0.753–0.757)	0.681 (0.678–0.684)	0.813 (0.811–0.815)	0.782 (0.778–0.786)	0.775 (0.772–0.778)	0.790 (0.788–0.792)
10 years	H. C-index	0.748 (0.746–0.752)	0.742 (0.739–0.745)	0.750 (0.748–0.752)	0.669 (0.667–0.671)	0.778 (0.776–0.780)	0.764 (0.761–0.767)	0.765 (0.763–0.767)	0.777 (0.775–0.779)
	U. C-index	0.736 (0.732–0.740)	0.739 (0.734–0.744)	0.727 (0.724–0.730)	0.630 (0.628–0.632)	0.746 (0.741–0.751)	0.728 (0.720–0.736)	0.738 (0.734–0.742)	0.759 (0.757–0.761)
	AUC-ROC	0.790 (0.787–0.793)	0.778 (0.771–0.785)	0.756 (0.753–0.759)	0.631 (0.628–0.634)	0.800 (0.796–0.804)	0.773 (0.764–0.780)	0.744 (0.741–0.747)	0.768 (0.765–0.771)

* CI denotes Confidence Interval. H. C-index and U. C-index denote the Harrell and Uno concordance indexes, respectively.

Figure 8.6: Discriminative Accuracy with Respect to the Primary and Secondary Outcomes in the External Validation Cohort (SEER, n=571,635).*

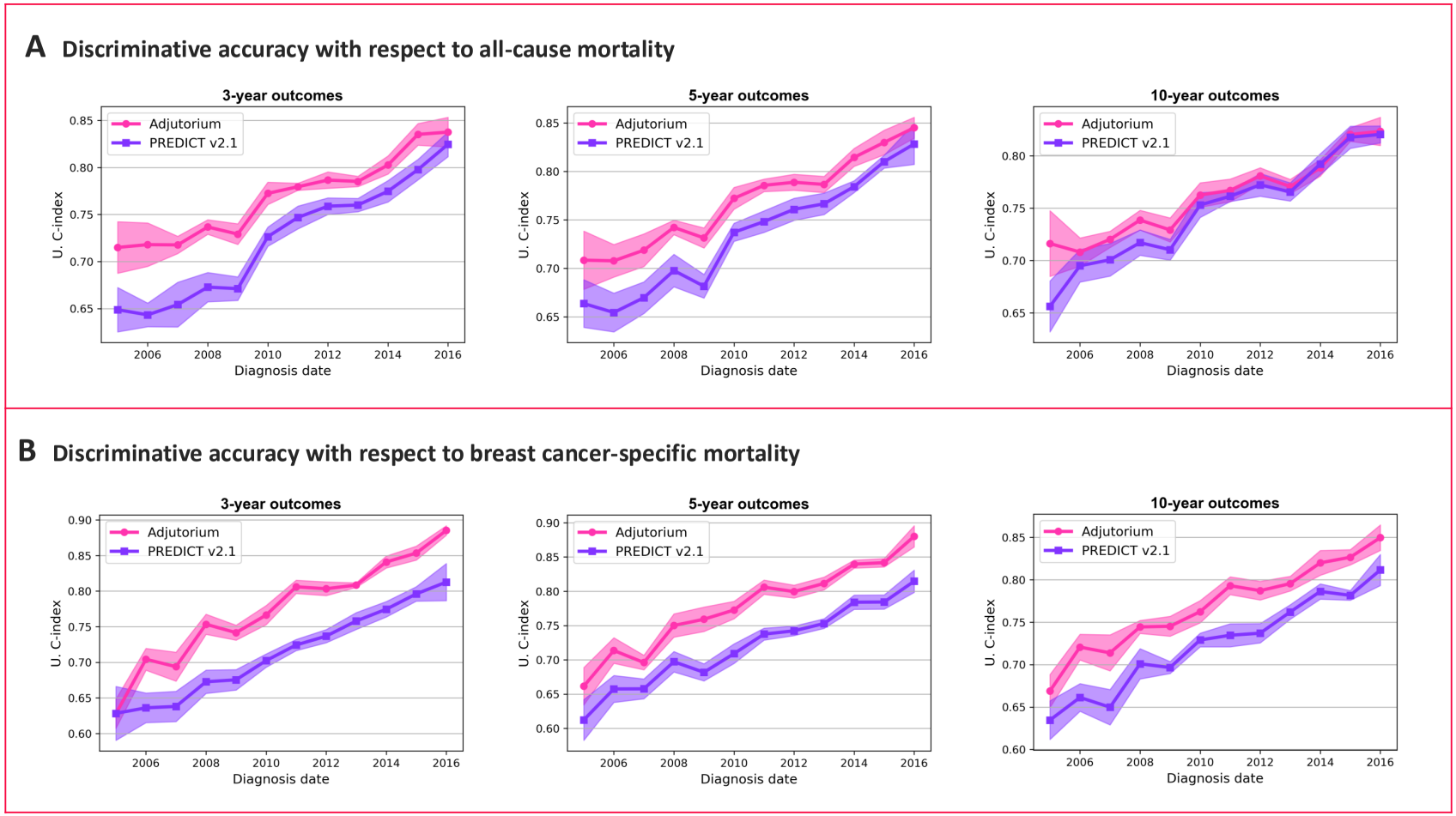


Figure 8.7: Discriminative accuracy evaluated in sub-cohorts of patients stratified by diagnosis date.

CI: 0.821-0.827) for 10-year outcomes, compared with 0.729 (95% CI: 0.726-0.732) by PREDICT v2.1, 0.756 (95% CI: 0.754-0.758) by NPI, and 0.784 (95% CI: 0.780-0.788) by the Cox PH model. The fact that the accuracy improvements were more significant in the secondary outcome is not surprising since all of the variables included in the model were breast cancer-related.

Adjutorium generalized well to the external validation cohort, with similar accuracy improvements for both the primary and secondary outcomes (Table 8.6). With respect to 10-year all-cause mortality, Adjutorium achieved an AUC-ROC of 0.790 (95% CI: 0.787-0.793), compared to 0.756 (95% CI: 0.753-0.759) by PREDICT, 0.631 (95% CI: 0.628-0.634) by NPI, and 0.778 (95% CI: 0.771-0.785) by the Cox PH model. Similar gains were achieved over the other time horizons (Table 8.6). For prediction of 10-year breast cancer-specific mortality, Adjutorium achieved an AUC-ROC of 0.800 (95% CI: 0.796-0.804), compared to 0.744 (95% CI: 0.741-0.747) by PREDICT, 0.768 (95% CI: 0.765-0.771) by NPI, and 0.773 (95% CI: 0.764-0.780) by Cox PH model.

Importantly, Adjutorium outperformed the Cox PH model fitted to the same development cohort, reflecting the *gain from modeling*, i.e., the gain achieved by using flexible machine learning models instead of standard regression. On the other hand, the gain achieved by the Cox PH model compared to PREDICT v2.1 in external validation reflects the *gain from information*, i.e., the gain achieved by using large-scale, representative data that enhance the accuracy and generalizability of the fitted models to other cohorts that might entail different demographic structure and outcomes.

Sensitivity Analysis. Internal and external validation on patient sample with complete and missing data demonstrated the robustness of Adjutorium to data missingness; the model performed well in cases with complete and missing data, outperforming other models by similar margins in both analyses. When validated on 21,164 patients (in the internal validation cohort) with complete data on all variables, the AUC-ROC accuracy of Adjutorium with respect to 10-year breast cancer-specific mortality was 0.811 (95% CI: 0.0.808-0.814), and 0.783 (95% CI: 0.780-0.786) for PREDICT v2.1. When validated on 57,996 patients with missing data on one or more variables, the AUC-ROC accuracy of Adjutorium was 0.829 (95% CI: 0.0.827-0.831), and 0.728 (95% CI: 0.725-0.731) for PREDICT v2.1. Adjutorium also displayed robustness to time-cohort effects; internal validation on sub-cohorts stratified by diagnosis dates from 2005 to 2016 showed that the

accuracy gains by Adjutorium are achieved for all diagnosis years Figure 8.7.

Subgroup Analysis. The accuracy improvements achieved by Adjutorium were consistent across all subgroups of patients stratified by age, HER2 status, ER status and tumour grade (Table 8.8). Improvements were greater in subgroups that are poorly served by current prognostic tools; the accuracy gains achieved by Adjutorium relative to PREDICT v2.1 were higher in elderly patients (age $>$ 65 yrs at diagnosis), patients with ER negative and HER2 negative breast cancer (Table 8.8), and patients with large tumours. This is likely due to the fact that data-driven machine learning captured nuanced interactions and non-linear patterns that were not incorporated in existing prognostic tools.

8.6 Discussion and Conclusions

In this Chapter, we developed and validated Adjutorium — a machine learning-based tool for predicting the individualized benefit of adjuvant therapies in breast cancer based on the AutoPrognosis framework presented in Chapter 4. Involving data from nearly 1 million individuals with breast cancer from the UK and US, this is one of the largest studies of its kind. We found that Adjutorium substantially outperforms one of the most widely used standards for clinical decision making, and critically is generalisable to distinct clinical settings. To our knowledge this is the first application of a machine learning model for prognostication in breast cancer, that has been shown to be generalisable across multiple nationally representative cohorts.

While several prognostication methods are available for supporting clinical decisions regarding adjuvant therapies in breast cancer, they have well recognized limitations particularly in terms of their accuracy in certain subgroups and their generalisability to other populations. We find that Adjutorium outperforms existing clinical decision support tools in terms of accuracy, and calibration to observed outcomes, across all patient groups. Additionally, it shows substantially improved performance in subgroups where existing clinical decision support tools are known perform poorly (e.g., older women with early cancer, HER negative and ER negative breast cancer) suggesting that using Adjutorium to support clinical decisions may lead to better treatment decisions, and poten-

			Adjutorium			PREDICT v2.1		
	No. of cases	Observed deaths	AUC-ROC	TP	FP	AUC-ROC	TP	FP
Age at diagnosis								
ER positive								
30 – 65 years	21,302	2,314	0.791	1,658	5,142	0.773	1,607	5,171
> 65 years	13,115	3,774	0.824	3,026	2,767	0.779	2,915	2,937
ER negative								
30 – 65 years	10,417	2,440	0.729	1,615	2,634	0.666	1,595	3,043
> 65 years	4,861	2,090	0.785	1,458	730	0.700	1,626	1,202
HER2 positive								
30 – 65 years	11,894	2,390	0.717	1,563	3,157	0.682	1,535	3,299
> 65 years	4,388	1,940	0.767	1,370	733	0.671	1,449	1,131
HER2 negative								
30 – 65 years	19,825	2,363	0.816	1,749	4,286	0.797	1,749	4,898
> 65 years	13,588	3,924	0.825	2,970	2,443	0.763	3,088	3,433
Grade I								
30 – 65 years	4,942	146	0.752	101	1,262	0.739	103	1,580
> 65 years	2,608	382	0.816	273	423	0.758	290	683
Grade II and III								
30 – 65 years	26,777	4,607	0.762	3,369	7,418	0.718	3,348	9,078
> 65 years	15,368	5,482	0.806	4,179	2,884	0.721	4,702	5,074

* FP and TP denote false positive and true positive cases, respectively.

Figure 8.8: Subgroup-level Discrimination with Respect to Breast cancer-specific 10-year Outcomes in Internal Validation*.

tially better outcomes in these subgroups. By contrast with other existing tools, Adjutorium is robust to missing data, and is able to make accurate predictions even when information on some of the prognostic factors is not available. This is an important advance, making our model more generalisable to settings where data on patients may be incomplete.

We find that Adjutorium not only outperforms PREDICT v2.1, but also a Cox proportional hazards model fit on the same training cohort. This suggests that gains in performance are achieved not only due to a larger representative set for training the models, but also due to the flexible nature of the machine learning algorithms applied. Our fitted model does not make any assumptions about the linearity of the patient risks as function of prognostic factors, or the proportionality of hazards over time. Additionally it is able to infer interactions, and non-linear associations in a data-driven fashion, as evident through the interpretable risk equations describing the machine learning model.

We acknowledge limitations of our model, which include the retrospective nature of our study which makes it difficult to assess changes in patient outcomes when using Adjutorium relative to existing tools. Another limitation is that our model does not predict outcomes such as recurrence, and currently does not incorporate gene expression based predictive information. However, these can be easily incorporated into our model. Also, Adjutorium does not explicitly derive treatment effects in a data-driven fashion, rather using estimates from meta-analyses on clinical trials.

REFERENCES

[1] Karen B DeSalvo, Ayame Nagatani Dinkler, and Lee Stevens. The us office of the national coordinator for health information technology: progress and promise for the future at the 10-year mark. *Annals of emergency medicine*, 66(5):507–510, 2015.

[2] Kate Ann Levin. Study design iii: Cross-sectional studies. *Evidence-based dentistry*, 7(1):24, 2006.

[3] E Kevin Kelloway and Lori Francis. Longitudinal research and data analysis. 2013.

[4] Aluísio JD Barros and Vânia N Hirakata. Alternatives for logistic regression in cross-sectional studies: an empirical comparison of models that directly estimate the prevalence ratio. *BMC medical research methodology*, 3(1):21, 2003.

[5] Fliss EM Murtagh, Neil S Sheerin, Julia Addington-Hall, and Irene J Higginson. Trajectories of illness in stage 5 chronic kidney disease: a longitudinal study of patient symptoms and concerns in the last year of life. *Clinical Journal of the American Society of Nephrology*, 6(7):1580–1590, 2011.

[6] Leticia Coutinho, Marcia Scazufca, and Paulo R Menezes. Methods for estimating prevalence ratios in cross-sectional studies. *Revista de saude publica*, 42(6):992–998, 2008.

[7] Stephen F Weng, Jenna Reps, Joe Kai, Jonathan M Garibaldi, and Nadeem Qureshi. Can machine-learning improve cardiovascular risk prediction using routine clinical data? *PloS one*, 12(4):e0174944, 2017.

[8] Stephen L Morgan and Christopher Winship. *Counterfactuals and causal inference*. Cambridge University Press, 2015.

[9] Riccardo Guidotti, Anna Monreale, Salvatore Ruggieri, Franco Turini, Fosca Giannotti, and Dino Pedreschi. A survey of methods for explaining black box models. *ACM computing surveys (CSUR)*, 51(5):93, 2019.

[10] Michael W Kattan, Kenneth R Hess, Mahul B Amin, Ying Lu, Karl GM Moons, Jeffrey E Gershenwald, Phyllis A Gimotty, Justin H Guinney, Susan Halabi, Alexander J Lazar, et al. American joint committee on cancer acceptance criteria for inclusion of risk models for individualized prognosis in the practice of precision medicine. *CA: a cancer journal for clinicians*, 66(5):370–374, 2016.

[11] Jared C Foster, Jeremy MG Taylor, and Stephen J Ruberg. Subgroup identification from randomized clinical trial data. *Statistics in medicine*, 30(24):2867–2880, 2011.

[12] Sheng Li and Yun Fu. Matching on balanced nonlinear representations for treatment effects estimation. In *Advances in Neural Information Processing Systems*, pages 930–940, 2017.

[13] Stefan Wager and Susan Athey. Estimation and inference of heterogeneous treatment effects using random forests. *Journal of the American Statistical Association*, (just-accepted), 2017.

[14] Uri Shalit, Fredrik Johansson, and David Sontag. Estimating individual treatment effect: generalization bounds and algorithms. pages 3076–3085, 2017.

[15] Ahmed M Alaa and Mihaela van der Schaar. Bayesian inference of individualized treatment effects using multi-task gaussian processes. *Advances in Neural Information Processing Systems (NIPS)*, 2017.

[16] James J Heckman. Sample selection bias as a specification error (with an application to the estimation of labor supply functions), 1977.

[17] Donald B Rubin. Causal inference using potential outcomes: Design, modeling, decisions. *Journal of the American Statistical Association*, 100(469):322–331, 2005.

[18] Paul R Rosenbaum and Donald B Rubin. Reducing bias in observational studies using subclassification on the propensity score. *Journal of the American statistical Association*, 79(387):516–524, 1984.

[19] Léon Bottou, Jonas Peters, Joaquin Quiñonero-Candela, Denis X Charles, D Max Chickering, Elon Portugaly, Dipankar Ray, Patrice Simard, and Ed Snelson. Counterfactual reasoning and learning systems: The example of computational advertising. *The Journal of Machine Learning Research*, 14(1):3207–3260, 2013.

[20] Ahmed M Alaa and Mihaela van der Schaar. Bayesian nonparametric causal inference: Information rates and learning algorithms. *Journal on Selected Topics in Signal Processing*, 2018.

[21] Sören Künzel, Jasjeet Sekhon, Peter Bickel, and Bin Yu. Meta-learners for estimating heterogeneous treatment effects using machine learning. *arXiv preprint arXiv:1706.03461*, 2017.

[22] Min Lu, Saad Sadiq, Daniel J Feaster, and Hemant Ishwaran. Estimating individual treatment effect in observational data using random forest methods. *Journal of Computational and Graphical Statistics*, 2017.

[23] Fredrik Johansson, Uri Shalit, and David Sontag. Learning representations for counterfactual inference. In *International Conference on Machine Learning*, pages 3020–3029, 2016.

[24] Jinsung Yoon, James Jordon, and Mihaela van der Schaar. Ganite: Estimation of individualized treatment effects using generative adversarial nets. *International Conference on Learning Representations (ICLR)*, 2018.

[25] Masashi Sugiyama, Matthias Krauledat, and Klaus-Robert MÃžller. Covariate shift adaptation by importance weighted cross validation. *Journal of Machine Learning Research*, 8(May):985–1005, 2007.

[26] O Atan, J Jordan, and M van der Schaar. Deep-treat: Learning optimal personalized treatments from observational data using neural networks. *AAAI*, 2018.

[27] Edward H Kennedy. Nonparametric causal effects based on incremental propensity score interventions. *Journal of the American Statistical Association*, 2018.

[28] Qinghua Zhang. Using wavelet network in nonparametric estimation. *IEEE Transactions on Neural networks*, 8(2):227–236, 1997.

[29] Aad W van der Vaart, J Harry van Zanten, et al. Reproducing kernel hilbert spaces of gaussian priors. In *Pushing the limits of contemporary statistics: contributions in honor of Jayanta K. Ghosh*, pages 200–222. Institute of Mathematical Statistics, 2008.

[30] Suzanne Sniekers, Aad van der Vaart, et al. Adaptive bayesian credible sets in regression with a gaussian process prior. *Electronic Journal of Statistics*, 9(2):2475–2527, 2015.

[31] Jennifer L Hill. Bayesian nonparametric modeling for causal inference. *Journal of Computational and Graphical Statistics*, 20(1):217–240, 2011.

[32] Charles J Stone. Optimal global rates of convergence for nonparametric regression. *The annals of statistics*, pages 1040–1053, 1982.

[33] Yun Yang, Surya T Tokdar, et al. Minimax-optimal nonparametric regression in high dimensions. *The Annals of Statistics*, 43(2):652–674, 2015.

[34] Yuhong Yang and Andrew Barron. Information-theoretic determination of minimax rates of convergence. *Annals of Statistics*, pages 1564–1599, 1999.

[35] Garvesh Raskutti, Bin Yu, and Martin J Wainwright. Lower bounds on minimax rates for nonparametric regression with additive sparsity and smoothness. In *Advances in Neural Information Processing Systems*, pages 1563–1570, 2009.

[36] Scott Powers, Junyang Qian, Kenneth Jung, Alejandro Schuler, Nigam H Shah, Trevor Hastie, and Robert Tibshirani. Some methods for heterogeneous treatment effect estimation in high-dimensions. *arXiv preprint arXiv:1707.00102*, 2017.

[37] Marıa Isabel Borrajo, Wenceslao González-Manteiga, and María Dolores Martínez-Miranda. Bandwidth selection for kernel density estimation with length-biased data. *Journal of Nonparametric Statistics*, 29(3):636–668, 2017.

[38] Hidetoshi Shimodaira. Improving predictive inference under covariate shift by weighting the log-likelihood function. *Journal of statistical planning and inference*, 90(2):227–244, 2000.

[39] Masashi Sugiyama and Amos J Storkey. Mixture regression for covariate shift. In *Advances in Neural Information Processing Systems*, pages 1337–1344, 2007.

[40] P Richard Hahn, Jared S Murray, and Carlos M Carvalho. Bayesian regression tree models for causal inference: regularization, confounding, and heterogeneous effects. 2017.

[41] Carl Edward Rasmussen and Christopher KI Williams. *Gaussian processes for machine learning*, volume 1. MIT press Cambridge, 2006.

[42] Mauricio A Alvarez, Lorenzo Rosasco, Neil D Lawrence, et al. Kernels for vector-valued functions: A review. *Foundations and Trends® in Machine Learning*, 4(3):195–266, 2012.

[43] Aad van der Vaart and Harry van Zanten. Information rates of nonparametric gaussian process methods. *Journal of Machine Learning Research*, 12:2095–2119, 2011.

[44] Aad W van der Vaart and J Harry van Zanten. Rates of contraction of posterior distributions based on gaussian process priors. *The Annals of Statistics*, pages 1435–1463, 2008.

[45] I. Castillo. Lower bounds for posterior rates with gaussian process priors. *Electron. J. Stat.*, 2:1281–1299, 2008.

[46] Jiayuan Huang, Arthur Gretton, Karsten M Borgwardt, Bernhard Schölkopf, and Alex J Smola. Correcting sample selection bias by unlabeled data. In *Advances in neural information processing systems*, pages 601–608, 2007.

[47] A Bhattacharyya. On some analogues of the amount of information and their use in statistical estimation. *Sankhyā: The Indian Journal of Statistics*, pages 1–14, 1946.

[48] Peter J Bickel, Chris A Klaassen, Peter J Bickel, Y Ritov, J Klaassen, Jon A Wellner, and YA'Acov Ritov. *Efficient and adaptive estimation for semiparametric models*, volume 2. Springer New York, 1998.

[49] James Robins, Lingling Li, Eric Tchetgen, Aad van der Vaart, et al. Higher order influence functions and minimax estimation of nonlinear functionals. In *Probability and Statistics: Essays in Honor of David A. Freedman*, pages 335–421. Institute of Mathematical Statistics, 2008.

[50] James M Robins. Optimal structural nested models for optimal sequential decisions. In *Proceedings of the second seattle Symposium in Biostatistics*, pages 189–326. Springer, 2004.

[51] Sandrine Dudoit and Mark J van der Laan. Asymptotics of cross-validated risk estimation in estimator selection and performance assessment. *Statistical Methodology*, 2(2):131–154, 2005.

[52] Ioannis Karatzas and Steven Shreve. *Brownian motion and stochastic calculus*, volume 113. Springer Science & Business Media, 2012.

[53] Kristin E Porter, Susan Gruber, Mark J Van Der Laan, and Jasjeet S Sekhon. The relative performance of targeted maximum likelihood estimators. *The International Journal of Biostatistics*, 7(1):1–34, 2011.

[54] Scott M Lundberg and Su-In Lee. A unified approach to interpreting model predictions. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 4765–4774, 2017.

[55] Zachary C Lipton. The mythos of model interpretability. *arXiv preprint arXiv:1606.03490*, 2016.

[56] Ahmed M. Alaa and Mihaela van der Schaar. Attentive state-space modeling of disease progression. *In Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

[57] Michael Schmidt and Hod Lipson. Distilling free-form natural laws from experimental data. *Science*, 324(5923):81–85, 2009.

[58] Yiqun Wang, Nicholas Wagner, and James M Rondinelli. Symbolic regression in materials science. *arXiv preprint arXiv:1901.04136*, 2019.

[59] Jinsung Yoon, James Jordon, and Mihaela van der Schaar. Invase: Instance-wise variable selection using neural networks. *International Conference on Representation Learning (ICLR)*, 2018.

[60] James Jordon, Jinsung Yoon, and Mihaela van der Schaar. Knockoffgan: Generating knockoffs for feature selection using generative adversarial networks. *International Conference on Representation Learning (ICLR)*, 2018.

[61] Avanti Shrikumar, Peyton Greenside, and Anshul Kundaje. Learning important features through propagating activation differences. In *International Conference on Machine Learning (ICML)*, pages 3145–3153. JMLR. org, 2017.

[62] Yin Lou, Rich Caruana, Johannes Gehrke, and Giles Hooker. Accurate intelligible models with pairwise interactions. In *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 623–631. ACM, 2013.

[63] Ethan Elenberg, Alexandros G Dimakis, Moran Feldman, and Amin Karbasi. Streaming weak submodularity: Interpreting neural networks on the fly. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 4044–4054, 2017.

[64] Kristofer Bouchard, Alejandro Bujan, Farbod Roosta-Khorasani, Shashanka Ubaru, Mr Prabhat, Antoine Snijders, Jian-Hua Mao, Edward Chang, Michael W Mahoney, and Sharmodeep Bhattacharya. Union of intersections (uoi) for interpretable data driven discovery and prediction. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 1078–1086, 2017.

[65] David Alvarez Melis and Tommi Jaakkola. Towards robust interpretability with self-explaining neural networks. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 7775–7784, 2018.

[66] Michael Tsang, Hanpeng Liu, Sanjay Purushotham, Pavankumar Murali, and Yan Liu. Neural interaction transparency (nit): Disentangling learned interactions for improved interpretability. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 5804–5813, 2018.

[67] Xin Zhang, Armando Solar-Lezama, and Rishabh Singh. Interpreting neural network judgments via minimal, stable, and symbolic corrections. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 4874–4885, 2018.

[68] Jianbo Chen, Le Song, Martin J Wainwright, and Michael I Jordan. Learning to explain: An information-theoretic perspective on model interpretation. *International Conference on Machine Learning (ICML)*, 2018.

[69] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. Why should i trust you?: Explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pages 1135–1144. ACM, 2016.

[70] Martha LL Abell and James P Braselton. *Mathematica by example*. Academic Press, 2017.

[71] Stephen Wolfram. Wolfram research. *Inc., Mathematica, Version*, 8:23, 2013.

[72] Aaron Meurer, Christopher P. Smith, Mateusz Paprocki, Ondřej Čertík, Sergey B. Kirpichev, Matthew Rocklin, AMiT Kumar, Sergiu Ivanov, Jason K. Moore, Sartaj Singh, Thilina Rathnayake, Sean Vig, Brian E. Granger, Richard P. Muller, Francesco Bonazzi, Harsh Gupta, Shivam Vats, Fredrik Johansson, Fabian Pedregosa, Matthew J. Curry, Andy R. Terrel, Štěpán Roučka, Ashutosh Saboo, Isuru Fernando, Sumith Kulal, Robert Cimrman, and Anthony Scopatz. Sympy: symbolic computing in python. *PeerJ Computer Science*, 3:e103, 2017.

[73] Pang Wei Koh and Percy Liang. Understanding black-box predictions via influence functions. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 1885–1894. JMLR. org, 2017.

[74] CS Meijer. On the g-function. *North-Holland*, 1946.

[75] CS Meijer. Uber whittakersche bezw. besselsche funktionen und deren produkte (english translation: About whittaker and bessel functions and their products). *Nieuw Archief voor Wiskunde*, 18(2):10–29, 1936.

[76] Richard Beals and Jacek Szmigielski. Meijer g-functions: a gentle introduction. *Notices of the AMS*, 60(7):866–872, 2013.

[77] Patryk Orzechowski, William La Cava, and Jason H Moore. Where are we now?: a large benchmark study of recent symbolic regression methods. In *Proceedings of the Genetic and Evolutionary Computation Conference*, pages 1183–1190. ACM, 2018.

[78] Telmo Menezes and Camille Roth. Symbolic regression of generative network models. *Scientific reports*, 4:6284, 2014.

[79] Ekaterina J Vladislavleva, Guido F Smits, and Dick Den Hertog. Order of nonlinearity as a complexity measure for models generated by symbolic regression via pareto genetic programming. *IEEE Transactions on Evolutionary Computation*, 13(2):333–349, 2009.

[80] Timothy Y Chow. What is a closed-form number? *The American mathematical monthly*, 106(5):440–448, 1999.

[81] Jonathan M Borwein, Richard E Crandall, et al. Closed forms: what they are and why we care. *Notices of the AMS*, 60(1):50–65, 2013.

[82] Andrei Nikolaevich Kolmogorov. On the representation of continuous functions of many variables by superposition of continuous functions of one variable and addition. In *Doklady Akademii Nauk*, volume 114, pages 953–956. Russian Academy of Sciences, 1957.

[83] Kurt Hornik, Maxwell Stinchcombe, and Halbert White. Multilayer feedforward networks are universal approximators. *Neural networks*, 2(5):359–366, 1989.

[84] Vera Kurkova. Kolmogorov's theorem and multilayer neural networks. *Neural networks*, 5(3):501–506, 1992.

[85] Federico Girosi and Tomaso Poggio. Representation properties of networks: Kolmogorov's theorem is irrelevant. *Neural Computation*, 1(4):465–469, 1989.

[86] Boris Igelnik and Neel Parikh. Kolmogorov's spline network. *IEEE transactions on neural networks*, 14(4):725–733, 2003.

[87] George Cybenko. Approximation by superpositions of a sigmoidal function. *Mathematics of control, signals and systems*, 2(4):303–314, 1989.

[88] David A Sprecher. A universal mapping for kolmogorov's superposition theorem. *Neural Networks*, 6(8):1089–1094, 1993.

[89] Trevor J Hastie. Generalized additive models. In *Statistical models*, pages 249–307. 2017.

[90] I. Gradshteyn and I. Ryzhik. Table of integrals, series, and products. *Academic press*, 2014.

[91] Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Deep inside convolutional networks: Visualising image classification models and saliency maps. *arXiv preprint arXiv:1312.6034*, 2013.

[92] T Stephens. Gplearn model, genetic programming, 2015.

[93] Gang Luo, Bryan L Stone, Michael D Johnson, Peter Tarczy-Hornoch, Adam B Wilcox, Sean D Mooney, Xiaoming Sheng, Peter J Haug, and Flory L Nkoy. Automating construction of machine learning models with clinical big data: proposal rationale and methods. *JMIR research protocols*, 6(8), 2017.

[94] Jasper Snoek, Hugo Larochelle, and Ryan P Adams. Practical bayesian optimization of machine learning algorithms. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 2951–2959, 2012.

[95] Liuyi Yao, Sheng Li, Yaliang Li, Mengdi Huai, Jing Gao, and Aidong Zhang. Representation learning for treatment effect estimation from observational data. In *Advances in Neural Information Processing Systems*, pages 2634–2644, 2018.

[96] Ahmed Alaa and Mihaela van der Schaar. Limits of estimating heterogeneous treatment effects: Guidelines for practical algorithm design. In *International Conference on Machine Learning*, pages 129–138, 2018.

[97] Vincent Dorie, Jennifer Hill, Uri Shalit, Marc Scott, and Dan Cervone. Automated versus do-it-yourself methods for causal inference: Lessons learned from a data analysis competition. *arXiv preprint arXiv:1707.02641*, 2017.

[98] Elizabeth A Stuart, Eva DuGoff, Michael Abrams, David Salkever, and Donald Steinwachs. Estimating causal effects in observational studies using electronic health data: challenges and (some) solutions. *Egems*, 1(3), 2013.

[99] Frank R Hampel, Elvezio M Ronchetti, Peter J Rousseeuw, and Werner A Stahel. *Robust statistics: the approach based on influence functions*, volume 196. John Wiley & Sons, 2011.

[100] Avital Oliver, Augustus Odena, Colin Raffel, Ekin D Cubuk, and Ian J Goodfellow. Realistic evaluation of semi-supervised learning algorithms. *International Conference on Learning Representations (ICLR)*, 2018.

[101] Lars Kotthoff, Chris Thornton, Holger H Hoos, Frank Hutter, and Kevin Leyton-Brown. Auto-weka 2.0: Automatic model selection and hyperparameter optimization in weka. *Journal of Machine Learning Research*, 17:1–5, 2016.

[102] Matthias Feurer, Aaron Klein, Katharina Eggensperger, Jost Springenberg, Manuel Blum, and Frank Hutter. Efficient and robust automated machine learning. In *Advances in Neural Information Processing Systems (NIPS)*, pages 2962–2970, 2015.

[103] Randal S Olson and Jason H Moore. Tpot: A tree-based pipeline optimization tool for automating machine learning. In *ICML Workshop on Automatic Machine Learning*, pages 66–74, 2016.

[104] Ziyu Wang, Masrour Zoghi, Frank Hutter, David Matheson, Nando De Freitas, et al. Bayesian optimization in high dimensions via random embeddings. In *IJCAI*, pages 1778–1784, 2013.

[105] Frank Hutter, Holger H Hoos, and Kevin Leyton-Brown. Sequential model-based optimization for general algorithm configuration. *LION*, 5:507–523, 2011.

[106] Kevin Swersky, David Duvenaud, Jasper Snoek, Frank Hutter, and Michael A Osborne. Raiders of the lost architecture: Kernels for bayesian optimization in conditional parameter spaces. *arXiv preprint arXiv:1409.4011*, 2014.

[107] Rodolphe Jenatton, Cedric Archambeau, Javier González, and Matthias Seeger. Bayesian optimization with tree-structured dependencies. In *International Conference on Machine Learning*, pages 1655–1664, 2017.

[108] James Bergstra, Rémi Bardenet, B Kégl, and Y Bengio. Implementations of algorithms for hyper-parameter optimization. In *NIPS Workshop on Bayesian optimization*, page 29, 2011.

[109] Soko Setoguchi, Sebastian Schneeweiss, M Alan Brookhart, Robert J Glynn, and E Francis Cook. Evaluating uses of data mining techniques in propensity score estimation: a simulation study. *Pharmacoepidemiology and drug safety*, 17(6):546–555, 2008.

[110] Alejandro Schuler, Ken Jung, Robert Tibshirani, Trevor Hastie, and Nigam Shah. Synthvalidation: Selecting the best causal inference method for a given dataset. *arXiv preprint arXiv:1711.00083*, 2017.

[111] Craig A Rolling and Yuhong Yang. Model selection for estimating treatment effects. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 76(4):749–769, 2014.

[112] Mark J Van der Laan, MJ Laan, and James M Robins. *Unified methods for censored longitudinal data and causality*. Springer Science & Business Media, 2003.

[113] Alejandro Schuler, Michael Baiocchi, Robert Tibshirani, and Nigam Shah. A comparison of methods for model selection when estimating individual treatment effects. *arXiv preprint arXiv:1804.05146*, 2018.

[114] Larry Goldstein and Karen Messer. Optimal plug-in estimators for nonparametric functional estimation. *The annals of statistics*, pages 1306–1328, 1992.

[115] James M Robins, Lingling Li, Rajarshi Mukherjee, Eric Tchetgen Tchetgen, Aad van der Vaart, et al. Minimax estimation of a functional on a structured high-dimensional model. *The Annals of Statistics*, 45(5):1951–1987, 2017.

[116] Aad van der Vaart. Higher order tangent spaces and influence functions. *Statistical Science*, pages 679–686, 2014.

[117] Michael J Blaha. The critical importance of risk score calibration, 2016.

[118] László Györfi, Michael Kohler, Adam Krzyzak, and Harro Walk. *A distribution-free theory of nonparametric regression*. Springer Science & Business Media, 2006.

[119] Kirthevasan Kandasamy, Jeff Schneider, and Barnabás Póczos. High dimensional bayesian optimisation and bandits via additive models. In *International Conference on Machine Learning (ICML)*, pages 295–304, 2015.

[120] Daniel Golovin, Benjamin Solnik, Subhodeep Moitra, Greg Kochanski, John Karro, and D Sculley. Google vizier: A service for black-box optimization. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1487–1495. ACM, 2017.

[121] Frank Hutter, Holger H Hoos, Kevin Leyton-Brown, and Thomas Stützle. Paramils: an automatic algorithm configuration framework. *Journal of Artificial Intelligence Research*, 36(1):267–306, 2009.

[122] Zi Wang, Chengtao Li, Stefanie Jegelka, and Pushmeet Kohli. Batched high-dimensional bayesian optimization via structural kernel learning. *International Conference on Machine Learning (ICML)*, 2017.

[123] Chris J Maddison, Daniel Tarlow, and Tom Minka. A* sampling. In *Advances in Neural Information Processing Systems*, pages 3086–3094, 2014.

[124] Tarun Kathuria, Amit Deshpande, and Pushmeet Kohli. Batched gaussian process bandit optimization via determinantal point processes. In *Advances in Neural Information Processing Systems*, pages 4206–4214, 2016.

[125] Aleksandar Nikolov. Randomized rounding for the largest simplex problem. In *Proceedings of the forty-seventh annual ACM symposium on Theory of computing*, pages 861–870. ACM, 2015.

[126] Donald B Rubin. Estimating causal effects of treatments in randomized and nonrandomized studies. *Journal of educational Psychology*, 66(5):688, 1974.

[127] Judea Pearl. *Causality*. Cambridge university press, 2009.

[128] Daniel B Wright, Kamala London, and Andy P Field. Using bootstrap estimation and the plug-in principle for clinical psychology data. *Journal of Experimental Psychopathology*, 2(2):jep–013611, 2011.

[129] Jessica M Franklin, Sebastian Schneeweiss, Jennifer M Polinski, and Jeremy A Rassen. Plasmode simulation for the evaluation of pharmacoepidemiologic methods in complex healthcare databases. *Computational statistics & data analysis*, 72:219–226, 2014.

[130] Luisa Turrin Fernholz. *Von Mises calculus for statistical functionals*, volume 19. Springer Science & Business Media, 2012.

[131] Nicholas T Longford. A fast scoring algorithm for maximum likelihood estimation in unbalanced mixed models with nested random effects. *Biometrika*, 74(4):817–827, 1987.

[132] Ayanendranath Basu, Ian R Harris, Nils L Hjort, and MC Jones. Robust and efficient estimation by minimising a density power divergence. *Biometrika*, 85(3):549–559, 1998.

[133] Donald B Rubin. On the limitations of comparative effectiveness research. *Statistics in medicine*, 29(19):1991–1995, 2010.

[134] Antonio R Linero and Yun Yang. Bayesian regression tree ensembles that adapt to smoothness and sparsity. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 80(5):1087–1110, 2018.

[135] Jennifer L Hill. 2016 atlantic causal inference conference competition: Is your satt where it's at?, 2016.

[136] Kenneth R Niswander. The collaborative perinatal study of the national institute of neurological diseases and stroke. *The Woman and Their Pregnancies*, 1972.

[137] Mary Ann Sevick, Jeanette M Trauth, Bruce S Ling, Roger T Anderson, Gretchen A Piatt, Amy M Kilbourne, and Robert M Goodman. Patients with complex chronic diseases: perspectives on supporting self-management. *Journal of general internal medicine*, 22(3):438–444, 2007.

[138] David Blumenthal and Marilyn Tavenner. The meaningful use regulation for electronic health records. *New England Journal of Medicine*, 363(6):501–504, 2010.

[139] Eric J Topol. High-performance medicine: the convergence of human and artificial intelligence. *Nature medicine*, 25(1):44, 2019.

[140] DW Coyne. Management of chronic kidney disease comorbidities. *CKD medscape CME expert column series*, (3), 2011.

[141] Jose M Valderas, Barbara Starfield, Bonnie Sibbald, Chris Salisbury, and Martin Roland. Defining comorbidity: implications for understanding health and health services. *The Annals of Family Medicine*, 7(4):357–363, 2009.

[142] Ahmed M Alaa and Mihaela van der Schaar. A hidden absorbing semi-markov model for informatively censored temporal data: Learning and inference. *Journal of Machine Learning Research*, 2018.

[143] Ilya Sutskever, Oriol Vinyals, and Quoc V Le. Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*, pages 3104–3112, 2014.

[144] Yu-Ying Liu, Shuang Li, Fuxin Li, Le Song, and James M Rehg. Efficient learning of continuous-time hidden markov models for disease progression. In *Advances in neural information processing systems*, pages 3600–3608, 2015.

[145] Hanjun Dai, Bo Dai, Yan-Ming Zhang, Shuang Li, and Le Song. Recurrent hidden semi-markov model. *International Conference on Learning Representations*, 2016.

[146] Xun Zheng, Manzil Zaheer, Amr Ahmed, Yuan Wang, Eric P Xing, and Alexander J Smola. State space lstm models with particle mcmc inference. *arXiv preprint arXiv:1711.11179*, 2017.

[147] Edward Choi, Mohammad Taha Bahadori, Andy Schuetz, Walter F Stewart, and Jimeng Sun. Doctor ai: Predicting clinical events via recurrent neural networks. In *Machine Learning for Healthcare Conference*, pages 301–318, 2016.

[148] Zachary C Lipton, David C Kale, Charles Elkan, and Randall Wetzel. Learning to diagnose with lstm recurrent neural networks. *International Conference on Learning Representations*, 2016.

[149] Bryan Lim and Mihaela van der Schaar. Disease-atlas: Navigating disease trajectories with deep learning. *Machine Learning for Healthcare Conference (MLHC)*, 2018.

[150] Edward Choi, Mohammad Taha Bahadori, Jimeng Sun, Joshua Kulas, Andy Schuetz, and Walter Stewart. Retain: An interpretable predictive model for healthcare using reverse time attention mechanism. In *Advances in Neural Information Processing Systems*, pages 3504–3512, 2016.

[151] Fenglong Ma, Radha Chitta, Jing Zhou, Quanzeng You, Tong Sun, and Jing Gao. Dipole: Diagnosis prediction in healthcare via attention-based bidirectional recurrent neural networks. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1903–1911. ACM, 2017.

[152] Bum Chul Kwon, Min-Je Choi, Joanne Taery Kim, Edward Choi, Young Bin Kim, Soonwook Kwon, Jimeng Sun, and Jaegul Choo. Retainvis: Visual analytics with interpretable and interactive recurrent neural networks on electronic medical records. *IEEE transactions on visualization and computer graphics*, 25(1):299–309, 2019.

[153] Xiang Wang, David Sontag, and Fei Wang. Unsupervised learning of disease progression models. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 85–94. ACM, 2014.

[154] Ahmed M Alaa, Scott Hu, and Mihaela van der Schaar. Learning from clinical judgments: Semi-markov-modulated marked hawkes processes for risk prognosis. *International Conference on Machine Learning*, 2017.

[155] Rahul G Krishnan, Uri Shalit, and David Sontag. Structured inference networks for nonlinear state space models. In *AAAI*, pages 2101–2109, 2017.

[156] Maximilian Karl, Maximilian Soelch, Justin Bayer, and Patrick van der Smagt. Deep variational bayes filters: Unsupervised learning of state space models from raw data. *arXiv preprint arXiv:1605.06432*, 2016.

[157] Matthew Johnson, David K Duvenaud, Alex Wiltschko, Ryan P Adams, and Sandeep R Datta. Composing graphical models with neural networks for structured representations and fast inference. In *Advances in neural information processing systems*, pages 2946–2954, 2016.

[158] Syama Sundar Rangapuram, Matthias W Seeger, Jan Gasthaus, Lorenzo Stella, Yuyang Wang, and Tim Januschowski. Deep state space models for time series forecasting. In *Advances in Neural Information Processing Systems*, pages 7796–7805, 2018.

[159] Junyoung Chung, Kyle Kastner, Laurent Dinh, Kratarth Goel, Aaron C Courville, and Yoshua Bengio. A recurrent latent variable model for sequential data. In *Advances in neural information processing systems*, pages 2980–2988, 2015.

[160] Marco Fraccaro, Søren Kaae Sønderby, Ulrich Paquet, and Ole Winther. Sequential neural models with stochastic layers. In *Advances in neural information processing systems*, pages 2199–2207, 2016.

[161] Justin Bayer and Christian Osendorfer. Learning stochastic recurrent networks. *arXiv preprint arXiv:1411.7610*, 2014.

[162] Allison A Eddy and Eric G Neilson. Chronic kidney disease progression. *Journal of the American Society of Nephrology*, 17(11):2964–2966, 2006.

[163] Frans MJ Willems, Yuri M Shtarkov, and Tjalling J Tjalkens. The context-tree weighting method: basic properties. *IEEE Transactions on Information Theory*, 41(3):653–664, 1995.

[164] Ron Begleiter, Ran El-Yaniv, and Golan Yona. On prediction using variable order markov models. *Journal of Artificial Intelligence Research*, 22:385–421, 2004.

[165] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. *International Conference on Learning Representations*, 2015.

[166] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, pages 6000–6010, 2017.

[167] Peter J Green and Sylvia Richardson. Hidden markov models and disease mapping. *Journal of the American statistical association*, 97(460):1055–1070, 2002.

[168] Andriy Mnih and Karol Gregor. Neural variational inference and learning in belief networks. *arXiv preprint arXiv:1402.0030*, 2014.

[169] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *International Conference on Learning Representations*, 2014.

[170] John Schulman, Nicolas Heess, Theophane Weber, and Pieter Abbeel. Gradient estimation using stochastic computation graphs. In *Advances in Neural Information Processing Systems*, pages 3528–3536, 2015.

[171] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[172] Rajesh Ranganath, Sean Gerrish, and David Blei. Black box variational inference. In *Artificial Intelligence and Statistics*, pages 814–822, 2014.

[173] Rhonda D Szczesniak, Dan Li, Weiji Su, Cole Brokamp, John Pestian, Michael Seid, and John P Clancy. Phenotypes of rapid cystic fibrosis lung disease progression during adolescence and young adulthood. *American journal of respiratory and critical care medicine*, 196(4):471–478, 2017.

[174] Don B Sanders, Lucas R Hoffman, Julia Emerson, Ronald L Gibson, Margaret Rosenfeld, Gregory J Redding, and Christopher H Goss. Return of fev1 after pulmonary exacerbation in children with cystic fibrosis. *Pediatric pulmonology*, 45(2):127–134, 2010.

[175] Andrew T Braun and Christian A Merlo. Cystic fibrosis lung transplantation. *Current opinion in pulmonary medicine*, 17(6):467–472, 2011.

[176] Amanda I Adler, Brian SF Shine, Parinya Chamnan, Charles S Haworth, and Diana Bilton. Genetic determinants and epidemiology of cystic fibrosis–related diabetes. *Diabetes care*, 31(9):1789–1794, 2008.

[177] Pascale Fanen, Adeline Wohlhuter-Haddad, and Alexandre Hinzpeter. Genetics of cystic fibrosis: Cftr mutation classifications toward genotype-based cf therapies. *The international journal of biochemistry & cell biology*, 52:94–102, 2014.

[178] Peter J Mogayzel Jr, Edward T Naureckas, Karen A Robinson, Cynthia Brady, Margaret Guill, Thomas Lahiri, Lisa Lubsch, Jane Matsui, Christopher M Oermann, Felix Ratjen, et al. Cystic fibrosis foundation pulmonary guideline*. pharmacologic approaches to prevention and eradication of initial pseudomonas aeruginosa infection. *Annals of the American Thoracic Society*, 11(10):1640–1650, 2014.

[179] Steven M Rowe, Drucy S Borowitz, Jane L Burns, John P Clancy, Scott H Donaldson, George Retsch-Bogart, Scott D Sagel, and Bonnie W Ramsey. Progress in cystic fibrosis and the cf therapeutics development network. *Thorax*, 67(10):882–890, 2012.

[180] Todd MacKenzie, Alex H Gifford, Kathryn A Sabadosa, Hebe B Quinton, Emily A Knapp, Christopher H Goss, and Bruce C Marshall. Longevity of patients with cystic fibrosis in 2000 to 2010 and beyond: Survival analysis of the cystic fibrosis foundation patient registrylifetime of patients with cystic fibrosis in 2000 to 2010 and beyond. *Annals of internal medicine*, 161(4):233–241, 2014.

[181] Patrick A Flume. Cystic fibrosis: when to consider lung transplantation? *Chest*, 113(5):1159–1162, 1998.

[182] Theodore G Liou, Frederick R Adler, Barbara C Cahill, Stacey C FitzSimmons, David Huang, Jonathan R Hibbs, and Bruce C Marshall. Survival effect of lung transplantation among patients with cystic fibrosis. *Jama*, 286(21):2683–2689, 2001.

[183] Markus Hofer, Christian Benden, Ilhan Inci, Christoph Schmid, Sarosh Irani, Rudolf Speich, Walter Weder, and Annette Boehler. True survival benefit of lung transplantation for cystic fibrosis patients: the zurich experience. *The Journal of Heart and Lung Transplantation*, 28(4):334–339, 2009.

[184] Nicole Mayer-Hamblett, Margaret Rosenfeld, Julia Emerson, Christopher H Goss, and Moira L Aitken. Developing cystic fibrosis lung transplant referral criteria using predictors of 2-year mortality. *American journal of respiratory and critical care medicine*, 166(12):1550–1555, 2002.

[185] Theodore G Liou, Frederick R Adler, and David Huang. Use of lung transplantation survival models to refine patient selection in cystic fibrosis. *American journal of respiratory and critical care medicine*, 171(9):1053–1059, 2005.

[186] David Weill, Christian Benden, Paul A Corris, John H Dark, R Duane Davis, Shaf Keshavjee, David J Lederer, Michael J Mulligan, G Alexander Patterson, Lianne G Singer, et al. A consensus document for the selection of lung transplant candidates: 2014an update from the pulmonary transplantation council of the international society for heart and lung transplantation, 2015.

[187] Jaime L Hook and David J Lederer. Selecting lung transplant candidates: where do current guidelines fall short? *Expert review of respiratory medicine*, 6(1):51–61, 2012.

[188] Tim Oliver Hirche, Christiane Knoop, H Hebestreit, Dee Shimmin, Amparo Solé, Joseph Stuart Elborn, Helmut Ellemunter, Paul Aurora, Michael Hogardt, Thomas Otto Friedrich Wagner, et al. Practical guidelines: lung transplantation in patients with cystic fibrosis. *Pulmonary medicine*, 2014, 2014.

[189] Eitan Kerem, Joseph Reisman, Mary Corey, Gerard J Canny, and Henry Levison. Prediction of mortality in patients with cystic fibrosis. *New England Journal of Medicine*, 326(18):1187–1191, 1992.

[190] Carlos E Milla and Warren J Warwick. Risk of death in cystic fibrosis patients with severely compromised lung function. *Chest*, 113(5):1230–1234, 1998.

[191] Gabriella Wojewodka, Juan B De Sanctis, Joanie Bernier, Julie Bérubé, Heather G Ahlgren, Jim Gruber, Jennifer Landry, Larry C Lands, Dao Nguyen, Simon Rousseau, et al. Candidate markers associated with the probability of future pulmonary exacerbations in cystic fibrosis patients. *PloS one*, 9(2):e88567, 2014.

[192] K Ramos, B Quon, S Heltshe, N Mayer-Hamblett, E Lease, M Aitken, N Weiss, and C Goss. Heterogeneity in survival among adult cystic fibrosis patients with fev1$<$30$\%$ of predicted in the united states. *CHEST*, 2017.

[193] Donald S Urquhart, Lena P Thia, Jackie Francis, S Ammani Prasad, Charlie Dawson, Colin Wallis, and Ian M Balfour-Lynn. Deaths in childhood from cystic fibrosis: 10-year analysis from two london specialist centres. *Archives of disease in childhood*, 98(2):123–127, 2013.

[194] Anne L Stephenson, Sanja Stanojevic, Jenna Sykes, and Pierre-Regis Burgel. The changing epidemiology and demography of cystic fibrosis. *La Presse Médicale*, 2017.

[195] Karen M Hayllar, SG Williams, Amelia E Wise, Shideh Pouria, Martin Lombard, Margaret E Hodson, and David Westaby. A prognostic model for the prediction of survival in cystic fibrosis. *Thorax*, 52(4):313–317, 1997.

[196] Theodore G Liou, Frederick R Adler, Stacey C FitzSimmons, Barbara C Cahill, Jonathan R Hibbs, and Bruce C Marshall. Predictive 5-year survivorship model of cystic fibrosis. *American journal of epidemiology*, 153(4):345–352, 2001.

[197] Roberto Buzzetti, Gianfranco Alicandro, Laura Minicucci, Sara Notarnicola, Maria Lucia Furnari, Gabriella Giordano, Vincenzina Lucidi, Enza Montemitro, Valeria Raia, Giuseppe Magazzù, et al. Validation of a predictive survival model in italian patients with cystic fibrosis. *Journal of Cystic Fibrosis*, 11(1):24–29, 2012.

[198] Shawn D Aaron, Anne L Stephenson, Donald W Cameron, and George A Whitmore. A statistical model to predict one-year risk of death in patients with cystic fibrosis. *Journal of clinical epidemiology*, 68(11):1336–1345, 2015.

[199] Cystic fibrosis trust, https://www.cysticfibrosis.org.uk/the-work-we-do/uk-cf-registry. *(accessed Oct 1, 2017)*.

[200] Cystic fibrosis trust, https://www.cysticfibrosis.org.uk/the-work-we-do/uk-cf-registry/reporting-and-resources. *(accessed Oct 2, 2017)*.

[201] Aliza K Fink, Deena R Loeffler, Bruce C Marshall, Christopher H Goss, and Wayne J Morgan. Data that empower: The success and promise of cf patient registries. *Pediatric Pulmonology*, 2017.

[202] L Nkam, J Lambert, A Latouche, G Bellis, PR Burgel, and MN Hocine. A 3-year prognostic score for adults with cystic fibrosis. *Journal of Cystic Fibrosis*, 2017.

[203] Daniel J Stekhoven and Peter Bühlmann. Missforestnon-parametric missing value imputation for mixed-type data. *Bioinformatics*, 28(1):112–118, 2011.

[204] Cormac McCarthy, Borislav D Dimitrov, Imran J Meurling, Cedric Gunaratnam, and Noel G McElvaney. The cf-able score: a novel clinical prediction rule for prognosis in patients with cystic fibrosis. *CHEST Journal*, 143(5):1358–1364, 2013.

[205] Borislav D Dimitrov and Mohd Hafiz Hj Jaidi. Cf-able-uk score: Modification and validation of a clinical prediction rule for prognosis in cystic fibrosis on data from uk cf registry, 2015.

[206] Jonathan B Orens, Marc Estenne, Selim Arcasoy, John V Conte, Paul Corris, Jim J Egan, Thomas Egan, Shaf Keshavjee, Christiane Knoop, Robert Kotloff, et al. International guidelines for the selection of lung transplant candidates: 2006 updatea consensus report from the pulmonary scientific council of the international society for heart and lung transplantation. *The Journal of heart and lung transplantation*, 25(7):745–755, 2006.

[207] Andrew T Braun, Elliott C Dasenbrook, Ashish S Shah, Jonathan B Orens, and Christian A Merlo. Impact of lung allocation score on survival in cystic fibrosis lung transplant recipients. *The Journal of Heart and Lung Transplantation*, 34(11):1436–1441, 2015.

[208] John A Swets et al. Measuring the accuracy of diagnostic systems. *Science*, 240(4857):1285–1293, 1988.

[209] Takaya Saito and Marc Rehmsmeier. The precision-recall plot is more informative than the roc plot when evaluating binary classifiers on imbalanced datasets. *PloS one*, 10(3):e0118432, 2015.

[210] Ronen Fluss, David Faraggi, and Benjamin Reiser. Estimation of the youden index and its associated cutoff point. *Biometrical journal*, 47(4):458–472, 2005.

[211] Douglas G Altman and J Martin Bland. Diagnostic tests. 1: Sensitivity and specificity. *BMJ: British Medical Journal*, 308(6943):1552, 1994.

[212] Jesse Davis and Mark Goadrich. The relationship between precision-recall and roc curves. In *Proceedings of the 23rd international conference on Machine learning*, pages 233–240. ACM, 2006.

[213] Mu Zhu. Recall, precision and average precision. *Department of Statistics and Actuarial Science, University of Waterloo, Waterloo*, 2:30, 2004.

[214] Peter Flach and Meelis Kull. Precision-recall-gain curves: Pr analysis done right. In *Advances in Neural Information Processing Systems*, pages 838–846, 2015.

[215] Steven M Rowe, Sonya L Heltshe, Tanja Gonska, Scott H Donaldson, Drucy Borowitz, Daniel Gelfond, Scott D Sagel, Umer Khan, Nicole Mayer-Hamblett, Jill M Van Dalfsen, et al. Clinical mechanism of the cystic fibrosis transmembrane conductance regulator potentiator ivacaftor in g551d-mediated cystic fibrosis. *American journal of respiratory and critical care medicine*, 190(2):175–184, 2014.

[216] Claire E Wainwright, J Stuart Elborn, Bonnie W Ramsey, Gautham Marigowda, Xiaohong Huang, Marco Cipolli, Carla Colombo, Jane C Davies, Kris De Boeck, Patrick A Flume, et al. Lumacaftor–ivacaftor in patients with cystic fibrosis homozygous for phe508del cftr. *New England Journal of Medicine*, 373(3):220–231, 2015.

[217] Parinya Chamnan, Brian SF Shine, Charles S Haworth, Diana Bilton, and Amanda I Adler. Diabetes as a determinant of mortality in cystic fibrosis. *Diabetes Care*, 33(2):311–316, 2010.

[218] Iven H Young and Peter TP Bye. Gas exchange in disease: asthma, chronic obstructive pulmonary disease, cystic fibrosis, and interstitial lung disease. *Comprehensive Physiology*, 2011.

[219] AE Ewence, S Malone, A Nutbourne, A Higton, and C Orchard. 302 a retrospective review of renal function and intravenous (iv) antibiotic use in an adult uk cystic fibrosis centre. *Journal of Cystic Fibrosis*, 16:S139, 2017.

[220] Hassan S Sheikh, Noel Dexter Tiangco, Christopher Harrell, and Robert L Vender. Severe hypercapnia in critically ill adult cystic fibrosis patients. *Journal of clinical medicine research*, 3(5):209, 2011.

[221] Ramos KJ, Quon BS, Heltshe SL, Mayer-Hamblett N, Lease ED, Aitken ML, Weiss NS, and Goss CH. Heterogeneity in survival amongadult cystic fibrosis patients with fev1$<$30$\%$ of predicted in the united states. *CHEST*, 151(6):1320–1328, 2017.

[222] Mark R Thomas and Gregory YH Lip. Novel risk markers and risk assessments for cardiovascular disease. *Circulation research*, 120(1):133–149, 2017.

[223] Paul M Ridker, Eleanor Danielson, FA Fonseca, Jacques Genest, Antonio M Gotto Jr, JJ Kastelein, Wolfgang Koenig, Peter Libby, Alberto J Lorenzatti, Jean G MacFadyen, et al. Rosuvastatin to prevent vascular events in men and women with elevated c-reactive protein. *New England Journal of Medicine*, 359(21):2195, 2008.

[224] Hilal Maradit Kremers, Cynthia S Crowson, Terry M Therneau, Veronique L Roger, and Sherine E Gabriel. High ten-year risk of cardiovascular disease in newly diagnosed rheumatoid arthritis patients: A population-based cohort study. *Arthritis & Rheumatology*, 58(8):2268–2274, 2008.

[225] Ralph B DAgostino, Ramachandran S Vasan, Michael J Pencina, Philip A Wolf, Mark Cobain, Joseph M Massaro, and William B Kannel. General cardiovascular risk profile for use in primary care: the framingham heart study. *Circulation*, 117(6):743–753, 2008.

[226] RM Conroy, K Pyörälä, AP el Fitzgerald, S Sans, A Menotti, Gui De Backer, Dirk De Bacquer, P Ducimetiere, P Jousilahti, U Keil, et al. Estimation of ten-year risk of fatal cardiovascular disease in europe: the score project. *European heart journal*, 24(11):987–1003, 2003.

[227] Lars Sjöström, Anna-Karin Lindroos, Markku Peltonen, Jarl Torgerson, Claude Bouchard, Björn Carlsson, Sven Dahlgren, Bo Larsson, Kristina Narbro, Carl David Sjöström, et al. Lifestyle, diabetes, and cardiovascular risk factors 10 years after bariatric surgery. *New England Journal of Medicine*, 351(26):2683–2693, 2004.

[228] Philip Greenland, Joseph S Alpert, George A Beller, Emelia J Benjamin, Matthew J Budoff, Zahi A Fayad, Elyse Foster, Mark A Hlatky, John McB Hodgson, Frederick G Kushner, et al. 2010 accf/aha guideline for assessment of cardiovascular risk in asymptomatic adults: a report of the american college of cardiology foundation/american heart association task force on practice guidelines developed in collaboration with the american society of echocardiography, american society of nuclear cardiology, society of atherosclerosis imaging and prevention, society for cardiovascular angiography and interventions, society of cardiovascular computed tomography, and society for cardiovascular magnetic resonance. *Journal of the American College of Cardiology*, 56(25):e50–e103, 2010.

[229] Massimo F Piepoli, Arno W Hoes, Stefan Agewall, Christian Albus, Carlos Brotons, Alberico L Catapano, Marie-Therese Cooney, Ugo Corrà, Bernard Cosyns, Christi Deaton, et al. 2016 european guidelines on cardiovascular disease prevention in clinical practice: The sixth joint task force of the european society of cardiology and other societies on cardiovascular disease prevention in clinical practice (constituted by representatives of 10 societies and by invited experts) developed with the special contribution of the european association for cardiovascular prevention & rehabilitation (eacpr). *Atherosclerosis*, 252:207–274, 2016.

[230] Julia Hippisley-Cox, Carol Coupland, Yana Vinogradova, John Robson, Rubin Minhas, Aziz Sheikh, and Peter Brindle. Predicting cardiovascular risk in england and wales: prospective derivation and validation of qrisk2. *Bmj*, 336(7659):1475–1482, 2008.

[231] Julia Hippisley-Cox, Carol Coupland, and Peter Brindle. Development and validation of qrisk3 risk prediction algorithms to estimate future risk of cardiovascular disease: prospective cohort study. *bmj*, 357:j2099, 2017.

[232] George CM Siontis, Ioanna Tzoulaki, Konstantinos C Siontis, and John PA Ioannidis. Comparisons of established risk prediction models for cardiovascular disease: systematic review. *Bmj*, 344:e3318, 2012.

[233] Ruth L Coleman, Richard J Stevens, Ravi Retnakaran, and Rury R Holman. Framingham, score, and decode risk equations do not provide reliable cardiovascular risk estimates in type 2 diabetes. *Diabetes care*, 30(5):1292–1293, 2007.

[234] P McEwan, JE Williams, JD Griffiths, A Bagust, JR Peters, P Hopkinson, and CJ Currie. Evaluating the performance of the framingham risk equations in a population with diabetes. *Diabetic medicine*, 21(4):318–323, 2004.

[235] Iciar Martín-Timón, Cristina Sevillano-Collantes, Amparo Segura-Galindo, and Francisco Javier del Cañizo-Gómez. Type 2 diabetes and cardiovascular disease: have all risk factors the same strength? *World journal of diabetes*, 5(4):444, 2014.

[236] John B Buse, Henry N Ginsberg, George L Bakris, Nathaniel G Clark, Fernando Costa, Robert Eckel, Vivian Fonseca, Hertzel C Gerstein, Scott Grundy, Richard W Nesto, et al. Primary prevention of cardiovascular diseases in people with diabetes mellitus: a scientific statement from the american heart association and the american diabetes association. *Circulation*, 115(1):114–126, 2007.

[237] Bharath Ambale-Venkatesh, Colin O Wu, Kiang Liu, WG Hundley, Robyn L McClelland, Antoinette S Gomes, Aaron R Folsom, Steven Shea, Eliseo Guallar, David A Bluemke, et al. Cardiovascular event prediction by machine learning: the multi-ethnic study of atherosclerosis. *Circulation research*, pages CIRCRESAHA–117, 2017.

[238] Tariq Ahmad, Lars H Lund, Pooja Rao, Rohit Ghosh, Prashant Warier, Benjamin Vaccaro, Ulf Dahlström, Christopher M O'Connor, G Michael Felker, and Nihar R Desai. Machine learning methods improve prognostication, identify clinically distinct phenotypes, and detect heterogeneity in response to therapy in a large cohort of heart failure patients. *Journal of the American Heart Association*, 7(8):e008081, 2018.

[239] Cathie Sudlow, John Gallacher, Naomi Allen, Valerie Beral, Paul Burton, John Danesh, Paul Downey, Paul Elliott, Jane Green, Martin Landray, et al. Uk biobank: an open access resource for identifying the causes of a wide range of complex diseases of middle and old age. *PLoS medicine*, 12(3):e1001779, 2015.

[240] Lyle J Palmer. Uk biobank: bank on it. *The Lancet*, 369(9578):1980–1982, 2007.

[241] Andrea Ganna and Erik Ingelsson. 5 year mortality predictors in 498 103 uk biobank participants: a prospective population-based study. *The Lancet*, 386(9993):533–540, 2015.

[242] Ligia Adamska, Naomi Allen, Robin Flaig, Cathie Sudlow, Michael Lay, and Martin Landray. Challenges of linking to routine healthcare records in uk biobank. *Trials*, 16(2):O68, 2015.

[243] David C Goff, Donald M Lloyd-Jones, Glen Bennett, Sean Coady, Ralph B DAgostino, Raymond Gibbons, Philip Greenland, Daniel T Lackland, Daniel Levy, Christopher J ODonnell, et al. 2013 acc/aha guideline on the assessment of cardiovascular risk: a report of the american college of cardiology/american heart association task force on practice guidelines. *Journal of the American College of Cardiology*, 63(25 Part B):2935–2959, 2014.

[244] Robert Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 267–288, 1996.

[245] Tianqi Chen and Carlos Guestrin. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, pages 785–794. ACM, 2016.

[246] Carolin Strobl, Anne-Laure Boulesteix, Thomas Kneib, Thomas Augustin, and Achim Zeileis. Conditional variable importance for random forests. *BMC bioinformatics*, 9(1):307, 2008.

[247] Ahmed M Alaa and Mihaela van der Schaar. Autoprognosis: Automated clinical prognostic modeling via bayesian optimization with structured kernel learning. *arXiv preprint arXiv:1802.07207*, 2018.

[248] Matthew A Allison, William R Hiatt, Alan T Hirsch, Joseph R Coll, and Michael H Criqui. A high ankle-brachial index is associated with increased cardiovascular disease morbidity and lower quality of life. *Journal of the American College of Cardiology*, 51(13):1292–1298, 2008.

[249] Bradford W Hesse, David E Nelson, Gary L Kreps, Robert T Croyle, Neeraj K Arora, Barbara K Rimer, and Kasisomayajula Viswanath. Trust and sources of health information: the impact of the internet and its implications for health care providers: findings from the first health information national trends survey. *Archives of internal medicine*, 165(22):2618–2624, 2005.

[250] Thomas A Gaziano, Cynthia R Young, Garrett Fitzmaurice, Sidney Atwood, and J Michael Gaziano. Laboratory-based versus non-laboratory-based method for assessment of cardiovascular disease risk: the nhanes i follow-up study cohort. *The Lancet*, 371(9616):923–931, 2008.

[251] Shanthi Mendis, Lars H Lindholm, Giuseppe Mancia, Judith Whitworth, Michael Alderman, Stephen Lim, and Tony Heagerty. World health organization (who) and international society of hypertension (ish) risk prediction charts: assessment of cardiovascular risk for prevention and control of cardiovascular disease in low and middle-income countries. *Journal of hypertension*, 25(8):1578–1582, 2007.

[252] Gerd Assmann, Paul Cullen, and Helmut Schulte. Simple scoring scheme for calculating the risk of acute coronary events based on the 10-year follow-up of the prospective cardiovascular münster (procam) study. *Circulation*, 105(3):310–315, 2002.

[253] K Eeg-Olofsson, Jan Cederholm, PM Nilsson, Björn Zethelius, A-M Svensson, S Gudbjörnsdóttir, and B Eliasson. New aspects of hba1c as a risk factor for cardiovascular diseases in type 2 diabetes: an observational study from the swedish national diabetes register (ndr). *Journal of internal medicine*, 268(5):471–482, 2010.

[254] Emerging Risk Factors Collaboration. C-reactive protein, fibrinogen, and cardiovascular disease prediction. *New England Journal of Medicine*, 367(14):1310–1320, 2012.

[255] Peter Willeit, Stephen Kaptoge, Paul Welsh, Adam S Butterworth, Rajiv Chowdhury, Sarah A Spackman, Lisa Pennells, Pei Gao, Stephen Burgess, Daniel F Freitag, et al. Natriuretic peptides and integrated risk assessment for cardiovascular disease: an individual-participant-data meta-analysis. *The Lancet Diabetes & Endocrinology*, 4(10):840–849, 2016.

[256] Christina Fitzmaurice, Christine Allen, Ryan M Barber, Lars Barregard, Zulfiqar A Bhutta, Hermann Brenner, Daniel J Dicker, Odgerel Chimed-Orchir, Rakhi Dandona, Lalit Dandona, et al. Global, regional, and national cancer incidence, mortality, years of life lost, years lived with disability, and disability-adjusted life-years for 32 cancer groups, 1990

to 2015: a systematic analysis for the global burden of disease study. *JAMA oncology*, 3(4):524–548, 2017.

[257] Freddie Bray, Jacques Ferlay, Isabelle Soerjomataram, Rebecca L Siegel, Lindsey A Torre, and Ahmedin Jemal. Global cancer statistics 2018: Globocan estimates of incidence and mortality worldwide for 36 cancers in 185 countries. *CA: a cancer journal for clinicians*, 68(6):394–424, 2018.

[258] Fangjian Guo, Yong-fang Kuo, Ya Chen Tina Shih, Sharon H Giordano, and Abbey B Berenson. Trends in breast cancer mortality by stage at diagnosis among young women in the u nited s tates. *Cancer*, 124(17):3500–3509, 2018.

[259] Joseph A Sparano, Robert J Gray, Peter M Ravdin, Della F Makower, Kathleen I Pritchard, Kathy S Albain, Daniel F Hayes, Jr CE Geyer, Elizabeth C Dees, Matthew P Goetz, et al. Clinical and genomic risk to guide the use of adjuvant therapy for breast cancer. *The New England journal of medicine*, 2019.

[260] W Fraser Symmans, Florentia Peintinger, Christos Hatzis, Radhika Rajan, Henry Kuerer, Vicente Valero, Lina Assad, Anna Poniecka, Bryan Hennessy, Marjorie Green, et al. Measurement of residual breast cancer burden to predict survival after neoadjuvant chemotherapy. *Journal of Clinical Oncology*, 25(28):4414–4422, 2007.

[261] Peter M Ravdin, Laura A Siminoff, Greg J Davis, Mary Beth Mercer, Joan Hewlett, Nancy Gerson, and Helen L Parker. Computer program to assist in making decisions about adjuvant therapy for women with early breast cancer. *Journal of clinical oncology*, 19(4):980–991, 2001.

[262] Gordon C Wishart, Elizabeth M Azzato, David C Greenberg, Jem Rashbass, Olive Kearins, Gill Lawrence, Carlos Caldas, and Paul DP Pharoah. Predict: a new uk prognostic model that predicts survival following surgery for invasive breast cancer. *Breast Cancer Research*, 12(1):R1, 2010.

[263] Francisco J Candido dos Reis, Gordon C Wishart, Ed M Dicks, David Greenberg, Jem Rashbass, Marjanka K Schmidt, Alexandra J van den Broek, Ian O Ellis, Andrew Green, Emad Rakha, et al. An updated predict breast cancer prognostication and treatment benefit prediction model with independent validation. *Breast Cancer Research*, 19(1):58, 2017.

[264] Shlomit Strulov Shachar and Hyman B Muss. Internet tools to enhance breast cancer care. *NPJ Breast Cancer*, 2:16011, 2016.

[265] Marissa C van Maaren, Cornelia D van Steenbeek, Paul DP Pharoah, Annemieke Witteveen, Gabe S Sonke, Luc JA Strobbe, Philip MP Poortmans, and Sabine Siesling. Validation of the online prediction tool predict v. 2.0 in the dutch breast cancer population. *European journal of cancer*, 86:364–372, 2017.

[266] Ivo A Olivotto, Chris D Bajdik, Peter M Ravdin, Caroline H Speers, Andrew J Coldman, Brian D Norris, Greg J Davis, Stephen K Chia, and Karen A Gelmon. Population-based validation of the prognostic model adjuvant! for early breast cancer. *Journal of Clinical Oncology*, 23(12):2716–2725, 2005.

[267] Nirmala Bhoo-Pathy, Cheng-Har Yip, Mikael Hartman, Nakul Saxena, Nur Aishah Taib, Gwo-Fuang Ho, Lai-Meng Looi, Awang M Bulgiba, Yolanda van der Graaf, and Helena M Verkooijen. Adjuvant! online is overoptimistic in predicting survival of asian breast cancer patients. *European Journal of Cancer*, 48(7):982–989, 2012.

[268] HE Campbell, MA Taylor, AL Harris, and AM Gray. An investigation into the performance of the adjuvant! online prognostic programme in early breast cancer for a cohort of patients in the united kingdom. *British journal of cancer*, 101(7):1074, 2009.

[269] Hui Miao, Mikael Hartman, Helena M Verkooijen, Nur Aishah Taib, Hoong-Seam Wong, Shridevi Subramaniam, Cheng-Har Yip, Ern-Yu Tan, Patrick Chan, Soo-Chin Lee, et al. Validation of the cancermath prognostic tool for breast cancer in southeast asia. *BMC cancer*, 16(1):820, 2016.

[270] Ziad Obermeyer and Ezekiel J Emanuel. Predicting the futurebig data, machine learning, and clinical medicine. *The New England journal of medicine*, 375(13):1216, 2016.

[271] Jonathan H Chen and Steven M Asch. Machine learning and prediction in medicinebeyond the peak of inflated expectations. *The New England journal of medicine*, 376(26):2507, 2017.

[272] AM Noone, N Howlader, M Krapcho, D Miller, A Brest, M Yu, J Ruhl, Z Tatalovich, A Mariotto, DR Lewis, et al. Seer cancer statistics review, 1975-2015. *Bethesda, MD: National Cancer Institute*, 2018.

[273] Marcus H Galea, Roger W Blamey, Christopher E Elston, and Ian O Ellis. The nottingham prognostic index in primary breast cancer. *Breast cancer research and treatment*, 22(3):207–219, 1992.

[274] James S Michaelson, L Leon Chen, Devon Bush, Allan Fong, Barbara Smith, and Jerry Younger. Improved web-based calculators for predicting breast carcinoma outcomes. *Breast cancer research and treatment*, 128(3):827–835, 2011.

[275] James H Ware, David Harrington, David J Hunter, and Ralph B D'Agostino Sr. Missing data, 2012.

[276] Zhongheng Zhang. Multiple imputation with multivariate imputation by chained equation (mice) package. *Annals of translational medicine*, 4(2), 2016.

[277] Early Breast Cancer Trialists' Collaborative Group et al. Comparisons between different polychemotherapy regimens for early breast cancer: meta-analyses of long-term outcome among 100 000 women in 123 randomised trials. *The Lancet*, 379(9814):432–444, 2012.

[278] NR Latimer, KR Abrams, and U Siebert. Two-stage estimation to adjust for treatment switching in randomised trials: a simulation study investigating the use of inverse probability weighting instead of re-censoring. *BMC medical research methodology*, 19(1):69, 2019.

[279] Andrew HS Lee and Ian O Ellis. The nottingham prognostic index for invasive carcinoma of the breast. *Pathology & Oncology Research*, 14(2):113–115, 2008.

[280] Jérôme Lambert and Sylvie Chevret. Summary measure of discrimination in survival models based on cumulative/dynamic time-dependent roc curves. *Statistical methods in medical research*, 25(5):2088–2102, 2016.

[281] Frank E Harrell Jr, Kerry L Lee, and Daniel B Mark. Multivariable prognostic models: issues in developing models, evaluating assumptions and adequacy, and measuring and reducing errors. *Statistics in medicine*, 15(4):361–387, 1996.

[282] Hajime Uno, Tianxi Cai, Michael J Pencina, Ralph B D'Agostino, and LJ Wei. On the c-statistics for evaluating overall adequacy of risk prediction procedures with censored survival data. *Statistics in medicine*, 30(10):1105–1117, 2011.

[283] RB D'agostino and Byung-Ho Nam. Evaluation of the performance of survival analysis models: discrimination and calibration measures. *Handbook of statistics*, 23:1–25, 2003.

[284] Sotiris B Kotsiantis, I Zaharakis, and P Pintelas. Supervised machine learning: A review of classification techniques. *Emerging artificial intelligence applications in computer engineering*, 160:3–24, 2007.