PROBABILISTIC MODELS FOR EXPLORING, PREDICTING, AND INFLUENCING HEALTH TRAJECTORIES

by

Peter F. Schulam

A dissertation submitted to Johns Hopkins University in conformity with the requirements for the degree of Doctor of Philosophy

Baltimore, Maryland

October, 2019

Abstract

Over the past decade, healthcare systems around the world have transitioned from paper to electronic health records. The majority of healthcare systems today now host large, on-premise clusters that support an institution-wide network of computers deployed at the point of care. A stream of transactions pass through this network each minute, recording information about what medications a patient is receiving, what procedures they have had, and the results of hundreds of physical examinations and laboratory tests. There is increasing pressure to leverage these repositories of data as a means to improve patient outcomes, drive down costs, or both. To date, however, there is no clear answer on how to best do this. In this thesis, we study two important problems that can help to accomplish these goals: disease subtyping and disease trajectory prediction. In disease subtyping, the goal is to better understand complex, heterogeneous diseases by discovering patient populations with similar symptoms and disease expression. As we discover and refine subtypes, we can integrate them into clinical practice to improve management and can use them to motivate new hypothesis-driven research into the genetic and molecular underpinnings of the disease. In disease trajectory prediction, our goal is to forecast how severe a patient's disease will become in the future. Tools to make accurate forecasts have clear implications for clinical decision support, but they can also improve our process for validating new therapies through *trial enrichment*. We identify several characteristics of EHR data that make it to difficult to do subtyping and disease trajectory prediction. The key contribution of this thesis is a collection of novel probabilistic models that address these challenges and make it possible to successfully solve the subtyping and disease trajectory prediction problems using EHR data.

Primary Readers

Suchi Saria (Primary Advisor)
John C. Malone Assistant Professor
Departments of Computer Science,
Applied Mathematics and Statistics,
and Health Policy and Management
Johns Hopkins University

Gregory D. Hager
Mandell Bellmore Professor
Department of Computer Science
Johns Hopkins University

David M. Blei
Professor
Departments of Statistics and Computer Science
Columbia University

Acknowledgments

I want to start by thanking the person that has been closest to the work in this dissertation. In a field that at times seems to value technical complexity above all, Suchi constantly urged me focus on how we could contribute to the larger scientific landscape beyond machine learning. Doing so is especially difficult as a young PhD student looking to quiet his insecurities by beating readers over the head with mathematical details. As I look back on my years at Johns Hopkins, I cannot imagine a better lesson to have taken away from my training here. Through sleepless nights and countless heated debates, Suchi and I have forged a deep working relationship (and a lasting friendship, I'd like to think). It is something that I hope to continue, and that I will always treasure.

I would also like to thank other members of the academic community at Hopkins. My labmates Katie, Kirill, Hossein, Andong, Adarsh, Noam, and Roy have all been incredible colleagues over the past 6 years. I especially want to thank Katie for her willingness to answer an endless stream of medical questions (I still think Hopkins should award you an MD-PhD). Noam for answering my naive software engineering questions as I got up to speed with "real-world" systems. And Adarsh for the thought-partnership and collaborations; I look forward to more in the future. I had the opportunity to work with a number of excellent medical scientists at Hopkins. Thank you to Dr. Laura Hummers, Dr. Fred Wigley, and Dr. Colin Ligon for their patience and for the stimulating discussions that motivated many ideas in this thesis. The administrative staff in the computer science department have gone above and beyond to help me navigate the university during my time here. Thank you to Javonnia Thomas for the many friendly conversations and help with reimbursements over the years. And thank you to Zack Burwell for his patience and help during the graduation process.

Equally important to this undertaking have been all of the people in my life that supported me along the way. I quite literally could not have completed this process without my wife, Jenny. From ordering food to the computer science department when I couldn't make it home for dinner (and sometimes breakfast), to the "tough love" that got me back on my feet when I was feeling defeated, she was a constant source of support and motivation through many sleepless nights and work-filled weekends. My son, Peter Harold, although a new addition to the team, has made it easy to shrug off the many stumbles and setbacks along the way by greeting me every evening with his beaming smile and unconditional love. I also want to thank my parents Kim and Peter for their patience and support over the past 30 years. You've both taught me the importance of putting family first, having an impact through work, and being kind. And finally, thanks to my sister, Mia, for always reminding me what it looks like to work hard and passionately.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

Historically, documentation of healthcare care delivery was stored using paper records. In the last decade, new legislature has pushed providers to adopt *electronic health record* (EHR) systems. The Health Information Technology for Economic and Clinical Health (HITECH) Act pushed healthcare providers to transition from paper-based records to digitized databases and to implement new programs around "meaningful use" of that data. Although the transition is still in progress, the majority of healthcare systems today now host large, on-premise clusters that support an institution-wide network of computers deployed at the point of care. Nearly everything that happens in a hospital passes through this software. Doctors review patient status, enter notes, and place orders for laboratory tests and medication. Nurses check out prescribed medications using automated dispensers, request consultations, and enter patient vitals into massive, patient-specific spreadsheets. Valuable information that was once locked away in filing cabinets is now available to analyze and process using computers. The best way to leverage that data to improve patient outcomes and to drive down costs, however, is still an open question.

The idea of using computer programs and data to impact care has been around for decades and preceded the widespread adoption of EHR systems. To highlight one example, Michael Pozen and his colleagues fit a logistic regression model to predict the probability of acute ischemic heart disease in patients that arrived at the emergency department (ED) [Pozen et al., 1980, 1984]. Although the statistical methods are relatively simple, this work is remarkable for a number of reasons. First, the

work focused not on a biological question, but instead on an operational inefficiency in care delivery. When patients arrived in the ED with symptoms of ischemic heart disease, Pozen and his colleagues observed that physicians responsible for triaging these patients would often overestimate a patient's risk. The consequence of this was that the cardiac care unit (CCU) was often filled to capacity with patients who did not actually require the elevated care, which then made it harder to act quickly when a patient arrived in critical condition. Pozen and his colleagues wondered whether statistical estimates of severity could reduce CCU admission rates. Second, the authors of this study worked hard to develop a solution that was as closely integrated with workflow as possible. For eleven months, research assistants across six EDs manually collected 59 clinical features for 2801 patients. The authors used this data to fit a logistic regression model and implemented it on handheld calculators that ED physicians could use while treating patients. Pozen et al. [1984] report results on a prospective trial demonstrating a reduction in CCU capacity with no change in outcomes for cardiac patients. These investigators went to remarkable lengths to "instrument" care delivery and to integrate their solution back into the workflow of providers on the frontline, and the effort lead to a real difference in care delivery.

Using EHR systems, work like that which was done by Pozen and his colleagues would be made considerably easier. Many of the features that they collected are often routinely stored in the EHR, and the predictive tool could be deployed through the EHR-provider interface. Nevertheless, EHR data brings new challenges not encountered by early studies like the example above. Unlike studies with a carefully designed data collection protocol, EHR data is collected as a by-product. EHR systems are primarily a tool for coordinating care and for enabling administrative and financial processes. This introduces new statistical and computational challenges that we must address in order to maximize the value of this information. In this thesis, we identify and describe solutions to to several of these challenges.

1.1 Overview of Contributions

We focus on two core problems in this thesis: subtyping and disease trajectory prediction. Both problems are difficult to solve because of a variety of biases and technical challenges posed by EHR data. In the following sections, we motivate and describe these important problems, outline the challenges that make them difficult to solve using EHR data, and preview the key contributions in this thesis that address those challenges.

1.1.1 Discovering Disease Subtypes from Clinical Trajectory Data

Many diseases have no single canonical description; each patient is seemingly unique in how they manifest the disease. For instance, asthma is functionally defined by acute inflamation of the airways, but the severity of those symptoms, how they worsen or improve over time, and how they are triggered vary widely across asthmatic individuals (e.g. Wenzel et al. 1999). Although inhaled glucocorticosteroids effectively treat the symptoms, we do not have a clear understanding of the underlying biological mechanisms that drive the observed heterogeneity across individuals. In these *complex, heterogeneous diseases* we do not fully understand the "root causes", which makes it difficult to effectively treat more than the symptoms alone.

In Chapters 2 and 3, we develop new computational tools to help discover the root causes of heterogeneous diseases by searching for *subtypes* [Saria and Goldenberg, 2015]; subpopulations of patients that have similar clinical characteristics. The hypothesis underlying subtype-driven research is that individuals with similar disease expression are likely to share biological mechanisms driving the disease. As we discover and refine subtypes in a heterogeneous disease, we can begin to further our understanding of it by searching for genetic or molecular differences across those subtypes. Using EHR data we can study larger populations, which can accelerate subtype discovery and help to more quickly find "root causes" of heterogeneous diseases.

Disease subtyping is a natural idea, and has been implemented using manual case review (e.g. Barr et al. 1999a) and using automated clustering procedures of patient attributes (e.g. Haldar et al. 2008, Moore et al. 2010). In this thesis, we show how to use *clinical trajectory data* extracted

from EHRs to drive subtyping research [Schulam et al., 2015]. A clinical trajectory is time series data reflecting a patient's state over time. For instance, if a disease affects the respiratory system then we might use the results of spirometry tests over time to monitor how lung function changes throughout the course of the disease. Our proposal is to discover disease subtypes by searching for groups of patients with similar disease progression patterns by leveraging clinical trajectory data. Clinical trajectory data adds time as an important new dimension to the subtyping problem. Past approaches that use snapshots of patient state may be underpowered because two subtypes can look similar at a given point of the disease course. To successfully learn subtypes using clinical trajectory data from EHRs, we must tackle an important and common type of bias.

EHR databases are a "convenience sample"; they are not as curated as those collected in prospective studies. Subjects in an EHR study may differ in a number of important ways that influence their clinical presentation, but that are unrelated to the pathophysiology driving the disease. For example, patients may have incomplete medical histories (e.g. past surgeries). Or may live in widely varying conditions at home, which is typically not reflected in health records (along with other social determinants of health). If these important factors are not recorded in the EHR, we cannot easily "control" for them using standard techniques; i.e. we cannot remove their effects and isolate change in patient state due to the disease alone. When looking at EHR data, we do not see a clear picture of the disease but instead see one that is convolved with myriad observed and unobserved nuisance factors. We refer to these nuisance factors and their effects as *unobserved heterogeneity*. The result is that there can be considerable differences between the raw EHR data of two patients with the same subtype [Schulam et al., 2015].

We solve this problem by jointly modeling both the effects of the disease subtype and of the nuisance factors on an individual's clinical trajectory. Although we do not directly observe nuisance factors, we show that we can instead model their aggregate effects at various "resolutions" [Schulam et al., 2015]. For example, although we may not always record variables that reflect an individual's full medical history and living conditions at home, we know that the combined effects of those factors are approximately constant over time. We refer to these as individual-specific long term

effects. Similarly, although we do not observe all transient factors that might affect an individual's health (e.g. an infection), we know that the effects of these factors are only active over short time periods. We refer to these effects as individual-specific short term effects. Our approach searches for patients with similar clinical trajectories while simultaneously estimating and removing individual-specific long and short term effects that would otherwise bias the subtypes that we learn. We show that this approach helps us to discover more clinically significant disease trajectory subtypes than state-of-the-art alternatives that do not account for unobserved heterogeneity.

1.1.2 Making Accurate and Reliable Disease Trajectory Predictions

Alvin Feinstein argued that a fundamental scientific challenge in clinical medicine is to develop accurate procedures for predicting patient outcomes [Feinstein, 1983]. This claim reflects the idea that the core of successfully managing and treating patients is a process of estimating and mitigating risks [Spiegelhalter, 1986]. Predictive models in clinical medicine have been studied for decades, and are most commonly built using statistical regression techniques [Steyerberg, 2009, Harrell, 2015]. In classical statistical regression, the goal is to determine a mathematical relationship between a collection of measured input variables (e.g. age, gender, blood pressure) and an outcome (e.g. the time to disease recurrence or the value of a future test result).

Predictions have also started to play a larger role in the design and implementation of clinical trials through the idea of *enrichment*. The key idea behind enrichment is that we design enrollment criteria based on the results of prognostic indicators [Temple, 2010, Freidlin and Korn, 2014]. For example, if an experimental drug is thought to lower the risk of heart attacks, then we can make the trial more efficient (i.e. require fewer subjects) by only enrolling those at highest risk [Simon and Maitournam, 2004]. In many cases, the prognostic indicators are single risk factors or biomarkers (e.g. family history or the result of a genetic test). A relatively new idea is to instead build predictive models for trial enrichment. Instead of relying on a single test result or biomarker, this approach pools together information across a variety of measurements and uses the predicted risk as "synthetic" biomarker. This idea has been explored, for instance, in the context of trials targeting

skin disease in scleroderma [Maurer et al., 2015].

EHRs capture a wide range of clinical data on large patient populations for long periods of time, and so they are an attractive source of training data for fitting predictive models. Moreover, because EHR data so closely mirrors the interactions between a patient and the healthcare system, a predictive model fit to this data has the potential to be especially effective for guiding patient management and driving trial enrichment (as opposed to models that are fit on more curated datasets that are not similar to the environments in which the model will be used). To use EHR data effectively, however, there are a number of obstacles that we must address:

Unobserved heterogeneity. Classical statistical regression models (which include linear models, random forests, and neural networks) all require a sufficiently informative collection of input variables in order to achieve good performance. For example, if an individual's prognosis heavily depends on the presence of several genes, then we must include measures of those genes in our model (or some correlated variable). Without those inputs, there is a cap on how well the model will perform. EHR data, however, is commonly affected by unobserved heterogeneity; i.e. unmeasured factors that affect our clinical observations. There is therefore a fundamental limit on the accuracy of classical regression models when applied to EHR data.

Policy shift. Patients in EHR data are being actively treated, and these treatment decisions partially determine the distribution of our training data. When we fit a regression model to this data, we are assuming that the policy in the training data remains unchanged in the test data as well. There are at least two ways that such a model is unreliable. First, our model can perform poorly if patients are treated differently than in the training data (i.e. if the assumption underlying our regression model is violated). This is not uncommon; treatment policies can shift across hospitals, across providers, and even over time at the same hospital. The second way that our model can be unreliable is subtle and does not require a change in how patients are treated. It stems from a mismatch between what our predictive model learns and how users often incorporate model predictions into their workflow. Chapter 6 further elaborates on this second type of unreliability.

Sparse and high dimensional data. In many diseases, there are multiple risk factors and biomarker trajectories that we want to track over time. For example, in the complex autoimmune disease scleroderma Varga et al. [2012], physicians monitor a variety of test results that each reflect the health of a different organ system (e.g. the lungs, vasculature, skin, and so on). These trajectories are often sparsely and irregularly samples, and the markers are not always sampled at the same time. Joint generative models can handle the missing data, and are a natural approach for modeling these trajectories. This approach, however, introduces several additional challenges. First, generative models are sensitive to incorrect statistical assumptions and it is difficult to accurately model interactions between a large collection of trajectories. Second, it becomes increasingly expensive to perform the computations needed to learn and make predictions with the model as we include additional biomarker trajectories.

In Chapter 4, we build on the ideas developed for subtyping to describe a new framework for predicting an individual's disease progression that is accurate even when there is unboserved heterogeneity. Our approach leverages observed predictors (e.g. age, gender, race, and so on) along with an individual-specific hierarchy of unobserved factors [Schulam and Saria, 2015]. As we gather more information about a patient, our approach dynamically updates estimates of subtype, individual-specific long term, and individual-specific short term factors. At any given point in time, our approach combines current estimates of these factors with observed predictors to produce more accurate forecasts of a patient's future disease activity. In Chapter 5, we describe an extension of our framework that makes it simpler and more efficient to leverage high dimensional clinical trajectory data to predict disease progression. We use a modular, two-stage approach that first fits individual biomarker trajectory models and then learns to share inferences across those individual models in a way that maximizes predictive performance [Schulam and Saria, 2016].

In Chapter 5, we demonstrate that our approach to predicting disease progression has strong clinical utility by fitting a state-of-the-art model of lung disease progression in scleroderma, a complex and heterogeneous autoimmune disease [Varga et al., 2012]. These results demonstrate the

added predictive value of both solving the unobserved heterogeneity problem and of integrating information across many biomarker trajectories. In clinical practice, these improvements in predicting disease progression can make it easier for clinicians to better anticipate their patients needs and initiate treatment early enough to be most effective. In medical research, we can use the improved predictions as more effective "synthetic biomarkers" to drive trial enrichment. In complex heterogeneous diseases, where the rate of trial failure is especially high due to poor power [Temple, 2010], there is an opportunity to drive down costs and improve the quality of the data generated in new trials.

In Chapter 6 we address the *policy shift* issue by blending ideas from predictive modeling and causal inference [Schulam and Saria, 2017]. Fitting causal models to sparse and irregularly sampled continuous-time disease trajectory data is challenging because both the measurement policy driving when we record biomarkers and the treatment policy can introduce confounding. We describe a procedure that estimates causal models of continuous-time trajectory data in this challenging setting, and lay out a set of assumptions that are sufficient to prove its correctness. Moreover, we motivate our continuous-time causal model by showing that discretizing, or binning, continuous-time trajectories to create discrete-time trajectories can introduce confounding bias. In our experiments, we show that causal disease trajectory models are less sensitive to policy shift and therefore more reliable. Finally, we demonstrate how our causal disease trajectory models can be used for planning patient management by building a tool to predict the trajectory of kidney function under various forms of dialysis for patients in the ICU.

Our approach to building reliable predictive models resolves several long-standing issues with machine learning in clinical medicine. First, it is common practice to validate predictive models using data from a target hospital before applying it in that new environment. This is because models can fail to generalize from training data for many reasons (e.g. different case mix or changes in treatment policies). The methods in Chapter 6 show how to build models that generalize even when treatment policies change. Many of the same ideas can also be applied to mitigate other reasons for poor generalization. Models fit with these techniques can help to remove some of the

"due diligence" required before deploying models in clinical practice.

Finally, there are also implications for how predictive models are incorporated into a clinician's workflow. Steyerberg [2009] summarizes the issue with standard predictive models:

> Unfortunately, choice of therapy based on prognostic models will directly affect the validity of such models, but nevertheless prognostic models are needed even if they are self-destroying. In some sense, not only the information on the patient is dynamic, but so is any prognostic model, because it should be based on current and not on past treatment protocols.

In short, classical predictive models are "one-shot" because the distribution of future data changes once a clinician chooses treatment based on its output. Because predictive models trained using the methods in Chapter 6 are robust to changes in treatment policy, they remain valid even after a clinician uses the model to guide therapy. This makes it safer and easier to deploy predictive models at the point of care because they do not need to be updated over time.

Chapter 2

Disease Trajectory Subtypes

Disease subtyping is the process of stratifying a population of individuals with a shared disease into subgroups that exhibit similar clinical traits; a task that is analogous to clustering in machine learning. Under the assumption that individuals with similar traits share an underlying disease mechanism, disease subtyping can help to propose candidate subgroups of individuals that should be investigated for biological differences. Understanding these differences can shed light on the mechanisms specific to each group. Observable traits useful for identifying sub-populations of similar patients are called *phenotypes*. Once subtypes are linked to a distinct underlying pathobiological mechanism, they are then referred to as *endotypes* [Anderson, 2008].

Traditionally, disease subtyping research has been conducted as a by-product of clinical experience. A clinician may notice the presence of subgroups, and may perform a more thorough retrospective or prospective study to confirm their existence (e.g. Barr et al. 1999b). Recently, however, literature in the medical community has noted the need for more objective methods for discovering subtypes [De Keulenaer and Brutsaert, 2009]. Growing repositories of health data stored in electronic health record (EHR) databases and patient registries [Blumenthal, 2009, Shea and Hripcsak, 2010] present an exciting opportunity to identify disease subtypes in an objective, data-driven manner using tools from machine learning that can help to tackle the problem of combing through these massive databases. In this chapter, we describe such a tool, the Probabilistic Subtyping Model (PSM), that is designed to discover subtypes of complex, systemic diseases using

longitudinal clinical marker trajectories collected in EHR databases and patient registries.

Discovering and refining disease subtypes using clinical trajectories can benefit both the practice and the science of medicine. Clinically, associating patients with disease trajectory subtypes can help to reduce uncertainty in expected outcome of an individual's case, thereby improving treatment. Subtype trajectories can inform therapies and aid in making prognoses and forecasts about expected costs of care [Chang et al., 2011]. Scientifically, disease subtypes can help to improve the effectiveness of clinical trials [Gundlapalli et al., 2008], drive the design of new genome-wide association studies [Kho et al., 2011, Kohane, 2011], and allow medical scientists to view related diseases through a more fine-grained lens that can lead to insights that connect their causes and developmental pathways [Hoshida et al., 2007]. Disease subtyping is especially useful for complex, heterogeneous diseases where mechanism is often poorly understood. Examples of disease subtyping research include work in autism (e.g. State and Sestan 2012), cardiovascular disease (e.g. De Keulenaer and Brutsaert 2009), and Parkinson's disease (e.g. Lewis et al. 2005).

Clinicians commonly characterize complex disease using the level of disease activity present in an array of organ systems. They typically measure the influence of a disease on an organ using clinical tests that quantify the extent to which that organ's function has been affected by the disease. The results of these tests, which we refer to as *illness severity markers* (s-markers for short), are being routinely collected over the course of care for large numbers of patients within EHR databases and patient registries. For a single individual, the time series formed by the sequence of these s-markers constitutes a *disease activity trajectory*. Operating under the hypothesis that individuals with similar disease activity trajectories are more likely to share mechanism, this chapter shows how to cluster individuals according to their longitudinal clinical marker data and learn the associated *prototypical disease activity trajectories* (i.e. a continuous-time curve characterizing the expected s-marker values over time) for each subtype.

2.1 Controlling for Unobserved Heterogeneity

The s-marker trajectories recorded in EHR databases are influenced by many factors such as age and co-existing conditions that are unrelated to the underlying disease mechanism [Lötvall et al., 2011]. We call the effects of these additional factors *nuisance variability*. In order to correctly cluster individuals and uncover disease subtypes that are likely candidates for endotyping Lötvall et al. [2011], we must model and explain away nuisance variability. In many cases, however, these additional factors are not recorded in our data (e.g. parts of the patient's medical history might be missing). When nuisance variability is caused by unmeasured factors, we refer to it as *unobserved heterogeneity*, which we cannot control for by directly modeling the underlying factors.

We account for and remove unobserved heterogeneity by modeling the aggregate effects of unobserved factors at various resolutions using latent variables. First, we use a population-level regression on to observed covariates— such as demographic characteristics or co-existing conditions— to account for variability in s-marker values across individuals due to measured factors. For example, lung function as measured by the forced expiratory volume (FEV) test is well-known to be worse in smokers than in non-smokers [Camilli et al., 1987]. Second, we use individual-specific parameters to account for variability across individuals that is not predicted using the observed covariates. This form of variability may last throughout the course of an individual's disease (e.g. the individual may have an unusually weak respiratory system) or may be episodic (e.g. periods during which an individual is recovering from a cold). After accounting for unobserved heterogeneity, we cluster the time series formed by the residual activity to induce subtypes. We hypothesize that differences across such clusters are more likely candidates for endotype investigations.

2.2 Contributions

In this chapter, we describe the Probabilistic Subtyping Model (PSM), a novel model for discovering disease subtypes and associated prototypical disease activity trajectories using observational data that is routinely collected in electronic health records (EHRs). To tackle unobserved heterogeneity,

PSM learns clinical trajectory subtypes by simultaneously learning the subtypes and the effects of unobserved nuisance factors. PSM controls for (1)nuisance variability due to observed factors, (2) individual-specific long term unobserved heterogeneity, and (3) individual-specific short term unobserved heterogeneity. Moreover, PSM learns clinical trajectory subtypes using sparse and irregularly sampled time series, which is common in obserational EHR data. To evaluate PSM, we use real and simulated data to demonstrate that, by accounting for unobserved heterogeneity, PSM can accurately impute missing s-markers and accurately recover ground truth clinical trajectory subtypes. Finally, we discuss novel subtypes discovered using PSM in the context of the complex autoimmune disease, scleroderma [Varga et al., 2012].

2.3 Related Work

A large body of work has focused on identifying subtypes using genetic data (e.g. Chen et al. 2011). Enabled by the increasing availability of EHRs, researchers have also recently started to leverage clinical markers to conduct subtype investigations. Chen et al. [2007] use the clinarray— a vector containing summary statistics of all available clinical markers for an individual— to discover distinct phenotypes among patients with similar diseases. The clinarray summarizes longitudinal clinical markers using a single statistic, which ignores the pattern of progression over time. More recently, Ho et al. [2014] used tensor factorization as an alternative approach to summarizing high-dimensional vectors created from EHRs [Ho et al., 2014]. Others have used cross-sectional data to piece together population disease progression trajectories (e.g. Ross and Dy 2013), but do not model individual trajectories. Another approach to phenotyping using time series data, often applied in the acute care setting, is to segment an individual's time series into windows in order to discover transient traits that are expressed over shorter durations (minutes, hours, or days). For example, Saria et al. propose a probabilistic framework for discovering traits from physiologic time series that have similar shape characteristics [Saria et al., 2011] or dynamics characteristics [Saria et al., 2010]. Lasko et al. [2013] use deep learning to induce an over-complete dictionary in order to define latent traits over shorter segments of clinical markers [Lasko et al., 2013]. Beyond s-markers,

others have used ICD-9 codes— codes indicating the presence or absence of a condition— to study comorbidity patterns over time among patients with a shared disease (e.g. Doshi-Velez et al. 2014). ICD-9 codes are further removed from the biological processes measured by quantitative tests. Moreover, the notion of disease severity is more difficult to infer from codes, which only record binary presence/absence information.

Latent class mixed models (LCMMs) are a family of methods designed to discover subgroup structure in longitudinal datasets using fixed and random effects (e.g., Muthén and Shedden 1999, McCulloch et al. 2002, Nagin and Odgers 2010). Random effects are typically used in linear models and allow an individual's coefficients to be probabilistically perturbed from the population's. This alters the model's fit to the individual over the entire observation period. Modeling s-marker data for chronic diseases, where data are collected over tens of years, requires accounting for additional influences such as those due to transient disease activity. The task of modeling variability between related time series has been explored in other contexts (e.g. Listgarten et al. 2006, Fox et al. 2011). These typically assume regularly sampled time series and model properties that are different from those in our application.

Finally, previous work in the machine learning literature has also looked at relaxing the assumption of regularly sampled data. Marlin et al. [2012] cluster irregular clinical time series from in-hospital patients to improve mortality prediction. They use Gaussian process priors that allow unobserved measurements to be marginalized [Marlin et al., 2012]. Lasko et al. [2013] also address irregular measurements by using MAP estimates of Gaussian processes to impute sparse time series [Lasko et al., 2013]. Neither of these methods account for unobserved heterogeneity.

2.4 Probabilistic Subtyping Model

We define a generative model for a collection of M individuals with associated s-marker sequences. For each individual i, the s-marker sequence has N_i measurement times and values, which are denoted as $t_i \in \mathbb{R}^{N_i}$ and $y_i \in \mathbb{R}^{N_i}$ respectively. In addition to the measurement times and values, each individual is assumed to have a vector of d covariates $x_i \in \mathbb{R}^d$. The s-marker values $y_{1:M}$ are

random variables, and we assume that $t_{1:M}$ and $x_{1:M}$ are fixed and known. The major conceptual pieces of the model are the subtype mixture model, covariate-dependent nuisance variability, individual-specific long-term nuisance variability, and individual-specific short-term nuisance variability. We describe each of these pieces in turn. The graphical model in Figure 4.1 shows the relevant hyperparameters, random variables, and dependencies.

2.4.1 Background: B-Splines

A common approach to fitting nonlinear functions of time while maintaining a linear dependence on model parameters is to use a basis expansion. Such an expansion defines some non-linear function $f(t)$ as a linear combination of other functions $\phi_1(t), \ldots, \phi_p(t)$:

$$y = f_\beta(t) = \sum_{i=1}^{p} \beta_i \phi_i(t) = \Phi^\top(t)\beta, \tag{2.1}$$

where $\phi_1, \ldots, \phi_p$ are bases in a vector space of nonlinear functions and $\Phi(t) \in \mathbb{R}^p$ is the vector containing the values of the p basis functions evaluated at time t. The benefit of this formulation is that the function f is linear in the model parameters β, making it relatively easy to fit complex models. B-splines are a particular family of basis functions that we can use to parameterize nonlinear functions. Others include polynomial bases and radial basis functions. However, there are two advantages to using B-splines. First, each basis function is non-zero only over a compact interval of the real line, which improves statistical stability and also allows for computational speed ups that take advantage of sparse basis matrices [Gelman et al., 2014]. This is in contrast to polynomials, where each basis takes non-zero values globally. The second advantage is that the family of functions parameterized by B-splines are not infinitely differentiable (in contrast to radial basis functions) and therefore not smooth [Gelman et al., 2014]. This bias is often more realistic when modeling biological data. Moreover, because B-splines are linear in their parameters, we can use the well-developed machinery of linear regression for learning. Ch. 20 in Gelman et al. [2014] and Ch. 5 in Friedman et al. [2001] have additional details.

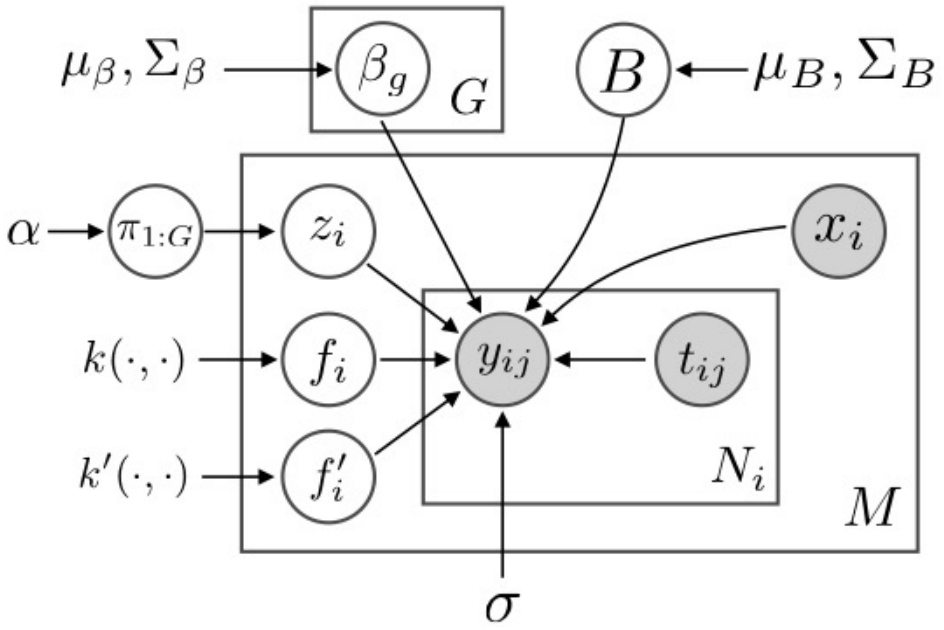


Figure 2.1: Graphical model for PSM.

Penalized B-splines

In practice, the parameters of a B-spline model are fit using a penalized least squares criterion. The penalty is typically introduced in order to control the smoothness of the fit. For data y measured at times t with corresponding basis matrix $\Phi(t) = [\Phi(t_1), \ldots, \Phi(t_n)]^\top$, we minimize the following objective:

$$J(\beta) = \|y - \Phi(t)\|_2^2 + \rho\beta^\top \Omega\beta, \tag{2.2}$$

where Ω is, for example, a first-order differences matrix as described by Eilers and Marx [1996]. The penalized objective is still quadratic in β and so can be easily minimized.

2.4.2 Subtype Mixture Model

Each individual is assumed to belong to one of G latent groups (representing a disease subtype). The random variable $z_i \in \{1, \ldots, G\}$ encodes subtype membership, and is drawn from a multinomial distribution with probability vector $\pi_{1:G}$. The probabilities $\pi_{1:G}$ are modeled as a Dirichlet random

variable with symmetric concentration parameter α. Formally, we have:

$$\pi_{1:G} \sim \mathsf{Dirichlet}(\alpha) \tag{2.3}$$

$$z_i \mid \pi_{1:G} \sim \mathsf{Categorical}(\pi_{1:G}). \tag{2.4}$$

We model each of the subtype prototypical disease activity trajectories using B-splines. Each subtype's disease activity trajectory is parameterized by a vector of coefficients $\beta \in \mathbb{R}^p$. The coefficients $\beta_{1:G}$ are modeled as independent random vectors drawn from a multivariate normal distribution:

$$\beta_g \sim \mathsf{Normal}(\mu_\beta, \Sigma_\beta). \tag{2.5}$$

2.4.3 Covariate-Dependent Heterogeneity

When one or more covariates are available that are known to influence clinical test results, but are posited to be unrelated to the underlying disease mechanism, the influences can be accounted for using covariate-dependent effects. For example, smoking status can partially explain why an individual's forced expiratory volume declines more rapidly, or African American race may be associated with especially severe scleroderma-related skin fibrosis. By fitting a standard mixture of B-spline regressions to s-marker data directly, the subtype random variable z_i may incorrectly capture correlations among groups of individuals with similar covariates. By including covariate-dependent effects, we can adjust for these correlations and free the subtype mixture model to capture *residual correlations* that are more likely to be due to a common underlying disease mechanism.

We model covariate-dependent effects by using x_i to predict a vector of coefficients $\beta_x \in \mathbb{R}^p$ for the same B-spline basis Φ that we use to model the subtype disease trajectories. To predict the elements of β_x, we use p linear functions of x. We stack the coefficients of these linear functions in the rows of a matrix $B \in \mathbb{R}^{p \times d}$. We model each row B_k of B as multivariate normal random

variable

$$B_k \sim \mathsf{Normal}(\mu_B, \Sigma_B). \tag{2.6}$$

Modeling Treatments

If there are treatments that can alter long-term course and the points at which they are administered vary widely across individuals, treatments become additional sources of nuisance variability that we can model using time-dependent covariates. In scleroderma, our disease of interest, and many other systemic diseases, no known drugs modify the long-term course of the disease, and so we do not tackle this issue in this chapter.

2.4.4 Individual-Specific Long Term Unobserved Heterogeneity

An individual may express additional variability over the entire observation period beyond what is explained away using covariates. For example, an individual may have an unusually weak respiratory system and so may have a lower baseline value (intercept), which may not be reflected in any measured covariates. We model this form of variability using samples from a *long-term Gaussian process*. Each individual i has a separate long-term function

$$f_i \sim \mathsf{GP}(0, k), \tag{2.7}$$

where k is a covariance function with a weak dependence on time. Practically, this means that two samples from the same patient will be correlated even if taken at different points in the disease trajectory. The constant covariance is one example of such a covariance:

$$k(t_1, t_2) = \nu^2. \tag{2.8}$$

This is equivalent to drawing an individual-specific random intercept from a normal distribution with variance ν^2.

2.4.5 Individual-Specific Short Term Unobserved Heterogeneity

Finally, an individual may experience episodic disease activity that only affects a handful of measurements within a small time window. For example, there may be periods during which an individual is recovering from a cold, which temporarily weakens his or her respiratory system and causes a temporary drop in forced expiratory volume. We model these transient deviations using a *short-term Gaussian process*. Each individual i has a separate short-term function

$$f_i' \sim \mathsf{GP}(0, k'), \tag{2.9}$$

where k' is a covariance function that assigns stronger correlations to samples measured close in time. One example of a short-term covariance function is the Ornstein-Uhlenbeck kernel:

$$k'(t_1, t_2) = a^2 \exp\left\{\frac{|t_1 - t_2|}{w^2}\right\}. \tag{2.10}$$

As a increases, we encode the assumption that transient events can have an increasing effect on an individual's trajectory. The parameter w encodes the expected time range over which we expect these events to occur (i.e. smaller w means that the events last for a shorter period of time).

2.4.6 S-marker Measurement Model

Given fixed model parameters $\Theta = \{\pi_{1:G}, \beta_{1:G}, B, k, k'\}$ and conditioned on the subtype z_i, long-term individual effects f_i, and short-term individual effects f_i', we model an individual's measurements y_i as iid samples from a normal distribution

$$y_{ij} \mid z_i, f_i, f_i' \sim \mathsf{Normal}\Big(\Phi(t_{ij})Bx_i + \Phi(t_{ij})\beta_{z_i} + f_i(t_{ij}) + f_i(t_{ij}), \sigma^2\Big). \tag{2.11}$$

2.5 Learning and Inference

In this section, we assume that the number of subtypes G, long-term covariance function k, short-term covariance function k', and measurement variance σ are fixed upfront. To choose these values

we use domain knowledge and model selection procedures (e.g. BIC, cross validation, or held out training data).

2.5.1 Learning

To learn the remaining model parameters $\Theta = \{\pi_{1:G}, \beta_{1:G}, B\}$, we use expectation- maximization (EM) to optimize the joint log density of $y_{1:M}$ and Θ:

$$\log p(y_{1:M}, \Theta) = \sum_{i=1}^{M} \log p(y_i \mid \Theta) + \log p(\pi_{1:G}) + \log p(\beta_{1:G}) + \log p(B). \tag{2.12}$$

To calculate the density of y_i we marginalize over the individual's latent variables z_i, f_i, and f_i'. Given z_i, it is straightforward to marginalize over f_i and f_i' since $p(y_i, f_i, f_i' \mid z_i)$ is a multivariate normal distribution:

$$y_i \mid z_i \sim \mathsf{MultNormal}\Big(\Phi(t)Bx_i + \Phi(t)\beta_{z_i}, K_i + K_i' + \sigma^2 \mathsf{I}_{N_i}\Big), \tag{2.13}$$

where we have defined the covariance matrices

$$\big[K_i\big]_{jk} = k(t_{ij}, t_{ik}) \tag{2.14}$$

$$\big[K_i'\big]_{jk} = k'(t_{ij}, t_{ik}) \tag{2.15}$$

and used I_{N_i} to denote the identity matrix with N_i rows and columns. To marginalize over z_i, we simply mix the G multivariate normals:

$$p(y_i \mid \Theta) = \sum_{g=1}^{G} \pi_g p(y_i \mid z_i). \tag{2.16}$$

Expectation Step

Because we assume the covariance functions k and k' are fixed, we only need to compute the posterior distribution of z_i given y_i for each individual. Fixing Θ, the posterior distribution of z_i is

$$q_i(g) \propto \pi_g p(y_i \mid z_i = g). \tag{2.17}$$

Maximization Step

Conceptually, there are two parts to the maximization step. First, we must reestimate the subtype prior probabilities $\pi_{1:G}$. Second, we must reestimate the parameters that determine the expected value of y_i ($\beta_{1:G}$ and B).

Prior Probabilities To reestimate $\pi_{1:G}$, we consider the terms of the complete-data log likelihood that depend on the prior probabilities:

$$J_1(\pi_{1:G}) = \sum_{i=1}^{M} \log p(z_i \mid \pi_{1:G}) + \log p(\pi_{1:G}) \tag{2.18}$$

$$= \sum_{i=1}^{M} \left(\sum_{g=1}^{G} 1_{z_i=g} \log \pi_g \right) + \sum_{g=1}^{G} (\alpha - 1) \log \pi_g + C. \tag{2.19}$$

After computing the expected value of J_π with respect to the posteriors $\{q_i\}_{i=1}^M$, we can maximize Q_π subject to the constraint that $\sum_{g=1}^G \pi_g = 1$ by setting

$$\pi_g = \frac{\sum_{i=1}^M q_i(g) + \alpha - 1}{\sum_{g'=1}^G \pi_{g'}}. \tag{2.20}$$

Mean Parameters To compute new estimates of the mean parameters (B and $\beta_{1:G}$), we introduce some additional notation. First, define $b = \mathsf{vec_r}(B) \in \mathbb{R}^{pd}$ where $\mathsf{vec_r}$ denotes the operation of stacking the rows of B into a single vector (i.e. *flattening* the matrix along its rows). The flattened

vector has prior mean and covariance

$$\mu_b = [\underbrace{\mu_B^\top, \ldots, \mu_B^\top}_{p \text{ times}}]^\top \tag{2.21}$$

$$\Sigma_b = \mathsf{I}_p \otimes \Sigma_B, \tag{2.22}$$

where $\otimes$ denotes the Kronecker product. Next, for each individual i we define the matrices

$$C_i = \Phi(t_i)[\mathsf{I}_p \otimes x_i^\top] \tag{2.23}$$

$$D_i = \Phi(t_i) \tag{2.24}$$

$$S_i = K_i + K_i' + \sigma^2 \mathsf{I}_{N_i}. \tag{2.25}$$

Using this notation, we can write the conditional distribution of y_i given z_i as

$$y_i \mid z_i \sim \mathsf{MultNormal}(C_i b + D_i \beta_{z_i}, S_i). \tag{2.26}$$

The complete-data log likelihood as a function of b and $\beta_{1:G}$ is therefore

$$J_2(b, \beta_{1:G}) = \sum_{i=1}^M \log p(y_i \mid z_i) + \log p(b) + \sum_{g=1}^G \log p(\beta_g) \tag{2.27}$$

$$= \sum_{i=1}^M -\frac{1}{2}(y_i - C_i b - D_i \beta_{z_i})^\top S_i^{-1}(y_i - C_i b - D_i \beta_{z_i})$$

$$+ -\frac{1}{2}(\mu_b - b)^\top \Sigma_b^{-1}(\mu_b - b) + \sum_{g=1}^G -\frac{1}{2}(\mu_\beta - \beta_g)^\top \Sigma_\beta^{-1}(\mu_\beta - \beta_g). \tag{2.28}$$

The partial derivatives are

$$\frac{\partial J_2}{\partial b} = \sum_{i=1}^M C_i^\top S_i^{-1}(y_i - C_i b - D_i \beta_{z_i}) + \Sigma_b^{-1}(\mu_b - b) \tag{2.29}$$

$$\frac{\partial J_2}{\partial \beta_g} = \sum_{i=1}^M 1_{z_i = g} D_i^\top S_i^{-1}(y_i - C_i b - D_i \beta_g) + \Sigma_\beta^{-1}(\mu_\beta - \beta_g). \tag{2.30}$$

Taking expectations with respect to the posteriors $\{q_i\}_{i=1}^M$ and setting these partial derivatives to

zero, we obtain the following fixed-point equations that we can iterate until the values of b and $\beta_{1:G}$ converge:

$$b = \Big(\sum_{i=1}^{M} C_i^\top S_i^{-1} C_i + \Sigma_b^{-1}\Big)^{-1} \Big(\sum_{i=1}^{M} C_i^\top S_i^{-1}(y_i - D_i \mathbb{E}_{q_i}[\beta_{z_i}]) + \Sigma_b^{-1}\mu_b\Big) \tag{2.31}$$

$$\beta_g = \Big(\sum_{i=1}^{M} q_i(g) D_i^\top S_i^{-1} D_i + \Sigma_\beta^{-1}\Big)^{-1} \Big(\sum_{i=1}^{M} q_i(g) D_i^\top S_i^{-1}(y_i - C_i b) + \Sigma_\beta^{-1}\mu_\beta\Big). \tag{2.32}$$

2.5.2 Scalability

PSM is designed to aid in the subtype discovery process when large electronic health databases are available for analysis, so the scalability of the learning algorithm is a natural concern. The primary computational bottleneck of the PSM learning procedure is the E-step, which may be expensive due to (1) the number of individuals in the analysis, or (2) the inversion of the individual covariance matrices Σ_i (in Equation 2.17). The computational complexity due to large M can be offset by parallelizing the E-step because the individual-specific latent variables are conditionally independent given Θ. Inverting S_i has computational complexity $\mathcal{O}(N_i^3)$. Because we study disease activity over the course of 10-20 years and because visit rates typically do not exceed 12 per year, the number of measurements N_i is typically on the order of 100-200 measurements, which is cheap.

2.5.3 Inference and Prediction

Given parameters Θ and observed data y_i at times t_i, we can compute a posterior distribution over the subtypes z_i using Equation 2.17. In addition to the subtype, we can compute an estimate of the individual's long-term and short-term components (f_i and f_i' respectively). Given z_i, the long-term individual variability at time t_* has a normal distribution with mean and variance

$$\mathbb{E}[f_i(t_*) \mid z_i, y_i] = K_* S_i^{-1}(y_i - C_i b - D_i \beta_{z_i}) \tag{2.33}$$

$$\mathsf{Var}(f_i(t_*) \mid z_i, y_i) = K_{**} - K_* S_i^{-1} K_*^\top, \tag{2.34}$$

where $K_{**} = k(t_*, t_*)$, $K_* \in \mathbb{R}^{1 \times N_i}$, and $[K_*]_{1i} = k(t_*, t_{ij})$. Similarly, given z_i the short-term individual variability at time t_* has a normal distribution with mean and variance

$$\mathbb{E}[f_i'(t_*) \mid z_i, y_i] = K_*' S_i^{-1}(y_i - C_i b - D_i \beta_{z_i}) \tag{2.35}$$

$$\mathsf{Var}(f_i'(t_*) \mid z_i, y_i) = K_{**}' - K_*' S_i^{-1} {K_*'}^\top, \tag{2.36}$$

where $K_{**}' = k'(t_*, t_*)$, $K_*' \in \mathbb{R}^{1 \times N_i}$, and $[K_*']_{1i} = k'(t_*, t_{ij})$.

Finally, we can use PSM to impute missing measurements by computing the posterior distribution over an individual's s-markers at time t_*. Define the composite covariance function $k'' = k + k'$, then the conditional distribution of y_* at time t_* given z_i is a normal distribution with mean and variance

$$\mathbb{E}[y_* \mid z_i, y_i] = C_* b + D_* \beta_{z_i} + K_*'' S_i^{-1}(y_i - C_i b - D_i \beta_{z_i}) \tag{2.37}$$

$$\mathsf{Var}(y_* \mid z_i, y_i) = K_{**}'' - K_*'' S_i^{-1} {K_*''}^\top, \tag{2.38}$$

where $K_{**}' = k'(t_*, t_*)$, $K_*' \in \mathbb{R}^{1 \times N_i}$, $[K_*']_{1i} = k'(t_*, t_{ij})$, and

$$C_* = \Phi(t_*)(\mathsf{I}_p \otimes x_i^\top) \tag{2.39}$$

$$D_* = \Phi(t_*). \tag{2.40}$$

2.5.4 Estimating Kernel Parameters

The long-term and shor-term covariance functions k and k' are important sources of domain knowledge. By choosing these functions carefully, we can place priors over the type of latent individual-specific variability that we want to explain away when searching for subtypes. In some cases, however, we may only want to choose a family of covariance kernels and would therefore need to estimate the parameters from data. In this section, we briefly describe an extension to the EM algorithm described above that includes updates to kernel parameters.

Expectation Step

We extend the expectation step above by computing a joint posterior distribution over the subtype z_i, long-term individual variability $f_i \in \mathbb{R}^{N_i}$, and short-term individual variability $f_i' \in \mathbb{R}^{N_i}$. Let f_i'' denote the concatenation of f_i and f_i', then the distribution over f_i'' given z_i and y_i is a multivariate normal with mean and covariance

$$\mu_i''(z_i) = \sigma^{-2}\Sigma_i'' A_i^\top (y_i - \Phi(t_i)Bx_i - \Phi(t_i)\beta_{z_i}) \tag{2.41}$$

$$\Sigma_i''(z_i) = \left(\sigma^{-2}A_i^\top A_i + \begin{pmatrix} K_i^{-1} & 0 \\ 0 & K_i'^{-1} \end{pmatrix}\right)^{-1}. \tag{2.42}$$

We see that the posterior distribution over f_i'' is a mixture of multivariate normals.

Maximization Step

To update our estimates of the kernel parameters given the posterior distribution of f_i'', we maximize the expected log priors $\log p(f_i)$ and $\log p(f_i')$ with respect to the kernel hyperparameters θ and θ'. In general, there is no closed form maximizer of the likelihood of a GP with respect to the kernel parameters (e.g. for the length scale of an RBF kernel), so we can instead use zero or first order searches to maximize

$$\sum_{i=1}^{M} \mathbb{E}[\log p(f_i)] = \sum_{i=1}^{M} -\frac{1}{2}\log|K_i| - \frac{1}{2}\mathsf{tr}\Big(K_i^{-1}\sum_{g=1}^{G} q(z_i)(\mu_i\mu_i^\top + \Sigma_i)\Big) + C_i. \tag{2.43}$$

Estimating marginal covariances. Many covariance kernels have a parameter that determines the marginal covariance of the Gaussian process. In the special case where the remaining parameters are fixed, we can update this marginal covariance parameter with a single expression in the maximization step. Suppose we can write the prior covariance of f_i as $\nu^2 K_i$, where ν^2 is the marginal covariance parameter, then the value of ν^2 that maximizes Equation 2.43 is

$$\hat{\nu}^2 = \frac{\sum_{i=1}^{M}\mathsf{tr}\Big(K_i^{-1}\sum_{g=1}^{G} q(z_i)(\mu_i\mu_i^\top + \Sigma_i)\Big)}{\sum_{i=1}^{M} n_i}. \tag{2.44}$$

Estimating σ^2. Estimating the noise variance σ^2 is straightforward once we've computed the posterior over z_i, f_i, and f_i'. The sufficient statistic for re-estimating σ^2 conditioned on the latent variables is simply the mean squared error:

$$\sigma^2 = \frac{\sum_{i=1}^{M} \|(y_i - \Phi(t_i)Bx_i - \Phi(t_i)\beta_{z_i} - f_i - f_i')\|_2^2}{\sum_{i=1}^{M} N_i} \tag{2.45}$$

To update σ^2 in the EM algorithm, we compute the expected value of each summand in the numerator under $q(z_i, f_i'')$, which is

$$\sum_{g=1}^{G} q_i(g)\Big(\|(y_i - \Phi(t_i)Bx_i - \Phi(t_i)\beta_g - A_i\mu''(g))\|_2^2 + \mathsf{tr}\big(A_i\Sigma''(g)A_i^\top\big)\Big) \tag{2.46}$$

2.6 Missing Data Assumptions

Trajectories in continuous-time can be thought of as random *functions* $F(\cdot)$ (Gaussian processes are an example of a family of distributions over random functions). Although the function specifies infinitely many values, to learn continuous-time models we maximize the probability of a finite set of observations (or a penalized version of this objective). In *observational* health care data, we need to be careful that we do not bias our likelihood-based learners by unduly ignoring the dependence between the finite set of times at which we observe the trajectory and the trajectory's values at those times. For example, if the trajectory is more likely to be sampled when its value is low, then our model will learn that trajectories with high values are less likely than they actually are.

The aim of this section is to posit a set of assumptions about continuous trajectory observation times that are (1) substantively reasonable, and (2) justify the use of standard likelihood-based learning. At a high-level, we assume that trajectory observation times are functions of the previous observation times and the values of the trajectory sampled at those times. These assumptions are more formally encoded in the graphical model shown in Figure 2.2, which expresses dependencies for an individual with three trajectory observations. In the figure, $F(\cdot)$ denotes the full trajectory, $\{T_1, T_2, T_3\}$ are random variables denoting the times at which the trajectory is sampled, and $\{Y_1^\star, Y_2^\star, Y_3^\star\}$ are the observed data. The conditional probability distribution of any $Y_i^\star$ given the

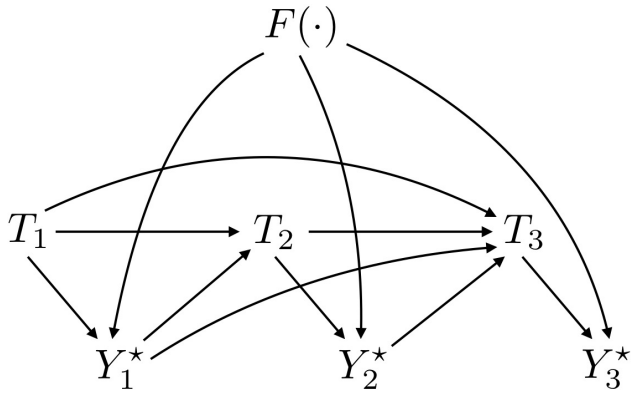


Figure 2.2: Example missing data mechanism in continuous-time.

trajectory and associated observation time is simply:

$$p(Y_i^\star = y_i^\star \mid T_i = t_i, F = f) = 1_{f(t_i)=y_i^\star}. \tag{2.47}$$

These assumptions are reasonable in many healthcare settings. For example, in an ICU where a patient is constantly under supervision, we can reasonably assume that clinical marker measurements are made at times that depend on the previous observations (e.g. the individual is thought to be at risk and so measurements are taken more frequently) and on previous observation times (e.g. a measurement has not been recorded in a while, so we should collect a new one). In the outpatient setting, an individual with a particular disease that is being actively managed by a physician will have follow-up visits scheduled either routinely or more frequently if the physician is especially concerned. On the other hand, modeling the progression of a disease such as the flu using information from a general practitioner's office may not satisfy our assumption because individual's with less severe manifestations are less likely to visit.

Conditioned on these assumptions about the dependencies between the trajectory, observation times, and observed values, we want to justify likelihood-based learning. Suppose we have a trajectory model with parameters Θ that allows us to compute the probability of any finite set of trajectory values. For example, we can compute $p_\Theta(F(t_1) = y_1^\star, F(t_2) = y_2^\star, F(t_3) = y_3^\star)$. The observed data, however, are the observation times and sampled values: $\{T_{1:n}, Y_{1:n}^\star\}$. Proper

likelihood-based learning requires that we maximize:

$$p(T_{1:n} = t_{1:n}, Y^{\star}_{1:n} = y^{\star}_{1:n}). \tag{2.48}$$

However, this expression is determined by both the observation time mechanism and the trajectory model. Our goal is to show that this can be factored into two terms: one that depends on the observed data and the observation time mechanism parameters, and the other that depends on the sampled trajectory values and the trajectory model parameters Θ. To do this, we first see that Equation 2.48 can be written as

$$\int p(F = f)p(T_{1:n} = t_{1:n}, Y^{\star}_{1:n} = y^{\star}_{1:n} \mid F = f)dF. \tag{2.49}$$

The integrand in Equation 2.49 can be now be factored further to obtain

$$p(F = f)\prod_{i=1}^{n} p(T_i = t_i \mid \mathcal{H}_i)p(Y^{\star}_i = y^{\star}_i \mid T_i = t_i, F = f), \tag{2.50}$$

where $\mathcal{H}_i$ is defined to be the previous $i-1$ observation times and sampled trajectory values. Note that the first term in the product of Equation 2.50 can be pulled out of the integral, allowing us to write Equation 2.49 as

$$\left[\prod_{i=1}^{n} p(T_i = t_i \mid \mathcal{H}_i)\right]\left[\int p(F = f)\prod_{i=1}^{n} p(Y^{\star}_i = y^{\star}_i \mid T_i = t_i, F = f)dF\right]. \tag{2.51}$$

The left-hand factor above depends only on the observation time mechanism and the observed data. Moreover, the right-hand factor depends only on the trajectory model and the sampled trajectory

values, which we now show:

$$\begin{aligned}
&\int p(F = f) \prod_{i=1}^{n} p(Y_i^\star = y_i^\star \mid T_i = t_i, F = f) dF \\
&\qquad = \int p(F = f) \prod_{i=1}^{n} 1_{f(t_i)=y_i^\star} dF \\
&\qquad = \int p(F = f) 1_{f(t_1)=y_1^\star,\ldots,f(t_n)=y_n^\star} dF \\
&\qquad = p_\Theta(f(t_1) = y_1^\star, \ldots, f(t_n) = y_n^\star). \qquad (2.52)
\end{aligned}$$

We therefore see that, given our observation time mechanism assumptions, maximizing the likelihood of the sampled trajectory values under our trajectory model is equivalent to maximizing the "proper" likelihood in Equation 2.48 with respect to the model parameters Θ. This result aligns with Theorems 7.1 and 8.1 found in Rubin's original paper on missing data [Rubin, 1976].

2.7 Experiments

The purpose of PSM is to discover subtypes, useful for tasks such as developing tailored treatment plans and advancing understanding of underlying disease mechanisms. Exploratory clustering method evaluations are typically two-fold. Quantitatively, we want to measure the model's fit to data. Qualitatively, we want to judge the insights that the model conveys. We therefore evaluate PSM using two experiments. First, we investigate PSM's ability to predict unobserved s-marker measurements. In the absence of ground-truth subtypes that we can use to compute a cluster-based metric, we instead measure the generalizability of PSM by evaluating posterior predictions of held-out s-marker observations.[1] Our second experiment involves qualitative analyses of the subtypes discovered by PSM. We evaluate the clinical merit of the prototypical disease activity trajectories discovered, and discuss follow-up clinical investigations that have stemmed from our results.

[1] Alternatively, one may use held-out data log-likelihood, but we choose predictive accuracy because the task of prediction is natural in the clinical setting and it is therefore easier to interpret the significance of the results.

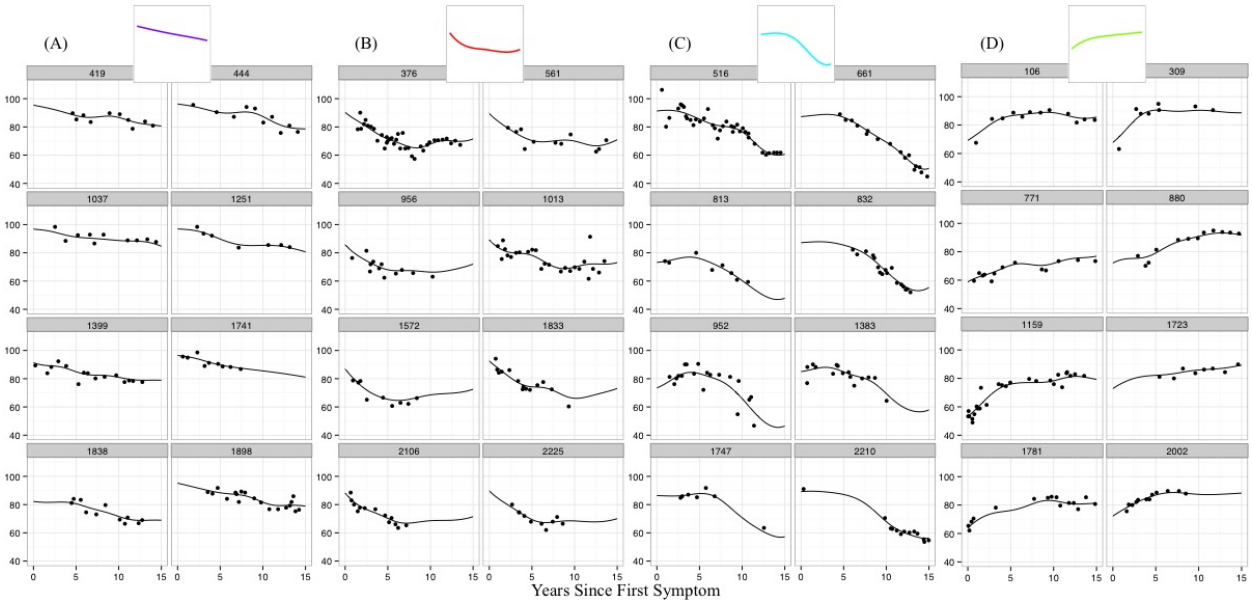


Figure 2.3: Example model fits to individual pFVC trajectories. Samples from four subtype candidates are displayed; one subtype per 4 by 2 block of individuals. Solid lines show full model fit (computed using Equation 2.37). Dots show observed pFVC values. Solid lines at the top of each block show the prototypical s-marker trajectory for that subtype.

2.7.1 Scleroderma S-markers

Scleroderma is a multi-system autoimmune disease resulting in insults that include damage to the skin, pulmonary system, and circulatory system. For our experiments we choose four datasets, each containing the s-marker corresponding to one of the major organ systems implicated in scleroderma. *Total skin score* (TSS) scores the thickness of the skin, which is used as a surrogate measure for the degree of fibrosis. An increased TSS indicates more extensive fibrosis across the individual's body. *Percent of predicted forced vital capacity* (pFVC) measures the volume of air expelled from the lung after maximum inhalation. pFVC is a measure of restricted ventilatory defect, which may reflect the severity of interstitial lung disease (ILD). *Percent of predicted diffusing capacity* (pDLCO) measures the efficiency of oxygen diffusion from the lungs to the bloodstream. Decreases in pDLCO indicate a defect in gas exchange, which is associated with development of pulmonary arterial hypertension (PAH). Finally, *right ventricular systolic pressure* (RVSP) measures systolic pressure in the chamber of the heart that directly pumps blood through the pulmonary vasculature. Significantly increased systolic pressure suggests an increased risk of heart failure due to pulmonary arterial hypertension.

We focus here on the analysis of subtypes for each of the complications individually. If subtypes exist, then a natural follow-up is to identify whether a common mechanism might jointly influence trajectories for two or more organ systems, but this relies on first developing an understanding of the individual s-markers; the focus of our analyses below.

2.7.2 Unobserved S-marker Prediction

Our first experiment evaluates the accuracy of s-marker value predictions at unobserved times for models that account for varying levels of nuisance variability. The first model includes covariate effects and group/subtype effects (C+G). The covariates provided to us were gender, African American race, and age at disease onset, which are well-known risk factors for severity in scleroderma [Varga et al., 2012]. The second model includes individual-specific long-term effects in addition to covariate and group effects (C+G+L). Finally, PSM includes covariate, group, long-term, and short-term effects.

To choose the number of groups G for each model, we use BIC as follows: we randomly generate five folds of the data by subsampling 75% of the individuals without replacement. For each model and for each choice of G, we compute the average BIC across the five folds and choose the number of clusters that results in the largest sequential drop in BIC (i.e. we search for the "elbow").

For each model, we use $p = 5$ bases and weak priors for the parameters: $\alpha = 2$, $\mu_\beta = \mu_B = 0$, and $\Sigma_\beta = \Sigma_B = \mathsf{diag}(10^5)$. For the long-term and short-term Gaussian processes, we use the following covariance function:

$$k(t_1, t_2) = \nu(1 + t_1 t_2) \tag{2.53}$$

$$k'(t_1, t_2) = a^2 * \exp\left\{-\frac{1}{2\ell^2}(t_1 - t_2)^2\right\}. \tag{2.54}$$

To choose the hyperparameters of for k and k' (ν, a, and ℓ) and the measurement noise σ^2, we perform a grid search using heldout log likelihood as the selection criterion. We search over $\nu \in \{10^{-5}, 10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}\}$. For a, ℓ, and σ^2 we search over the values $\{1, 2, 3, 4, 5\}$. We time-aligned each individual using years since first scleroderma related symptom. We also truncate

time at 15 years following the first scleroderma related symptom. We include an individual in the experiment if they have at least four measurements of a particular s-marker. Using these criteria, we included 1,011, 1,177, 1,114, and 504 individuals for TSS, pFVC, pDLCO, and RVSP respectively. Within this subset of individuals, the average number of measurements for each s-marker over the 15 year period is: 9.1 for TSS, 8.6 for pFVC, 8.3 for pDLCO, and 6.0 for RVSP. The average time in years between observed measurements for each s-marker is: 0.9 for TSS, 0.9 for pFVC, 1.0 for pDLCO, and 1.3 for RVSP.

To estimate prediction error, we split the individual trajectories into 10 groups and use 10-fold cross validation. We fit PSM on nine of the ten folds, and predict on the tenth fold. We select four s-marker observations from each held-out trajectory and assign each to a point-level fold. For each of the point-level folds, we condition on the remaining s-marker observations and use the MAP estimates of the held-out observations as our predictions (Equation 2.37). We compute the root mean squared error (RMSE) for each point-level fold across the trajectory-level folds, and finally compute the mean RMSE and standard errors across point-level folds.

We report the RMSEs and standard errors in Table 2.1. We see that PSM significantly outperforms the alternative models for TSS, pFVC, and RVSP. Although PSM has smallest RMSE for pDLCO, the difference is not statistically significant. Figure 2.3 displays model fits to individual trajectories sampled from four of the discovered candidate subtypes (one subtype per 4 by 2 block of individuals) for the pFVC s-marker. Note that blocks display overall similar behavior, but that long-term and short-term variability tailor predictions for each individual using the observed markers.

S-marker	C+G	C+G+L	PSM
TSS	5.32 ± 0.18	5.41 ± 0.07	$^{*}\mathbf{4.43} \pm 0.14$
pFVC	9.27 ± 0.49	9.34 ± 0.46	$^{*}\mathbf{7.69} \pm 0.39$
pDLCO	15.03 ± 1.82	15.13 ± 1.93	$\mathbf{14.08} \pm 1.77$
RVSP	12.21 ± 0.50	12.11 ± 0.44	$^{*}\mathbf{10.89} \pm 0.27$

Table 2.1: RMSE with standard errors for s-marker prediction. Bold shows best performance on s-marker; * shows statistical significance ($p \leq 0.05$).

2.7.3 Simulated Data Trajectory Estimate Accuracy

Another natural question is whether modeling these sources of variability reduces bias in our estimates of the prototypical trajectories. In other words, do the individual deviations cancel out so that PSM offers no benefit over less expressive models like C+G? For this, we turn to simulated data and investigate whether PSM recovers prototypical trajectories more accurately than C+G and C+G+L.

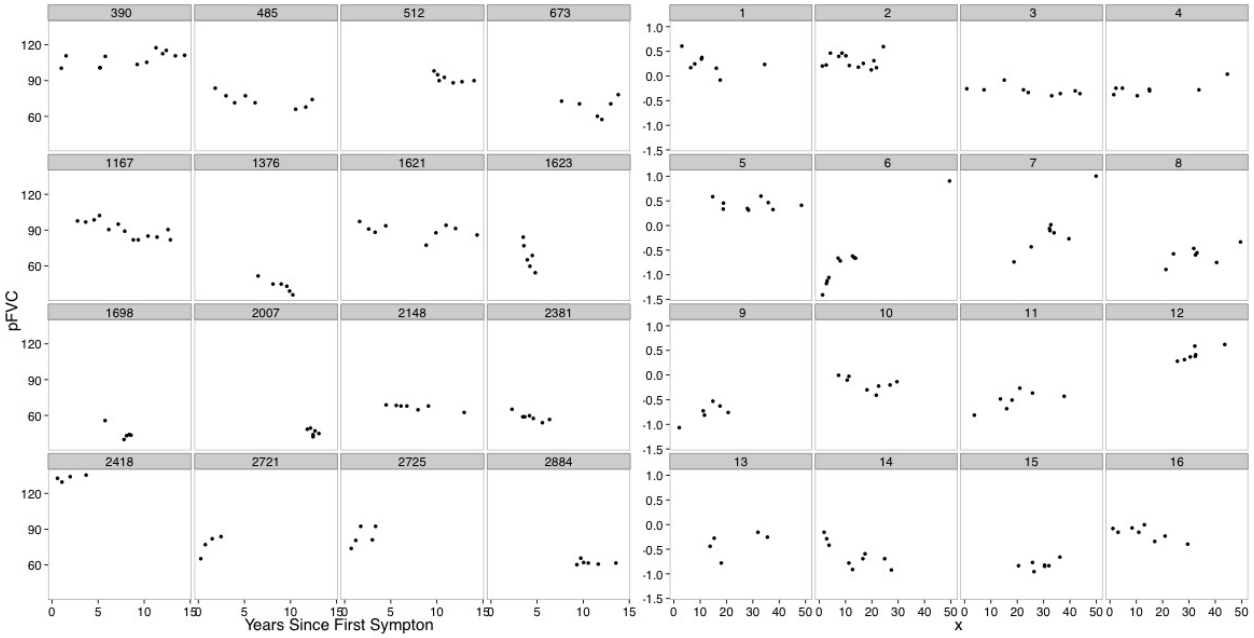
The simulation model samples observation time-stamps by sampling the N_i from a Poisson distribution, and, conditioned on N_i, samples $t_i \in \mathbb{R}^{N_i}$ from a Gaussian mixture model. The y_i are then sampled from a subtype mixture model with a hierarchy of individual-specific long-term, short-term, and iid noise.

To simulate an s-marker trajectory, we use an observation model and a measurement model. The observation model is used to select measurement times for each simulated trajectory i. We define a minimum number of observations N' and and an average number of additional observations λ_N. The number of observations for individual i is then sampled as

$$N_\delta \sim \mathsf{Poisson}(\lambda_N) \tag{2.55}$$

$$N_i \leftarrow N' + N_\delta. \tag{2.56}$$

Given the number of observations N_i, we use a Gaussian mixture model with an individual-specific number of components to sample the actual measurement times $t_i \in \mathbb{R}^{N_i}$. This is designed to replicate the sporadic observation patterns we have observed in our data. We have noted that many individuals have clusters of measurement activity rather than a consistent sampling frequency. Each curve has at least one Gaussian, and the number of additional Gaussians is drawn from a Poisson with mean parameter λ_g. For each Gaussian, location and scale parameters are drawn from a uniform distribution and inverse gamma distribution respectively. A mixing distribution is then drawn randomly from a Dirichlet distribution with concentration α. Finally, the N_i measurement times are sampled from the individual's Gaussian mixture model. The observation model generates



(A) Sample pFVC trajectories. (B) Sample simulated trajectories.

Figure 2.4: Comparison of pFVC trajectories and simulated trajectories.

measurement values by sampling from a subtype mixture model with a hierarchy of long-term, short-term, and iid noise as in PSM. Figures 2.4A and 2.4B show a sample of pFVC trajectories and simulated trajectories respectively. For each simulation, we sampled 200 s-marker trajectories from 4 subtypes with coefficients β_g sampled from a multivariate normal with mean 0 and identity covariate matrix. We set $N' = 4$ and $\lambda_N = 8$. Each individual-specific Gaussian mixture model had at least 1 component with the number of additional components drawn from Poiss(1). The scales for each Gaussian were drawn from $\mathsf{InverseGamma}(8, 8)$. In the Gaussian process, we set the bandwidth $\ell = 2.5$.

We compare the same three models used above, but do not simulate covariates and so they are not included. To measure bias, we find the alignment between estimated and true trajectories that minimizes the RMSE averaged across each estimated-true pair; to compute RMSE between two curves, we use a discrete approximation.

The iid noise $\sigma = 0.1$ for all simulations, and the left column of Table 2.2 shows the amplitude a of short-term individual variability and standard deviation of individual-specific intercept terms ν. Individual specific slope variance was set to 10^{-4} for all simulations. We see that as more

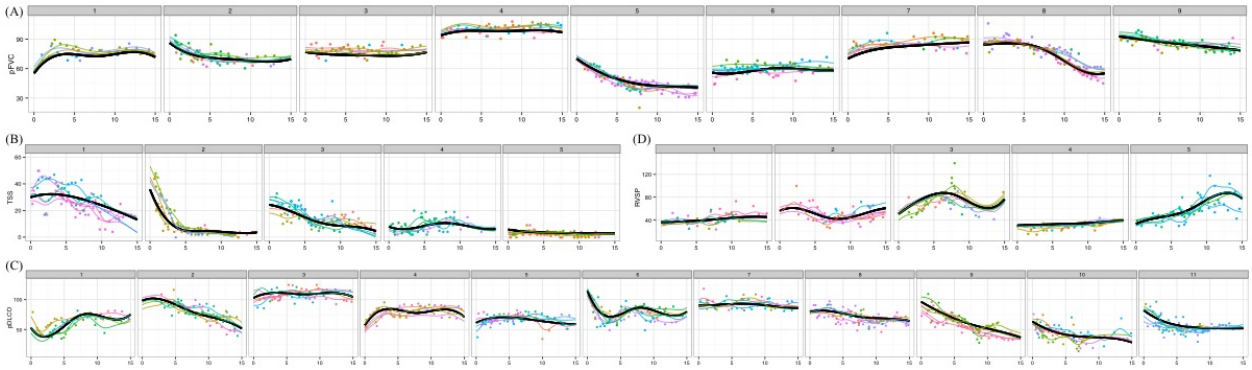


Figure 2.5: Discovered subtypes for all four s-markers. Panel (A) shows pFVC, panel (B) shows TSS, panel (C) shows pDLCO, and panel (D) shows RVSP. Prototypical s-marker trajectories are shown in black, and individuals sampled from the subtype are shown in color. Colored lines show the individualized s-marker trajectory, and colored points show the observed s-markers. Best viewed in color.

nuisance variability is added, PSM is able to recover less biased trajectory estimates. In Section A4 of the supplementary material, we provide plots that show example individual trajectories from these experiments and how they contribute to the bias of prototypical trajectory estimates.

(ν, a)	G	G+L	PSM
$(0.00, 0.10)$	0.27 ± 0.06	$^*\mathbf{0.04} \pm 0.01$	0.07 ± 0.06
$(0.00, 0.15)$	0.34 ± 0.05	$^*\mathbf{0.05} \pm 0.01$	0.09 ± 0.06
$(0.10, 0.15)$	0.34 ± 0.04	$\mathbf{0.11} \pm 0.05$	0.14 ± 0.08
$(0.15, 0.15)$	0.34 ± 0.05	0.18 ± 0.04	$^*\mathbf{0.13} \pm 0.06$
$(0.20, 0.15)$	0.36 ± 0.05	0.25 ± 0.07	$^*\mathbf{0.14} \pm 0.07$
$(0.25, 0.15)$	0.36 ± 0.06	0.32 ± 0.07	$^*\mathbf{0.18} \pm 0.04$

Table 2.2: Estimated trajectory RMSE and standard errors (computed over 20 replications) for simulations. Bold indicates best performance, statistical significance is indicated using * ($p \leq 0.05$).

2.7.4 Discovered Subtypes

We now present a qualitative discussion of the discovered subtypes. For all results below, we use the same setup as in the heldout s-marker prediction experiments. We begin with pFVC. Figure 2.5A displays the clusters we learn using PSM on pFVC trajectories of 1,177 individuals with $G = 9$ (chosen using BIC), and Figure 2.3 displays individual trajectory fits from 4 of the 9 clusters. We first focus on the subtypes displayed in panels (A), (B), and (C) of Figure 2.3. Each of these show distinct patterns of decline: individuals in (A) have a steady, linear progression, those in (B) decline quickly within the first five years and then stabilize, and those in (C) are stable for the first five to

ten years and then decline rapidly. Many of these individuals have at least one measurement that drops by more than 7% from the previous observation, which is clinically considered to suggest interstitial lung disease (see, for example, Beretta et al. 2007). It is clear, however, that they display unique patterns of decline, which has raised the question of whether individuals with these different subtypes differ with respect to their antibody profiles (since scleroderma is an autoimmune disease).

We now turn to panel (D). Fibrosis in the lungs due to end-stage interstitial lung disease is thought of as non-reversible damage. The individuals shown in panel (D), however, begin with inhibited pulmonary function, but slowly recover. This pattern of recovery warrants additional investigation; it is possible that in these patients, the initial insult is not due to end-stage lung disease, but rather due to other causes of restricted ventilatory defect such as inflammation. Recognizing this pattern may alter clinical management of these patients. Moreover, if it is the case that these patients all share a common comorbidity at the onset of disease, then it may suggest the presence of another subtype whose underlying mechanism triggers the comorbid condition.

Figure 2.5B displays the clusters learned by PSM for Total Skin Score (TSS) using $M = 1,011$ individuals with $G = 5$ (selected using BIC). TSS is a well-studied s-marker in scleroderma, and is used for one of the primary clinical classification criteria. Individuals with more extensive skin disease have higher TSS scores. Traditionally, skin disease in scleroderma is defined as limited (minimal involvement at the system level) or diffuse (systemic level involvement) [Varga et al., 2012]. Here we see five clusters: clusters 4 and 5 exhibit limited involvement, and clusters 1, 2, and 3 indicate different patterns of diffuse skin disease.

Figure 2.5C displays the clusters learned using PSM for pDLCO using $M = 1,114$ individuals with $G = 11$ (chosen using BIC). Clinicians monitor pDLCO to detect the onset of pulmonary arterial hypertension (PAH), one of the most prominent sources of mortality among patients with scleroderma. Once pDLCO is low enough, an individual will typically be screened using additional diagnostic tests [Steen and Medsger, 2003]. It is clear from Figure 2.5D, however, that there are several patterns of decline (seen in clusters 2, 8, 9, 10, and 11). Understanding whether particular

patterns of pDLCO decline are more predictive of PAH may help to develop more effective clinical heuristics.

Finally, Figure 2.5D displays the clusters learned using PSM for RVSP using $M = 504$ individuals with $G = 5$ (chosen using BIC). The RVSP results are noisier than the others, which may be due to the inherent noise in the measurement process. RVSP is measured using an echocardiogram, and is inaccurate when the true underlying systolic pressure is between 30 and 45 mmHg (millimeters mercury). We note that there are two groups (1 and 4) with stable, healthy pressures, which are presumably individuals with no serious circulatory complications. The remaining three clusters (2, 3, and 5) are more difficult to interpret because they are not associated with any known patterns of heart involvement; this may be attributable to the noisiness of the observations, or other phenomena that are as yet unknown. These remain the subject of future clinical follow up.

Joint Analysis of S-Markers

We have focused our analyses using PSM on single s-marker subtypes. A natural follow-up question is whether we can infer clusters across multiple s-markers. If we are analyzing K s-marker types, then, as presented, PSM assumes the following joint distribution over s-marker sequences y_i^k and memberships z_i^k:

$$\prod_{k=1}^{K} p(y_i^k \mid z_i^k, \Theta) p(z_i^k) \tag{2.57}$$

One alternative is to replace the fully factored distribution over $z_i^1, \ldots, z_i^K$ with a complete joint $p(z_i^1, \ldots, z_i^K)$ to induce correlations across s-marker types. We can inspect the posterior distribution over $z_i^1, \ldots, z_i^K$ to discover clusters defined over multiple s-markers.

A simpler alternative that does not depend modeling all s-markers jointly would be to represent each individual using a vector of categorical variables indicating the PSM-discovered subtype for each s-marker (e.g. [tss-type-1, pfvc-type-3, dlco-type-2, ...]). We can then discover clusters across multiple s-markers using a distance-based clustering method, such as hierarchical agglomerative clustering (HAC).

2.8 Discussion

This chapter describes the Probabilistic Subtyping Model, a novel method for clustering time series of clinical markers obtained from observational EHR data to discover patient subtypes with similar patterns of disease progression over time. We introduced the concept of unobserved heterogeneity— effects due to factors such as age, co-existing conditions, and genetic profiles— that affect the observed clinical test results, but are unrelated to the underlying disease mechanism and may not always be recorded in our data. The hierarchy of latent variables in PSM allows us to simultaneously estimate each individual's latent subtype and any unobserved factors. This allows us to remove the effects of those unobserved factors, and to discover more meaningful clinical trajectory subtypes. The subtypes that PSM discovers can be used to guide both medical practice and research. As we discover and refine subtypes for complex diseases, clinicians can associate individual patients with specific subtypes, and use the canonical disease trajectory to guide and improve decision-making. Subtypes are also valuable guides for basic medical research. Investigations that seek to understand the biological reasons behind variation across subtypes can yield important insights into how to best target the pathological mechanisms that drive the disease.

Chapter 3

Continuous Subtyping: Disease Trajectory Maps

In Chapter 2, we introduced the Probabilistic Subtyping Model (PSM) for discovering discrete, latent disease trajectory subtypes within a larger population of individuals. The assumption that discrete subpopulations exist, however, may not always be appropriate. For instance, disease activity in the respiratory system might not be easily characterized as either "on" or "off". Instead, it might be easier to characterize disease activity as a continuum, or spectrum, along which we can place an individual based on the extent to which their respiratory system has been affected.

In this chapter, we study this alternative perspective to exploring health trajectory data. We describe the "Disease Trajectory Map", a latent variable model that projects sparse and irregularly sampled health trajectory data into a low-dimensional space using a nonlinear transformation. There are two key advantages of this approach over discrete latent variable models. First, by projecting the health trajectory data into a two or three dimensional space, we can visualize a heterogeneous population using scatter plots. This might reveal interesting structure in the population that might otherwise be difficult to spot using the raw time series data or clustering methods. Second, the low-dimensional representations are compact numerical summaries (i.e. features) that we can use to represent complex time series data in studies of the relationship between disease trajectory and outcomes.

3.1 Related Work

Clinical marker data extracted from EHRs is a by-product of an individual's interactions with the healthcare system. As a result, the time series are often irregularly sampled (the time between samples varies within and across individuals), and may be extremely sparse (it is not unusual to have a single observation for an individual). To aid the following discussion, we briefly introduce notation for this type of data. We use M to denote the number of individual disease trajectories recorded in a given dataset. For each individual, we use N_i to denote the number of observations. We collect the observation times for subject i into a column vector $t_i \triangleq [t_{i1}, \ldots, t_{iN_i}]^\top$ (sorted in non-decreasing order) and the corresponding measurements into a column vector $y_i \triangleq [y_{i1}, \ldots, y_{iN_i}]^\top$. Our goal is to embed the pair (t_i, y_i) into a low-dimensional vector space wherein similarity between two embeddings x_i and x_j) implies that the trajectories have similar shapes. This is commonly done using *basis representations* of the trajectories.

3.1.1 Fixed Basis Representations

In the statistics literature, (t_i, y_i) is often referred to as *unbalanced longitudinal data*, and is commonly analyzed using linear mixed models (LMMs) [Verbeke and Molenberghs, 2009]. In their simplest form, LMMs assume the following probabilistic model:

$$w_i \mid \Sigma \sim \mathsf{MultNormal}(0, \Sigma) \tag{3.1}$$

$$y_i \mid w_i, B_i, \mu, \sigma^2 \sim \mathsf{MultNormal}(\mu + Bw_i, \sigma^2 \mathsf{I}_{N_i}). \tag{3.2}$$

The matrix $B_i \in \mathbb{R}^{N_i \times d}$ is known as the *design matrix*, and can be used to capture non-linear relationships between the observation times t_i and measurements y_i. Its rows are comprised of d-dimensional basis expansions of each observation time $B_i = [b(t_{i1}), \cdots, b(t_{iN_i})]^\top$. Common choices of $b(\cdot)$ include polynomials, splines, wavelets, and Fourier series. The particular basis used is often carefully crafted by the analyst depending on the nature of the trajectories and on the desired structure (e.g. invariance to translations and scaling) in the representation [Brillinger,

2001]. The design matrix can therefore make the LMM remarkably flexible despite its simple parametric probabilistic assumptions. Moreover, the prior over w_i and the conjugate likelihood make it straightforward to fit μ, Σ, and σ^2 using EM or Bayesian posterior inference.

After estimating the model parameters, we can estimate the coefficients w_i of a given clinical marker trajectory using the posterior distribution, which embeds the trajectory in a Euclidean space. To flexibly capture complex trajectory shapes, however, the basis must be high-dimensional, which makes interpretability of the representations challenging. We can use low-dimensional summaries such as the projection on to a principal subspace, but these are not necessarily substantively meaningful. Indeed, much research has gone into developing principal direction post-processing techniques (e.g. Kaiser [1958]) or alternative estimators that enhance interpretability (e.g. Carvalho et al. 2012).

3.1.2 Data-Adaptive Basis Representations

A set of related, but more flexible, techniques comes from functional data analysis where observations are functions (i.e. trajectories) assumed to be sampled from a stochastic process and the goal is to find a parsimonious representation for the data [Ramsay et al., 2002]. Functional principal component analysis (FPCA), one of the most popular techniques in functional data analysis, expresses functional data in the orthonormal basis given by the eigenfunctions of the auto-covariance operator. This representation is optimal in the sense that no other representation captures more variation [Ramsay, 2006]. The idea itself can be traced back to early independent work by Karhunen and Loeve and is also referred to as the Karhunen-Loeve expansion [Watanabe, 1965]. While numerous variants of FPCA have been proposed, the one that is most relevant to the problem at hand is that of sparse FPCA [Castro et al., 1986, Rice and Wu, 2001] where we allow sparse, irregularly sampled data as is common in longitudinal data analysis. To deal with the sparsity, Rice and Wu [2001] used LMMs to model the auto-covariance operator. When the data are sparse and the number of bases is large, however, the estimate of the covariance matrix using LMMs can have high variance. James et al. [2000] addressed this by constraining the rank of the covariance matrices—we

will refer to this model as the reduced-rank LMM, but note that it is a variant of sparse FPCA. Although sparse FPCA represents trajectories using a data-driven basis, the basis is restricted to lie in a linear subspace of a fixed basis, which may be overly restrictive. Other approaches to learning a functional basis include Bayesian estimation of B-spline parameters (e.g. Bigelow and Dunson 2012) and placing priors over reproducing kernel Hilbert spaces (e.g. MacLehose and Dunson 2009). Although flexible, these two approaches do not learn a low-dimensional representation.

3.1.3 Cluster-Based Representations

Mixture models and clustering approaches are also commonly used to represent and discover structure in time series data. Marlin et al. [2012] cluster time series data from the intensive care unit (ICU) using a mixture model and use cluster membership to predict outcomes. Schulam and Saria [2015] describe a probabilistic model that represents trajectories using a hierarchy of features, which includes "subtype" or cluster membership. LMMs have also been extended to have nonparametric Dirichlet process priors over the coefficients (e.g. Kleinman and Ibrahim 1998), which implicitly induce clusters in the data. Although these approaches flexibly model trajectory data, the structure they recover is a partition, which does not allow us to compare all trajectories in a coherent way as we can in a vector space.

3.1.4 Lexicon-Based Representations

Another line of research has investigated the discovery of motifs or repeated patterns in continuous time-series data for the purposes of succinctly representing the data as a string of elements of the discovered lexicon. These include efforts in the speech processing community to identify sub-word units (parts of words comparable to phonemes) in a data-driven manner [Varadarajan et al., 2008, Levin et al., 2013]. In computational healthcare, Saria et al. [2011] propose a method for discovering deformable motifs that are repeated in continuous time series data. These methods are, in spirit, similar to discretization approaches such as symbolic aggregate approximation (SAX) [Lin et al., 2007] and piecewise aggregate approximation (PAA) [Keogh et al., 2001] that are popular in

data mining, and aim to find compact descriptions of sequential data, primarily for the purposes of indexing, search, anomaly detection, and information retrieval. The focus in this paper is to learn representations for entire trajectories rather than discover a lexicon. Furthermore, we focus on learning a representation in a vector space where similarities among trajectories are captured through the standard inner product on $\mathbb{R}^d$.

3.1.5 Contributions

Our approach to simultaneously answering these questions is to embed individual disease trajectories into a low-dimensional vector space wherein similarity in the embedded space implies that two individuals have similar trajectories. Such a representation would naturally answer our first question, and could also be used to answer the second by comparing distributions over representations across groups defined by different outcomes. To learn these representations, we introduce a novel probabilistic model of longitudinal data, which we term the Disease Trajectory Map (DTM). In particular, the DTM models the trajectory over time of a single *clinical marker*, which is an observation or measurement recorded over time by clinicians that is used to track the progression of a disease (see e.g. Schulam et al. [2015]). Examples of clinical markers are pulmonary function tests or creatinine laboratory test results, which track lung and kidney function respectively. The DTM discovers low-dimensional (e.g. 2D or 3D) latent representations of clinical marker trajectories that are easy to visualize. We describe a stochastic variational inference algorithm for estimating the posterior distribution over the parameters and individual-specific representations, which allows our model to be easily applied to large datasets. To demonstrate the DTM, we analyze clinical marker data collected on individuals with the complex autoimmune disease scleroderma (see e.g. Allanore et al. 2015). We find that the learned representations capture interesting subpopulations consistent with previous findings, and that the representations suggest associations with important clinical outcomes not captured by alternative representations.

3.2 Disease Trajectory Maps

To motivate Disease Trajectory Maps (DTM), we begin with the reduced-rank LMM proposed by James et al. [2000]. We show that the reduced-rank LMM defines a Gaussian process with a covariance function that linearly depends on trajectory-specific representations. To define DTMs, we then use the kernel trick to make the dependence non-linear. Let $\mu \in \mathbb{R}$ be the marginal mean of the observations, $F \in \mathbb{R}^{d \times q}$ be a rank-q matrix, and σ^2 be the variance of measurement errors. As a reminder, $y_i \in \mathbb{R}^{N_i}$ denotes the vector of observed trajectory measurements, $B_i \in \mathbb{R}^{N_i \times d}$ denotes the individual's design matrix, and $x_i \in \mathbb{R}^q$ denotes the individual's representation. James et al. [2000] define the reduced-rank LMM using the following conditional distribution:

$$y_i \mid B_i, x_i, \mu, F, \sigma^2 \sim \mathsf{MultNormal}(\mu + B_i F x_i, \sigma^2 \mathsf{I}_{N_i}). \tag{3.3}$$

They assume an isotropic normal prior over x_i and marginalize to obtain the observed-data log-likelihood, which is then optimized with respect to $\{\mu, F, \sigma^2\}$. As in Lawrence [2004], we instead optimize x_i and marginalize F. By assuming a normal prior $\mathsf{Normal}(0, \alpha \mathsf{I}_q)$ over the rows of F and marginalizing we obtain:

$$y_i \mid B_i, x_i, \mu, \sigma^2, \alpha \sim \mathsf{MultNormal}(\mu, \alpha \langle x_i, x_i \rangle B_i B_i^\top + \sigma^2 \mathsf{I}_{N_i}). \tag{3.4}$$

Note that by marginalizing over F, we induce a joint distribution over all trajectories in the dataset. Moreover, this joint distribution is a Gaussian process with mean μ and the following covariance function defined across trajectories that depends on times $\{t_i, t_j\}$ and representations $\{x_i, x_j\}$:

$$\mathsf{Cov}(y_i, y_j \mid B_i, B_j, x_i, x_j, \mu, \sigma^2, \alpha) = \alpha \langle x_i, x_j \rangle B_i B_j^\top + 1_{i=j} \sigma^2 \mathsf{I}_{N_i} \tag{3.5}$$

This reformulation of the reduced-rank LMM highlights that the covariance across trajectories i and j depends on the inner product between the two representations x_i and x_j, and suggests

that we can non-linearize the dependency with an inner product in an expanded feature space using the "kernel trick". Let $k(\cdot, \cdot)$ denote a non-linear kernel defined over the representations with parameters θ, then we have:

$$\mathsf{Cov}(y_i, y_j \mid B_i, B_j, x_i, x_j, \mu, \sigma^2, \theta) = k(x_i, x_j) B_i B_j^\top + 1_{i=j} \sigma^2 \mathsf{I}_{N_i}. \tag{3.6}$$

Let $y \triangleq [y_1^\top, \ldots, y_M^\top]^\top$ denote the column vector obtained by concatenating the measurement vectors from each trajectory. The joint distribution over y is a multivariate normal:

$$y \mid B_{1:M}, x_{1:M}, \mu, \sigma^2, \theta \sim \mathsf{MultNormal}(\mu, \Sigma_{\text{DTM}} + \sigma^2 \mathsf{I}_N), \tag{3.7}$$

where Σ_{DTM} is a covariance matrix that depends on the times $t_{1:m}$ (through design matrices $B_{1:m}$) and representations $x_{1:m}$. In particular, Σ_{DTM} is a block-structured matrix with M row blocks and M column blocks. The block at the i^{th} row and j^{th} column is the covariance between y_i and y_j defined in (3.6). Finally, we place isotropic Gaussian priors over x_i. We use Bayesian inference to obtain a posterior Gaussian process and to estimate the representations. We tune hyperparameters by maximizing the observed-data log likelihood. Note that our model is similar to the Bayesian GPLVM [Titsias and Lawrence, 2010], but models functional data instead of finite-dimensional vectors.

3.2.1 Learning and Inference

As formulated, the model scales poorly to large datasets. Inference within each iteration of an optimization algorithm, for example, requires storing and inverting Σ_{DTM}, which requires $O(N^2)$ space and $O(N^3)$ time respectively, where $N \triangleq \sum_{i=1}^M N_i$ is the number of clinical marker observations. For modern datasets, where N can be in the hundreds of thousands or millions, this is unacceptable. In this section, we approximate the log-likelihood using techniques from Hensman et al. [2013] that allow us to apply stochastic variational inference (SVI) [Hoffman et al., 2013].

Inducing Points

Recent work in scaling Gaussian processes to large datasets has focused on the idea of *inducing points* [Snelson and Ghahramani, 2005, Titsias, 2009], which are a relatively small number of artificial observations of a Gaussian process that approximately capture the information contained in the training data. In general, let $f \in \mathbb{R}^M$ denote observations of a GP at inputs $\{x_i\}_{i=1}^M$ and $u \in \mathbb{R}^P$ denote inducing point values at inputs $\{z_i\}_{i=1}^P$. Titsias [2009] constructs the inducing points as variational parameters by introducing an augmented probability model:

$$u \sim \mathsf{MultNormal}(0, K_{PP}) \tag{3.8}$$

$$f \mid u \sim \mathsf{MultNormal}(K_{MP}K_{PP}^{-1}u, \tilde{K}_{MM}), \tag{3.9}$$

where K_{PP} is the Gram matrix between inducing points, K_{MM} is the Gram matrix between observations, K_{MP} is the cross Gram matrix between observations and inducing points, and $\tilde{K}_{MM} \triangleq K_{MM} - K_{MP}K_{PP}^{-1}K_{PM}$. We can marginalize over u to construct a low-rank approximate covariance matrix, which is computationally cheaper to invert using the Woodbury identity. Alternatively, Hensman et al. [2013] extends these ideas by explicitly maintaining a variational distribution over u that d-separates the observations and satisfies the conditions required to apply SVI [Hoffman et al., 2013]. Let $y_f = f + \epsilon$ where $\epsilon \in \mathbb{R}^P$ is iid Gaussian noise with variance σ^2, then we use the following inequality to lower bound our data log-likelihood:

$$\log p(y_f \mid u) \geq \sum_{i=1}^{M} \mathbb{E}_{f_i|u}[\log p(y_{fi} \mid fi)]. \tag{3.10}$$

In the interest of space, we refer the interested reader to Hensman et al. [2013] for details.

Evidence Lower Bound

When marginalizing over the rows of F, we induced a Gaussian process over the trajectories, but by doing so we also implicitly induced a Gaussian process over the individual-specific basis coefficients. Let $w_i \triangleq Fx_i \in \mathbb{R}^d$ denote the basis weights implied by the mapping F and representation x_i in

the reduced-rank LMM, and let $w_{:,k}$ for $k \in [d]$ denote the k^{th} coefficient of all individuals in the dataset. After marginalizing the k^{th} row of F and applying the kernel trick, we see that the vector of coefficients $w_{:,k}$ has a Gaussian process distribution with mean zero and covariance function: $\mathsf{Cov}(w_{ik}, w_{jk}) = \alpha k(x_i, x_j)$. Moreover, the Gaussian processes across coefficients are statistically independent of one another. To lower bound the DTM log-likelihood, we introduce P inducing points u_k for each vector of coefficients $w_{:,k}$ with shared inducing point inputs $\{z_i\}_{i=1}^P$. To refer to all inducing points simultaneously, we use $U \triangleq [u_1, \ldots, u_d]$ and $u = \mathsf{vec_c}(U)$ to denote the "vectorized" form of U obtained by stacking its columns. Applying (3.10) we have:

$$\begin{aligned}\log p(y \mid U, x_{1:m}) &\geq \sum_{i=1}^{M} \mathbb{E}_{w_i|U,x_i)}[\log p(y_i \mid w_i)] \\ &= \sum_{i=1}^{M} \log \mathsf{MultNormal}(y_i \mid \mu + B_i U^\top K_{PP}^{-1} k_i, \sigma^2 \mathsf{I}_{N_i}) - \frac{\tilde{k}_{ii}}{2\sigma^2} \mathsf{tr}(B_i^\top B_i) \triangleq \sum_{i=1}^{M} \log \tilde{p}(y_i \mid U, x_i),\end{aligned}$$

where $k_i \triangleq [k(x_i, z_1), \ldots, k(x_i, z_P)]^\top$ and $\tilde{k}_{ii}$ is the i^{th} diagonal element of $\tilde{K}_{MM}$. We can then construct the variational lower bound on $\log p(y)$:

$$\log p(y) \geq \mathbb{E}\big[\log p(y \mid U, x_{1:M})\big] - \mathsf{KL}\big(q(U, x_{1:M}) \| p(U, x_{1:M})\big) \tag{3.11}$$

$$\geq \sum_{i=1}^{M} \mathbb{E}\big[\log \tilde{p}(y_i \mid U, x_i)\big] - \mathsf{KL}\big(q(U, x_{1:M}) \| p(U, x_{1:M})\big), \tag{3.12}$$

where we use the lower bound in (3.11). Finally, to make the lower bound concrete we specify the variational distribution $q(U, x_{1:M})$ to be a product of independent multivariate normal distributions:

$$q(U, x_{1:M}) \triangleq \mathsf{MultNormal}(\mathsf{vec_c}(U) \mid m, S) \prod_{i=1}^{M} \mathsf{MultNormal}(x_i \mid m_i, S_i), \tag{3.13}$$

where the variational parameters to be fit are m, S, and $\{m_i, S_i\}_{i=1}^M$.

Stochastic Optimization of the Lower Bound

To apply SVI, we must be able to compute the gradient of the expected value of $\log \tilde{p}(y_i \mid U, x_i)$ under the variational distributions. Because U and x_i are assumed to be independent in the

variational posteriors, we can analyze the expectation in either order. Fix x_i, then we see that $\log \tilde{p}(y_i \mid U, x_i)$ depends on U only through the mean of the Gaussian density, which is a quadratic term in the log likelihood. Because $q(U)$ is multivariate normal, we can compute the expectation in closed form.

$$\begin{aligned}
&\mathbb{E}_{q(U)}[\log \tilde{p}(y_i \mid U, x_i)] \\
&\quad = \mathbb{E}_{q(U)}[\log \mathsf{MultNormal}(y_i \mid \mu + B_i \otimes k_i^\top K_{PP}^{-1} u, \sigma^2 \mathsf{I}_{N_i})] - \frac{\tilde{k}_{ii}}{2\sigma^2} \mathsf{tr}(B_i^\top B_i) \\
&\quad = \log \mathsf{MultNormal}(y_i \mid \mu + C_i m, \sigma^2 \mathsf{I}_{N_i}) - \frac{1}{2\sigma^2} \mathsf{tr}(S C_i^\top C_i) - \frac{\tilde{k}_{ii}}{2\sigma^2} \mathsf{tr}(B_i^\top B_i),
\end{aligned}$$

where we have defined $C_i \triangleq B_i \otimes k_i K_{PP}^{-1}$ to be the *extended design matrix* and $\otimes$ is the Kronecker product. We now need to compute the expectation of this expression with respect to $q(x_i)$, which entails computing the expectations of k_i (a vector) and $k_i k_i^\top$ (a matrix). In this paper, we assume an RBF kernel, and so the elements of the vector and matrix are all exponentiated quadratic functions of x_i. This makes the expectations straightforward to compute given that $q(x_i)$ is multivariate normal.[1] We therefore see that the expected value of $\log \tilde{p}(y_i)$ can be computed in closed form under the assumed variational distribution.

We use the standard SVI algorithm to optimize the lower bound. We subsample the data, optimize the likelihood of each example in the batch with respect to the variational parameters over the representation (k_i, S_i), and compute approximate gradients of the global variational parameters (m, S) and the hyperparameters. The likelihood term is conjugate to the prior over U, and so we can compute the natural gradients with respect to the global variational parameters m and S [Hoffman et al., 2013, Hensman et al., 2013]. Additional details on the approximate objective and the gradients required for SVI are given in the supplement of Schulam and Arora [2016]. We provide details on initialization, minibatch selection, and learning rates for our experiments below.

[1] Other kernels can be used instead, but the expectations may not have closed form expressions.

Inference on New Trajectories

The variational distribution over the inducing point values u can be used to approximate a *posterior process* over the basis coefficients w_i [Hensman et al., 2013]. Therefore, given a representation x_i, we have that

$$w_{ik} \mid x_i, m, S \sim \mathsf{Normal}(k_i^\top K_{pp}^{-1} m_k, \tilde{k}_{ii} + k_i^\top K_{pp}^{-1} S_{kk} K_{PP}^{-1} k_i), \tag{3.14}$$

where m_k is the approximate posterior mean of the k^{th} column of U and S_{kk} is its covariance. The approximate joint posterior distribution over all coefficients can be shown to be multivariate normal. Let $\mu(x_i)$ be the mean of this distribution given representation x_i and $\Sigma(x_i)$ be the covariance, then the posterior predictive distribution over a new trajectory y_* given the representation x_* is

$$y_* \mid x_* \sim \mathsf{Normal}(\mu + B_* \mu(x_*), B_* \Sigma(x_*) B_*^\top + \sigma^2 \mathsf{I}_{N_*}). \tag{3.15}$$

We can then approximately marginalize with respect to the prior over x_* or a variational approximation of the posterior given a partial trajectory using a Monte Carlo estimate.

3.3 Experiments

We now use DTM to analyze clinical marker trajectories of individuals with the autoimmune disease, scleroderma [Allanore et al., 2015]. Scleroderma is a heterogeneous and complex chronic autoimmune disease. It can potentially affect many of the visceral organs, such as the heart, lungs, kidneys, and vasculature. Any given individual may experience only a subset of complications, and the timing of the symptoms relative to disease onset can vary considerably across individuals. Moreover, there are no known biomarkers that accurately predict an individual's disease course. Clinicians and medical researchers are therefore interested in characterizing and understanding disease progression patterns. Moreover, there are a number of clinical outcomes responsible for the majority of morbidity among patients with scleroderma. These include congestive heart failure,

pulmonary hypertension and pulmonary arterial hypertension, gastrointestinal complications, and myositis [Varga et al., 2012]. We use the DTM to study associations between these outcomes and disease trajectories.

We study two scleroderma clinical markers. The first is the percent of predicted forced vital capacity (PFVC); a pulmonary function test result measuring lung function. PFVC is recorded in percentage points, and a higher value (near 100) indicates that the individual's lungs are functioning as expected. The second clinical marker that we study is the total modified Rodnan skin score (TSS). Scleroderma is named after its effect on the skin, which becomes hard and fibrous during periods of high disease activity. Because it is the most clinically apparent symptom, many of the current sub-categorizations of scleroderma depend on an individual's pattern of skin disease activity over time [Varga et al., 2012]. To systematically monitor skin disease activity, clinicians use the TSS which is a quantitative score between 0 and 55 computed by evaluating skin thickness at 17 sites across the body (higher scores indicate more active skin disease).

3.3.1 Experimental Setup

For our experiments, we extract trajectories from the Johns Hopkins Hospital Scleroderma Center's patient registry; one of the largest in the world. For both PFVC and TSS, we study the trajectory from the time of first symptom until ten years of follow-up. The PFVC dataset contains trajectories for 2,323 individuals and the TSS dataset contains 2,239 individuals. The median number of observations per individuals is 3 for the PFVC data and 2 for the TSS data. The maximum number of observations is 55 and 22 for PFVC and TSS respectively.

We present two sets of results. First, we visualize groups of similar trajectories obtained by clustering the representations learned by DTM. Although not quantitative, we use these visualizations as a way to check that the DTM uncovers subpopulations that are consistent with what is currently known about scleroderma. Second, we use the learned representations of trajectories obtained using the LMM, the reduced-rank LMM (which we refer to as FPCA), and the DTM to statistically test for relationships between important clinical outcomes and learned disease trajec-

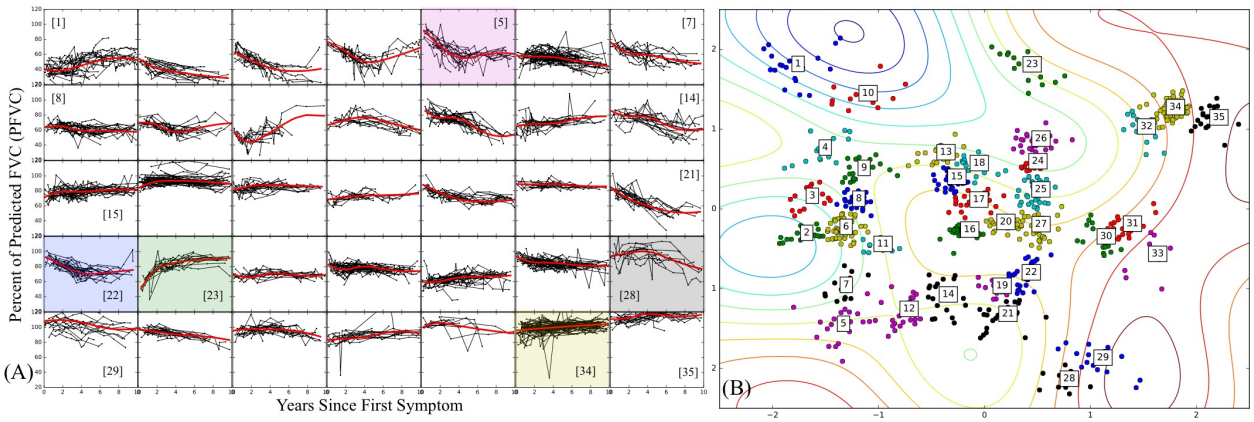


Figure 3.1: (A) Groups of PFVC trajectories obtained by hierarchical clustering of DTM representations. (B) Trajectory representations are color-coded and labeled according to groups shown in (A). Contours reflect posterior GP over the second B-spline coefficient (blue contours denote smaller values, red denote larger values).

tory representations.

For all experiments and all models, we use a common 5-dimensional B-spline basis composed of degree-2 polynomials (see e.g. Chapter 20 in Gelman et al. [2014]). We choose knots using the percentiles of observation times across the entire training set [Ramsay et al., 2002]. For LMM and FPCA, we use EM to fit model parameters. To fit the DTM, we use the LMM estimate to set the mean μ , noise σ^2, and average the diagonal elements of Σ to set the kernel scale α. Length-scales ℓ are set to 1. For these experiments, we do not learn the kernel hyperparameters during optimization. We initialize the variational means over x_i using the first two unit-scaled principal components of w_i estimated by LMM and set the variational covariances to be diagonal with standard deviation 0.1. For both PFVC and TSS, we use minibatches of size 25 and learn for a total of five epochs (passes over the training data). The initial learning rate for m and S is 0.1 and decays as t^{-1} for each epoch t.

3.3.2 Qualitative Analysis of Representations

The DTM returns approximate posteriors over the representations $\mathbf{x}_i$ for all individuals in the training set. We examine these posteriors for both the PFVC and TSS datasets to check for consistency with what is currently known about scleroderma disease trajectories. In Figure 3.1 (A) we show groups of trajectories uncovered by clustering the posterior means over the representations,

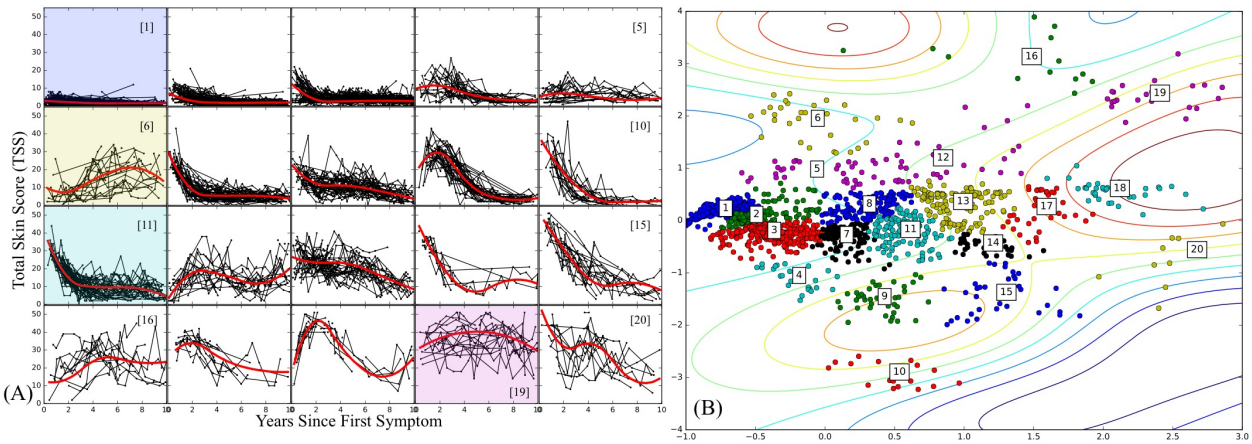


Figure 3.2: Same presentation as in Figure 3.1 but for TSS trajectories.

which are plotted in Figure 3.1 (B). Many of the groups shown here align with other work on scleroderma lung disease subtypes (e.g. Schulam et al. 2015). In particular, we see rapidly declining trajectories (group [5]), slowly declining trajectories (group [22]), recovering trajectories (group [23]), and stable trajectories (group [34]). Surprisingly, we also see a group of individuals who we describe as "late decliners" (group [28]). These individuals are stable for the first 5-6 years, but begin to decline thereafter. This is surprising because the onset of scleroderma-related lung disease is currently thought to occur early in the disease course [Varga et al., 2012]. In Figure 3.2 (A) we show clusters of TSS trajectories and the corresponding mean representations in Figure 3.2 (B). These trajectories corroborate what is currently known about skin disease in scleroderma. In particular, we see individuals who have minimal activity (e.g. group [1]) and individuals with early activity that later stabilizes (e.g. group [11]), which correspond to what are known as the limited and diffuse variants of scleroderma [Varga et al., 2012]. We also find that there are a number of individuals with increasing activity over time (group [6]) and some whose activity remains high over the ten year period (group [19]). These patterns are not currently considered to be canonical trajectories and warrant further investigation.

3.3.3 Associations between Representations and Clinical Outcomes

To quantitatively evaluate the low-dimensional representations learned by the DTM, we statistically test for relationships between the representations of clinical marker trajectories and impor-

Table 3.1: Disease Trajectory Held-out Log-Likelihoods

	PFVC		TSS	
Model	Subj. LL	Obs. LL	Subj. LL	Obs. LL
LMM	-17.59 (± 1.18)	-3.95 (± 0.04)	-13.63 (± 1.41)	-3.47 (± 0.05)
FPCA	-17.89 (± 1.19)	-4.03 (± 0.02)	-13.76 (± 1.42)	-3.47 (± 0.05)
DTM	-17.74 (± 1.23)	-3.98 (± 0.03)	-13.25 (± 1.38)	-3.32 (± 0.06)

Table 3.2: P-values under the null hypothesis that the distributions of trajectory representations are the same across individuals with and without clinical outcomes. Lower values indicate stronger support for rejection.

	PFVC			TSS		
Outcome	LMM	FPCA	DTM	LMM	FPCA	DTM
Congestive Heart Failure	0.170	0.081	0.013	0.107	0.383	0.189
Pulmonary Hypertension	0.270	*0.000	*0.000	0.485	0.606	0.564
Pulmonary Arterial Hypertension	0.013	0.020	*0.002	0.712	0.808	0.778
Gastrointestinal Complications	0.328	0.073	0.347	0.026	0.035	0.011
Myositis	0.337	*0.002	*0.004	*0.000	*0.002	*0.000
Interstitial Lung Disease	*0.000	*0.000	*0.000	0.553	0.515	0.495
Ulcers and Gangrene	0.410	0.714	0.514	0.573	0.316	*0.009

tant clinical outcomes. We compare the inferences of the hypothesis test with those made using representations derived from the LMM and FPCA baselines. For the LMM, we project w_i into its 2-dimensional principal subspace. For FPCA, we learn a rank-2 covariance, which learns 2-dimensional representations. To establish that the models are all equally expressive and achieve comparable generalization error, we present held-out data log-likelihoods in Table 3.1, which are estimated using 10-fold cross-validation. We see that the models are roughly equivalent with respect to generalization error.

To test associations between clinical outcomes and learned representations, we use a kernel

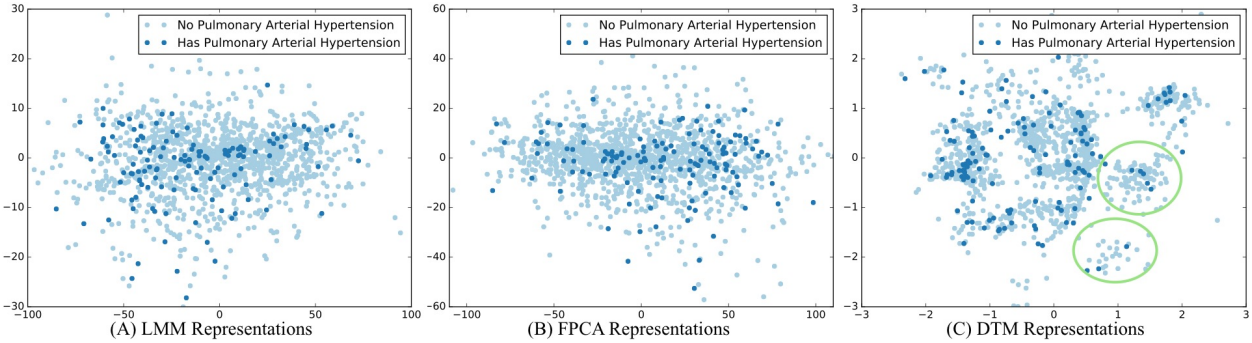


Figure 3.3: Scatter plots of PFVC representations for the three models color-coded by presence or absence of pulmonary arterial hypertension (PAH). Groups of trajectories with very few cases of PAH are circled in green.

density estimator test [Duong et al., 2012] to test the null hypothesis that the distributions across subgroups with and without the outcome are equivalent. The p-values obtained are listed in Table 3.2. As a point of reference, we include two clinical outcomes that should be clearly related to the two clinical markers. Interstitial lung disease is the most common cause of lung damage in scleroderma [Varga et al., 2012], and so we confirm that the null hypothesis is rejected for all three PFVC representations. Similarly, for TSS we expect ulcers and gangrene to be associated with severe skin disease. In this case, only the representations learned by DTM reveal this relationship. For the remaining outcomes, we see that FPCA and DTM reveal similar associations, but that only DTM suggests a relationship with pulmonary arterial hypertension (PAH). Presence of fibrosis (which drives lung disease progression) has been shown to be a risk factor in the development of PAH (see Chapter 36 of Varga et al. [2012]), but only the representations learned by DTM corroborate this association (see Figure 3.3).

3.4 Discussion

We presented the Disease Trajectory Map (DTM), a novel probabilistic model that learns low-dimensional embeddings of sparse and irregularly sampled clinical time series data. The DTM is a reformulation of the LMM. We derived it using an approach comparable to that of Lawrence [2004] in deriving the Gaussian process latent variable model (GPLVM) from probabilistic principal component analysis (PPCA) [Tipping and Bishop, 1999], and indeed the DTM can be interpreted as a "twin kernel" GPLVM (briefly discussed in the concluding paragraphs) over functional observations. The DTM can also be viewed as an LMM with a "warped" Gaussian prior over the random effects (see e.g. Damianou et al. [2015] for a discussion of distributions induced by mapping Gaussian random variables through non-linear maps). We demonstrated the model by analyzing data extracted from one of the nation's largest scleroderma patient registries, and found that the DTM discovers structure among trajectories that is consistent with previous findings and also uncovers several surprising disease trajectory shapes. We also explore associations between important clinical outcomes and the DTM's representations and found statistically significant differences in

representations between outcome-defined groups that were not uncovered by two sets of baseline representations.

Chapter 4

Dynamic Personalized Disease Trajectory Prediction

In Chapters 2 and 3, we described techniques for *exploring* disease trajectory data. Exploration can shed light on complex disease and can lead to new scientific hypotheses about the underlying mechanisms. In this chapter, we build on the ideas developed so far (those in PSM in particular) to develop tools for *predicting* disease trajectories. Predictions of disease activity trajectories are valuable for a number of reasons. For instance, the most effective treatments for many diseases often have strong side effects. With an accurate forecast of future progression, physicians can give more aggressive treatment to those who need it most. Predictions are also valuable for clinical trial enrichment. By enrolling patients at highest risk for particular outcomes, it may be possible to demonstrate that a therapy works using fewer subjects (i.e. the study has higher power). Accurately predicting disease progression can therefore have a large impact on clinical decision-making and also help to reduce the number of failed clinical trials in complex, heterogeneous diseases.

Predicting disease activity trajectories presents a number of challenges. As in subtyping, there are multiple *observed and latent factors* that cause heterogeneity across individuals. One possible factor is the individual's subtype. Depending on which mechanism is driving the disease, there may be very different overall trajectories (e.g. as in Figures 4.1a and 4.1b). If we knew the subtypes, it would be straightforward to fit separate models to each subpopulation. In most complex diseases,

however, the mechanisms are poorly understood and clear definitions of subtypes do not exist. Other important latent factors are *long-term, individual-specific* characteristics such as behavior, prior exposures, and genetic predispositions. For instance, a chronic smoker will typically have unhealthy lungs and so may have a trajectory that is consistently lower than a non-smoker's. Individual-specific long term factors may not always recorded reliably in electronic health data. Finally, an individual's trajectory might be influenced by *short-term factors*—e.g. an infection that temporarily harms respiratory function (similar to the "dips" in Figure 4.1c or the third row in Figure 4.1d). The causes of these short-term trends are also rarely recorded in electronic health record data. Without predictors that capture these many important factors, standard predictive models will not have a sufficiently rich set of inputs to make accurate predictions.

In this chapter, we describe a predictive model of disease activity trajectories that directly addresses these observed and latent sources of heterogeneity using components arranged into four conceptual layers: a population level (observed factors), a subpopulation level (unobserved, shared across individuals), individual-specific long term effects (unobserved, individual-specific), and individual-specific short term effects (unobserved, individual- and time-specific). Together, these four components allow individual trajectories to appear highly heterogeneous while simultaneously sharing statistical strength across observations at different "resolutions" of the data. When making predictions for a given individual, we use Bayesian inference to dynamically update our posterior belief over the unobserved components of this hierarchy given the clinical history, and use the posterior predictive to produce a trajectory estimate. We evaluate this approach by developing a state-of-the-art trajectory prediction tool for lung disease in scleroderma. We train the model using a large, national dataset containing individuals with scleroderma tracked over 20 years and compare our predictions against alternative state-of-the-art approaches. We find that our approach yields significant gains in predictive accuracy of disease activity trajectories. Importantly, the accuracy of our method improves over time as more clinical data is collected. This makes it possible to fully leverage the complete set of up-to-date information from a patient's medical history to, for example, guide clinical management or drive trial enrollment.

4.1 Related Work

The majority of predictive models in medicine explain variability in the target outcome by conditioning on observed risk factors alone. However, these do not account for latent sources of variability such as those discussed above. Further, these models are typically cross-sectional—they use features from data measured up until the current time to predict a clinical marker or outcome at a fixed point in the future. As an example, consider the mortality prediction model by Lee et al. [2003], where logistic regression is used to integrate features into a prediction about the probability of death within 30 days for a given patient. To predict the outcome at multiple time points, typically separate models are trained. Moreover, these models use data from a fixed-size window, rather than a growing history.

Researchers in the statistics and machine learning communities have proposed solutions that address a number of these limitations. Most related to our work is that by Rizopoulos [2011], where the focus is on making dynamical predictions about a time-to-event outcome (e.g. time until death). Their model updates predictions over time using all previously observed values of a longitudinally recorded marker. Besides conditioning on observed factors, they account for latent heterogeneity across individuals by allowing for individual-specific adjustments to the population-level model—e.g. for a longitudinal marker, deviations from the population baseline are modeled using random effects by sampling individual-specific intercepts from a common distribution. Other closely related work by Proust-Lima et al. [2014] tackle a similar problem as Rizopoulos [2011], but address heterogeneity using a mixture model.

Another common approach to dynamical predictions is to use Markov models such as order-p autoregressive models (AR-p), HMMs, state space models, and dynamic Bayesian networks (see e.g. Murphy 2012). Although such models naturally make dynamic predictions using the full history by forward-filtering, they typically assume discrete, regularly-spaced observation times. Gaussian processes (GPs) are a commonly used alternative for handling continuous-time observations—see Roberts et al. [2013] for a recent review of GP time series models. Since Gaussian processes are non-parametric generative models of functions, they naturally produce functional predictions

dynamically by using the posterior predictive conditioned on the observed data. Mixtures of GPs have been applied to model heterogeneity in the covariance structure across time series (e.g. Shi et al. 2005), however as noted Roberts et al. [2013], appropriate mean functions are critical for accurate forecasts using GPs. In our work, an individual's trajectory is expressed as a GP with a highly structured mean comprising population, subpopulation and individual-level components where some components are observed and others require inference.

More broadly, multi-level models have been applied in many fields to model heterogeneous collections of units that are organized within a hierarchy [Gelman and Hill, 2006]. For example, in predicting student grades over time, individuals within a school may have parameters sampled from the school-level model, and the school-level model parameters in turn may be sampled from a county-specific model. In our setting, the hierarchical structure—which individuals belong to the same subgroup—is not known *a priori*. Similar ideas are studied in multi-task learning, where relationships between distinct prediction tasks are used to encourage similar parameters. This has been applied to modeling trajectories by treating predictions at each time point as a separate task and enforcing similarity between sub-models close in time [Wang et al., 2012]. This approach is limited, however, in that it models a finite number of times. Others, more recently, have developed models for disease trajectories (see Ross and Dy 2013, Schulam et al. 2015 and references within) but these focus on retrospective analysis to discover disease etiology rather than dynamical prediction. Schulam et al. [2015] incorporate differences in trajectories due to subtypes and individual-specific factors. We build upon this work here. Finally, recommender systems also share information across individuals with the aim of tailoring predictions (see e.g. Marlin 2003, Adomavicius and Tuzhilin 2005, Sontag et al. 2012), but the task is otherwise distinct from ours.

4.2 Latent-Hierarchy Trajectory Model

To predict an individual's disease trajectory, we build off of the Probabilistic Subtyping Model (PSM) described in Chapter 2. As in the PSM, we model a measurement y_{ij} on individual i at time t_{ij} using a hierarchy observed and unobserved factors: a population component, a subpop-

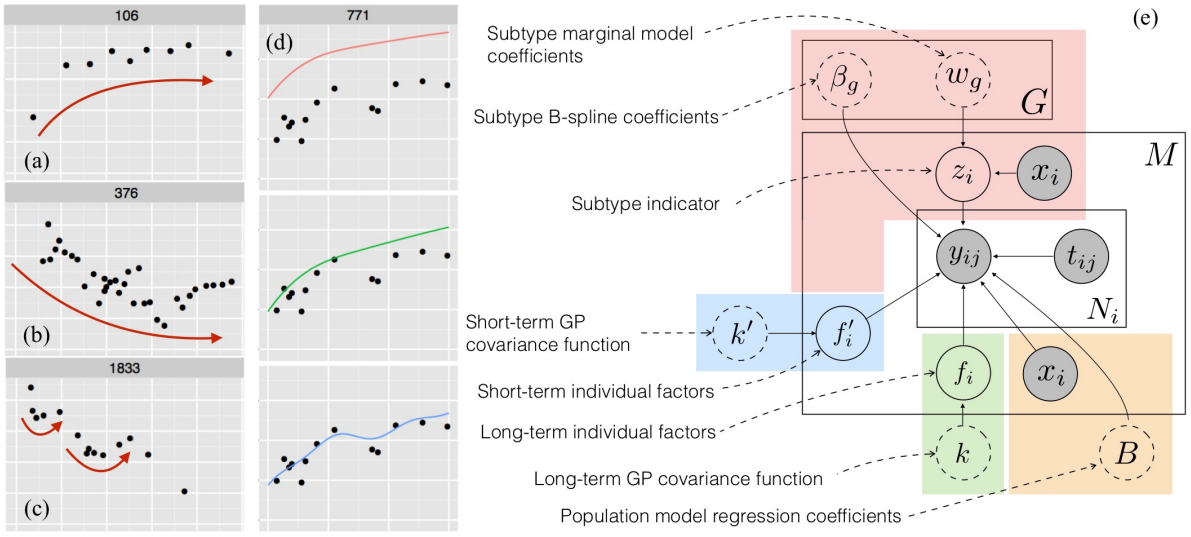


Figure 4.1: Plots (a-c) show example marker trajectories. Plot (d) shows adjustments to a population and subpopulation fit (row 1). Row 2 makes an individual-specific long-term adjustment. Row 3 makes short-term structured noise adjustments. Plot (e) shows the proposed graphical model. Levels in the hierarchy are color-coded. Model parameters are enclosed in dashed circles. Observed random variables are shaded.

ulation (subtype) component, an individual long-term component, and an individual short-term component. Specifically, we model y_{ij} using an additive model comprised of terms that use information at increasingly granular scales:

$$y_{ij} \mid z_i, f_i, f_i' \sim \text{Normal}\left(\underbrace{\Phi(t_{ij})Bx_i}_{\text{(A) population}} + \underbrace{\Phi(t_{ij})\beta_{z_i}}_{\text{(B) subpopulation}} + \underbrace{f_i(t_{ij})}_{\text{(C) long-term}} + \underbrace{f_i'(t_{ij})}_{\text{(D) short-term}}, \sigma^2\right) \tag{4.1}$$

The hierarchy that we use in the latent-hierarchy trajectory model (LTM) is similar to those developed for the PSM (Section 2.4). We briefly review these four components.

4.2.1 Population Level

The population model predicts aspects of an individual's disease activity trajectory using *observed* baseline characteristics (e.g. gender and race), which are represented using the feature vector $x_i \in \mathbb{R}^d$. This sub-model is shown within the orange box in Figure 4.1e. As in the PSM, we estimate p linear functions of x_i that determine the coefficients of a function within the span of B-spline bases $\Phi(t) \in \mathbb{R}^p$ (see Section 2.4.1 for a review of B-splines). We collect the coefficients of

these linear models into a matrix $B \in \mathbb{R}^{p \times d}$.

4.2.2 Subpopulation Level

We model an individual's subpopulation (or subtype) using a discrete-valued latent variable $z_i \in \{1, \ldots, G\}$, where G is the number of subtypes. We associate each subtype with a unique disease activity trajectory represented using a B-spline with bases $\Phi(t) \in \mathbb{R}^p$ and coefficients β_g. This component explains differences such as those observed between the trajectories in Figures 4.1a and 4.1b.

In many cases, features at baseline may be predictive of subtype. For example, in scleroderma, the types of antibody an individual produces (i.e. the presence of certain proteins in the blood) are correlated with certain trajectories. We can improve predictive performance by conditioning on baseline covariates to infer the subtype. To do this, we use a softmax linear regression to define feature-dependent marginal probabilities:

$$\log \pi_g(x_i) \leftarrow w_g^\top x_i - \log \sum_{g'=1}^{G} e^{w_{g'}^T x_i} \tag{4.2}$$

$$z_i \mid x_i \sim \mathsf{Categorical}(\pi_{1:G}(x_i)). \tag{4.3}$$

4.2.3 Individual Long-Term Level

This level models deviations from the population and subpopulation models using parameters that are learned dynamically as the individual's clinical history grows. This component can explain, for example, differences in overall health due to an unobserved characteristic such as chronic smoking, which may cause atypically lower lung function than what is predicted by the population and subpopulation components. Such an adjustment is illustrated across the first and second rows of Figure 4.1d. As in the PSM, we model this factor using a *long-term Gaussian process* with mean zero and covariance function k that is a relatively weak function of time:

$$f_i \sim \mathsf{GP}(0, k). \tag{4.4}$$

4.2.4 Individual Short-Term Level

Finally, the short-term component captures transient trends in the individual's disease trajectory. For example, an infection may cause an individual's lung function to temporarily appear more restricted than it actually is, which may cause short-term trends like those shown in Figure 4.1c and the third row of Figure 4.1d. As in the PSM, we model this factor using a *short-term Gaussian process* with mean zero and covariance function k' that assigns higher correlations to observations measured closer in time:

$$f'_i \sim \mathsf{GP}(0, k'). \tag{4.5}$$

4.3 Learning and Inference

As in the PSM, we assume that the number of subtypes G, the long-term covariance function k, and the short-term covariance function k' are fixed. In practice, we choose these using model selection procedures.

4.3.1 Learning

To estimate the remaining parameters $\Theta = \{w_{1:G}, \beta_{1:G}, B\}$, we maximize the log likelihood of the observed data using the expectation maximization (EM) algorithm.

Expectation Step

As in the PSM, we only need to compute the posterior distribution of z_i given y_i (because we assume that k and k' are fixed). The posterior is

$$q_i(g) \propto p(z_i = g \mid x_i) p(y_i \mid z_i = g). \tag{4.6}$$

Maximization Step

Given the posteriors over z_i, $\{q_i\}_{i=1}^M$, the method to reesimate the population regression parameters B and the subtype coefficients $\beta_{1:G}$ are the same in this model as in the PSM. One important difference is that we have replaced the marginal subtype probabilities $\pi_{1:G}$ with a softmax regression. To fit the softmax regression parameters, we maximize the following expected log likelihood

$$J(w_{1:G}) = \sum_{i=1}^{M} \mathbb{E}_{q_i}[\log p(z_i \mid x_i)] \tag{4.7}$$

$$= \sum_{i=1}^{M} \sum_{g=1}^{G} q_i(g) \left(w_g^\top x_i - \log \sum_{g'=1}^{G} e^{w_{g'}^\top x_i} \right). \tag{4.8}$$

There is no closed form solution to this maximization problem, so we use an iterative first-order optimization algorithm (e.g. L-BFGS). To iteratively compute the updates, we must compute the gradient of J with respect to w_g:

$$\frac{\partial J}{\partial w_g} = \sum_{i=1}^{M} q_i(g)(1 - \pi_g(x_i))x_i - (1 - q_i(g))\pi_g(x_i)x_i. \tag{4.9}$$

4.3.2 Inference and Prediction

Given model hyperparameters (G, k, and k') and estimated parameters ($w_{1:G}$, $\beta_{1:G}$, B), we can dynamically predict an individual's future trajectory for each subtype using the posterior predictive

$$p(y_* \mid y_i, z_i) \tag{4.10}$$

where y_* is the s-marker at some time t_* (usually in the future). Using Equation 4.6, we can average over Equation 4.10 using q_i or predict using the MAP estimate of z_i.

Given x_i and z_i, the s-marker trajectory y_i is a Gaussian process with mean and covariance functions

$$m(t; z_i) = \Phi(t)Bx_i + \Phi(t)\beta_{z_i} \tag{4.11}$$

$$v(t_1, t_2) = k(t_1, t_2) + k'(t_1, t_2) \tag{4.12}$$

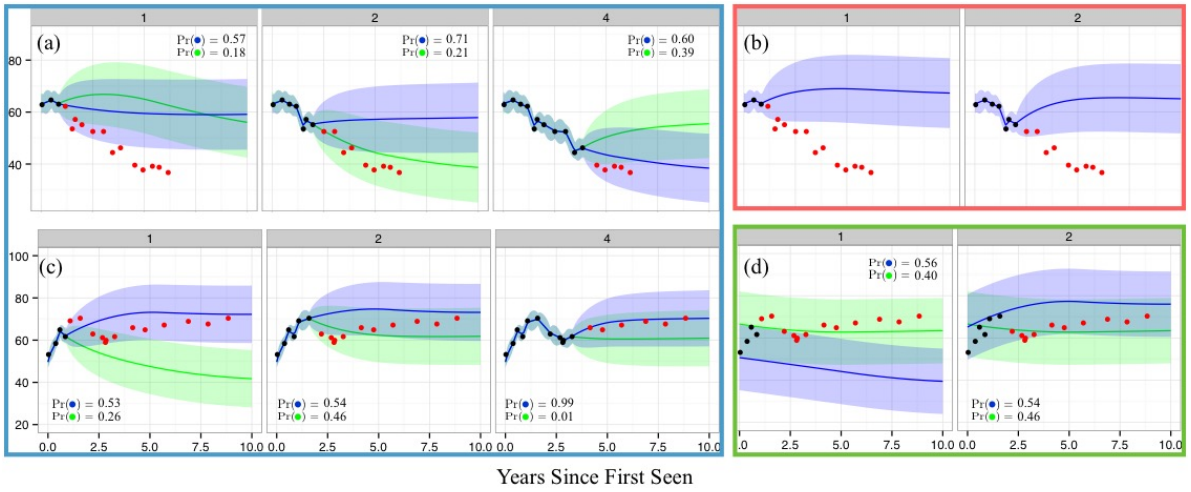


Figure 4.2: Plots (a) and (c) show dynamic predictions using the LTM for two individuals. Red markers are unobserved. Blue shows the trajectory predicted using the most likely subtype, and green shows the second most likely. Plot (b) shows dynamic predictions using the B-spline GP baseline. Plot (d) shows predictions made using the LTM without individual-specific adjustments.

It is therefore straightforward to compute Equation 4.10 using the Gaussian process prediction equations. Let $t_* \in \mathbb{R}^{N_*}$ denote a vector of measurement times at which we would like to predict the markers $y_* \in \mathbb{R}^{N_*}$. Next, define the following matrices:

$$K_i \in \mathbb{R}^{N_i \times N_i}, [K_i]_{jk} = v(t_{ij}, t_{ik}) \tag{4.13}$$

$$K_* \in \mathbb{R}^{N_* \times N_i}, [K_*]_{jk} = v(t_{*j}, t_{ik}) \tag{4.14}$$

$$K_{**} \in \mathbb{R}^{N_* \times N_*}, [K_{**}]_{jk} = v(t_{*j}, t_{*k}). \tag{4.15}$$

The posterior predictive of y_i given z_i is multivariate normal with mean and covariance

$$\mu_*(z_i) = m(t_*; z_i) + K_*(K_i + \sigma^2 \mathsf{I}_{N_i})^{-1}(y_i - m(t_i; z_i)) \tag{4.16}$$

$$\Sigma_* = K_{**} - K_*(K_i + \sigma^2 \mathsf{I}_{N_i})^{-1} K_*^\top. \tag{4.17}$$

4.4 Experiments

We demonstrate our approach by building a tool to predict the lung disease trajectories of individuals with scleroderma. Lung disease is currently the leading cause of death among scleroderma

patients, and is notoriously difficult to treat because there are few predictors of decline and there is tremendous variability across individual trajectories [Allanore et al., 2015]. Clinicians track lung severity using percent of predicted forced vital capacity (PFVC), which is expected to drop as the disease progresses. In addition, demographic variables and molecular test results are often available at baseline to aid prognoses. We train and validate our model using data from the Johns Hopkins Scleroderma Center patient registry, which is one of the largest in the world. To select individuals from the registry, we used the following criteria. First, we include individuals who were seen at the clinic within two years of their earliest scleroderma-related symptom. Second, we exclude all individuals with fewer than two PFVC measurements after their first visit. Finally, we exclude individuals who received a lung transplant. The dataset contains 672 individuals and a total of $4,992$ PFVC measurements.

For the population model, we use constant functions (i.e. observed covariates adjust an individual's intercept). The population covariates (x_i) are gender, African American race, and indicators of ACA and Scl-70 antibodies—two proteins believed to be connected to scleroderma-related lung disease. Note that all features are binary. For the subpopulation B-splines, we set boundary knots at 0 and 25 years (the maximum observation time in our data set is 23 years), use two interior knots that divide the time period from 0-25 years into three equally spaced chunks, and use quadratics as the piecewise components. These B-spline hyperparameters (knots and polynomial degree) are also used for all baseline models. We select $G = 9$ subtypes using BIC. The covariates in the subtype marginal model are the same used in the population model. For the long-term GP we use a constant covariance function $k(t_1, t_2) = \nu^2$ and for the short-term GP we use an OU covariance kernel $k(t_1, t_2) = a^2 \exp\{\ell^{-1}|t_1 - t_2|\}$. We set $\nu = 4$, $a = 6$, $\ell = 2$, and $\sigma^2 = 1$.

4.4.1 Baseline Models

First, to compare against typical approaches used in clinical medicine that condition on baseline covariates only (e.g. Khanna et al. 2011), we fit a regression model conditioned on all covariates

included in x_i above. The mean is parameterized using B-spline bases ($\Phi(t)$) as:

$$\hat{y} \mid \vec{x}_{iz} = \Phi(t)^\top \left(\vec{\beta}_0 + \sum_{j=1}^{d} x_{ij}\beta_j + \sum_{j=1}^{d} \sum_{k \neq j} x_{ij} x_{ik} \beta_{ij} \right). \tag{4.18}$$

The second baseline is similar to Rizopoulos [2011] and Shi et al. [2012] and extends the first baseline by accounting for individual-specific heterogeneity. The model has a mean function identical to the first baseline and individualizes predictions using a GP with covariance function $k + k'$ (using hyper-parameters as above). Another natural approach is to explain heterogeneity by using a mixture model similar to Proust-Lima et al. [2014]. However, a mixture model cannot adequately explain away individual-specific sources of variability that are unrelated to subtype and therefore fails to recover subtypes that capture canonical trajectories (we discuss this in detail in the supplemental section). The recovered subtypes from the full model do not suffer from this issue. To make the comparison fair and to understand the extent to which the individual-specific component contributes towards personalizing predictions, we create a mixture model (Proposed w/ no personalization) where the subtypes are fixed to be the same as those in the full model and the remaining parameters are learned. Note that this version does not contain the individual-specific component.

4.4.2 Evaluation Metrics

We make predictions after one, two, and four years of follow-up. Errors are summarized within four disjoint time periods: $(1, 2]$, $(2, 4]$, $(4, 8]$, and $(8, 25]$ years[1]. To measure error, we use the absolute difference between the prediction and a smoothed version of the individual's observed trajectory. We estimate mean absolute error (MAE) using 10-fold CV at the level of individuals (i.e. all of an individual's data is held-out), and test for statistically significant reductions in error using a one-sided, paired t-test. For all models, we use the MAP estimate of the individual's trajectory. In the models that include subtypes, this means that we choose the trajectory predicted by the most likely subtype under the posterior. Although this discards information from the posterior, in our experience clinicians find this choice to be more interpretable.

[1] After the eighth year, data becomes too sparse to further divide this time span.

4.4.3 Qualitative Results

In Figure 5.4 we present dynamically updated predictions for two patients (one per row, dynamic updates move left to right). Blue lines indicate the prediction under the most likely subtype and green lines indicate the prediction under the second most likely. The first individual (Figure 5.4a) is a 50-year-old, white woman with Scl-70 antibodies, which are thought to be associated with active lung disease. Within the first year, her disease seems stable, and the model predicts this course with 57% confidence. After another year of data, the model shifts 21% of its belief to a rapidly declining trajectory; likely in part due to the sudden dip in year 2. We contrast this with the behavior of the B-spline GP shown in Figure 5.4b, which has limited capacity to express individualized long-term behavior. We see that the model does not adequately adjust in light of the downward trend between years one and two. To illustrate the value of including individual-specific adjustments, we now turn to Figures 5.4c and 5.4d (which plot predictions made by the LTM with and without personalization respectively). This individual is a 60-year-old, white man that is Scl-70 negative, which makes declining lung function less likely. Both models use the same set of subtypes, but whereas the model without individual-specific adjustment does not consider the recovering subtype to be likely until after year two, the full model shifts the recovering subtype trajectory downward towards the man's initial PFVC value and identify the correct trajectory using a single year of data.

4.4.4 Quantitative Results

Table 5.1 reports MAE for the baselines and the LTM. We note that after observing two or more years of data, our model's errors are smaller than the two baselines (and statistically significantly so in all but one comparison). Although the B-spline GP improves over the first baseline, these results suggest that both subpopulation and individual-specific components enable more accurate predictions of an individual's future course as more data are observed. Moreover, by comparing the LTM with and without personalization, we see that subtypes alone are not sufficient and that individual-specific adjustments are critical. These improvements also have clinical significance. For

Model	$(1, 2]$	% Im.	$(2, 4]$	% Im.	$(4, 8]$	% Im.	$(8, 25]$	% Im.
Predictions using 1 year of data								
B-spline with Baseline Feats.	12.78		12.73		12.40		12.14	
B-spline + GP	5.49		7.70		**9.67**		**10.71**	
Proposed	**5.26**		***7.04**	8.6	10.17		12.12	
Proposed w/ no personalization	6.17		7.12		9.38		12.85	
Predictions using 2 years of data								
B-spline with Baseline Feats.			12.73		12.40		12.14	
B-spline + GP			5.88		8.65		10.02	
Proposed			***5.48**	6.8	***7.95**	8.1	**9.53**	
Proposed w/ no personalization			6.00		8.12		11.39	
Predictions using 4 years of data								
B-spline with Baseline Feats.					12.40		12.14	
B-spline + GP					6.00		8.88	
Proposed					***5.14**	14.3	***7.58**	14.3
Proposed w/ no personalization					5.75		9.16	

Table 4.1: MAE of PFVC predictions for the two baselines and the LTM. Bold numbers indicate best performance across models (* is stat. significant). "% Im." reports percent improvement over next best.

example, individuals who drop by more than 10 PFVC are candidates for aggressive immunosuppressive therapy. Out of the 7.5% of individuals in our data who decline by more than 10 PFVC, our model predicts such a decline at twice the true-positive rate of the B-spline GP (31% vs. 17%) and with a lower false-positive rate (81% vs. 90%).

4.5 Conclusion

We have described a hierarchical model for making individualized predictions of disease activity trajectories that accounts for both latent and observed sources of heterogeneity. We empirically demonstrated that using all elements of the LTM's hierarchy allows our model to dynamically personalize predictions and reduce error as more data about an individual is collected. Although our analysis focused on scleroderma, our approach is more broadly applicable to other complex, heterogeneous diseases [Craig, 2008]. Examples of such diseases include asthma [Lötvall et al., 2011], autism [Wiggins et al., 2012], and COPD [Castaldi et al., 2014]. There are several promising directions for further developing the ideas presented here. First, we observed that predictions are less accurate early in the disease course when little data is available to learn the individual-specific adjustments. To address this shortcoming, it may be possible to leverage time-dependent covariates

in addition to the baseline covariates used here. Second, the quality of our predictions depends upon the allowed types of individual-specific adjustments encoded in the model. More sophisticated models of individual variation may further improve performance. Moreover, approaches for automatically learning the class of possible adjustments would make it possible to apply our approach to new diseases more quickly.

Chapter 5

Coupling Trajectory Models to Improve Predictions

In this chapter, we extend the Latent-Hierarchy Trajectory Model (LTM) by conditioning on baseline information, previous observations of the trajectory, and additional time-dependent clinical markers (henceforth referred to as auxiliary markers) as they are collected. Like the target clinical marker, the auxiliary marker trajectories are typically heterogeneous and sampled at irregular (and sometimes sparse) times. This makes it difficult to condition on these inputs in a standard discriminative model. For instance, we might use the mean and variance of an auxiliary marker in the past six years as features in a predictive model, but these inputs will likely be noisy. They might be noisy because there are few observations (e.g. only one observation in the past six years), or because of the heterogeneity in the population. In some cases, there may be no past auxiliary markers at all and so these features will be missing.

One natural approach to handling the sparsity issue is to build a joint model of both the auxiliary markers and the target marker. Once we have a joint model, we can make predictions by computing the conditional distribution of the target marker given the auxiliary markers. For instance, Rizopoulos and Ghosh [2011] propose a joint linear mixed effects model (LMM) over multiple markers. Proust-Lima et al. [2014] describe an alternative approach using a discrete mixture model. Joint model approaches raise a number of different issues. First, joint models often

scale poorly as more auxiliary markers are included (e.g. the number of model parameters in an LMM grows quadratically). Second, as we include new auxiliary markers we must reformulate the joint model and take care to faithfully capture statistical dependencies between all markers. This grows increasingly difficult as the number of markers grows.

This chapter describes a scalable framework for predicting a target marker trajectory that allows us to include multiple auxiliary clinical marker histories as inputs. Our approach makes it easy to handle irregular sampling patterns across input markers (as in joint models). Moreover, the number of parameters and computational complexity scales linearly with the number of markers, which makes it possible to apply our approach in high-dimensional settings where many different marker types are available. Finally, our approach minimizes the impact of incorrectly specifying the statistical dependencies across markers by learning to share inferences across each marker-specific model in a way that maximizes predictive performance of the target marker. We apply our approach to the problem of predicting lung disease trajectories in scleroderma, a complex autoimmune disease. We show that our approach significantly improves over state-of-the-art baselines in predictive accuracy and we provide a qualitative analysis of our model's output to build intuition for why it works. In addition, we show that our model is clinically relevant by demonstrating improvements over the baseline models in early detection of individuals who develop aggressive lung disease (defined as a clinically significant drop in lung function).

5.1 Related Work

Most predictive models used in medicine are cross-sectional—they use features from data measured up until the current time to predict a clinical marker or outcome at a fixed point in the future. As an example, consider the mortality prediction model by Lee et al. [2003], where logistic regression is used to integrate features into a prediction about the probability of death within 30 days for a given patient. To predict the outcome at multiple time points, it is common to fit separate models (e.g., Wang et al. 2012, Zhou et al. 2011). Moreover, these models are trained to use features extracted from a fixed-size window, rather than a dynamically growing history. These models also tackle

heterogeneity in a limited way—any differences across individuals must be explained by observed features alone.

The statistics and machine learning communities have proposed solutions that address a number of these limitations. Most related to our work is that by Rizopoulos [2011], where the focus is on making dynamical predictions about a time-to-event outcome (e.g. time until death) using all previously observed values of a longitudinally recorded marker. As more data is collected, they dynamically update posterior distributions over individual-specific longitudinal model parameters, which serve as time-varying features for the time-to-event prediction. Proust-Lima et al. [2014] tackles the same task but uses a mixture model over the longitudinal data. As more observations are collected, the posterior over a set of classes is updated, each of which has a distinct set of time-to-event model parameters. These are both state-of-the-art models for the task of dynamical disease trajectory prediction; we will revisit them in our experimental section where we use the approaches as baselines.

More broadly, a common approach to dynamical prediction is to use Markov models such as order-p autoregressive models (AR-p), HMMs, state space models, and dynamic Bayesian networks (e.g. Hassan and Nath 2005, Quinn et al. 2009, Murphy 2002). While such models naturally make dynamic predictions using the full history by forward-filtering, they typically assume discrete, regularly-spaced observation times. Gaussian processes (GPs) are a commonly used alternative for handling continuous-time observations—see Roberts et al. [2013] for a recent review of GP time series models. Since Gaussian processes are non-parametric generative models of functions, they naturally produce functional predictions dynamically through the posterior predictive distribution. A number of authors have proposed variants of GPs that account for heterogeneity in the mean function (e.g. Lázaro-Gredilla et al. 2012, Shi et al. 2012) and the covariance function (e.g. Shi et al. 2005). To scale GPs to multivariate time series, however, requires careful selection of the covariance function, which can be challenging in high-dimensional settings (e.g., Dürichen et al. 2015). Recent work by Liu and Hauskrecht [2014] combines the advantages of Markov models (e.g. AR processes and state space models) and Gaussian processes to make predictions of clinical

laboratory test results. Although the task is similar, we focus on making predictions in highly heterogeneous populations, which their approach does not explicitly address.

Methods from functional data analysis (FDA) can also be used to analyze continuous-time trajectories (see, e.g., Ramsay 2006). A common approach in FDA is to project irregular observations on to a functional basis, such as B-splines, and then analyze the time series in coefficient space. A drawback of this approach is that the coefficient estimates can have high variance when a time series has too few observations, which is common in clinical data. James and Sugar [2003] address this issue using a low-rank parameterization of the mean and covariance functions. This work is closely related to ours, but focuses on retrospective analysis of longitudinal data rather than dynamic prediction. Another related line of work in the FDA literature is function-to-function regression (e.g., Oliva et al. 2015). In most approaches to function-to-function regression (FFR) the input and output are defined on fixed domains. In contrast, our problem requires updated predictions as the clinical history continues to grow; both the input and output domains are therefore constantly changing. In addition, FFR methods typically assume that the data are densely sampled, which is rare in health care.

In the disease progression modeling literature (see Mould 2012 for a recent review), there is an extensive body of work on methods for analysis of patient time series to discover canonical trajectories of progression. These focus on retrospective analysis to discover disease etiology rather than dynamical prediction of an individual's trajectory (e.g., Jackson et al. 2003, Wang et al. 2014). Recently, others have extended disease progression modeling to incorporate heterogeneity in disease trajectories due to subtypes (see Schulam et al. 2015 and references within). The focus of these contributions, however, like in other disease progression modeling work, is on discovery rather than prediction.

5.2 Coupled Latent-Hierarchy Trajectory Model

Our goal is to predict a *continuous function* modeling the future trajectory of a *target clinical marker* (e.g. PFVC) that tracks disease progression in a specific organ. To make our predictions,

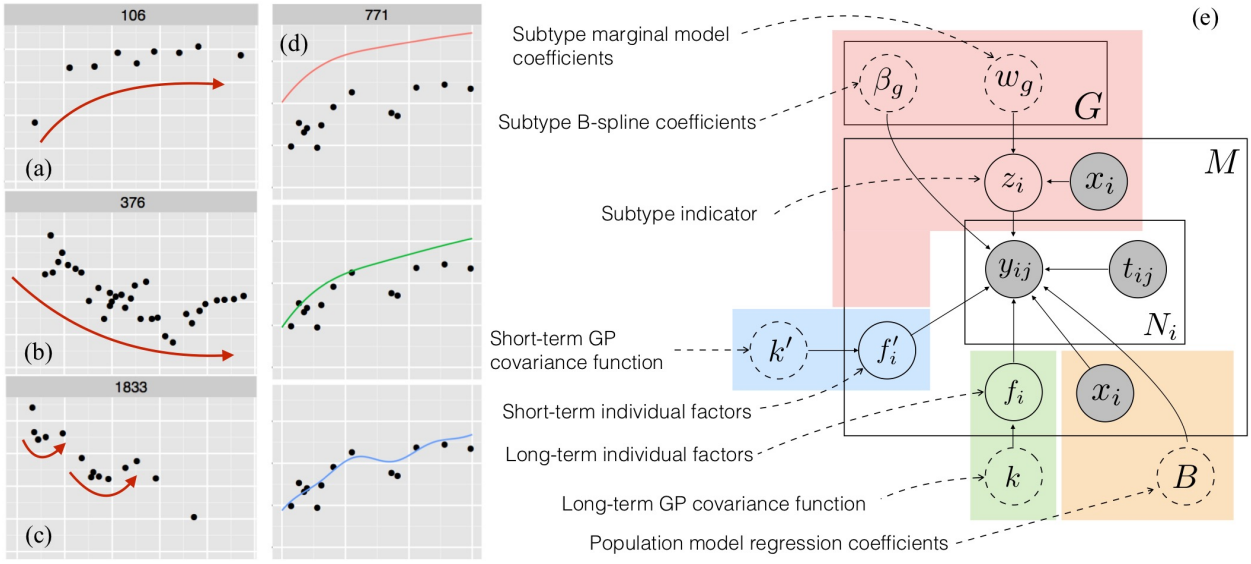


Figure 5.1: Plots (a-c) show example marker trajectories. Plot (d) shows four individuals with adjustments to a population and subpopulation fit (row 1). Row 2 makes an individual-specific long-term adjustment. Row 3 makes individual-specific short-term adjustments. To simplify, we only show mean functions; posterior uncertainty intervals are omitted.

we will use a collection of baseline (i.e. static) markers measured when an individual first presents, the previously observed values of the target marker, and the previously observed values of a collection of *auxiliary clinical markers* tracking related organ systems. See Figure 5.5a-d for example applications; the posterior distribution over the PFVC values (blue and green shaded regions) are conditioned upon baseline markers (e.g. gender and race), the observed PFVC values (black points), and auxiliary marker histories (e.g. TSS). We learn our model from a database of clinical histories of individuals, which are comprised of the individuals' baseline information and irregularly sampled trajectories of both the target and auxiliary markers. Formally, our model will estimate the following conditional distribution (notation is described in the subsequent paragraph):

$$\mathcal{D}(i,t) \triangleq p(y_i(\cdot) \mid y_{i,\leq t}, y_{1:C,i,\leq t}, x_i). \tag{5.1}$$

Notation. For an individual i, we denote each target marker observation using y_{ij} and its measurement time using t_{ij} where $j \in \{1, \ldots, N_i\}$. We use $y_i \in \mathbb{R}^{N_i}$ and $t_i \in \mathbb{R}^{N_i}$ to denote all of individual i's marker values and measurement times respectively. We assume that the target

Input: Baseline predictors $x_{1:M}$

Target marker trajectories $y_{1:M}$

Auxiliary marker trajectories $y_{1:C,1:M}$

Stage 1: Fit Marker-Dependent Factors (LTM)

Fit target marker latent variable model to define factor $\ell_{i,\leq t}(\cdot)$

For $c \in \{1, \ldots, C\}$

Fit auxiliary marker latent variable model to define factor $\ell_{c,i,\leq t}(\cdot)$

Stage 2: Fit latent factor CRF regression model

For each regularization penalty λ

Fit CRF on training data:

Initialize singleton factor weights for $\phi(\cdot)$, $\ell_{i,\leq t}(\cdot)$, and $\ell_{c,i,\leq t}(\cdot)$ for $c \in \{1, \ldots, C\}$

Initialize pairwise factor weights for $\psi(\cdot, \cdot)$

Fit CRF parameters by maximizing

$$J(\gamma, \gamma_{1:C}, \theta, \theta_{1:C}, \eta_{1:C}) \triangleq \sum_{i=1}^{M} \sum_{t \in \mathcal{T}} \log p(y_{i,>t} \mid y_{i,\leq t}, y_{1:C,i,\leq t}) - \lambda(\|\theta, \theta_{1:C}, \eta_{1:C}\|_1)$$

Evaluate penalty on development data

Refit CRF using full training set with best λ

Figure 5.2: Two-stage procedure for fitting the Coupled Latent Trajectory Model (C-LTM).

marker observations are noisy observations of a latent continuous-time function (the trajectory), which we denote using $y_i(\cdot)$. Each individual has baseline (static) information collected into a vector, which we denote using x_i. We use C to denote the number of auxiliary marker types, N_{ci} to denote the number of observations of the c^{th} type, and use y_{cij} and t_{cij} to denote individual i's j^{th} measurement of marker type c. We use $y_{ci} \in \mathbb{R}^{N_{ci}}$ and $t_{ci} \in \mathbb{R}^{N_{ci}}$ to denote the vector containing all of individual i's c^{th} marker values and times respectively. We will also frequently need to refer to the vector of marker values observed up until a time t, which we denote using $y_{i,\leq t}$ ($y_{ci,\leq t}$ for auxiliary markers). Similarly, for markers observed after a time t, we use $y_{i,>t}$ ($y_{ci,>t}$ for auxiliary markers). The term $y_{1:C,i,\leq t}$ refers to all auxiliary markers measured on individual i up until time t.

At a high-level, we will model Eq. 5.1 by first assuming that each clinical marker trajectory (both target and auxiliary) can be d-separated (rendered conditionally independent) of all other marker types given a marker type-specific latent variable. We denote these latent variables using

z_i for the target marker and z_{ci} for auxiliary marker c, and will describe them further later in this section. Under this assumption, we can write Eq. 5.1 as

$$\begin{aligned}
\mathcal{D}(i,t) &= \sum_{z_i} p(y_i(\cdot) \mid z_i, y_{i,\leq t}, x_i) p(z_i \mid y_{i,\leq t}, y_{1:C,i,\leq t}, x_i) \\
&\propto \sum_{z_i} p(y_i(\cdot) \mid z_i, y_{i,\leq t}, x_i) p(y_{i,\leq t} \mid z_i, \vec{x}_i) p(z_i \mid y_{1:C,i,\leq t}, x_i) \\
&\propto \sum_{z_i} \underbrace{p(y_i(\cdot) \mid z_i, y_{i,\leq t}, x_i)}_{\substack{\text{LTM predictive,} \\ \text{Eq. 4.16}}} \underbrace{p(y_{i,\leq t} \mid z_i, x_i)}_{\substack{\text{LTM likelihood,} \\ \text{Eq. 2.13}}} \sum_{z_{1:C,i}} \underbrace{p(z_i, z_{1:C,i} \mid x_i)}_{\substack{\text{Coupling Model,} \\ \text{Section 5.2.2,} \\ \text{Eq. 5.9}}} \prod_{c=1}^{C} \underbrace{p(y_{ci,\leq t} \mid z_{ci}, x_i)}_{\substack{\text{LTM likelihood,} \\ \text{Eq. 2.13 and 2.16}}}.
\end{aligned} \tag{5.2}$$

We will learn this parameterization of $\mathcal{D}(i,t)$ in two stages. The model for the target and each of the auxiliary markers are learned independently during the first stage; using these, the LTM predictive and likelihood terms are computed in Eq. 5.2. We treat the target and auxiliary markers as instances of the Latent Trajectory Model (LTM); another latent variable model can be used if better suited to the domain. The coupling model is learned in the second stage, and is described in Section 5.2.2. We refer to the model created by combining these components as the Coupled Latent Trajectory Model (C-LTM), which we describe in Section 5.2.3. An overview of the procedure used to fit the C-LTM is shown in Figure 5.2.

5.2.1 Background: Conditional Random Fields

Conditional random fields (CRFs) provide a framework for modeling and learning the joint distribution of a collection of random variables conditioned on some set of observations (see e.g. Murphy [2012]). The parameterization is identical to that of Markov random fields (MRF), but the factors that define the distribution can be functions of the observations (this allows the distribution to vary depending on the values of the observations). For some output y, input x and parameters θ, the conditional probability is defined to be:

$$p(y \mid x, \theta) = \frac{1}{Z(x,\theta)} \prod_c \psi_c(y_c \mid x, \theta), \quad Z(x,\theta) \triangleq \sum_{y'} \prod_c \psi_c(y'_c \mid x, \theta), \tag{5.3}$$

where $\psi_c(y_c \mid x, \theta)$ is a non-negative factor that can be interpreted as scoring the configuration of the subset of variables y_c given the observations x and parameters θ. The term $Z(x, \theta)$ is called the *partition function* and ensures that the distribution is normalized. When we can write

$$\log \psi_c(y_c \mid x, \theta) = \theta_c^\top f_c(y_c, x) \iff \psi_c(y_c \mid x, \theta) = \exp\left\{\theta_c^\top f_c(y_c, x)\right\}, \tag{5.4}$$

where f_c extracts some vector of features from the observations x and the target y_c, then we say that the CRF is a log-linear model. Log-linear models have a number of desirable properties, the most relevant to this work being the ease with which we can differentiate the log-likelihood with respect to model parameters. To compute the derivative with respect to θ_c (the parameters corresponding to the c^{th} factor) we have:

$$\frac{\partial \log p(y \mid x, \theta)}{\partial \theta_c} = f_c(y_c, x) - \frac{\partial \log Z(x, \theta)}{\partial \theta_c}. \tag{5.5}$$

To compute the partial derivative in the second term on the RHS, first note that

$$\frac{\partial Z(x, \theta)}{\partial \theta_c} = \sum_{y'} \left(\prod_{d \neq c} \psi_d(y'_d \mid x, \theta_d) \right) \frac{\partial \psi_c(y'_c \mid x, \theta_c)}{\partial \theta_c} \tag{5.6}$$

$$= \sum_{y'} \left(\prod_{d \neq c} \psi_d(y'_d \mid x, \theta_d) \right) \psi_c(y'_c \mid x, \theta_d) f_c(y'_c, x). \tag{5.7}$$

This implies that the partial derivative of $\log Z(x, \theta)$ is simply:

$$\frac{\partial \log Z(x, \theta)}{\partial \theta_c} = \frac{1}{Z(x, \theta)} \frac{\partial Z(x, \theta)}{\partial \theta_c} = \mathbb{E}_y \left[f_c(y_c, x) \mid x \right] \tag{5.8}$$

This means that the gradient of the log-likelihood with respect to a set of parameters θ_c is the difference between the observed features $f_c(y, x)$ and their expectation under the current set of parameters θ. To learn the weights, we can apply gradient-based algorithms to optimize the likelihood of a set of observed training input-output pairs. In addition, a regularizer is often added to the objective to discourage complexity or induce sparsity. We will use these ideas in the derivation of our learning algorithm. See Ch. 19 in Murphy [2012] for further details.

5.2.2 Coupling Model

The Coupled Latent Trajectory Model (C-LTM) seeks to learn and capture correlations across trajectories of different marker types. In scleroderma, for example, an individual with worse lung trajectories (e.g. the rapidly declining lung trajectory subtype) is more likely to have a severe skin disease trajectory. In the C-LTM these types of dependencies are captured by the term $p(z_i, z_{1:C,i} \mid x_i)$ shown in Eq. 5.2. We parameterize this distribution using a conditional random field with singleton and pairwise factors defined over z_i and $z_{1:C,i}$. Singleton factors can depend on the baseline covariates x_i. Pairwise factors are defined only between the clinical marker random variables z_i and each of the auxiliary marker latent variables z_{ci}. Both are parameterized linearly. The coupling model therefore has the following form:

$$\begin{aligned}\log p(z_i, z_{1:C,i} \mid x_i) &\propto \phi(z_i, x_i) + \sum_{c=1}^{C} \phi(z_{ci}, x_i) + \psi(z_i, z_{ci}) \\ &= \theta^\top f(z_i, x_i) + \sum_{c=1}^{C} \theta_c^\top f_c(z_{ci}, x_i) + \eta_c^\top g_c(z_i, z_{ci}). \end{aligned} \tag{5.9}$$

5.2.3 Predicting Trajectories using the C-LTM

To predict trajectories (i.e. compute Eq. 5.1), we combine the LTM likelihood (Eq. 2.13), the LTM predictive (Eq. 4.16), and the coupling model (Eq. 5.9). Let $\ell_{i,\leq t}(z_i)$ stand as shorthand for $\log p(y_{i,\leq t} \mid z_i, x_i)$ and $\ell_{ci,\leq t}(z_{c,i})$ stand as shorthand for $\log p(y_{ci,\leq t} \mid z_{ci}, x_i)$, then we see that

$$\mathcal{D}(i,t) \propto \sum_{z_i} p(y_i(\cdot) \mid z_i, y_{i,\leq t}, x_i) \sum_{z_{1:C,i}} u(z_i, z_{1:C,i} \mid \mathcal{H}(i,t)), \tag{5.10}$$

where we have defined $\mathcal{H}(i,t)$ to be the set of information contained in the clinical history of individual i at time t: $\{y_{i,\leq t}, y_{1:C,i,\leq t}, x_i\}$, and used $u(z_i, z_{1:C,i} \mid \mathcal{H}(i,t))$ to denote the following unnormalized weight assigned to all values of the latent variables given the history:

$$\begin{aligned} &u(z_i, z_{1:C,i} \mid \mathcal{H}(i,t)) \triangleq \\ &\quad \exp\left\{ \ell_{i,\leq t}(z_i) + \theta^\top f(z_i, x_i) + \sum_{c=1}^{C} \ell_{ci,\leq t}(z_{ci}) + \theta_c^\top f_c(z_{ci}, x_i) + \eta_c^\top g_c(z_i, z_{ci}) \right\}, \end{aligned} \tag{5.11}$$

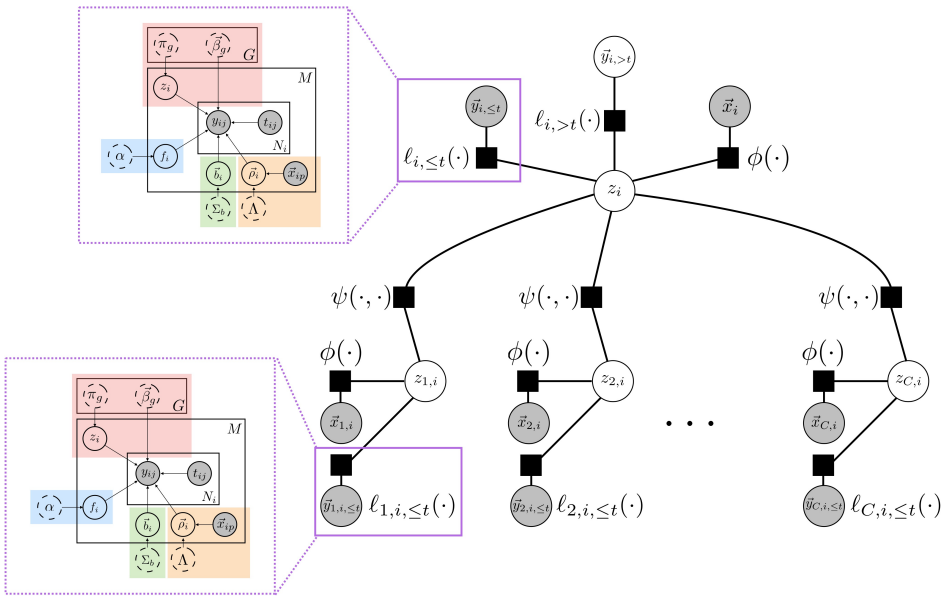


Figure 5.3: The factor graph of the coupled latent trajectory model. Empty nodes denote latent random variables, and shaded nodes denote observed variables. The latent trajectory model (LTM, described in Chapter 4) acts as a data-driven factor linking observed target and auxiliary marker histories into predictions.

To make $\mathcal{D}(i,t)$ a proper distribution, we normalize $u(z_i, z_{1:C,i} \mid \mathcal{H}(i,t))$ to obtain

$$p(z_i \mid \mathcal{H}(i,t)) = \frac{\sum_{z_{1:C}} u(z_i, z_{1:C} \mid \mathcal{H}(i,t))}{\sum_z \sum_{z_{1:C}} u(z, z_{1:C} \mid \mathcal{H}(i,t))} \triangleq \frac{Z'_{i,t}(z_i)}{Z_{i,t}}. \tag{5.12}$$

then we can write $\mathcal{D}(i,t)$ (Eq. 5.1) as

$$\mathcal{D}(i,t) = \sum_{z_i} p(y_i(\cdot) \mid z_i, y_{i,\leq t}, x_i)\, p(z_i \mid \mathcal{H}(i,t)). \tag{5.13}$$

Intuitively, we see that the predictive distribution under C-LTM is simply a weighted combination of the subtype-specific predictive distributions under LTM (Eq. 4.16). Moreover, the distribution $p(z_i \mid \mathcal{H}(i,t))$ is the marginal distribution over z_i in a conditional random field with structure similar to the coupling model (Eq. 5.9) but augmented with additional singleton factors defined by the LTM likelihood functions given the marker trajectory histories. The LTM likelihood factors in Eq. 5.11 are added into the model unchanged, but additional parameters $\{\gamma, \gamma_{1:C}\}$ can be included to reweight those terms (a similar idea is used in Raina et al. [2003]).[1] The factor graph

[1]When using a penalty, we can center the weights at 1 so that the default behavior is to leave the likelihood factors

for this conditional random field is shown in Figure 5.3. Note that the weight $p(z_i \mid \mathcal{H}(i,t))$ can be efficiently computed in time linear in the number of auxiliary markers using the junction tree algorithm. This is in contrast to a typical joint model over $z_i, z_{1:C,i}$, which would scale exponentially in the number of auxiliary markers.

The C-LTM offers a number of advantages for predictive modeling of disease trajectories in domains where many other related marker trajectories are available. First, it allows irregularly and sparsely sampled trajectories to be neatly summarized using modularized, single-marker generative models. These can capture important latent factors and account for marker-specific measurement models and noise processes. Second, we can discriminatively use auxiliary marker trajectory histories when modeling Eq. 5.1 instead of specifying a joint generative model, which sidesteps the challenges associated with correctly specifying dependencies between many different marker types. Finally, the model can be used in continuous time and dynamically updates predictions as new markers arrive.

5.2.4 Learning the C-LTM

We have described two components of our approach: the Latent Trajectory Model (LTM) and the coupling model. When these components are combined as shown in Section 5.2.3, then we obtain the C-LTM. The C-LTM has two conceptually distinct sets of parameters. The first set are those belonging to the individually trained LTMs for each marker type. To learn these, we can use the EM algorithm described in Schulam and Saria [2015]. To learn the parameters for the C-LTM, we keep the single-marker model parameters fixed (e.g. those learned for the LTM), and use a standard gradient-based CRF learning algorithm (as described in Section 5.2.1) to optimize the penalized log-likelihood of example trajectory predictions.

To learn the parameters of the latent-factor CRF regression, we directly maximize the conditional probability of future target markers given previously observed target markers, previously observed auxiliary markers, and static baseline covariates on a collection of examples extracted from retrospective data. Suppose we are given records containing the target marker, auxiliary markers,

unchanged as in Eq. 5.11

and baseline covariates for M individuals. We choose a collection of times $\mathcal{T}$ that will be used to create training examples of history-future pairs. For example, we may choose $\mathcal{T} = \{1, 2\}$ because early management decisions are made using prognoses at years 1 and 2. We emphasize, however, that the model is *not* restricted to making predictions at years 1 and 2; it can make predictions at arbitrary times. The times $\mathcal{T}$ are simply used to create training instances. We also note that it is possible to train specialized models for different time periods. For example, we may train one model for making predictions in the first 2 years and another for beyond 4 years. Given the M records and times $\mathcal{T}$, we define the objective:

$$J(\gamma, \gamma_{1:C}, \theta, \theta_{1:C}, \eta_{1:C}) = \sum_{i=1}^{M} \sum_{t \in \mathcal{T}} \log p(y_{i,>t} \mid \mathcal{H}(i,t)) \tag{5.14}$$

$$= \sum_{i=1}^{M} \sum_{t \in \mathcal{T}} \log \left(\sum_{z_i} \underbrace{p(y_{i,>t} \mid z_i, x_i)}_{\text{(A)}} \underbrace{p(z_i \mid \mathcal{H}(i,t))}_{\text{(B)}} \right), \tag{5.15}$$

where (A) is the subtype-specific multivariate normal likelihood in Eq. 2.16 and (B) is the conditional distribution over z_i shown in Eq. 5.12. To learn the parameters, we maximize this objective with respect to θ, $\theta_{1:C}$, and $\eta_{1:C}$ using gradient-based methods (e.g. `L-BFGS`). In our experiments, we optimize a regularized version of the objective, but for simplicity this section discusses the computations required to compute the gradient of Eq. 5.14 only. Consider a single summand of Eq. 5.14

$$\log p(\vec{y}_{i,>t} \mid \mathcal{H}(i,t)) = \log \left(\sum_{z_i} p(\vec{y}_{i,>t} \mid z_i, x_i) p(z_i \mid \mathcal{H}(i,t)) \right). \tag{5.16}$$

To reiterate, the parameters of the density $p(\vec{y}_{i,>t} \mid z_i, x_i)$ are assumed to have been learned in a separate step (e.g. using the EM algorithm from Section 2.5.1), and so we are only concerned with estimating the parameters of the singleton and pairwise factors in the CRF: $\theta, \theta_{1:C}, \eta_{1:C}$.

Gradient of the Objective

We derive the gradient for a single summand of the objective (Eq. 5.14), which are combined additively to form the full gradient used at each iteration. Although our model is log-linear over all latent variables z_i and $z_{1:C,i}$, Eq. 5.16 is not linear in the parameters because we marginalize over the unobserved auxiliary subtypes $z_{1:C}$. We therefore have that the partial derivative of Eq. 5.16 with respect to any parameter θ_k is:

$$\frac{\partial \log p(y_{i,>t} \mid \mathcal{H}(i,t))}{\partial \theta_k} = \frac{\left(\sum_{z_i} p(y_{i,>t} \mid z_i, x_i) \frac{\partial p(z_i \mid \mathcal{H}(i,t))}{\partial \theta_k}\right)}{p(y_{i,>t} \mid \mathcal{H}(i,t))}. \tag{5.17}$$

To complete the expression for the partial derivative, we need to compute the partial derivative of the probability of a given target marker latent variable z_i with respect to the parameter θ_k. We have that:

$$\frac{\partial p(z_i \mid \mathcal{H}(i,t))}{\partial \theta_k} = \frac{\partial}{\partial \theta_k} \frac{Z'_{i,t}(z_i)}{Z_{i,t}} = \frac{1}{Z_{i,t}} \frac{\partial Z'_{i,t}(z_i)}{\partial \theta_k} + Z'_{i,t}(z_i) \frac{\partial Z_{i,t}^{-1}}{\partial \theta_k}. \tag{5.18}$$

We can now leverage identities from the theory of log-linear models to continue with the derivation. In particular, recall that log-linear models are in the exponential family of distributions. As a consequence, we can consider the parameters $\theta, \theta_{1:C}, \eta_{1:C}$ as the *natural parameters* of the distribution. The corresponding *sufficient statistics* are therefore the factors in the log-linear model:

$$T(z_i, z_{1:C,i}, x_i) = [f^\top(z_i, x_i), f_1^\top(z_{1,i}, x_i), \ldots, f_C^\top(z_{C,i}, x_i), g_1^\top(z_i, z_{1,i}), \ldots, g_C^\top(z_i, z_{C,i})]^\top.$$

An important property of exponential families is that the gradient of the log-normalizing-constant with respect to the natural parameters is simply the expected value of the sufficient statistics computed using the current value of the natural parameters. Note that both $Z'_{i,t}(z_i)$ and $Z_{i,t}$ are normalizing constants of exponential family distributions. In the case of $Z_{i,t}$ this is trivial to see because it is the normalizing constant of our log-linear model. In the case of $Z'_{i,t}(z_i)$ we see that it is the normalizing constant of a log-linear model over the auxiliary marker latent variables $z_{1:C}$

given *both* z_i and the clinical history $\mathcal{H}(i,t)$. We therefore have:

$$\frac{\partial \log Z'_{i,t}(z_i)}{\partial \theta_k} = \mathbb{E}\left[T(z_i, z_{1:C,i}, x_i)_k \mid z_i, \mathcal{H}(i,t)\right]$$

$$\implies \frac{\partial Z'_{i,t}(z_i)}{\partial \theta_k} = Z'_{i,t}(z_i)\mathbb{E}\left[T(z_i, z_{1:C,i}, x_i)_k \mid z_i, \mathcal{H}(i,t)\right], \tag{5.19}$$

$$\frac{\partial \log Z_{i,t}}{\partial \theta_k} = \mathbb{E}\left[T(z_i, z_{1:C,i}, x_i)_k \mid \mathcal{H}(i,t)\right]$$

$$\implies \frac{\partial Z_{i,t}^{-1}}{\partial \theta_k} = -\frac{1}{Z_{i,t}}\mathbb{E}\left[T(z_i, z_{1:C,i}, x_i)_k \mid \mathcal{H}(i,t)\right], \tag{5.20}$$

where we have used $T(z_i, z_{1:C,i}, x_i)_k$ to denote the feature (or sufficient statistic) corresponding to the parameter θ_k. By plugging these partial derivatives back into Eq. 5.18, we have

$$\frac{\partial}{\partial \theta_k}\frac{Z'_{i,t}(z_i)}{Z_{i,t}} = \frac{Z'_{i,t}(z_i)}{Z_{i,t}}\left(\mathbb{E}\left[T(z_i, z_{1:C,i}, x_i)_k \mid z_i, \mathcal{H}(i,t)\right] - \mathbb{E}\left[T(z_i, z_{1:C,i}, x_i)_k \mid \mathcal{H}(i,t)\right]\right) \tag{5.21}$$

$$= p(z_i \mid \mathcal{H}(i,t))\left(\mathbb{E}_\Theta\left[T(z_i, z_{1:C,i}, x_i)_k \mid z_i, \mathcal{H}(i,t)\right] - \mathbb{E}_\Theta\left[T(z_i, z_{1:C,i}, x_i)_k \mid \mathcal{H}(i,t)\right]\right). \tag{5.22}$$

In words, we see that the partial derivative with respect to a parameter θ_k is the expected value of its corresponding feature given that we have observed the target marker latent variable z and clinical history $\mathcal{H}(i,t)$ minus the expected value of the feature given only the clinical history $\mathcal{H}(i,t)$. The difference is then weighted by the probability of observing the target marker latent variable given the clinical history. By plugging this expression back into Eq. 5.17, we arrive at the final expression for the partial derivative of a single summand with respect to θ_k:

$$\frac{\partial \log p(y_{i,>t} \mid \mathcal{H}(i,t))}{\partial \theta_k} \tag{5.23}$$

$$= \sum_{z_i} \frac{p(y_{i,>t} \mid z_i)p(z_i \mid \mathcal{H}(i,t))}{p(y_{i,>t} \mid \mathcal{H}(i,t))}\left(\mathbb{E}\left[T(z_i, z_{1:C,i}, x_i)_k \mid z_i, \mathcal{H}(i,t)\right] - \mathbb{E}\left[T(z_i, z_{1:C,i}, x_i)_k \mid \mathcal{H}(i,t)\right]\right)$$

$$= \sum_{z_i} p(z_i \mid y_{i,>t}, \mathcal{H}(i,t))\left(\mathbb{E}\left[T(z_i, z_{1:C,i}, x_i)_k \mid z_i, \mathcal{H}(i,t)\right] - \mathbb{E}\left[T(z_i, z_{1:C,i}, x_i)_k \mid \mathcal{H}(i,t)\right]\right). \tag{5.24}$$

The partial derivative has a nice interpretation. Each summand has similar structure to the partial derivative of $p(z_i \mid \mathcal{H}(i,t))$ (Eq. 5.21), but the weight conditioned on only the clinical history has been replaced with a weight conditioned on both the clinical history *and* the future target

marker trajectory. The partial derivatives of the summands of the objective in Eq. 5.14 are added together to obtain the partial derivative with respect to the objective. These partial derivatives are combined to form a gradient, which is easily plugged into existing first-order optimization routines. Optionally, the objective can be augmented with a regularizer to restrict the complexity of the model or to encourage a sparse solution to the learning problem.

Scalability

The EM algorithm used to learn the parameters of the LTM poses no serious challenges to scalability. The primary computational burden lies in the E-step wherein sufficient statistics from all individuals are computed and collected. This is linear in the number of patient records being analyzed, but since the inference required to compute the sufficient statistics can be performed independently for each individual given the current parameter estimates, the E-step can be easily parallelized to offset slow learning due to large numbers of patient records. For any given individual, the E-step is dominated by the inversion of the $N_i \times N_i$ covariance matrix. We do not expect this to be problematic, however, because clinical markers in chronic diseases are observed at a maximum rate of 12 times per year. Moreover, such diseases occur over periods on the order of tens of years. Therefore, the number of measurements will be at most on the order of 100-200.

Learning the parameters of the CRF requires a sweep through all $M|\mathcal{T}|$ training instances in order to compute and aggregate the gradient at each iteration. The primary computational burden is computing the expected values of the features (Eq. 5.21), however, the tree-structured graphical model shown in Figure 5.3 allows the junction tree algorithm to run in time linear in the number of auxiliary markers. On a standard laptop, we are able to train the model on 772 patients (5,458 PFVC measurements) in 10-20 minutes.

Online inference for predicting a given individual's future trajectory is also computationally straightforward. The key quantities are (1) the weights $p(z_i \mid \mathcal{H}(i,t))$ in Eq. 5.13, which are easily computed using the junction tree algorithm in time linear in the number of auxiliary markers, and (2) the subtype-specific predictive densities $p(y_i(\cdot) \mid z_i, y_{i,\leq t}, x_i)$, which have the same complexity

as the E-step in the LTM learning algorithm.

5.3 Experiments

We demonstrate our approach by building a tool for predicting lung disease trajectories for individuals with scleroderma. Lung disease is currently the leading cause of death among scleroderma patients, and is notoriously difficult to treat due to the lack of accurate predictors of decline and tremendous variability across individual trajectories [Allanore et al., 2015]. Clinicians use percent of predicted forced vital capacity (PFVC) to track lung severity, which is expected to drop as the disease progresses. In addition, they collect demographic information and other clinical marker values that measure the impact of disease on the different organ systems involved in scleroderma.

5.3.1 Data Description

We train and validate our model using data from the Johns Hopkins Scleroderma Center patient registry, one of largest collections of clinical scleroderma data in the world. Demographic information is collected during the patient's first visit to the clinic. PFVC and other clinical markers are collected during routine visits thereafter. To select individuals from the registry, we used the following criteria. First, we include individuals who were seen at the clinic within two years of their earliest scleroderma-related symptom[2] (1,186 individuals). Second, we exclude all individuals with fewer than two PFVC measurements after first being seen by the clinic (398 individuals). Finally, we exclude individuals who received a lung transplant (16 individuals) because their natural trajectory is altered by the intervention. Transplants are rare so removing patients with transplants should not introduce significant bias. We note that other interventions are common in scleroderma, but none have been proven to significantly alter the long-term course of the disease. For example, steroids are commonly administered, but there have been no randomized controlled trials confirming its effects on patients with scleroderma-related lung disease—see, for example, Ch. 35 in Varga et al. [2012]. Immunosuppressants are also commonly used to treat scleroderma-related lung dis-

[2]Date of first symptom is established during the first encounter by both the patient and clinician.

ease, but the proven effects are modest and have only been demonstrated over the course of one year [Tashkin et al., 2006]. We assume that these types of transient interventions are well-modeled by the individual-specific short-term component, and so we do not explicitly model the treatment effects of steroids or immunosuppressants in our data. Others have developed methods for estimating treatment effects from observational time series (see Athey et al. and references within; more recently, see Xu et al. [2016] for an application using functional data). Treatment effects can be incorporated within the trajectory likelihood in diseases where treatments are suspected to alter long term trajectory. We leave this more general case as a direction for future work. Our final data set contains 772 individuals and a total of $5,458$ PFVC measurements tracking individuals over a period of 20 years. The first, second, and third quartiles of the total number of PFVC measurements for an individual are 3, 5, and 9 respectively. The maximum number of PFVC measurements for one individual is 63. The first, second, and third quartiles of the measurement times are 1 year, 2.8 years, and 5.9 years. The first, second, and third quartiles of elapsed time between measurements are 0.4 years, 0.7 years, and 1.10 years. The minimum and maximum elapsed time is 0.002 years and 16.4 years respectively.

The baseline demographic information includes gender and African American race, both of which have been shown to be associated with disease severity in scleroderma [Allanore et al., 2015]. Antibody data are also collected at baseline, but since these are only available for a small subset of individuals, we do not include that data here. For time-dependent predictors, we include 5 auxiliary clinical markers. Three of the auxiliary markers are similar to PFVC in that they are continuous-valued test results used to measure the health of organ systems. We include: percent of predicted forced expiratory volume in one second (PFEV1), which measures the force with which air is expelled from the lungs; percent of predicted diffusing capacity (PDLCO), which measures the efficiency of oxygen diffusion from the lungs to the bloodstream; and total skin score (TSS), which is a cumulative measure of the thickness of the skin at various points on the body. In addition, we include 2 severity scores—clinical Likert-scaled judgements of organ damage severity: Raynaud's phenomenon (RP) severity score, which measures the severity of damage to the extremities by issues

related to the vasculature, and GI severity score that measures the severity of damage to the GI tract. For the interested reader, a more detailed discussion of these markers and their relationship to the disease can be found in Varga et al. [2012].

5.3.2 Experimental Setup

For the 4 continuous-valued clinical markers (PFVC, PFEV1, PDLCO, TSS) we use the LTM and for the 2 severity scores (GI and RP) we use a simpler model that we will describe later. For the population model, we use constant functions (i.e. the basis expansion $\Phi_p(t)$ contains an intercept term whose coefficient is determined by baseline covariates). For the subpopulation B-splines, we set boundary knots at 0 and 25 years (the maximum observation time in our data set is 23 years), use two interior knots that divide the time period from 0-25 years into three equally spaced chunks, and use quadratics as the piecewise components. For the individual-specific long-term basis Φ_ℓ, we use the same basis as the population model (constant functions).

We divide our data into 10 folds and use log-likelihood on the first fold for tuning hyperparameters. For PFVC, we select $G = 9$ subtypes using BIC. For the kernel hyperparameters $\Theta_1 = \{\Sigma_b, \alpha, \ell, \sigma^2\}$ we set $\Sigma_b \in \mathbb{R}$ to be 16.0, which corresponds to the variance of individual-specific intercepts. We set $\alpha = 6$, $\ell = 2$, and $\sigma^2 = 1$ using a grid search over values chosen using domain knowledge. Qualitatively, these make sense; we expect transient deviations to last around 2 years and to change PFVC by around ± 6 units. Finally, we penalize the expected log-likelihood with respect to $\vec{\beta}_{1:G}$ as in Eq. 2.2 and set the weight $\rho = 0.01$, which was chosen based on the clinical interpretability of the learned subtype trajectories. The remaining 9 folds were used for our cross-validation experiments. The parameters of each trajectory model are estimated independently for each fold (e.g. the B-spline coefficients of the subtype trajectories). For the severity scores, which are Likert-scaled and not continuous, we use a simple naive Bayes generative model wherein the latent "class" is an indicator of whether the individual ever reaches a high severity level (a cut-off in the severity scale determined by clinical collaborators). Severity score observations are treated as iid draws from a class-specific multinomial distribution (i.e. the likelihood for

these auxiliary markers is a multinomial distribution over severity scores). Finally, we estimate the parameters of the C-LTM by maximizing the objective in Eq. 5.14 augmented with an $L1$ regularizer. We optimize the objective using the Orthant-Wise Limited-memory Quasi-Newton (`OWL-QN`) algorithm [Andrew and Gao, 2007]. To generate training examples for the C-LTM, we use times $\mathcal{T} = \{1, 2, 4\}$ (the first three quintiles of observation times in our data) to fit three different models. We choose time points earlier in the disease course because this is when it is most valuable to leverage all available information. In our cross-validated experimental results below, we estimate the penalty of the $L1$ regularization term in each fold by splitting a portion of the training data into a development set. We sweep the penalty from 1.0×10^{-7} to 1.0×10^{-1} and choose based on development set performance.

5.3.3 Baselines

As a first baseline, we fit a regression model using static predictors only (features in $\vec{x}_i$). This is to compare against typical approaches in clinical prediction which rely only on observed features to predict disease progression (e.g. Khanna et al. 2011). The regression function is as follows, where $\Phi(t)$ is a B-spline basis:

$$\hat{y} \mid \vec{x}_{iz} = \Phi(t)^\top \left(\vec{\beta}_0 + \sum_{j=1}^{d} x_{ij}\beta_j + \sum_{j=1}^{d} \sum_{k \neq j} x_{ij} x_{ik} \beta_{ij} \right). \tag{5.25}$$

The following baselines reflect state-of-the-art approaches for dynamical prediction. The focus for each of these models, as discussed in the related work section, is on dynamical prediction of single marker trajectories using the marker history and static measurements collected during the first visit. The second baseline, like Rizopoulos [2011] and Shi et al. [2012], defines a single mean function parameterized in the same way as the first baseline and models individual-specific variations using a GP with the same kernel as in Chapter 4 (using hyper-parameters). The third baseline is a mixture of B-splines, which models subpopulations that can express different trajectory shapes (as in Proust-Lima et al. 2014). Finally, we use the LTM (no coupling to auxiliary markers) as a baseline. All B-spline bases used in these baseline models are parameterized in the same way as

the C-LTM (described above).

5.3.4 Evaluation

Prediction accuracy for all models is measured using the absolute error between the predicted and a smoothed version of the individual's observed trajectory. We make predictions after one, two, and four years of follow-up, which are summarized using averages computed in the second year of follow-up ($t \in (1, 2]$), in the third and fourth year of follow-up ($t \in (2, 4]$), fifth to eighth year of follow-up ($t \in (4, 8]$), and beyond the eighth year of follow-up ($t \in (8, 25]$)[3]. Mean absolute errors (MAE) and standard errors are estimated using 9-fold CV[4] at the level of individuals (i.e. all of an individual's data is held-out). Significance tests are computed against baselines using a paired t-test with point-wise predictions aggregated across folds.

5.3.5 Results

In this section, we present four sets of results. The first two are qualitative, and demonstrate the advantages of the C-LTM over the baseline models using examples. In the first qualitative analysis, we compare predictions made by C-LTM to those made by the B-spline mixture and the B-spline + GP. In the second qualitative analysis, we compare the C-LTM inferences with those from the LTM, which is a state-of-the-art single-marker model. The second two results are quantitative. The first compares predictive accuracies between the baseline models and the C-LTM. The second investigates clinical utility by using each model to predict a severity score that we use to detect individuals with aggressive lung disease.

Visual Comparison to Baselines

In Figures 5.4a, 5.4b, and 5.4c, we show the dynamic predictions made using the C-LTM, the B-spline mixture, and the B-spline + GP baselines on a sample patient.[5] For each model, we show 95% posterior intervals for the future trajectory. For the C-LTM and B-spline mixture, the most

[3] After the eighth year, data becomes too sparse to further divide this time span.
[4] Recall that the first of 10 folds is used for hyperparameter estimates.
[5] This patient was selected as an exemplar for the types of errors commonly made by the baseline models.

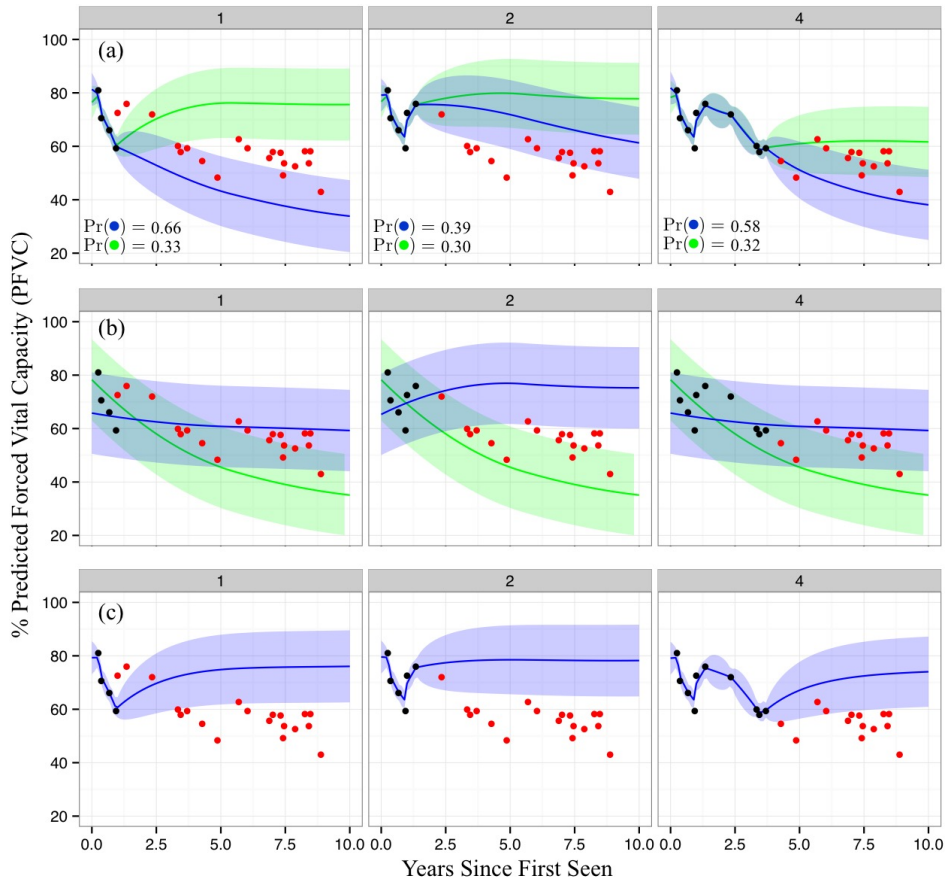


Figure 5.4: Examples of predictions made using 1, 2, and 4 years of data (moving across columns from left to right). Plot (a) shows dynamic predictions using C-LTM. Red markers are unobserved. Blue shows the trajectory predicted using the most likely subtype, and green shows the second most likely. Plot (b) shows dynamic predictions for the B-spline mixture baseline. Plot (c) shows the same for the B-spline + GP baseline.

likely subtype is shown in blue and the second most likely is in green. The B-spline mixture (Figure 5.4b) cannot explain individual-specific sources of variation (e.g. short-term deviations from the mixture mean) and so over-reacts to the slight rise in PFVC seen in the last two observed (black) measurements in the second panel (year 2). The B-spline + GP (Figure 5.4c) cannot capture long-term differences in trajectory means (e.g. due to subtypes) and so pulls back to the population mean over time even after four years of data suggest a declining trajectory. On the other hand, at year 1 the C-LTM (Figure 5.4a) maintains the hypothesis that the individual may decline or return to stability (correctly putting most weight on the former). After 2 years of data, the temporary recovery seems to have caused confidence in the declining trajectory to fall (going from 66% to

39%), but the top-weighted hypothesis is still correct. After 4 years of data, the model again becomes confident in the declining trajectory. Clinically, this robustness to short-term changes is important. After having seen the recovery between years 1 and 2, a clinician may become less immediately concerned with the individual's future lung disease, possibly delaying immunotherapy until a rapid decline becomes more evident. Note that the B-spline mixture, on the other hand, over-reacts to the recovery and predicts that the individual will continue to recover.

Analysis of Example Inferences

In Figure 5.5a-d, we show the C-LTM's target and auxiliary marker inferences for four different patients. For the target marker (PFVC) and auxiliary markers (TSS, PDLCO, and PFEV1), we show the most likely (blue) and second most likely (green) subtype and their corresponding trajectories. For the RP and GI severity score markers, we show the most likely severity class (high versus low). The dashed lines indicate the threshold at which high and low are determined based on judgements by our clinical collaborators. For PFVC, PFEV1, and PDLCO lower values indicate more severe progression. For TSS, higher values indicate severe progression. In Figures 5.5e-h, we show the predictions made by LTM to visually compare against predictions made using the baseline markers and PFVC history only (i.e. that do not leverage information from auxiliary markers).

In Figure 5.5a, we see a 55 year-old woman who presents with mildly impaired lung function (approximately 65 PFVC), but seems to recover over the course of the first year to reach a PFVC above 75 (considered by clinicians to be relatively healthy). Using this information alone, one may suspect that she will not have future lung issues. Indeed, this is what LTM predicts as shown in Figure 5.5e. By examining her auxiliary markers, however, we see that the picture is less clear. In particular, PFEV1 (a clinical marker closely related to PFVC) both decreases and increases over that period. C-LTM infers a mildly declining trajectory for PFEV1. In addition, PDLCO is also noisy and overall low, which suggests that the blood is not efficiently absorbing oxygen. This can happen for a number of reasons, but active lung disease is one of them. Finally, we see that her initial skin score is quite high and C-LTM projects it to stay high for the next few years, which is

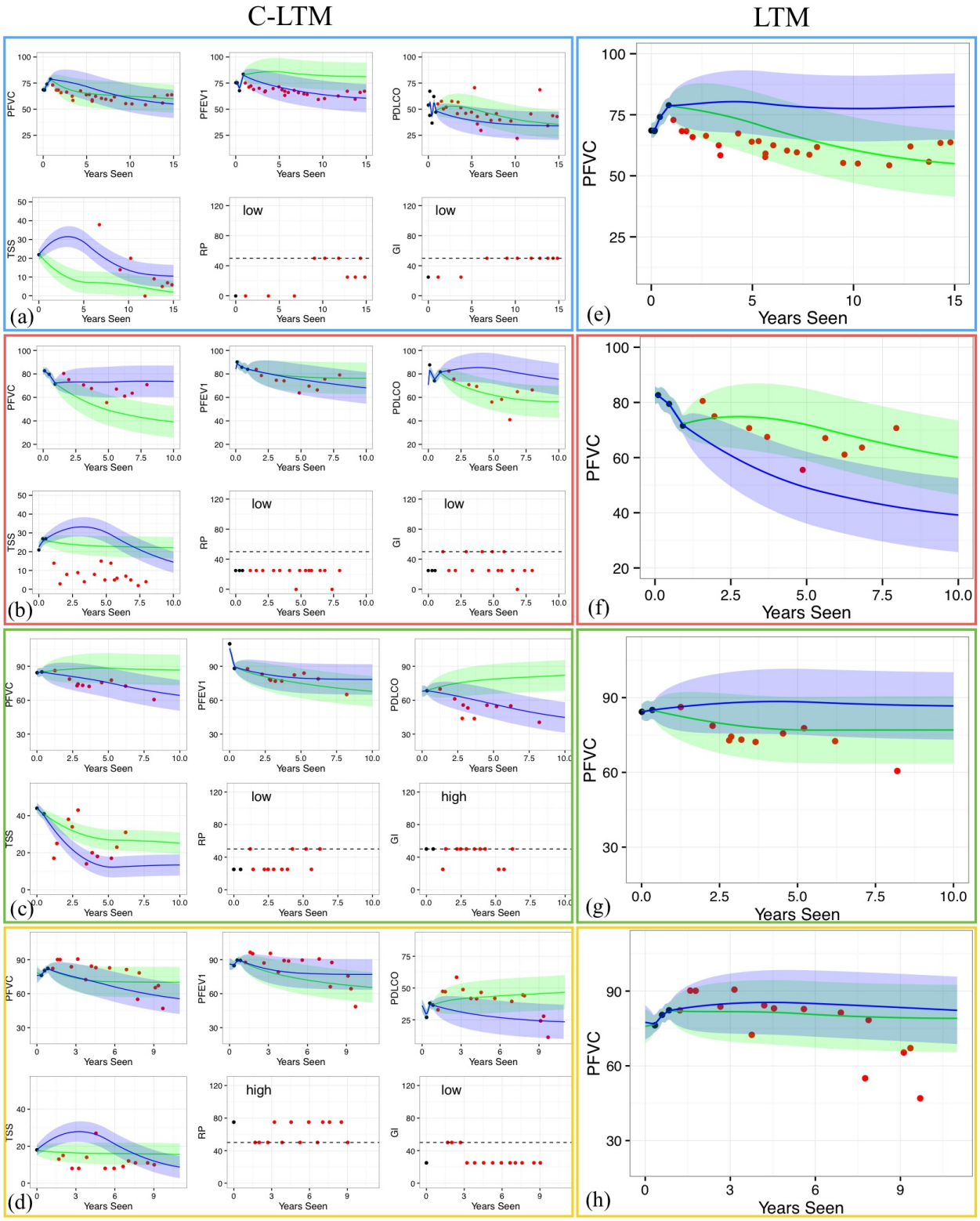


Figure 5.5: The predicted PFVC trajectory and the auxiliary markers are shown for two different patients. Red markers are unobserved. For the auxiliary markers TSS, PFEV1, and PDLCO we show the most likely (blue) and second most likely (green) subtype and their corresponding trajectories. For the RP and GI severity scores, we show the most likely severity class (high versus low). The dashed lines indicate the threshold at which high and low are determined clinically.

associated with active lung disease. We see that C-LTM has successfully incorporated inferences about the future trends of the auxiliary markers and correctly predicts that this woman's PFVC will decline after this initial improvement.

In Figure 5.5b, we see a 75 year-old white woman who presents with healthy lung function (approximately 85 PFVC), but is consistently declining over the course of the first year by nearly 15 PFVC. A clinical rule of thumb is that a drop in 10 PFVC over the course of a year warrants close monitoring for active lung disease. We see that LTM extrapolates this initial trend and predicts that this individual will continue to decline rapidly (Figure 5.5f). Just as in the previous example, however, the auxiliary markers paint a more complete picture of this individual. In the first few PFEV1 observations, we see that this decline is not quite as pronounced and the progression is predicted to be more mild. In PDLCO we see that oxygen is absorbed into the blood at healthy levels and also predicted to remain stable (although incorrectly in this case). Finally, C-LTM predicts that the RP and GI severity scores will remain low, which also supports the prediction that this woman will stabilize. Note that in this example C-LTM overestimates the course of PDLCO and TSS. Although the model still makes the correct prediction for PFVC in spite of this mistake, it highlights that the performance of our approach may be further improved with better auxiliary marker inferences. As research in systems biology yields new insights into modeling specific measurements more precisely, the modular architecture of C-LTM makes it possible to improve overall performance by incorporating improved versions of the target or auxiliary marker models.

In Figure 5.5c, we see a 76 year-old white woman that presents with healthy lung function (just under 90 PFVC), which also appears to be stable given the subsequent test result taken later that same year. The LTM predicts that this individual's most likely course is to remain stable. From the PFEV1 trajectory, however, we see that there was a large initial loss in PFEV1, which, together with the unusually high skin score (TSS) suggests that this woman's disease is active. The activity in the other organ systems allows the C-LTM to offset the stability seen in the first two PFVC measurements and correctly predict the consistently declining lung trajectory.

Predictions using 1 year of data				
Model	$(1, 2]$	$(2, 4]$	$(4, 8]$	$(8, 25]$
B-spline with Baseline Feats.	13.17 (0.43)	14.07 (0.61)	14.34 (0.65)	14.12 (1.04)
B-spline + GP	5.57 (0.24)	8.40 (0.19)	10.88 (0.42)	**11.74** (0.76)
B-spline Mixture	6.31 (0.22)	7.59 (0.36)	**9.82** (0.46)	13.77 (0.55)
LTM	5.70 (0.30)	8.02 (0.41)	11.17 (0.72)	13.93 (0.67)
C-LTM	★♣♦♠**5.12** (0.20)	★♣♦♠**6.88** (0.27)	★♣♠9.95 (0.51)	★13.70 (1.08)
Predictions using 2 years of data				
B-spline with Baseline Feats.		14.07 (0.61)	14.34 (0.65)	14.12 (1.04)
B-spline + GP		6.51 (0.19)	9.79 (0.35)	10.95 (0.68)
B-spline Mixture		6.17 (0.29)	8.34 (0.36)	12.19 (0.48)
LTM		5.74 (0.29)	8.08 (0.37)	**10.89** (0.62)
C-LTM		★♣♦♠**5.58** (0.34)	★♣**7.99** (0.61)	★♦11.27 (1.02)
Predictions using 4 years of data				
B-spline with Baseline Feats.			14.34 (0.65)	14.12 (1.04)
B-spline + GP			6.60 (0.24)	9.53 (0.56)
B-spline Mixture			6.00 (0.37)	10.11 (0.56)
LTM			**4.88** (0.28)	8.65 (0.59)
C-LTM			★♣♦5.04 (0.42)	★♣♦♠**8.07** (0.35)

Table 5.1: Mean absolute error of PFVC predictions for the B-spline with baseline features, the B-spline + GP, LTM, and C-LTM. Bold numbers indicate best performance across baseline models and C-LTM. ★ indicates statistically significant improvement against the B-spline model with baseline features only using a paired t-test ($\alpha = 0.05$). ♣ indicates statistical significance compared against the B-spline + GP. ♦ indicates statistical significance compared against the B-spline mixture. ♠ indicates statistical significance compared against LTM.

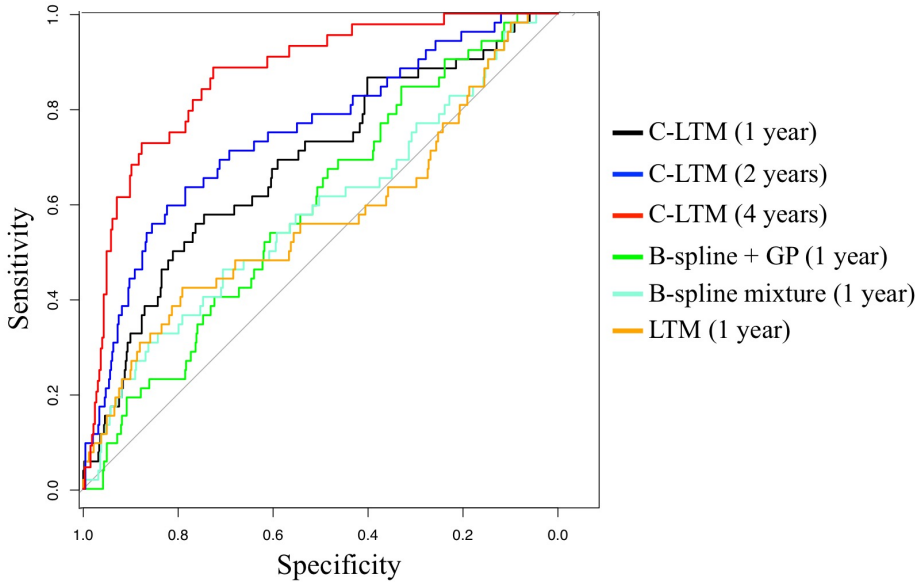
Finally, in Figure 5.5d, we see a 67 year-old African American man with mildly impaired lung function early in the disease course (around 75 PFVC) that seems to recover over the next one or two years to a healthier 85 PFVC. In Figure 5.5h, we see that the LTM predicts that a stable trajectory thereafter is likely. By considering other organ systems, however, we see that this man's blood-oxygen diffusion is severely limited early in the disease course (nearly 25% of the predicted DLCO). Moreover, we see that the this individual's Raynaud's phenomenon severity score is high early on and correctly predicted to remain that way. The low PDLCO and high RP severity score point to active vasculature disease, which is hypothesized to cause late deterioration in lung function. We see that C-LTM correctly uses this evidence to predict an accurate disease trajectory.

Predictive Accuracy

In Table 5.1, we report performance of the C-LTM, LTM, and the three other baseline models. First, we note that the C-LTM statistically significantly outperforms the B-spline with baseline features for all predictions. This baseline makes static predictions using baseline information only, and cannot adapt to an individual as new data becomes available. Moreover, after an initial amount

Model / Years of Data	1	2	4
B-spline + GP	0.59	0.63	0.74
B-spline mixture	0.58	0.63	0.76
LTM	0.57	0.71	0.84
C-LTM	**0.68**	**0.75**	**0.87**

(a) AUCs for detecting declining individuals.



(b) ROCs comparing B-spline + GP at 1 year, B-spline mixture at 1 year, LTM at 1 year, and C-LTM at years 1, 2, and 4.

Figure 5.6: Declining individual detection results.

of data has been collected on an individual, C-LTM statistically significantly outperforms all other models. This is not surprising. When compared to the LTM, we see that C-LTM benefits from leveraging information from auxiliary markers. As more information is collected, both models are able to the individual and provide comparable predictions. The B-spline mixture is not able to personalize beyond capturing long-term trends across subpopulations, so we see that it becomes less competitive compared to both C-LTM and LTM as more data are collected. Finally, the B-spline + GP cannot capture long-term trends specific to subpopulations (as we saw in Section 5.3.5), and so we see that it does poorly when making predictions.

Clinical Utility

One may naturally wonder whether the observed improvements in MAE reported above translate to practical benefits in the clinic. In the examples shown in Figure 5.5, we have walked through cases where the model makes predictions that would seem unlikely if we were to consider PFVC alone. This suggests that the model can augment expert clinical judgement and may serve to protect against incorrect extrapolations. In this section, we further elaborate upon this intuition by studying clinical utility quantitatively. In particular, we compare how well the B-spline + GP, B-spline mixture, LTM, and C-LTM are able to detect individuals who will have rapidly declining lung function. It is notoriously difficult to predict which scleroderma patients will rapidly decline using only information from early in the disease course. In addition to improving prognoses, more accurate detection of rapidly declining lung function can help to improve the recruitment for clinical trials evaluating drugs for scleroderma-related lung disease. If we include many individuals in a study who are predicted to have active lung disease but do not, the results of the study are blurred because both arms of the trial may include many individuals without active lung disease.

To test how well these different models can detect individuals that will experience rapidly declining lung function, we use the predictions of future PFVC measurements to produce a score. The score is defined to be the difference between the individual's first PFVC measurement and the minimum predicted value in the future—this will be higher for individuals on whom a model predicts deteriorating lung function and lower for those predicted to be stable. To label an individual as declining, we require that they (1) have at least one observation within the first year of being seen by the clinic, (2) have 3 years between their first and last measurements, (3) have at least 4 PFVC measurements, and (4) have an initial PFVC measurement that is 20 PFVC higher than their last measurement. Requirements (2) and (3) are to ensure that the trajectory can be reliably annotated as declining or not. For each model, we make predictions at years 1, 2, and 4 and compute the score described above for each individual. We then compute the AUC for each model at each year. Table 5.6a displays the results of this experiment. We see that C-LTM achieves higher AUC at all years than the baseline models. Figure 5.6b displays the ROCs for the B-Spline + GP (green), B-Spline

mixture (cyan), LTM (orange), and the C-LTM (black) at year 1 and also includes the ROCs for the C-LTM model at years 2 (blue) and 4 (red) to visualize how performance improves as more data is added. Clinically, an AUC of 0.87 for predicting individuals with lung decline after—on average—four years of data is high and has not been shown previously.

5.4 Discussion

The goal of personalized (also called precision) medicine is to develop tools that help to tailor prognoses to the characteristics and unique medical history of the individual. In this paper, we describe an approach to personalized prognosis that uses an integrative analysis of multiple clinical marker histories from the individual's medical records. Our approach combines single-marker latent variable models (the LTM) with a CRF coupling model to make more accurate predictions about the future trajectory of a target clinical marker.

The coupled model (C-LTM) has several advantages. First, the marker-specific LTMs account for marker trajectory shapes using components at the population, subpopulation, individual long-term, and individual short-term levels, which simultaneously allows for heterogeneity across and within individuals, and enables statistical strength to be shared across observations at different "resolutions" of the data. Within an individual marker model, the population and subpopulation components are learned offline, while estimates of the individual-specific parameters are refined over the course of the disease as data accrues for that individual. Second, our coupling model allows us to condition both the target and auxiliary marker histories to make predictions about the future target marker trajectory. We therefore use the marker-specific latent variable models to neatly summarize and extract information from the irregularly sampled and sparse data, while simultaneously sidestepping the issue of jointly modeling both the target and auxiliary markers. Our conditional formulation is less sensitive to misspecified dependencies between different marker types and can also be easily scaled to diseases with a large number of auxiliary markers. Finally, our model aligns with clinical practice; predictions are dynamically updated in continuous time as new marker observations are measured. We also note that our description of the method and the

experimental results focus on predicting the trajectory of a single clinical marker, but multiple latent factor regression models can be easily fit so that many markers can be simultaneously predicted. Using this extension, we only need to maintain different CRF parameters; the latent variable models are shared since they are fit independently as a precursor to learning the CRF.

There are several shortcomings of our approach that are promising directions for future research. First, the model implicitly assumes that the data generating process is noninformative (i.e. missing data is missing at random [Little and Rubin, 2014]). This is appropriate for clinical markers that are routinely collected, but additional machinery would be required to model markers whose missingness is informative. For example, in some cases, additional measurements may be made due to clinical suspicion caused by factors that are not clearly document in the health record. Researchers have begun to explore more complex missing-data mechanisms for disease trajectory modeling (see e.g. Lange et al. [2015]), and it will be important to incorporate these ideas into the framework discussed here to integrate the full set of markers measured during a clinical visit. Another shortcoming is our focus on discrete latent factors of the auxiliary marker trajectories. Continuous-valued latent factors may also be useful, but would make learning and inference in the latent factor CRF more challenging.

There are also several other immediate opportunities for improving the model. Auxiliary markers are integrated via separate marker-specific generative models. While we incorporated two different types of models—trajectory and maximum-severity based—both of which were data driven, existing and new clinical knowledge should be brought to bear to improve these models, which we expect will improve predictions of the target trajectories. Further, in this work, we focused on modeling the dependency of the target subtype on the auxiliary markers. In addition, estimates of the individual-specific long-term and short-term components may also benefit from conditioning on the auxiliary markers. Finally, the parameters for the pairwise potentials learned in our model may serve as a means for generating hypotheses about the co-evolution of organ-specific trajectories.

The ideas described here also open up other longer-term directions for future work. The C-LTM does not account for the effects of treatment on an individual's long-term trajectory. In many

chronic conditions, as is the case for scleroderma, drugs only provide short-term relief (accounted for in our model by the individual-specific adjustments). However, if treatments that alter long-term course are available and commonly prescribed, then these should be included within the model as an additional component that influences the trajectory. Learning these treatment effects from noisy electronic health record data (e.g., Xu et al. 2016) present an exciting and challenging direction for future work.

We have demonstrated our model by developing a prognostic tool for predicting lung disease trajectories in patients with scleroderma, an autoimmune disease. We showed that the C-LTM makes more accurate predictions than state-of-the-art approaches. Accurate tools for prognosis can allow clinicians and patients to more actively manage their disease. These tools can also help to enrich clinical trials, which commonly fail in complex, heterogeneous diseases due to inadequate power. While we have focused model development and evaluation on scleroderma, this work is broadly applicable to other complex diseases [Craig, 2008], many of which track disease activity using clinical scales of severity. The C-LTM is most directly applicable to complex and heterogeneous chronic diseases. Examples of such diseases include lupus, multiple sclerosis, inflammatory bowel disease (IBD), chronic obstructive pulmonary disease (COPD), and asthma. Extending the ideas in this chapter to these other diseases is an opportunity to address important open challenges in precision medicine.

Chapter 6

Causal Trajectory Models and Reliable Decision Support

Decision-makers in medicine are often faced with the challenge of estimating what is likely to happen when they take an action. One use of such an estimate is to evaluate *risk*; e.g. is this patient likely to die if I do not intervene? Another use is "what if?" reasoning to compare outcomes under alternative actions; e.g. would changing therapy improve this patient's outcome? In this chapter, we discuss conditions under which predictive models (such as those in Chapters 4 and 5) can be used to reliably answer such questions. In general, we show that standard supervised learning algorithms can give unreliable, and even dangerous, answers to these important questions.

Consider, for instance, recent work to predict risk of death for hospitalized patients with pneumonia [Caruana et al., 2015]. The predictions were intended to help doctors decide whether a patient should be sent home or kept in the hospital (i.e. send home if low-risk, otherwise treat further). A subtle assumption underlying the intended use is that the risk prediction should represent the patient's probability of dying *without further treatment*. After fitting the model, Caruana et al. [2015] found that asthma counterintuitively reduced the probability of death due to pneumonia. They traced the result back to a policy at the hospitals that supplied the training data. At those hospitals, asthmatics with pneumonia were directly admitted to the intensive care unit (ICU), therefore receiving more aggressive treatment.

The relationship between asthma and risk appeared counterintuitive because of a *mismatch* between the policy at test time and the treatment policy in the training data. If asthmatics are always given more care, it is natural to learn that risk of death is lower for asthmatics. This relationship is surprising, however, if we interpret the model's prediction as risk of death without further treatment. In general, we expect that models used in decision support tools will frequently be applied in settings where the policy is different from the one in the training data. This motivates our definition of a *stable learning algorithm*: one that produces a model that is independent of the policy in the training data. Because the learner does not depend on the policy in the training data, it is robust to *policy shifts* that induce a mismatch between the train and test distributions. Learning algorithms that are not stable produce models that are unreliable, and can even lead to harmful decisions; e.g. "send a patient with asthma home because they have low-risk of death".

To design stable learning algorithms, we show how to use learning objectives that predict *counterfactuals*, which are collections of random variables $\{Y[a] : a \in \mathcal{C}\}$ used in the *potential outcomes* framework [Neyman, 1923, 1990, Rubin, 1978]. Counterfactuals model the outcome Y after an action a is taken from a set of choices $\mathcal{C}$. Counterfactual predictions are broadly applicable to a number of decision-support tasks. In medicine, for instance, when evaluating a patient's risk of death Y to determine whether they should be treated aggressively, we want an estimate of how they will fare *without* treatment. This can be done by predicting the counterfactual $Y[\varnothing]$, where $\varnothing$ stands for "do nothing". In online marketing, to decide whether we should display ad a_1 or a_2, we may want an estimate of click-through Y under each (i.e. predict $Y[a_1]$ and $Y[a_2]$).

To build stable predictive models in temporal settings, we develop the Counterfactual Gaussian Process (CGP) to predict the counterfactual future progression of continuous-time trajectories under sequences of future actions. The CGP can be learned from and applied to time series data where actions are taken and outcomes are measured at irregular time points; a generalization of discrete time series. Figure 6.1 illustrates an application of the CGP. We show an individual with a lung disease, and would like to predict her future lung capacity (y-axis). Panel (a) shows the *history* in the red box, which includes previous lung capacity measurements (black dots) and

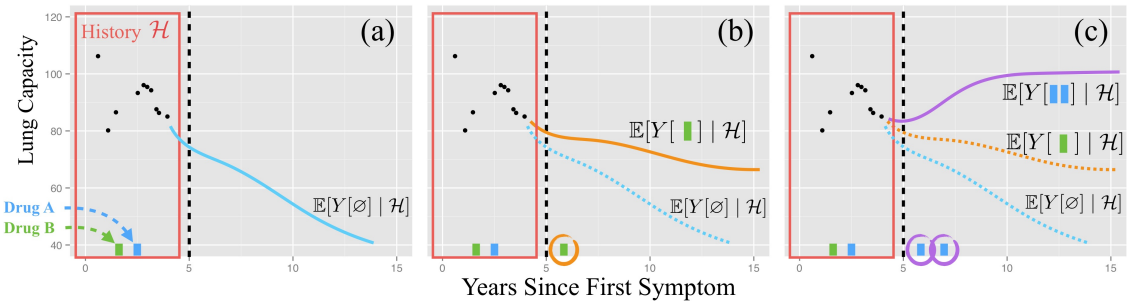


Figure 6.1: Best viewed in color. An illustration of the counterfactual GP applied to health care. The red box in (a) shows previous lung capacity measurements (black dots) and treatments (the history). Panels (a)-(c) show the type of predictions we would like to make. We use $Y[a]$ to represent the potential outcome under action a.

previous treatments (green and blue bars). The blue counterfactual trajectory shows what might occur under *no action*, which can be used to evaluate this individual's risk. In panel (b), we show the counterfactual trajectory under a single future green treatment. Panel (c) illustrates "what if?" reasoning by overlaying counterfactual trajectories under two different action sequences; in this case it seems that two future doses of the blue drug may lead to a better outcome than a single dose of green.

6.1 Contributions

Our key methodological contribution is the Counterfactual Gaussian process (CGP), a model that predicts how a continuous-time trajectory will progress under sequences of actions. We derive an adjusted maximum likelihood objective that learns the CGP from *observational traces*; irregularly sampled sequences of actions and outcomes denoted using $\mathcal{D} = \{\{(y_{ij}, a_{ij}, t_{ij})\}_{j=1}^{n_i}\}_{i=1}^{m}$, where $y_{ij} \in \mathbb{R} \cup \{\varnothing\}$, $a_{ij} \in \mathcal{C} \cup \{\varnothing\}$, and $t_{ij} \in [0, \tau]$.[1] Our objective accounts for and removes the effects of the policy used to choose actions in the observational traces, making the algorithm *stable to policy shift*. We derive the objective by jointly modeling observed actions and outcomes using a marked point process (MPP; see e.g., Daley and Vere-Jones 2007), and show how it correctly learns the CGP under a set of assumptions analagous to those required to learn counterfactual models in

[1] y_{ij} and a_{ij} may be the null variable $\varnothing$ to allow for the possibility that an action is taken but no outcome is observed and vice versa. $[0, \tau]$ denotes a fixed period of time over which the trajectories are observed.

other settings.

We demonstrate the CGP on two decision-support tasks. First, we use the CGP to predict risk: the likelihood of a poor outcome given that we do not intervene. We show that the CGP is stable and does not depend on the action policy in the training data. On the other hand, we show that predictions made by a model trained using a classical supervised learning objective is unstable. In our second experiment, we use data from a real intensive care unit (ICU) to learn the CGP, and qualitatively demonstrate how the CGP can be used to compare counterfactuals and answer "what if?" questions, which could offer medical decision-makers a powerful new tool for individualized treatment planning.

6.2 Related Work

Decision support is a rich field; because our main methodological contribution is a counterfactual model for time series data, we limit the scope of our discussion of related work to this area.

6.2.1 Causal Inference

Counterfactual models stem from causal inference. In that literature, the difference between the counterfactual outcomes if an action had been taken and if it had not been taken is defined as the *causal effect* of the action (see e.g., Pearl 2009 or Morgan and Winship 2014). *Potential outcomes* are commonly used to formalize counterfactuals and obtain causal effect estimates [Neyman, 1923, 1990, Rubin, 1978]. Potential outcomes are often applied to cross-sectional data; see, for instance, the examples in Morgan and Winship 2014. Recent examples from the machine learning literature are Bottou et al. [2013] and Johansson et al. [2016].

6.2.2 Potential Outcomes in Discrete Time

Potential outcomes have also been used to estimate the causal effect of a sequence of actions in discrete time on a final outcome (e.g. Robins 1986, Robins and Hernán 2009, Taubman et al. 2009). The key challenge in the sequential setting is to account for feedback between intermediate

outcomes that determine future treatment. Conversely, Brodersen et al. [2015] estimate the effect that a *single discrete intervention* has on a *discrete* time series. Recent work on optimal dynamic treatment regimes uses the sequential potential outcomes framework proposed by Robins [1986] to learn lists of discrete-time treatment rules that optimize a scalar outcome. Algorithms for learning these rules often use action-value functions (Q-learning; e.g., Nahum-Shani et al. 2012). Alternatively, A-learning is a semiparametric approach that directly learns the relative difference in value between alternative actions [Murphy, 2003].

6.2.3 Potential Outcomes in Continuous Time

Others have extended the potential outcomes framework in Robins [1986] to learn causal effects of actions taken in continuous-time on a single final outcome using observational data. Lok [2008] proposes an estimator based on structural nested models [Robins, 1992] that learns the instantaneous effect of administering a single type of treatment. Arjas and Parner [2004] develop an alternative framework for causal inference using Bayesian posterior predictive distributions to estimate the effects of actions in continuous time on a final outcome. Both Lok [2008] and Arjas and Parner [2004] use marked point processes to formalize assumptions that make it possible to learn causal effects from continuous-time observational data. We build on these ideas to learn causal effects of actions on continuous-time *trajectories* instead of a single outcome. There has also been recent work on building expressive models of treatment effects in continuous time. Xu et al. [2016] propose a Bayesian nonparametric approach to estimating individual-specific treatment effects of discrete but irregularly spaced actions, and Soleimani et al. [2017] model the effects of continuous-time, continuous-valued actions. Causal effects in continuous-time have also been studied using differential equations. Mooij et al. [2013] formalize an analog of Pearl's "do" operation for deterministic ordinary differential equations. Sokol and Hansen [2014] make similar contributions for stochastic differential equations by studying limits of discrete-time non-parametric structural equation models [Pearl, 2009]. Cunningham et al. [2012] introduce the Causal Gaussian Process, but their use of the term "causal" is different from ours, and refers to a constraint that holds for sample paths of

the GP.

6.2.4 Reinforcement Learning

Reinforcement learning (RL) algorithms learn from data where actions and observations are interleaved in discrete time (see e.g., Sutton and Barto 1998). In RL, however, the focus is on learning a *policy* (a map from states to actions) that optimizes the expected reward, rather than a model that predicts the effects of the agent's actions on future observations. In model-based RL, a model of an action's effect on the subsequent state is produced as a by-product either offline before optimizing the policy (e.g., Ng et al. 2006) or incrementally as the agent interacts with its environment. In most RL problems, however, learning algorithms rely on active experimentation to collect samples. This is not always possible; for example, in healthcare we cannot actively experiment on patients, and so we must rely on retrospective observational data. In RL, a related problem known as off-policy evaluation also uses retrospective observational data (see e.g., Dudík et al. 2011, Swaminathan and Joachims 2015, Jiang and Li 2016, Păduraru et al. 2012, Doroudi et al. 2017). The goal is to use state-action-reward sequences generated by an agent operating under an unknown policy to estimate the expected reward of a target policy. Off-policy algorithms typically use action-value function approximation, importance reweighting, or doubly robust combinations of the two to estimate the expected reward.

6.3 Counterfactual Models from Observational Traces

Counterfactual GPs build on ideas from potential outcomes [Neyman, 1923, 1990, Rubin, 1978], Gaussian processes [Rasmussen and Williams, 2006], and marked point processes [Daley and Vere-Jones, 2007]. In the interest of space, we review potential outcomes and marked point processes, but refer the reader to Rasmussen and Williams [2006] for background on GPs.

6.3.1 Background: Potential Outcomes

To formalize counterfactuals, we adopt the potential outcomes framework [Neyman, 1923, 1990, Rubin, 1978], which uses a collection of random variables $\{Y[a] : a \in \mathcal{C}\}$ to model the outcome after each action a from a set of choices $\mathcal{C}$. To make counterfactual predictions, we must learn the distribution $P(Y[a] \mid X)$ for each action $a \in \mathcal{C}$ given features X. If we can freely experiment by repeatedly taking actions and recording the effects, then it is straightforward to fit a predictive model. Conducting experiments, however, may not be possible. Alternatively, we can use observational data, where we have example actions A, outcomes Y, and features X, but do not know how actions were chosen. Note the difference between the action a and the random variable A that models the *observed actions* in our data; the notation $Y[a]$ serves to distinguish between the observed distribution $P(Y \mid A, X)$ and the target distribution $P(Y[a] \mid X)$.

In general, we can only use observational data to estimate $P(Y \mid A, X)$. Under two assumptions, however, we can show that this conditional distribution is equivalent to the counterfactual model $P(Y[a] \mid X)$. The first is known as the Consistency Assumption.

Assumption 1 (Consistency)**.** *Let Y be the observed outcome, $A \in \mathcal{C}$ be the observed action, and $Y[a]$ be the potential outcome for action $a \in \mathcal{C}$, then:* $(Y \triangleq Y[a]) \mid A = a$.

Under consistency, we have that $P(Y \mid A = a) = P(Y[a] \mid A = a)$. Now, the potential outcome $Y[a]$ may depend on the action A, so in general $P(Y[a] \mid A = a) \neq P(Y[a])$. The next assumption posits that the features X include all possible *confounders* [Morgan and Winship, 2014], which are sufficient to d-separate $Y[a]$ and A.

Assumption 2 (No Unmeasured Confounders (NUC))**.** *Let Y be the observed outcome, $A \in \mathcal{C}$ be the observed action, X be a vector containing all potential confounders, and $Y[a]$ be the potential outcome under action $a \in \mathcal{C}$, then:* $(Y[a] \perp A) \mid X$.

Under Assumptions 1 and 2, $P(Y \mid A, X) = P(Y[a] \mid X)$. An extension of Assumption 2 introduced by Robins [1997] known as *sequential NUC* allows us to estimate the effect of a sequence of actions in discrete time on a single outcome. In continuous-time settings, where both the type

and *timing* of actions may be statistically dependent on the potential outcomes, Assumption 2 (and sequential NUC) cannot be applied as-is. We will describe an alternative that serves a similar role for CGPs.

6.3.2 Background: Marked Point Processes

Point processes are distributions over sequences of timestamps $\{T_i\}_{i=1}^N$, which we call points, and a marked point process (MPP) is a point process where each point is annotated with an additional random variable X_i, called its mark. For example, a point T might represent the arrival time of a customer, and X the amount that she spent at the store. We emphasize that both the annotated points (T_i, X_i) and the number of points N are random variables.

A point process can be characterized as a counting process $\{N_t : t \geq 0\}$ that counts the number of points that occured up to and including time t: $N_t = \sum_{i=1}^N \mathbb{I}_{(T_i \leq t)}$. By definition, this processes can only take integer values, and $N_t \geq N_s$ if $t \geq s$. In addition, it is commonly assumed that $N_0 = 0$ and that $\Delta N_t = \lim_{\delta \to 0^+} N_t - N_{t-\delta} \in \{0, 1\}$. We can parameterize a point process using a probabilistic model of ΔN_t given the history of the process $\mathcal{H}_{t^-}$ up to but not including time t (we use t^- to denote the left limit of t). Using the Doob-Meyer decomposition [Daley and Vere-Jones, 2007], we can write $\Delta N_t = \Delta M_t + \Delta \Lambda_t$, where M_t is a martingale, Λ_t is a cumulative intensity function, and

$$P(\Delta N_t = 1 \mid \mathcal{H}_{t^-}) = \mathbb{E}\left[\Delta N_t \mid \mathcal{H}_{t^-}\right] = \mathbb{E}\left[\Delta M_t \mid \mathcal{H}_{t^-}\right] + \Delta \Lambda_t(\mathcal{H}_{t^-}) = 0 + \Delta \Lambda_t(\mathcal{H}_{t^-}),$$

which shows that we can parameterize the point process using the conditional intensity function $\lambda^*(t)\,\mathrm{d}t \triangleq \Delta \Lambda_t(\mathcal{H}_{t^-})$. The star superscript on the intensity function serves as a reminder that it depends on the history $\mathcal{H}_{t^-}$. For example, in non-homogeneous Poisson processes $\lambda^*(t)$ is a function of time that does not depend on the history. On the other hand, a Hawkes process is an example of a point process where $\lambda^*(t)$ *does* depend on the history [Hawkes, 1971]. MPPs are defined by an intensity that is a function of both the time t and the mark x: $\lambda^*(t, x) = \lambda^*(t) p^*(x \mid t)$. We have written the joint intensity in a factored form, where $\lambda^*(t)$ is the intensity of *any* point occuring

(that is, the mark is unspecified), and $p^*(x \mid t)$ is the pdf of the observed mark given the point's time. For an MPP, the history $\mathcal{H}_t$ contains each prior point's time and mark.

6.3.3 Counterfactual Gaussian Processes

Let $\{Y_t : t \in [0, \tau]\}$ denote a continuous-time stochastic process, where $Y_t \in \mathbb{R}$, and $[0, \tau]$ defines the interval over which the process is defined. We will assume that the process is observed at a discrete set of irregular and random times $\{(y_j, t_j)\}_{j=1}^n$. We use $\mathcal{C}$ to denote the set of possible *action types*, $a \in \mathcal{C}$ to denote the elements of the set, and define an action to be a 2-tuple (a, t) specifying an action type $a \in \mathcal{C}$ and a time $t \in [0, \tau]$ at which it is taken. To refer to multiple actions, we use $\mathbf{a} = [(a_1, t_1), \ldots, (a_n, t_n)]$. Finally, we define the history $\mathcal{H}_t$ at a time $t \in [0, \tau]$ to be a list of all previous observations of the process and all previous actions. Our goal is to model the counterfactual:

$$P(\{Y_s[\mathbf{a}] : s > t\} \mid \mathcal{H}_t), \text{ where } \mathbf{a} = \{(a_j, t_j) : t_j > t\}_{j=1}^m. \tag{6.1}$$

To learn the counterfactual model, we will use *traces* $\mathcal{D} \triangleq \{\mathbf{h}_i = \{(t_{ij}, y_{ij}, a_{ij})\}_{j=1}^{n_i}\}_{i=1}^m$, where $y_{ij} \in \mathbb{R} \cup \{\varnothing\}$, $a_{ij} \in \mathcal{C} \cup \{\varnothing\}$, and $t_{ij} \in [0, \tau]$. Our approach is to model $\mathcal{D}$ using a marked point process (MPP), which we learn using the traces. Using Assumption 1 and two additional assumptions defined below, the estimated MPP recovers the counterfactual model in Equation 6.1.

We define the MPP mark space as the Cartesian product of the outcome space $\mathbb{R}$ and the set of action types $\mathcal{C}$. To allow either the outcome or the action (but not both) to be the null variable $\varnothing$, we introduce binary random variables $z_y \in \{0, 1\}$ and $z_a \in \{0, 1\}$ to indicate when the outcome y and action a are not $\varnothing$. Formally, the mark space is $\mathcal{X} = (\mathbb{R} \cup \{\varnothing\}) \times (\mathcal{C} \cup \{\varnothing\}) \times \{0, 1\} \times \{0, 1\}$. We can then write the MPP intensity as

$$\lambda^*(t, y, a, z_y, z_a) = \underbrace{\lambda^*(t) p^*(z_y, z_a \mid t)}_{\text{[A] Event model}} \quad \underbrace{p^*(y \mid t, z_y)}_{\text{[B] Outcome model (GP)}} \quad \underbrace{p^*(a \mid y, t, z_a)}_{\text{[C] Action model}}, \tag{6.2}$$

where we have again used the $*$ superscript as a reminder that the hazard function and densities

above are implicitly conditioned on the history $\mathcal{H}_{t^-}$. The parameterization of the event and action models can be chosen to reflect domain knowledge about how the timing of events and choice of action depend on the history. The outcome model is parameterized using a GP (or any elaboration such as a hierarchical GP or mixture of GPs), and can be treated as a standard regression model that predicts how the future trajectory will progress given the previous actions and outcome observations.

Learning

To learn the CGP, we maximize the likelihood of observational traces over a fixed interval $[0, \tau]$. Let $\boldsymbol{\theta}$ denote the model parameters, then the likelihood for a single trace is

$$\ell(\boldsymbol{\theta}) = \sum_{j=1}^{n} \log p^*_\theta(y_j \mid t_j, z_{yj}) + \sum_{j=1}^{n} \log \lambda^*_\theta(t_j) p^*_\theta(a_j, z_{yj}, z_{aj} \mid t_j, y_j) - \int_0^\tau \lambda^*_\theta(s)\, \mathrm{d}s. \tag{6.3}$$

We assume that traces are independent, and so can learn from multiple traces by maximizing the sum of the individual-trace log likelihoods with respect to $\boldsymbol{\theta}$. We refer to Equation 6.3 as the adjusted maximum likelihood objective. We see that the first term fits the GP to the outcome data, and the second term acts as an adjustment to account for dependencies between future outcomes and the timing and types of actions that were observed in the training data.

Connection to Target Counterfactual

By maximizing Equation 6.3, we obtain a statistical model of the observational traces $\mathcal{D}$. In general, the statistical model may not recover the target counterfactual model (Equation 6.1). To connect the CGP to Equation 6.1, we describe two additional assumptions. The first assumption is an alternative to Assumption 2.

Assumption 3 (Continuous-Time NUC)**.** *For all times t and all histories $\mathcal{H}_{t^-}$, the densities $\lambda^*(t)$, $p^*(z_y, z_a \mid t)$, and $p^*(a \mid y, t, z_a)$ do not depend on $Y_s[\mathbf{a}]$ for all times $s > t$ and all actions $\mathbf{a}$.*

The key implication of this assumption is that the policy used to choose actions in the observational data did not depend on any unobserved information that is predictive of the future potential outcomes.

	Regime A		Regime B		Regime C	
	Baseline GP	CGP	Baseline GP	CGP	Baseline GP	CGP
Risk Score Δ from A	0.000	0.000	0.083	0.001	0.162	0.128
Kendall's τ from A	1.000	1.000	0.857	0.998	0.640	0.562
AUC	0.853	0.872	0.832	0.872	0.806	0.829

Table 6.1: Results measuring reliability for simulated data experiments. See Section 6.4.1 for details.

Assumption 4 (Non-Informative Measurement Times)**.** *For all times t and any history $\mathcal{H}_{t^-}$, the following holds:* $p^*(y \mid t, z_y = 1)\,\mathrm{d}y = P(Y_t \in \mathrm{d}y \mid \mathcal{H}_{t^-})$.

Under Assumptions 1, 3, and 4, we can show that Equation 6.1 is equivalent to the GP used to model $p^*(y \mid t, z_y = 1)$. In the interest of space, the argument for this equivalence is deferred to Section 6.7, which uses potential outcomes. We make an equivalent argument in Section 6.8 using the language of causal Bayesian networks instead. Note that these assumptions are not statistically testable (see e.g., Pearl 2009).

6.4 Experiments

We demonstrate the CGP on two decision-support tasks. First, we use the CGP for risk prediction and show that it is stable; i.e. its predictions are insensitive to the action policy in the training data. Classical supervised learning algorithms, however, are unstable (they depend on the action policy) and this can make them unreliable decision-support tools. Second, we show how the CGP can be used to compare counterfactuals and ask "what if?" questions for individualized treatment planning by learning the effects of dialysis on creatinine levels using real data from an intensive care unit (ICU).

6.4.1 Reliable Risk Prediction with CGPs

We first show how the CGP can be used for reliable risk prediction, where the objective is to predict the likelihood of an adverse event. In this section, we use simulated data so that we can evaluate using the true risk on test data. For concreteness, we frame our experiment within a healthcare

setting, but the ideas can be more broadly applied. Suppose that a clinician records a real-valued measurement over time that reflects an individual's health, which we call a *severity marker*. We consider the individual to *not* be at risk if the severity marker is unlikely to fall below a particular threshold in the future without intervention. As discussed by Caruana et al. [2015], modeling risk can help caregivers decide whether they need to intervene.

We simulate the value of a severity marker recorded over a period of 24 hours in the hospital; high values indicate that the patient is healthy. A natural approach to predicting risk at time t is to model the conditional distribution of the severity marker's future trajectory given the history up until time t; i.e. $P(\{Y_s : s > t\} \mid \mathcal{H}_t)$. We use this as our baseline. As an alternative, we use the CGP to explicitly model the counterfactual "What if we do not treat this patient?"; i.e. $P(\{Y_s[\varnothing] : s > t\} \mid \mathcal{H}_t)$. For all experiments, we consider a single decision time $t = 12$hrs. To quantify risk, we use the negative of each model's predicted value at the end of 24 hours, normalized to lie in $[0, 1]$.

Data

We simulate training and test data from three regimes. In regimes A and B, we simulate severity marker trajectories that are treated by policies π_A and π_B respectively, which are both unknown to the baseline model and CGP at train time. Both π_A and π_B are designed to satisfy Assumptions 1, 3, and 4. In regime C, we use a policy that *does not* satisfy these assumptions. This regime will demonstrate the importance of verifying whether the assumptions hold when applying the CGP. We train both the baseline model and CGP on data simulated from all three regimes. We test all models on a common set of trajectories treated up until $t = 12$hrs with policy π_A and report how risk predictions vary as a function of action policy in the training data.

Simulator

For each patient, we randomly sample outcome measurement times from a homogeneous Poisson process with with constant intensity λ over the 24 hour period. Given the measurement times, outcomes are sampled from a mixture of three GPs. The covariance function is shared between all

classes, and is defined using a Matérn 3/2 kernel (variance 0.2^2, lengthscale 8.0) and independent Gaussian noise (scale 0.1) added to each observation. Each class has a distinct mean function parameterized using a 5-dimensional, order-3 B-spline. The first class has a declining mean trajectory, the second has a trajectory that declines then stabilizes, and the third has a stable trajectory.[2] All classes are equally likely *a priori*. At each measurement time, the treatment policy π determines a probability p of treatment administration (we use only a single treatment type). The treatments increase the severity marker by a constant amount for 2 hours. If two or more actions occur within 2 hours of one another, the effects do not add up (i.e. it is as though only one treatment is active). Additional details about the simulator and policies can be found in Section 6.9.

Model

For both the baseline GP and CGP, we use a mixture of three GPs (as was used to simulate the data). We assume that the mean function coefficients, the covariance parameters, and the treatment effect size are unknown and must be learned. We emphasize that both the baseline GP and CGP have identical forms, but are trained using different objectives; the baseline marginalizes over future actions, inducing a dependence on the treatment policy in the training data, while the CGP explicitly controls for them while learning. For both the baseline model and CGP, we analytically sum over the mixture component likelihoods to obtain a closed form expression for the likelihood, which we optimize using BFGS [Nocedal and Wright, 2006]. Predictions for both models are made using the posterior predictive mean given data and interventions up until 12 hours. Additional details are deferred to Section 6.10.

Results

We find that the baseline GP's risk scores are unstable across regimes A, B, and C. The CGP is stable across regimes A and B, but unstable in regime C, where our assumptions are violated. In Table 6.1, the first row shows the average difference in risk scores (which take values in $[0, 1]$) produced by the models trained in each regime and produced by the models trained in regime A.

[2]The exact B-spline coefficients can be found in the simulation code included in the supplement.

In row 1, column B we see that the baseline GP's risk scores differ for the same person on average by around eight points ($\Delta = 0.083$). From the perspective of a decision-maker, this behavior could make the system appear less reliable. Intuitively, the risk for a given patient should not depend on the policy used to determine treatments in retrospective data. On the other hand, the CGP's scores change very little when trained on different regimes ($\Delta = 0.001$), as long as Assumptions 1, 3, and 4 are satisfied.

A cynical reader might ask: even if the risk scores are unstable, perhaps it has no consequences on the downstream decision-making task? In the second row of Table 6.1, we report Kendall's τ computed between each regime and regime A using the risk scores to rank the patient's in the test data according to severity (i.e. scores closer to 1 are more severe). In the third row, we report the AUC for both models trained in each regime on the common test set. We label a patient as "at risk" if the last marker value in the untreated trajectory is below zero, and "not at risk" otherwise. In row 2, column B we see that the CGP has a high rank correlation ($\tau = 0.998$) between the two regimes where the policies satisfy our key assumptions. The baseline GP model trained on regime B, however, has a lower rank correlation of $\tau = 0.857$ with the risk scores produced by the same model trained on regime A. Similarly, in row three, columns A and B, we see that the CGP's AUC is unchanged (AUC $= 0.872$). The baseline GP, however, is unstable and creates a risk score with poorer discrimination in regime B (AUC $= 0.832$) than in regime A (AUC $= 0.853$). Although we illustrate stability of the CGP compared to the baseline GP using two regimes, this property is not specific to the particular choice of policies used in regimes A and B; the issue persists as we generate different training data by varying the distribution over the action choices.

Finally, the results in column C highlight the importance of Assumptions 1, 3, and 4. The policy π_C *does not* satisfy these assumptions, and we see that the risk scores for the CGP are different when fit in regime C than when fit in regime A ($\Delta = 0.128$); the CGP is unstable if the assumptions do not hold. Similarly, in row 2 the CGP's rank correlation degrades ($\tau = 0.562$), and in row 3 the AUC decreases to 0.829. Note that the baseline GP continues to be unstable when fit in regime C.

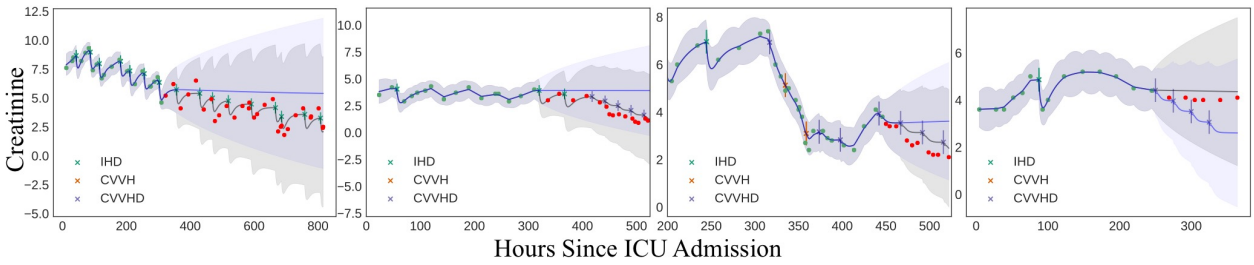


Figure 6.2: Example factual (grey) and counterfactual (blue) predictions on real ICU data using the CGP.

Conclusions

These results have important implications for the practice of building predictive models for decision support. Classical supervised learning algorithms can be unreliable due to an implicit dependence on the action policy in the training data, which is usually different from the assumed action policy at test time (e.g. what will happen if we do not treat?). Note that this issue is not resolved by training only on individuals who are not treated because selection bias creates a mismatch between our train and test distributions. From a broader perspective, supervised learning can be unreliable because it captures features of the training distribution that may change (e.g. relationships caused by the action policy). Although we have used a counterfactual model to account for and remove these relationships to achieve stability, there may be other approaches that achieve the same effect (e.g., Dyagilev and Saria 2016). Recent related work by Gong et al. [2016] on covariate shift aims to learn only the components of the source distribution that will generalize to the target distribution. As predictive models are becoming more widely used in domains like healthcare where safety is critical (e.g. Li-wei et al. 2015, Schulam and Saria 2015, Alaa et al. 2016, Wiens et al. 2016, Cheng et al. 2017), the framework we describe here is increasingly pertinent.

6.4.2 "What if?" Reasoning for Individualized Treatment Planning

To demonstrate how the CGP can be used for individualized treatment planning, we extract observational creatinine traces from the publicly available MIMIC-II database [Saeed et al., 2011]. Creatinine is a compound produced as a by-product of the chemical reaction in the body that breaks down creatine to fuel muscles. Healthy kidneys normally filter creatinine out of the body,

which can otherwise be toxic in large concentrations. During kidney failure, however, creatinine levels rise and the compound must be extracted using a medical procedure called dialysis.

We extract patients in the database who tested positive for abnormal creatinine levels, which is a sign of kidney failure. We also extract the times at which three different types of dialysis were given to each individual: intermittent hemodialysis (IHD), continuous veno-venous hemofiltration (CVVH), and continuous veno-venous hemodialysis (CVVHD). The data set includes a total of 428 individuals, with an average of 34 ($\pm$12) creatinine observations each. We shuffle the data and use 300 traces for training, 50 for validation and model selection, and 78 for testing.

Model

We parameterize the outcome model of the CGP using a mixture of GPs. We always condition on the initial creatinine measurement and model the deviation from that initial value. The mean for each class is zero (i.e. we assume there is no deviation from the initial value on average). We parameterize the covariance function using the sum of two non-stationary kernel functions. Let $\phi : t \to [1, t, t^2]^\top \in \mathbb{R}^3$ denote the quadratic polynomial basis, then the first kernel is $k_1(t_1, t_2) = \phi^\top(t_1)\Sigma\phi(t_2)$, where $\Sigma \in \mathbb{R}^{3\times3}$ is a positive-definite symmetric matrix parameterizing the kernel. The second kernel is the covariance function of the integrated Ornstein-Uhlenbeck (IOU) process (see e.g., Taylor et al. 1994), which is parameterized by two scalars α and ν and defined as

$$k_{\text{IOU}}(t_1, t_2) = \frac{\nu^2}{2\alpha^3}\left(2\alpha\min(t_1, t_2) + e^{-\alpha t_1} + e^{-\alpha t_2} - 1 - e^{-\alpha|t_1-t_2|}\right).$$

The IOU covariance corresponds to the random trajectory of a particle whose velocity drifts according to an OU process. We assume that each creatinine measurement is observed with independent Gaussian noise with scale σ. Each class in the mixture has a unique set of covariance parameters. To model the treatment effects in the outcome model, we define a short-term function and long-term response function. If an action is taken at time t_0, the outcome $\delta = t - t_0$ hours later will be additively affected by the response function $g(\delta; h_1, a, b, h_2, r) = g_s(\delta; h_1, a, b) + g_\ell(\delta; h_2, r)$, where $h_1, h_2 \in \mathbb{R}$ and $a, b, r \in \mathbb{R}^+$. The short-term and long-term response functions are defined as

$g_s(\delta; h_1, a, b) = \frac{h_1 a}{a-b}\left(e^{-b\cdot t} - e^{-a\cdot t}\right)$, and $g_\ell(\delta : h_2, r) = h_2 \cdot (1.0 - e^{-r\cdot t})$. The two response functions are included in the mean function of the GP, and each class in the mixture has a unique set of response function parameters. We assume that Assumptions 1, 3, and 4 hold, and that the event and action models have separate parameters, so can remain unspecified when estimating the outcome model. We fit the CGP outcome model using Equation 6.3, and select the number of classes in the mixture using fit on the validation data (we choose three components).

Results

Figure 6.2 demonstrates how the CGP can be used to do "what if?" reasoning for treatment planning. Each panel in the figure shows data for an individual drawn from the test set. The green points show measurements on which we condition to obtain a posterior distribution over mixture class membership and the individual's latent trajectory under each class. The red points are unobserved, future measurements. In grey, we show predictions under the *factual* sequence of actions extracted from the MIMIC-II database. Treatment times are shown using vertical bars marked with an "x" (color indicates which type of treatment was given). In blue, we show the CGP's *counterfactual* predictions under an alternative sequence of actions. The posterior predictive trajectory is shown for the MAP mixture class (mean is shown by a solid grey/blue line, 95% credible intervals are shaded).

We qualitatively discuss the CGP's counterfactual predictions, but cannot quantitatively evaluate them without prospective experimental data from the ICU. We can, however, measure fit on the factual data and compare to baselines to evaluate our modeling decisions. Our CGP's outcome model allows for heterogeneity in the covariance parameters and the response functions. We compare this choice to two alternatives. The first is a mixture of three GPs that *does not* model treatment effects. The second is a single GP that *does* model treatment effects. Over a 24-hour horizon, the CGP's mean absolute error (MAE) is 0.39 (95% CI: 0.38-0.40),[3], and for predictions between 24 and 48 hours in the future the MAE is 0.62 (95% CI: 0.60-0.64). The pairwise mean difference between the first baseline's absolute errors and the CGP's is 0.07 (0.06, 0.08) for 24

[3] 95% confidence intervals computed using the pivotal bootstrap are shown in parentheses

hours, and 0.09 (0.08, 0.10) for 24-48 hours. The mean difference between the second baseline's absolute errors and the CGP's is 0.04 (0.04, 0.05) for 24 hours and 0.03 (0.02, 0.04) for 24-48 hours. The improvements over the baselines suggest that modeling treatments and heterogeneity with a mixture of GPs for the outcome model are useful for this problem.

Figure 6.2 shows factual and counterfactual predictions made by the CGP. In the first (left-most) panel, the patient is factually administered IHD about once a day, and is responsive to the treatment (creatinine steadily improves). We query the CGP to estimate how the individual *would have* responded had the IHD treatment been stopped early. The model reasonably predicts that we would have seen no further improvement in creatinine. The second panel shows a similar case. In the third panel, an individual with erratic creatinine levels receives CVVHD for the last 100 hours and is responsive to the treatment. As before, the CGP counterfactually predicts that she would not have improved had CVVHD not been given. Interestingly, panel four shows the opposite situation: the individual did not receive treatment and did not improve for the last 100 hours, but the CGP counterfactually predicts an improvement in creatinine as in panel 3 under daily CVVHD.

6.5 Discussion

We have shown that classical supervised learning algorithms are, in general, not stable with respect to the action policy in the training data. The models they learn may therefore be unreliable, and even dangerous, decision-support tools. As a safer alternative, this paper advocates for using *stable* learning algorithms that are independent of the action policy in the training data. To design stable learning algorithms, we showed how to use potential outcomes [Neyman, 1923, 1990, Rubin, 1978] and *counterfactual learning objectives* (like the one in Equation 6.3). We introduced the Counterfactual Gaussian Process (CGP) as a decision-support tool for scenarios where outcomes are measured and actions are taken at irregular, discrete points in continuous-time. The CGP builds on previous ideas in continuous-time causal inference (e.g. Robins 1997, Arjas and Parner 2004, Lok 2008), but is unique in that it can predict the full counterfactual *trajectory* of a time-dependent outcome. We designed an adjusted maximum likelihood algorithm for learning the CGP

from *observational traces* by modeling them using a marked point process (MPP), and described three structural assumptions that are sufficient to show that the algorithm correctly recovers the CGP.

We empirically demonstrated the CGP on two decision-support tasks. First, we showed that the CGP can be used to make reliable risk predictions that are stable with respect to the action policies used in the training data. This is critical because an action policy can cause a predictive model fit using classical supervised learning to capture relationships between the features and outcome (risk) that lead to poor downstream decisions and that are difficult to diagnose. In the second set of experiments, we showed how the CGP can be used to compare counterfactuals and answer "what if?" questions, which could offer decision-makers a powerful new tool for individualized treatment planning. We demonstrated this capability by learning the effects of dialysis on creatinine trajectories using real ICU data and predicting counterfactual progressions under alternative dialysis treatment plans.

These results suggest a number of new questions and directions for future work. First, the validity of the CGP is conditioned upon a set of assumptions (this is true for all counterfactual models). In general, these assumptions are not testable. The reliability of approaches using counterfactual models therefore critically depends on the plausibility of those assumptions in light of domain knowledge. Formal procedures, such as sensitivity analyses (e.g., Robins et al. 2000, Scharfstein et al. 2014), that can identify when causal assumptions conflict with a data set will help to make these methods more easily applied in practice. In addition, there may be other sets of structural assumptions beyond those presented that allow us to learn counterfactual GPs from non-experimental data. For instance, the back door and front door criteria are two separate sets of structural assumptions discussed by Pearl [2009] in the context of estimating parameters of causal Bayesian networks from observational data.

More broadly, this work has implications for recent pushes to introduce safety, accountability, and transparency into machine learning systems. We have shown that learning algorithms sensitive to certain factors in the training data (the action policy, in this case) can make a system less reliable.

In this paper, we used the potential outcomes framework and counterfactuals to characterize and account for such factors, but there may be other ways to do this that depend on fewer or more realistic assumptions (e.g., Dyagilev and Saria 2016). Moreover, removing these nuisance factors is complementary to other system design goals such as interpretability (e.g., Ribeiro et al. 2016).

6.6 Why Continuous Time?

Counterfactual models in discrete time have been studied extensively, so why do we need new models and assumptions for modeling data in continuous time? Could we simply *discretize* the continuous time data. In this section, we show that discretization can bias predictive models for decision-making, which can lead to incorrect or harmful downstream decisions. We refer to this phenomenon as *discretization bias*.

Discretization bias is a form of confounding caused by *grouping individual observations* into equally sized bins and creating an aggregate observation by summarizing those observations (e.g. using the average). This preprocessing step treats groups of observations as exchangeable, and drops information about their specific values and measurement times. If the dropped information is used in the policy, then estimates of the *outcome model* may be biased.

6.6.1 Simulated Markov Decision Process

To study discretization bias, we how how this common preprocessing step can affect the policy learned by model-based solutions to discrete-time Markov decision problems. In a discrete-time MDP, the outcome model predicts future values of a time series given the history, which includes both previous observations and any actions or interventions applied to the system. When we estimate an outcome model to derive a policy for an MDP, we want to estimate a causal model. We show that discretization can confound this causal model.

Let Y_k be the k^{th} observation of a discrete-time time series and let U_k denote the k^{th} action,

then

$$P(Y_k \mid \mathcal{Y}_k, \mathcal{U}_k) \tag{6.4}$$

is the outcome model, where $\mathcal{Y}_k$ is all previous observations $[Y_1, \ldots, Y_{k-1}]$ and $\mathcal{U}_k$ is all previous actions $[U_1, \ldots, U_{k-1}]$. We assume that the distribution of actions U_k is determined by a *policy* π, that may depend on $\mathcal{Y}_k$, $\mathcal{U}_k$, and Y_k, which we call the *history* and denote using $\mathcal{H}_k$.

To show how discretization can confound estimates of Equation 6.4, we simulate time series data from a two-dimensional discrete-time Gaussian hidden Markov model (SG-HMM). We define a Gaussian discrete-time hidden Markov model by discretizing a stationary linear continuous-time hidden Markov model. A stationary linear continuous-time model hidden Markov model is parameterized by five matrices: F, G, Q, H, and R. The first three matrices describe the system *dynamics*, and the final two matrices describe the *observation model*. The dynamics are characterized using the Itô stochastic differential equation

$$dX(t) = FX(t)dt + GU(t) + Ld\beta(t), \tag{6.5}$$

where $U(t)$ is an input process and L is the Cholesky factorization of the positive definite covariance matrix Q and $\beta(t)$ is Brownian motion. The observation model is a simple multivariate normal distribution:

$$Y(t) \mid X(t) \sim \mathcal{N}(HX(t), R). \tag{6.6}$$

Continuous to Discrete Conversion

A Gaussian discrete-time hidden Markov model is also parameterized by five matrices: A, B, C, H, and R. As before, the first three matrices define the dynamics model and the last two define the obsrvation model. For $k \in [0, \ldots, n_k]$, we define the distributions of the discrete-time random

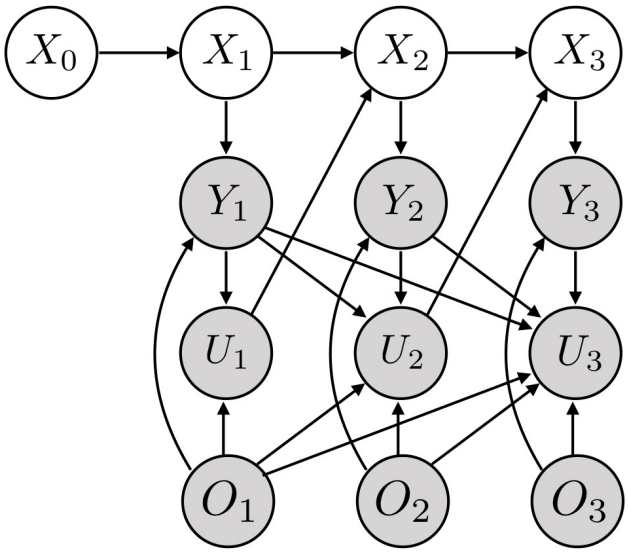


Figure 6.3: Graphical model for simulated data.

variables

$$X_k \mid X_{k-1} \sim \mathcal{N}(AX_{k-1} + BU_{k-1}, C) \tag{6.7}$$

$$Y_k \mid X_k \sim \mathcal{N}(HX_k, R). \tag{6.8}$$

To discretize the continuous-time model described above, we need to define a mapping from (F, G, Q, H, R) to (A, B, C, H, R). The mapping depends on the *timestep* of the discretization, which we will denote using δ. The timestep determines the intervals at which we observe the continuous-time process. Suppose that $X(t)$ is defined on the interval $[0, T]$ and that δ is chosen such that $n_k\delta = T$, then the discretization defines the random variables

$$X_k \triangleq X(k\delta) \tag{6.9}$$

$$Y_k \triangleq Y(k\delta) \tag{6.10}$$

for $k \in [0, \ldots, n_k]$. Using these definitions, we first calculate the conditional distribution of X_k

given X_{k-1}. This conditional distribution has mean and covariance

$$\mu = \Phi(\delta)X_{k-1} + \int_0^\delta \Phi(\delta - s)GU(k\delta - \delta + s)ds \tag{6.11}$$

$$\Sigma = \int_0^\delta \Phi(\delta - s)Q\Phi^T(\delta - s)ds, \tag{6.12}$$

where $\Phi(s) = \exp\{Fs\}$; the matrix exponential of Fs (see, e.g., Särkkä and Solin 2014). We therefore see that $A = \Phi(\delta)$ and that $C = \Sigma$ in our mapping from continuous-time to discrete-time. To define the matrix B, we make the assumption that $U(t)$ is defined on the grid $[0, \delta, 2\delta, \ldots, n_k\delta]$ using a sequence of $n_k + 1$ values $U_0, U_1, \ldots, U_k$:

$$U(t) = \sum_{i=0}^{n_k} U_i \delta_{i\delta}(t), \tag{6.13}$$

where $\delta_{i\delta}$ is the Dirac delta function centered at $i\delta$. If $U(t)$ has this form, then the integral in the expression for the conditional expected value μ of X_k given X_{k-1} is

$$\int_0^\delta \Phi(\delta - s)GU(k\delta - \delta + s)ds = \Phi(\delta)GU_{k-1}. \tag{6.14}$$

Therefore, we have $B = \Phi(\delta)G$. To complete the mapping, note that H and R do not need to be modified for the continuous to discrete conversion. In summary, we have

$$A = \Phi(\delta) \tag{6.15}$$

$$B = \Phi(\delta)G \tag{6.16}$$

$$C = \int_0^\delta \Phi(\delta - s)Q\Phi^T(\delta - s)ds \tag{6.17}$$

$$H = H \tag{6.18}$$

$$R = R. \tag{6.19}$$

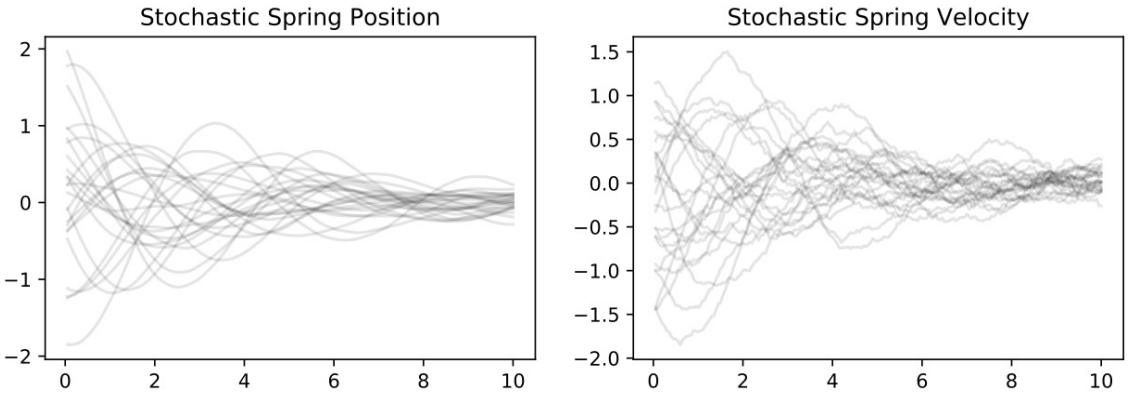


Figure 6.4: Sample trajectories from the stochastic spring model.

Stochastic Spring Model

For our experiments, we define the continuous-time model using a stochastic spring model. The dynamics of the stochastic spring depend on two parameters ν and γ, which we set to 1.0 and 0.5 respectively. The model is parameterized using

$$F = \begin{bmatrix} 0.0 & 1.0 \\ -\nu^2 & -\gamma \end{bmatrix} \tag{6.20}$$

$$G = \begin{bmatrix} 0.0 \\ 0.5 \end{bmatrix} \tag{6.21}$$

$$Q = \begin{bmatrix} 10^{-8} & 0.0 \\ 0.0 & 10^{-2} \end{bmatrix} \tag{6.22}$$

$$H = \begin{bmatrix} 1.0 & 0.0 \end{bmatrix} \tag{6.23}$$

$$R = \begin{bmatrix} 10^{-4} \end{bmatrix}. \tag{6.24}$$

To simulate from the model, we draw an initial state $X(0)$ from a two-dimensional normal distribution with zero mean and identity covariance. Figure 6.4 shows a sample of simulated trajectories from the stochastic spring model.

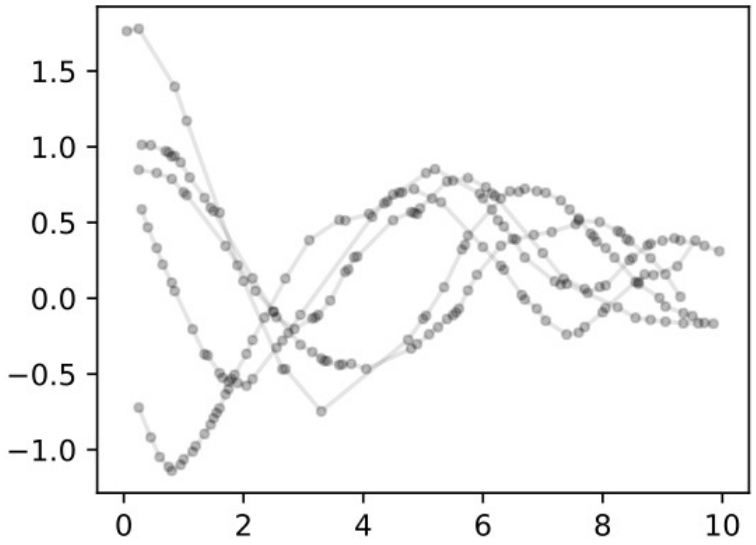


Figure 6.5: Examples of the data used to learn B.

Simulating Actions

The actions $U_k \in \{0, 1\}$ are chosen dynamically based on the history of observed measurements $[Y_1, \ldots, Y_k]$. In particular, each U_k has a Bernoulli distribution with a mean parameter that is computed using a weighted average of the previously observed measurements. Let π_k denote the expected value of U_k, then

$$\log \frac{\pi_k}{1 - \pi_k} = \beta_0 + \beta_1 \sum_{i=1}^{k} w_i Y_i. \tag{6.25}$$

To compute the weights w_i, we define a parameter $\alpha \in [0, 1]$. In a history of k measurements, the weight for the i^{th} measurement Y_i is

$$w_i \propto \alpha^{k-i}. \tag{6.26}$$

The weights are normalized to sum to one. We see that when $\alpha = 0$, the history has no effect (i.e. the log odds are 0). On the other hand, when $\alpha = 1$ all of the previous measurements are weighted equally. When $\alpha \in (0, 1)$, the more recent measurements are given more weight.

Missing Data in Histories

In our experiment, we randomly "drop" measurements Y_k with probability $p_m = 0.8$. When a measurement is dropped, we replace it with the null value $\varnothing$. To account for missing data in the average used to compute π_k, we define the variable $O_k = 1$ when Y_k is observed (i.e. it is not $\varnothing$), and $O_k = 0$ otherwise. We then modify the weight for measurement Y_i to be

$$w_i \propto O_i \alpha^{k-i}. \tag{6.27}$$

Coarsened Discrete Models

When time series data is dicretized, the observations are first grouped into a sequence of equal-sized windows and then an aggregate measurement is computed from the measurements that fall into the bin. The average (arithmetic mean) is typically used as the aggregation method.

Recall that δ is the unknown step size of the discrete-time system that generated our data. To formalize the discretization preprocessing step, we introduce the idea of *coarsening*, which is an operation on a discrete-time model that produces another discrete-time model with a timestep Δ that is larger than δ. Coarsening depends on an integer $n_c \geq 2$, which we refer to as the *factor*. Given the coarsening factor n_c, we define new states X'_k and observations Y'_k that are obtained by stacking n_c consecutive states (or observations) together to form a larger vector:

$$X'_k = \left[X^T_{n_c(k-1)+1}, \ldots, X^T_{n_c(k-1)+n_c}\right]^T \tag{6.28}$$

$$Y'_k = \left[Y^T_{n_c(k-1)+1}, \ldots, Y^T_{n_c(k-1)+n_c}\right]^T \tag{6.29}$$

If (A, B, C, H, R) are the parameters of the original discrete-time HMM, then the distribution over the coarsened states X'_k and Y'_k is also a discrete-time HMM with parameters (A', B', C', H', R') that depend only on (A, B, C, H, R). The coarsened dynamics and measurement matrices have the

following block structure:

$$[A']_{ij} = \begin{cases} A^i & \text{if } j = n_c, \\ 0 & \text{otherwise.} \end{cases} \tag{6.30}$$

$$[B']_{ij} = \begin{cases} 0 & \text{if } i < j, \\ A^{i-j}B & \text{otherwise.} \end{cases} \tag{6.31}$$

$$[C']_{ij} = \begin{cases} C & \text{if } i = j = 1, \\ A^{i-1}C(A^{i-1})^T + C & \text{if } i = j > 1, \\ [C']_{ii}(A^{j-i})^T & \text{if } i < j, \\ A^{i-j}[C']_{jj} & \text{if } i > j. \end{cases} \tag{6.32}$$

The coarsened measurement model parameters H' and R' also have block matrix structure:

$$[H']_{ij} = \begin{cases} H & \text{if } i = j, \\ 0 & \text{otherwise.} \end{cases} \tag{6.33}$$

$$[R']_{ij} = \begin{cases} R & = \text{ if } i == j, \\ 0 & = \text{ otherwise.} \end{cases} \tag{6.34}$$

Coarsening and Discretization

To discretize data, there is typically an aggregation step that summarizes a collection of observations that fall into the same bin. One of the most common aggregation operations is taking the average of all observations in a bin, and this is how we aggregate in our simulation experiment. Since averaging is a linear operation, we see that preprocessing with discretization defines a new coarsened discrete-time HMM with a new measurement model

$$H'' = [n_c^{-1}, \ldots, n_c^{-1}]\, H' \tag{6.35}$$

$$R'' = \sum_{i=1}^{n_c} n_c^{-2} R^2. \tag{6.36}$$

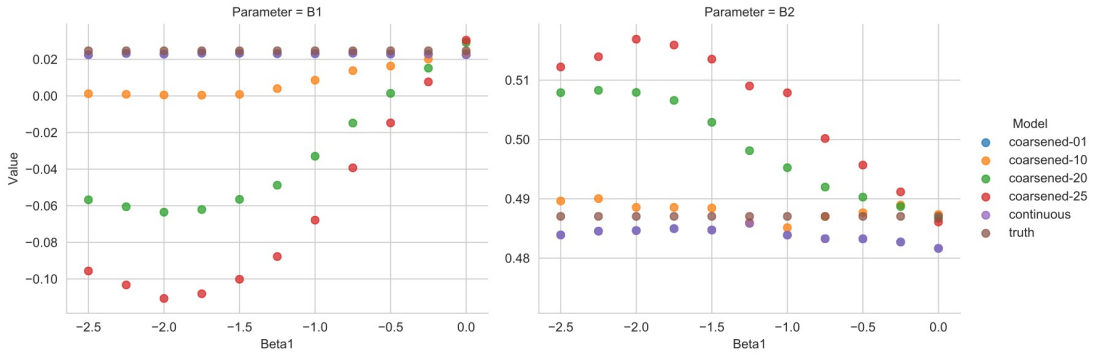


Figure 6.6: Action effect estimates (y-axis) for each model under policies with varying levels of dependence on the history (x-axis). The `coarsened-01` and `continuous` models produce the exact same estimates, and so are overlaid in the plots.

6.6.2 Demonstrating Discretization Bias

We fit five models using maximum likelihood: four discrete-time models with coarsening factors 1, 10, 20, and 25 (when $m = 1$, the coarsened model is the original SG-HMM), and one continuous-time model. Because the SG-HMM is derived by discretizing a continuous-time model, all five models depend on the same underlying parameters. Maximizing the likelihood of an SG-HMM is a non-convex optimization problem, so we simplify by assuming that all parameters are known *except* for the matrix B. Estimating B is a concave optimization problem, which simplifies estimation and helps to isolate the effects of discretization.

Figure 6.6 displays the results of the simulation experiment. We see that as $\beta_1 \to 0$, both elements of B (listed as `B1` and `B2`) are accurately estimated for all models. As β_1 becomes more negative, however, we see that the models with larger coarsening factors m are biased. In particular, note that the models with $m = 20$ and $m = 25$ learn that actions have the opposite effect (i.e. `B1` is negative instead of positive). On the other hand, the continuous-time model gives the exact same, unbiased, estimates of the parameters as the true model with coarsening factor $m = 1$.

6.7 Equivalence of MPP Outcome Model and Counterfactual Model

At a given time t, we want to make predictions about the potential outcomes that we will measure at a set of future query times $\mathbf{q} = [s_1, \ldots, s_m]$ given a specified future sequence of actions $\mathbf{a}$. This

can be written formally as

$$P(\{Y_s[\mathbf{a}] : s \in \mathbf{q}\} \mid \mathcal{H}_t) \tag{6.37}$$

Without loss of generality, we can use the chain rule to factor this joint distribution over the potential outcomes. We choose a factorization in time order; that is, a potential outcome is conditioned on all potential outcomes at earlier times. We now describe a sequence of steps that we can apply to each factor in the product.

$$P(\{Y_s[\mathbf{a}] : s \in \mathbf{q}\} \mid \mathcal{H}_t) = \prod_{i=1}^{m} P(Y_{s_i}[\mathbf{a}] \mid \{Y_s[\mathbf{a}] : s \in \mathbf{q}, s < s_i\}, \mathcal{H}_t). \tag{6.38}$$

Using Assumption 3, we can introduce random variables for marked points that have the same timing and actions as the proposed sequence of actions without changing the probability. Recall our assumption that actions can only affect future values of the outcome, so we only need to introduce marked points for actions taken at earlier times. Formally, we introduce the set of marked points for the potential outcome at each time s_i

$$\mathbf{A}_i = \{(t', \varnothing, a, 0, 1) : (t', a) \in \mathbf{a}, t' < s_i\}. \tag{6.39}$$

We can then write

$$P(Y_{s_i}[\mathbf{a}] \mid \{Y_s[\mathbf{a}] : s \in \mathbf{q}, s < s_i\}, \mathcal{H}_t) = P(Y_{s_i}[\mathbf{a}] \mid \mathbf{A}_i, \{Y_s[\mathbf{a}] : s \in \mathbf{q}, s < s_i\}, \mathcal{H}_t). \tag{6.40}$$

To show that $P(Y[a] \mid A = a, X = x) = P(Y[a] \mid X = x)$ in Section 6.3, we use Assumption 2 to remove the random variable A from the conditioning information without changing the probability statement. We reverse that logic here by adding $\mathbf{A}_i$.

Now, under Assumption 1, after conditioning on $\mathbf{A}_i$, we can replace the potential outcome $Y_{s_i}[\mathbf{a}]$

with Y_{s_i}. We therefore have

$$P(Y_{s_i}[\mathbf{a}] \mid \mathbf{A}_i, \{Y_s[\mathbf{a}] : s \in \mathbf{q}, s < s_i\}, \mathcal{H}_t) = P(Y_{s_i} \mid \mathbf{A}_i, \{Y_s[\mathbf{a}] : s \in \mathbf{q}, s < s_i\}, \mathcal{H}_t). \tag{6.41}$$

Similarly, because the set of proposed actions affecting the outcome at time s_i contain all actions that affect the outcome at earlier times $s < s_i$, we can invoke Assumption 1 again and replace all potential outcomes at earlier times with the value of the observed process at that time.

$$P(Y_{s_i} \mid \mathbf{A}_i, \{Y_s[\mathbf{a}] : s \in \mathbf{q}, s < s_i\}, \mathcal{H}_t) = P(Y_{s_i} \mid \mathbf{A}_i, \{Y_s : s \in \mathbf{q}, s < s_i\}, \mathcal{H}_t).$$

Next, Assumption 4 posits that the outcome model $p^*(y \mid t', z_y = 1)$ is the density of $P(Y_{t'} \mid \mathcal{H}_t)$, which implies that the mark $(t', y, \varnothing, 1, 0)$ is equivalent to the event $(Y_{t'} \in \mathrm{d}y)$. Therefore, for each s_i define

$$\mathbf{O}_i = \{(s, Y_s, \varnothing, 1, 0) : s \in \mathbf{q}, s < s_i\}. \tag{6.42}$$

Using this definition, we can write

$$P(Y_{s_i} \mid \mathbf{A}_i, \{Y_s : s \in \mathbf{q}, s < s_i\}, \mathcal{H}_t) = (Y_{s_i} \mid \mathbf{A}_i, \mathbf{O}_i, \mathcal{H}_t).$$

The set of information $(\mathbf{A}_i, \mathbf{O}_i, \mathcal{H}_t)$ is a valid history of the marked point process $\mathcal{H}_{s_i}^-$ up to but not including time s_i. We can therefore replace all information after the conditioning bar in each factor of Equation 6.38 with $\mathcal{H}_{s_i^-}$.

$$P(Y_{s_i} \mid \mathbf{A}_i, \mathbf{O}_i, \mathcal{H}_t) = P(Y_{s_i} \mid \mathcal{H}_{s_i}^-). \tag{6.43}$$

Finally, by applying Assumption 4 again, we have

$$P(Y_{s_i} \in \mathrm{d}y \mid \mathcal{H}_{s_i}^-) = p^*(y \mid s_i, z_y = 1)\,\mathrm{d}y. \tag{6.44}$$

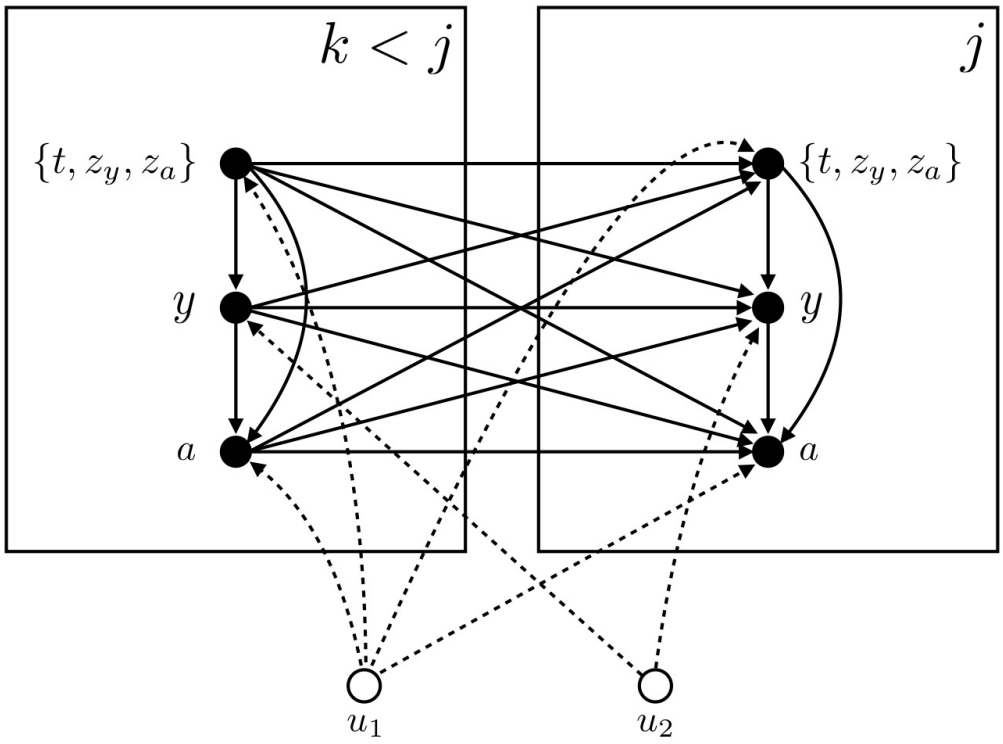


Figure 6.7: The causal Bayesian network for the counterfactual GP.

The potential outcome query can therefore be answered using the outcome model, which we can estimate from data.

6.8 Causal Bayesian Network

We can also characterize our key assumptions using causal Bayesian networks [Pearl, 2009]. Let $\{(t_j, z_{y,j}, z_{a,j}, y_j, a_j)\}_{j\geq 1}$ be a countable sequence of tuples of variables (a marked point process can be characterized as a countable sequence of points and marks). Recall that t_j is an event time, $z_{y,j}$ is a binary random variable indicating whether an outcome is measured, $z_{a,j}$ is a binary random variable indicating whether an action is taken, $y_j \in \mathcal{R} \cup \{\varnothing\}$ is an outcome measurement, and $a_j \in \mathcal{C} \cup \{\varnothing\}$ is an action (the last two variables are $\varnothing$ when the respective indicator is 0).

We define the directed acyclic graph $\mathcal{G}$ with nodes $\mathcal{V} \triangleq \cup_{j\geq 1}\{t_j, z_{y,j}, z_{a,j}, y_j, a_j\}$ and edge set $\mathcal{E}$ to be the causal Bayesian network for the counterfactual GP. For any variables $v_1 \in \{t_j, z_{y,j}, z_{a,j}, y_j, a_j\}$ and $v_2 \in \{t_k, z_{y,k}, z_{a,k}, y_k, a_k\}$, the edge $(v_1 \to v_k) \in \mathcal{E}$ if $j < k$ or if $j = k$ and v_1 is a parent of v_2 in the right-most plate of Figure 6.7. We allow the variables $\{(t_j, z_{y,j}, z_{a,j}, a_j)\}_{j=1}^{\infty}$ to depend on

a common unobserved parent u_1, and the outcomes $\{y_j\}_{j=1}^{\infty}$ to depend on a common unobserved parent u_2. The DAG in Figure 6.7 sketches the causal Bayesian network. For any index j, we show the edges present between all variables at times $k < j$.

We now formulate our causal query, and show that it is identified using observational traces sampled from the distribution implied by the causal Bayesian network. For any time $t \in [0, \tau]$, our goal is to predict the values of future outcomes under a hypothetical sequence of future actions given the history up until time t. Define $\mathcal{H}_t = \cup_{j:t_j<t}\{t_j, z_{y,j}, z_{a,j}, y_j, a_j\}$ to be the sequence of n actions taken and outcomes measured prior to time t, and define $\mathcal{F}_t$ to be a sequence of m tuples corresponding to future actions and measurements. The variables in $\mathcal{H}_t \cup \mathcal{F}_t$ are connected using the edge set definition described above. Let $\boldsymbol{t}$ denote the m future time points, $\boldsymbol{z}_y$ the future measurement indicators, $\boldsymbol{z}_a$ the future action indicators, $\boldsymbol{y}$ the future outcomes, and $\boldsymbol{a}$ the future actions. Our goal is to show that the following query is identified:

$$p(\boldsymbol{y} \mid \mathrm{do}(\boldsymbol{t}, \boldsymbol{z}_y, \boldsymbol{z}_a, \boldsymbol{a}), \mathcal{H}_t) = \prod_{j=1}^{m} p(y_j \mid \bar{\boldsymbol{y}}_{:j}, \mathrm{do}(\boldsymbol{t}, \boldsymbol{z}_y, \boldsymbol{z}_a, \boldsymbol{a}), \mathcal{H}_t), \tag{6.45}$$

where $\bar{\boldsymbol{y}}_{:j}$ denotes the vector of future outcomes before the j^{th}. We will also use $\bar{\boldsymbol{y}}_{j:}$ to denote all outcomes measured after the j^{th} (this notation will be used for the other variables as well). First, consider any factor in the expression above. We define the future and past intervened-on variables at time t_j as

$$\boldsymbol{f}_j \triangleq \{a_j, \bar{\boldsymbol{t}}_{j:}, \bar{\boldsymbol{z}}_{y,j:}, \bar{\boldsymbol{z}}_{a,j:}, \bar{\boldsymbol{a}}_{j:}\} \tag{6.46}$$

$$\boldsymbol{p}_j \triangleq \{\bar{\boldsymbol{t}}_{:j}, \bar{\boldsymbol{z}}_{y,:j}, \bar{\boldsymbol{z}}_{a,:j}, \bar{\boldsymbol{a}}_{:j}, t_j, z_{y,j}, z_{a,j}\}. \tag{6.47}$$

Using these shorthand definitions, we first prove the following equivalence

$$p(y_j \mid \bar{\boldsymbol{y}}_{:j}, \mathrm{do}(\boldsymbol{p}_j), \mathrm{do}(\boldsymbol{f}_j), \mathcal{H}_t) = p(y_j \mid \bar{\boldsymbol{y}}_{:j}, \mathrm{do}(\boldsymbol{p}_j), \mathcal{H}_t). \tag{6.48}$$

Intuitively, we are showing that actions taken after y_j is measured do not affect its value. To justify the equality, we use "Rule 3" from Pearl's do-calculus (see Chapter 3 in Pearl 2009). We must

show that y_j is d-separated from $\boldsymbol{f}_j$ in the mutilated DAG where all incoming edges to nodes in $\boldsymbol{p}_j$ and $\boldsymbol{f}_j$ have been removed. To show d-separation, let $v \in \boldsymbol{f}_j \setminus \{a_j\}$ be some future intervened-on variable at time step $k > j$. Since all incoming edges have been removed, all paths starting at v must be outgoing. Outgoing edges for v in the original DAG either point to an outcome y_ℓ for $\ell \geq k$ or some other intervened-on variable $v' \in \boldsymbol{f}_j \setminus \{a_j, v\}$. The latter are removed in the mutilated graph, so the only edges outgoing from v must point to an outcome y_ℓ for $\ell \geq k$. This implies that all paths starting at v must begin with an edge $v \to y_\ell$ for some $\ell \geq k$. Because y_ℓ is unobserved, the only unblocked paths must then follow an outgoing edge (otherwise it would be a collider). All outgoing edges from variables y_ℓ for $\ell \geq k$ can only point to outcomes $y_{\ell'}$ for $\ell' > \ell$, which in turn must point to $y_{\ell''}$ for $\ell'' > \ell'$, and so on. Therefore, any path starting from v must pass through outcomes y at strictly increasing times. Eventually, we will reach the final outcome, where there are no outgoing edges, ending the path. We can conclude that no paths starting at v can reach y_j. A similar argument shows that no path starting from a_j can reach y_j.

Next, we use "Rule 2" from the do-calculus to prove that

$$p(y_j \mid \bar{\boldsymbol{y}}_{:j}, \mathrm{do}(\boldsymbol{p}_j), \mathcal{H}_t) = p(y_j \mid \bar{\boldsymbol{y}}_{:j}, \boldsymbol{p}_j, \mathcal{H}_t). \tag{6.49}$$

This requires showing that y_j is d-separated from $\boldsymbol{p}_j$ in the mutilated graph where all outgoing edges from $v \in \boldsymbol{p}_j$ have been removed. For any $v \in \boldsymbol{p}_j$, there are two types of incoming edges. The first are edges originating from observed direct parents of v, and the second is the edge originating from the unobserved variable u_1. Any path from v to y_j must start with one of these edge types, and therefore all that start with an edge to an observed parent of v will be blocked, and any unblocked path must start by going through u_1. Now, u_1 has no parents and any path must then have a second edge from u_1 to one of its children, which are all times t_k, indicators $z_{y,k}$ or $z_{a,k}$, and actions a_k. We will analyze these possibilities using two cases. First, the second edge could go from u_1 to a time t_k where $k \leq j$, indicator $z_{y,k}$ or $z_{a,k}$ where $k \leq j$, or to an action a_k where $k < j$. The only possible next step is to go through an incoming edge where the origin is not u_1; all such edges will be blocked, and so cannot reach y_j. In the second case, an edge could go from u_1 to a time

or indicator at step $k > j$, or an action at step $k \geq j$. These variables are unobserved, and so the only valid next step is to follow an outgoing edge. Subsequent steps must all also follow outgoing edges by the same logic, and so the path can never return to y_j. We therefore can conclude that there are no paths from $v \in \boldsymbol{p}_j$ to y_j in the mutilated graph, so the equality holds. Together, the two inequalities show

$$p(\boldsymbol{y} \mid \text{do}(\boldsymbol{t}, \boldsymbol{z}_y, \boldsymbol{z}_a, \boldsymbol{a}), \mathcal{H}_t) = \prod_{j=1}^{m} p(y_j \mid \bar{\boldsymbol{y}}_{:j}, \boldsymbol{p}_j, \mathcal{H}_t). \tag{6.50}$$

This shows that the structural dependencies encoded in the graph shown in Figure 6.7 can be used in place of Assumption 3. In addition, we no longer need Assumption 1 (consistency), which highlights an interesting difference between the potential outcomes and causal Bayesian network frameworks. In Pearl's causal DAGs, consistency is in fact a theorem derived from the axioms of the framework, whereas it is assumed in the potential outcomes framework. This is shown in Corollary 7.3.2 in Pearl [2009], which follows from the Composition axiom and the definition of a "null" intervention. Intuitively, the fact that consistency is a theorem in Pearl's framework reflects the assumption that the parent-child relationships in the DAG are sufficiently stable, autonomous, or "local" [Pearl, 2009]. See Section 7.2.4 in Pearl [2009] for further information. Finally, Assumption 4 remains unchanged and simply allows us to treat measured outcomes y_j as unbiased samples of the process Y_{t_j}.

6.9 Simulation and Policy Details

For each patient, we randomly sample outcome measurement times from a homogeneous Poisson process with with constant intensity λ over the 24 hour period. Given the measurement times, outcomes are sampled from a mixture of three GPs. The covariance function is shared between all classes, and is defined using a Matérn 3/2 kernel (variance 0.2^2, lengthscale 8.0) and independent Gaussian noise (scale 0.1) added to each observation. Each class has a distinct mean function parameterized using a 5-dimensional, order-3 B-spline. The first class has a declining mean trajectory,

the second has a trajectory that declines then stabilizes, and the third has a stable trajectory.[4] All classes are equally likely *a priori*. At each measurement time, the treatment policy π determines a probability p of treatment administration (we use only a single treatment type). The treatments increase the severity marker by a constant amount for 2 hours. If two or more actions occur within 2 hours of one another, the effects do not add up (i.e. it is as though only one treatment is active). Additional details about the simulator and policies can be found in the supplement.

Policies π_A and π_B determine a probability of treatment at each outcome measurement time. They each use the average of the observed outcomes over the previous two hours, which we denote using $\hat{y}_{(t-2):t}$, as a feature, which is then multiplied by a weight $w_A = -0.5$ ($w_B = 0.5$ for regime B) and passed through the inverse logit to determine a probabilty. The policy π_C for regime C depends on the patient's latent class. The probability of treatment at any time t is $p = \alpha_z \sigma(w_A \cdot \hat{y}_{(t-2):t})$, where $\alpha_z \in (0, 1)$ is a weight that depends on the latent class z. We set $\alpha_1 = 0.2$, $\alpha_2 = 0.9$, and $\alpha_3 = 0.5$.

6.10 Mixture Estimation Details

For both the simulated and real data experiments, we analytically sum over the component-specific densities to obtain an explicit mixture density involving no latent variables. We then estimate the parameters using maximum likelihood. The likelihood surface is highly non-convex. To account for this, we used different parameter initialization strategies for the simulated and real data.

On the simulated data experiments, the mixture components for both the CGP and baseline GP are primarily distinguished by the mean functions. We initialize the mean parameters for both the baseline GP and CGP by first fitting a linear mixed model with B-spline bases using the EM algorithm, computing MAP estimates of trace-specific coefficients, clustering the coefficients, and initializing with the cluster centers.

On the real data, traces have similar mean behavior (trajectories drift around the initial creatinine value), but differed by length and amplitude of variations from the mean. We therefore

[4] The exact B-spline coefficients can be found in the simulation code included in the supplement.

centered each trace around its initial creatinine measurement (which we condition on), and use a mean function that includes only the short-term and long-term response functions. For each mixture, the response function parameters are initialized randomly: parameters a, b, and r are initialized using a LogNormal(mean $= 0.0, \text{std} = 0.1$); heights h_1 and h_2 are initialized using a Normal(mean $= 0.0, \text{std} = 0.1$). For each mixture, Σ (L300) is initialized to the identity matrix; α and ν are drawn from a LogNormal(mean $= 0.0, \text{std} = 0.1$).

Chapter 7

Conclusion

As electronic health record systems become more widespread, there is increasing pressure on providers to use that data in ways that improve patient outcomes, drive down costs, or both. It is still an open question how to best accomplish these goals. In this thesis, we studied disease trajectory subtyping and prediction. Subtyping has the potential to further our understanding of the underlying biological mechanisms that drive a disease, which can guide clinical decisions and lead to new therapies [Saria and Goldenberg, 2015]. Accurate predictive models of disease trajectories can improve clinical decision-making [Feinstein, 1983, Spiegelhalter, 1986] and can also lead to new clinical trial enrichment strategies [Simon and Maitournam, 2004, Temple, 2010, Freidlin and Korn, 2014]. We showed that there are several important types of bias that we must account for in order to successfully tackle subtyping and prediction using EHR data, and we proposed novel methods for doing so.

There has been a tremendous amount of growth in the machine learning for healthcare literature over the past several years. How do the ideas in this thesis tie into recent trends? One trend is an intense focus on applying deep neural networks to healthcare data (e.g. Lipton et al. 2015, Rajkomar et al. 2018, Tomašev et al. 2019). Although we did not use deep neural networks in our experiments, many of the findings are still relevant. First, none of these methods consider policy shift and so are vulnerable to the failures discussed in Chapter 6. Second, unobserved heterogeneity is not a problem that can be solved by richer models, so neural networks still need to account for

this issue. Recurrent neural networks (RNNs) make predictions using the full history of a time series, and so they could, in principle, learn to output the posterior predictive distribution of the hierarchical latent variable model that we propose in Chapters 4 and 5. It is not clear, however, whether RNNs will actually replicate this behavior in practice (the inductive bias may not be strong enough). Moreover, to apply RNNs to sparse and irregularly observed disease trajectory data, we must first discretize the data (recent work by Chen et al. 2018 relaxes this constraint, but this approach is still nascent). As we showed in Chapter 6, discretizing by binning a continous time series can introduce confounding and make the model sensitive to policy shift (and therefore make it less reliable). Moreover, state-of-the-art techniques for handling missing data in clinical trajectory data induces a dependence on measurement policies [Lipton et al., 2016], which, like policy shift, can make the model less reliable. Causal inference and missing data are closely related problems (see e.g. Tsiatis 2007), so the ideas from Chapter 6 may help to remove the dependence on measurement policy.

Another recent trend in the machine learning for healthcare literature is a growing interest in using reinforcement learning to automatically learn optimal treatment policies from EHR data. For example, Prasad et al. [2017] learn a policy for weaning off of mechanical ventilators, and Komorowski et al. [2018] learn a policy for giving fluids to septic patients. Surprisingly, few of these studies address unobserved confounding and the potential effect it can have on the policies that we learn. Moreover, practioners often discretize time series data prior to applying reinforcement learning algorithms, which can introduce confounding bias (Chapter 6). Although confounding was recently mentioned in a short comment by Gottesman et al. [2019], the issue has still not been fully addressed by the RL community. The work in this thesis shows that there are still a number of complexities that make unsupervised and supervised machine learning on EHR data difficult. In our opinion, the machine learning for healthcare community should further map out and understand these complexities before solving the more challenging problem of using reinforcement learning to automatically treat patients. Many of the findings in the unsupervised and supervised settings will likely be applicable to reinforcement learning, and will help to build confidence that we are

developing safe and reliable technologies.

Finally, the broader machine learning community has recently started to investigate the reliability of predictive models. Much of the work in this area has focused on model reliability in response to adversarial attacks (e.g. Goodfellow et al. 2015). In this thesis, we instead focus on reliability under shifts in the distribution of the data. We proposed using causal models as a means to *proactively* account for shifts before seeing any data from the test distribution. Others have generalized and improved on these ideas (e.g. Subbaswamy and Saria 2018, Subbaswamy et al. 2019, Rothenhäusler et al. 2018). As an alternative, there has been an independent stream of work that uses samples from multiple distributions to learn robust causal models (e.g. Gong et al. 2016, Parascandolo et al. 2018) with fewer upfront assumptions. This alternative approach may be valuable in healthcare as we begin to aggregate datasets across EHRs from multiple institutions. A promising line of future research is to blend the essential ideas from these two approaches. A combined strategy would benefit from the ability to inject domain knowledge through causal graphs, and from the flexibility of data-driven approaches that adapt to shifts observed in our data.

Looking beyond recent trends in the machine learning community, there are increasing efforts within hospitals to deploy EHR-based decision support tools. For example, data from the EHR can be used to generate alerts and warn providers that a patient is at risk of an adverse event. The majority of work studying these systems, however, uses modified versions of static clinical criteria to generate alerts and have not demonstrated any clinical benefit (e.g. Al-Jaghbeer et al. 2018, Downing et al. 2019). Although predictive models are recognized as a powerful alternative for generating alerts that may lead to better outcomes, there are no studies to date that have conclusively demonstrated improvements. Judging by the literature, supervised learning in healthcare is far from being a solved problem. There is still a considerable amount of ground to cover in order to understand how to build machine learning systems that meaningfully augment provider decisions and lead to improved patient outcomes. In our opinion, this is the most exciting future direction for the machine learning for healthcare community. By actively engaging with stakeholders across healthcare institutions (e.g. medical, quality improvement, and financial teams) and deploying real

systems to solve their most pressing problems, our community has an opportunity to see where some of the fundamental ideas in the machine learning literature fall short. Just as internet search and advertising has driven innovation in machine learning over the past several decades, we believe that pushing the boundaries of applied ML in high-impact areas such as healthcare will spur the next generation of foundational ideas in our field.

References

G. Adomavicius and A. Tuzhilin. Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *Knowledge and Data Engineering, IEEE Transactions on*, 17(6):734–749, 2005.

M. Al-Jaghbeer, D. Dealmeida, A. Bilderback, R. Ambrosino, and J.A. Kellum. Clinical decision support for in-hospital aki. *Journal of the American Society of Nephrology*, 29(2):654–660, 2018.

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

Y. Allanore, R. Simms, O. Distler, M. Trojanowska, J. Pope, C.P. Denton, and J. Varga. Systemic sclerosis. *Nature Reviews Disease Primers*, 2015.

Gary P Anderson. Endotyping asthma: new insights into key pathogenic mechanisms in a complex, heterogeneous disease. *The Lancet*, 372(9643):1107–1119, 2008.

G. Andrew and J. Gao. Scalable training of l1-regularized log-linear models. In *International Conference on Machine Learning (ICML)*, 2007.

E. Arjas and J. Parner. Causal reasoning from longitudinal data. *Scandinavian Journal of Statistics*, 31(2):171–187, 2004.

S Barr, A Zonana-Nacach, L Magder, and M Petri. Patterns of disease activity in systemic lupus

erythematosus. *Arthritis & Rheumatism: Official Journal of the American College of Rheumatology*, 42(12):2682–2688, 1999a.

Susan G. Barr, Abraham Zonana-Nacach, Laurence S. Magder, and Michelle Petri. Patterns of disease activity in systemic lupus erythematosus. *Arthritis and Rheumatism*, 42(12):2682–2688, 1999b.

Lorenzo Beretta, Monica Caronni, Massimo Raimondi, Alessandra Ponti, Tiziana Viscuso, Laura Origgi, and Raffaella Scorza. Oral cyclophosphamide improves pulmonary function in scleroderma patients with fibrosing alveolitis: experience in one centre. *Clinical rheumatology*, 26(2):168–172, 2007.

Jamie L Bigelow and David B Dunson. Bayesian semiparametric joint models for functional predictors. *Journal of the American Statistical Association*, 2012.

David Blumenthal. Stimulating the adoption of health information technology. *New England Journal of Medicine*, 360(15):1477–1479, 2009.

L. Bottou, J. Peters, J.Q. Candela, D.X. Charles, M. Chickering, E. Portugaly, D. Ray, P.Y. Simard, and E. Snelson. Counterfactual reasoning and learning systems: the example of computational advertising. *Journal of Machine Learning Research (JMLR)*, 14(1):3207–3260, 2013.

David R Brillinger. *Time series: data analysis and theory*, volume 36. SIAM, 2001.

K.H. Brodersen, F. Gallusser, J. Koehler, N. Remy, and S.L. Scott. Inferring causal impact using bayesian structural time-series models. *The Annals of Applied Statistics*, 9(1):247–274, 2015.

Anthony E Camilli, Benjamin Burrows, Ronald J Knudson, Sarah K Lyle, and Michael D Lebowitz. Longitudinal changes in forced expiratory volume in one second in adults. effects of smoking and smoking cessation. *The American review of respiratory disease*, 135(4):794–799, 1987.

R. Caruana, Y. Lou, J. Gehrke, P. Koch, M. Sturm, and N. Elhadad. Intelligible models for healthcare: Predicting pneumonia risk and hospital 30-day readmission. In *International Conference on Knowledge Discovery and Data Mining (KDD)*, pages 1721–1730. ACM, 2015.

Carvalho et al. High-dimensional sparse factor modeling: applications in gene expression genomics. *Journal of the American Statistical Association*, 2012.

P.J. Castaldi et al. Cluster analysis in the copdgene study identifies subtypes of smokers with distinct patterns of airway disease and emphysema. *Thorax*, 2014.

PE Castro, WH Lawton, and EA Sylvestre. Principal modes of variation for processes with continuous sample curves. *Technometrics*, 28(4):329–337, 1986.

H. Y. Chang, J. M. Clark, and J. P. Weiner. Morbidity trajectories as predictors of utilization: multi-year disease patterns in taiwan's national health insurance program. *Medical care*, 49(10): 918–923, Oct 2011.

D. P. Chen, S. C. Weber, P. S. Constantinou, T. A. Ferris, H. J. Lowe, and A. J. Butte. Clinical arrays of laboratory measures, or "clinarrays", built from an electronic health record enable disease subtyping by severity. *AMIA Annual Symposium Proceedings Archive*, pages 115–119, 2007.

Minhua Chen, Aimee Zaas, Christopher Woods, Geoffrey S. Ginsburg, Joseph Lucas, David Dunson, and Lawrence Carin. Predicting viral infection from high-dimensional biomarker trajectories. *Journal of the American Statistical Association*, 106(496), 2011.

T.Q. Chen, Y. Rubanova, J. Bettencourt, and D.K. Duvenaud. Neural ordinary differential equations. In *Advances in Neural Information Processing Systems (NIPS)*, pages 6571–6583, 2018.

L.F. Cheng, G. Darnell, C. Chivers, M.E. Draugelis, K. Li, and B.E. Engelhardt. Sparse multi-output Gaussian processes for medical time series prediction. *arXiv preprint arXiv:1703.09112*, 2017.

J. Craig. Complex diseases: Research and applications. *Nature Education*, 1(1):184, 2008.

J. Cunningham, Z. Ghahramani, and C.E. Rasmussen. Gaussian processes for time-marked time-series data. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, pages 255–263, 2012.

D.J. Daley and D. Vere-Jones. *An Introduction to the Theory of Point Processes*. Springer Science & Business Media, 2007.

A. C. Damianou, M. K Titsias, and N. D. Lawrence. Variational inference for latent variables and uncertain inputs in gaussian processes. *JMLR*, 2, 2015.

G. W. De Keulenaer and D. L. Brutsaert. The heart failure spectrum: time for a phenotype-oriented approach. *Circulation*, 119(24):3044–3046, Jun 23 2009.

S. Doroudi, P.S. Thomas, and E. Brunskill. Importance sampling for fair policy selection. In *Uncertainty in Artificial Intelligence (UAI)*, 2017.

F. Doshi-Velez, Y. Ge, and I. Kohane. Comorbidity clusters in autism spectrum disorders: an electronic health record time-series analysis. *Pediatrics*, 133(1):e54–63, Jan 2014.

N.L. Downing, J. Rolnick, S.F. Poole, E. Hall, A.J. Wessels, P. Heidenreich, and L. Shieh. Electronic health record-based clinical decision support alert for severe sepsis: a randomised evaluation. *BMJ Quality & Safety*, pages bmjqs–2018, 2019.

M. Dudík, J. Langford, and L. Li. Doubly robust policy evaluation and learning. In *International Conference on Machine Learning (ICML)*, 2011.

T. Duong, B. Goud, and K. Schauer. Closed-form density-based framework for automatic detection of cellular morphology changes. *Proceedings of the National Academy of Sciences*, 109(22):8382–8387, 2012.

R. Dürichen, M.A.F. Pimentel, L. Clifton, A. Schweikard, and D.A. Clifton. Multitask gaussian processes for multivariate physiological time-series analysis. *Biomedical Engineering, IEEE Transactions on*, 62(1):314–322, 2015.

K. Dyagilev and S. Saria. Learning (predictive) risk scores in the presence of censoring due to interventions. *Machine Learning*, 102(3):323–348, 2016.

P.H.C Eilers and B.D. Marx. Flexible smoothing with b-splines and penalties. *Statistical Science*, pages 89–102, 1996.

A.R. Feinstein. An additional basic science for clinical medicine: I. the constraining fundamental paradigms. *Annals of Internal Medicine*, 99(3):393–397, 1983.

Emily B Fox, Erik B Sudderth, Michael I Jordan, and Alan S Willsky. Joint modeling of multiple related time series via the beta process. *arXiv preprint arXiv:1111.4226*, 2011.

B. Freidlin and E.L. Korn. Biomarker enrichment strategies: matching trial design to biomarker credentials. *Nature reviews Clinical oncology*, 11(2):81, 2014.

J. Friedman, T. Hastie, and R. Tibshirani. *The Elements of Statistical Learning*. Springer, 2001.

A. Gelman and J. Hill. *Data analysis using regression and multilevel/hierarchical models*. Cambridge University Press, 2006.

A. Gelman, J.B. Carlin, H.S. Stern, and D.B. Rubin. *Bayesian Data Analysis*. Taylor & Francis, 2014.

M. Gong, K. Zhang, T. Liu, D. Tao, C. Glymour, and B. Schölkopf. Domain adaptation with conditional transferable components. In *International Conference on Machine Learning (ICML)*, pages 2839–2848, 2016.

I.J. Goodfellow, J. Shlens, and C. Szegedy. Explaining and harnessing adversarial examples. *International Conference on Learning Representations (ICLR)*, 2015.

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

Adi V Gundlapalli, Brett R South, Shobha Phansalkar, Anita Y Kinney, Shuying Shen, Sylvain Delisle, Trish Perl, and Matthew H Samore. Application of natural language processing to va electronic health records to identify phenotypic characteristics for clinical and research purposes. *Summit on translational bioinformatics*, 2008:36, 2008.

P. Haldar, I. Pavord, D. Shaw, M. Berry, M. Thomas, C. Brightling, A. Wardlaw, and R. Green. Cluster analysis and clinical asthma phenotypes. *American journal of respiratory and critical care medicine*, 178(3):218–224, 2008.

F.E. Harrell. *Regression modeling strategies: with applications to linear models, logistic and ordinal regression, and survival analysis*. Springer, 2015.

M.R. Hassan and B. Nath. Stock market forecasting using hidden markov model: a new approach. In *Intelligent Systems Design and Applications, 2005. ISDA'05. Proceedings. 5th International Conference on*, pages 192–196. IEEE, 2005.

A.G. Hawkes. Spectra of some self-exciting and mutually exciting point processes. *Biometrika*, pages 83–90, 1971.

J. Hensman, N. Fusi, and N.D. Lawrence. Gaussian processes for big data. *arXiv:1309.6835*, 2013.

Joyce C Ho, Joydeep Ghosh, and Jimeng Sun. Marble: high-throughput phenotyping from electronic health records via sparse nonnegative tensor factorization. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 115–124. ACM, 2014.

M.D. Hoffman, D.M. Blei, C. Wang, and J. Paisley. Stochastic variational inference. *JMLR*, 14(1): 1303–1347, 2013.

Yujin Hoshida, Jean-Philippe Brunet, Pablo Tamayo, Todd R. Golub, and Jill P. Mesirov. Subclass mapping: identifying common subtypes in independent disease data sets. *PloS one*, 2(11):e1195, 2007.

C.H. Jackson, L.D. Sharples, S.G. Thompson, S.W. Duffy, and E. Couto. Multistate markov models for disease progression with classification error. *Journal of the Royal Statistical Society: Series D (The Statistician)*, 52(2):193–209, 2003.

G.M. James and C.A. Sugar. Clustering for sparsely sampled functional data. *Journal of the American Statistical Association*, 98(462):397–408, 2003.

G.M. James, T.J. Hastie, and C.A. Sugar. Principal component models for sparse functional data. *Biometrika*, 87(3):587–602, 2000.

N. Jiang and L. Li. Doubly robust off-policy value evaluation for reinforcement learning. In *International Conference on Machine Learning (ICML)*, pages 652–661, 2016.

F.D. Johansson, U. Shalit, and D. Sontag. Learning representations for counterfactual inference. In *International Conference on Machine Learning (ICML)*, 2016.

H.F. Kaiser. The varimax criterion for analytic rotation in factor analysis. *Psychometrika*, 23(3): 187–200, 1958.

E. Keogh et al. Locally adaptive dimensionality reduction for indexing large time series databases. *ACM SIGMOD Record*, 30(2):151–162, 2001.

D. Khanna et al. Clinical course of lung physiology in patients with scleroderma and interstitial lung disease: analysis of the scleroderma lung study placebo group. *Arthritis & Rheumatism*, 63 (10):3078–3085, 2011.

A. N. Kho, J. A. Pacheco, P. L. Peissig, L. Rasmussen, K. M. Newton, N. Weston, P. K. Crane, J. Pathak, C. G. Chute, S. J. Bielinski, I. J. Kullo, R. Li, T. A. Manolio, R. L. Chisholm, and J. C. Denny. Electronic medical records for genetic research: results of the emerge consortium. *Science translational medicine*, 3(79):79re1, Apr 20 2011.

K.P Kleinman and J.G. Ibrahim. A semiparametric bayesian approach to the random effects model. *Biometrics*, pages 921–938, 1998.

Isaac S. Kohane. Using electronic health records to drive discovery in disease genomics. *Nature Reviews Genetics*, 12(6):417–428, 2011.

M. Komorowski, L.A. Celi, O. Badawi, A.C. Gordon, and A.A. Faisal. The artificial intelligence clinician learns optimal treatment strategies for sepsis in intensive care. *Nature Medicine*, 24(11): 1716, 2018.

J.M. Lange et al. A joint model for multistate disease processes and random informative observation times, with applications to electronic medical records data. *Biometrics*, 71(1):90–101, 2015.

Thomas A. Lasko, Joshua C. Denny, and Mia A. Levy. Computational phenotype discovery using unsupervised feature learning over noisy, sparse, and irregular clinical data. *PloS one*, 8(6): e66341, 2013.

N.D. Lawrence. Gaussian process latent variable models for visualisation of high dimensional data. *Advances in neural information processing systems*, 16(3):329–336, 2004.

M. Lázaro-Gredilla, S. Van Vaerenbergh, and N.D. Lawrence. Overlapping mixtures of gaussian processes for the data association problem. *Pattern Recognition*, 45(4):1386–1395, 2012.

D.S. Lee, P.C. Austin, J.L. Rouleau, P.P. Liu, D. Naimark, and J.V. Tu. Predicting mortality among patients hospitalized for heart failure: derivation and validation of a clinical model. *Journal of the American Medical Association*, 290(19):2581–2587, 2003.

K. Levin, K. Henry, A. Jansen, and K. Livescu. Fixed-dimensional acoustic embeddings of variable-length segments in low-resource settings. In *ASRU*, pages 410–415. IEEE, 2013.

S. J. Lewis, T. Foltynie, A. D. Blackwell, T. W. Robbins, A. M. Owen, and R. A. Barker. Heterogeneity of parkinson's disease in the early clinical stages using a data driven approach. *Journal of neurology, neurosurgery, and psychiatry*, 76(3):343–348, Mar 2005.

H.L Li-wei, R.P. Adams, L. Mayaud, G.B. Moody, A. Malhotra, R.G. Mark, and S. Nemati. A physiological time series dynamics-based approach to patient monitoring and outcome prediction. *IEEE Journal of Biomedical and Health Informatics*, 19(3):1068–1076, 2015.

Jessica Lin, Eamonn Keogh, Li Wei, and Stefano Lonardi. Experiencing SAX: a novel symbolic representation of time series. *Data Mining and knowledge discovery*, 15(2):107–144, 2007.

Zachary C Lipton, David C Kale, and Randall Wetzel. Modeling missing data in clinical time series with rnns. In *Machine Learning for Healthcare (MLHC)*, 2016.

Z.C. Lipton, D.C. Kale, C. Elkan, and R. Wetzel. Learning to diagnose with lstm recurrent neural networks. In *International Conference on Learning Representations (ICLR)*, 2015.

Jennifer Listgarten, Radford M Neal, Sam T Roweis, Rachel Puckrin, and Sean Cutler. Bayesian detection of infrequent differences in sets of time series with shared structure. In *Advances in neural information processing systems*, pages 905–912, 2006.

R.J.A. Little and D.B. Rubin. *Statistical Analysis with Missing Data*. John Wiley & Sons, 2014.

Z. Liu and M. Hauskrecht. Clinical time series prediction: Toward a hierarchical dynamical system framework. *Artificial Intelligence in Medicine*, 2014.

J.J. Lok. Statistical modeling of causal effects in continuous time. *The Annals of Statistics*, pages 1464–1507, 2008.

J. Lötvall, C.A. Akdis, L.B. Bacharier, L. Bjermer, T.B. Casale, A. Custovic, R.F. Lemanske, A.J. Wardlaw, S.E. Wenzel, and P.A. Greenberger. Asthma endotypes: a new approach to classification of disease entities within the asthma syndrome. *Journal of Allergy and Clinical Immunology*, 127(2):355–360, 2011.

Richard F MacLehose and David B Dunson. Nonparametric bayes kernel-based priors for functional data analysis. *Statistica Sinica*, pages 611–629, 2009.

B.M. Marlin. Modeling user rating profiles for collaborative filtering. In *Advances in neural information processing systems*, 2003.

B.M. Marlin, D.C. Kale, R.G. Khemani, and R.C. Wetzel. Unsupervised pattern discovery in electronic health care data using probabilistic clustering models. In *ACM SIGHIT International Health Informatics Symposium*, pages 389–398. ACM, 2012.

B. Maurer, N. Graf, B.A. Michel, U. Müller-Ladner, L. Czirják, C. Denton, A. Tyndall, C. Metzig, V. Lanius, D. Khanna, et al. Prediction of worsening of skin fibrosis in patients with diffuse cutaneous systemic sclerosis using the eustar database. *Annals of the rheumatic diseases*, 74(6): 1124–1131, 2015.

CE McCulloch, H. Lin, EH Slate, and BW Turnbull. Discovering subpopulation structure with latent class mixed models. *Statistics in medicine*, 21(3):417–429, 2002.

J.M. Mooij, D. Janzing, and B. Schölkopf. From ordinary differential equations to structural causal models: the deterministic case. In *Uncertainty in Artificial Intelligence (UAI)*, 2013.

W Moore, D. Meyers, S. Wenzel, W. Teague, H. Li, X. Li, R. D'Agostino Jr, M. Castro, D. Curran-Everett, A. Fitzpatrick, et al. Identification of asthma phenotypes using cluster analysis in the severe asthma research program. *American journal of respiratory and critical care medicine*, 181 (4):315–323, 2010.

S.L. Morgan and C. Winship. *Counterfactuals and causal inference*. Cambridge University Press, 2014.

D.R. Mould. Models for disease progression: new approaches and uses. *Clinical Pharmacology & Therapeutics*, 92(1):125–131, 2012.

K.P. Murphy. *Dynamic bayesian networks: representation, inference and learning*. PhD thesis, University of California, Berkeley, 2002.

K.P. Murphy. *Machine learning: a probabilistic perspective*. MIT press, 2012.

S.A. Murphy. Optimal dynamic treatment regimes. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 65(2):331–355, 2003.

Bengt Muthén and Kerby Shedden. Finite mixture modeling with mixture outcomes using the em algorithm. *Biometrics*, 55(2):463–469, 1999.

Daniel S. Nagin and Candice L. Odgers. Group-based trajectory modeling in clinical research. *Annual Review of Clinical Psychology*, 6:109–138, 2010.

I. Nahum-Shani, M. Qian, D. Almirall, W.E. Pelham, B. Gnagy, G.A. Fabiano, J.G. Waxmonsky, J. Yu, and S.A. Murphy. Q-learning: A data analysis method for constructing adaptive interventions. *Psychological Methods*, 17(4):478, 2012.

J. Neyman. Sur les applications de la théorie des probabilités aux experiences agricoles: Essai des principes. *Roczniki Nauk Rolniczych*, 10:1–51, 1923.

J. Neyman. On the application of probability theory to agricultural experiments. *Statistical Science*, 5(4):465–472, 1990.

A.Y. Ng, A. Coates, M. Diel, V. Ganapathi, J. Schulte, B. Tse, E. Berger, and E. Liang. Autonomous inverted helicopter flight via reinforcement learning. In *Experimental Robotics IX*, pages 363–372. Springer, 2006.

J. Nocedal and S.J. Wright. Numerical optimization 2nd, 2006.

J.B. Oliva, W. Neiswanger, B. Póczos, E.P. Xing, H. Trac, S. Ho, and J.G. Schneider. Fast function to function regression. In *Conference on Artificial Intelligence and Statistics (AISTATS)*, 2015.

C. Păduraru, D. Precup, J. Pineau, and G. Comănici. An empirical analysis of off-policy learning in discrete mdps. In *Workshop on Reinforcement Learning*, page 89, 2012.

G. Parascandolo, N. Kilbertus, M. Rojas-Carulla, and B. Schölkopf. Learning independent causal mechanisms. *International Conference on Machine Learning (ICML)*, 2018.

J. Pearl. *Causality: models, reasoning and inference*. Cambridge University Press, 2009.

M.W. Pozen, R.B. D'Agostino, J.B. Mitchell, D.M. Rosenfeld, J.T. Guglielmino, M.L. Schwartz, N. Teebagy, J.M. Valentine, and W.B. Hood. The usefulness of a predictive instrument to reduce inappropriate admissions to the coronary care unit. *Annals of internal medicine*, 92(2_Part_1): 238–242, 1980.

M.W. Pozen, R.B. D'Agostino, H.P. Selker, P.A. Sytkowski, and W.B. Hood. A predictive instrument to improve coronary-care-unit admission practices in acute ischemic heart disease: a prospective multicenter clinical trial. *New England Journal of Medicine*, 310(20):1273–1278, 1984.

N. Prasad, L. Cheng, C. Chivers, M. Draugelis, and B.E. Engelhardt. A reinforcement learning approach to weaning of mechanical ventilation in intensive care units. *Uncertainty in Artificial Intelligence (UAI)*, 2017.

C. Proust-Lima, M. Séne, J.M.G Taylor, and H. Jacqmin-Gadda. Joint latent class models for longitudinal and time-to-event data: A review. *Statistical Methods in Medical Research*, 23(1): 74–90, 2014.

J.A. Quinn, C.K. Williams, and N. McIntosh. Factorial switching linear dynamical systems applied to physiological condition monitoring. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 31(9):1537–1551, 2009.

R. Raina, Y. Shen, A. Mccallum, and A.Y. Ng. Classification with hybrid generative/discriminative models. In *Advances in Neural Information Processing Systems (NIPS)*, 2003.

A. Rajkomar, E. Oren, K. Chen, A.M. Dai, N. Hajaj, M. Hardt, P.J. Liu, X. Liu, J. Marcus, M. Sun, et al. Scalable and accurate deep learning with electronic health records. *NPJ Digital Medicine*, 1(1):18, 2018.

James Ramsay et al. *Applied functional data analysis: methods and case studies*. Springer, 2002.

J.O. Ramsay. *Functional Data Analysis*. Wiley Online Library, 2006.

C.E. Rasmussen and C.K.I. Williams. Gaussian processes for machine learning. *the MIT Press*, 2 (3):4, 2006.

M.T. Ribeiro, S. Singh, and C. Guestrin. Why should i trust you?: Explaining the predictions of any classifier. In *International Conference on Knowledge Discovery and Data Mining (KDD)*, pages 1135–1144. ACM, 2016.

J.A. Rice and C.O. Wu. Nonparametric mixed effects models for unequally sampled noisy curves. *Biometrics*, 57(1):253–259, 2001.

D. Rizopoulos. Dynamic predictions and prospective accuracy in joint models for longitudinal and time-to-event data. *Biometrics*, 67(3):819–829, 2011.

D. Rizopoulos and P. Ghosh. A bayesian semiparametric multivariate joint model for multiple longitudinal outcomes and a time-to-event. *Statistics in medicine*, 30(12):1366–1380, 2011.

S. Roberts, M. Osborne, M. Ebden, S. Reece, N. Gibson, and S. Aigrain. Gaussian processes for time-series modelling. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 371(1984):20110550, 2013.

J.M. Robins. A new approach to causal inference in mortality studies with a sustained exposure period—application to control of the healthy worker survivor effect. *Mathematical Modelling*, 7 (9-12):1393–1512, 1986.

J.M. Robins. Estimation of the time-dependent accelerated failure time model in the presence of confounding factors. *Biometrika*, 79(2):321–334, 1992.

J.M. Robins. Causal inference from complex longitudinal data. In *Latent variable modeling and applications to causality*, pages 69–117. Springer, 1997.

J.M. Robins and M.A. Hernán. Estimation of the causal effects of time-varying exposures. *Longitudinal data analysis*, pages 553–599, 2009.

J.M. Robins, A. Rotnitzky, and D.O. Scharfstein. Sensitivity analysis for selection bias and unmeasured confounding in missing data and causal inference models. In *Statistical models in epidemiology, the environment, and clinical trials*, pages 1–94. Springer, 2000.

J. Ross and J. Dy. Nonparametric mixture of gaussian processes with constraints. In *International Conference on Machine Learning (ICML)*, pages 1346–1354, 2013.

D. Rothenhäusler, N. Meinshausen, P. Bühlmann, and J. Peters. Anchor regression: heterogeneous data meets causality. *arXiv preprint arXiv:1801.06229*, 2018.

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

D.B. Rubin. Bayesian inference for causal effects: The role of randomization. *The Annals of statistics*, pages 34–58, 1978.

M. Saeed, M. Villarroel, A.T. Reisner, G. Clifford, L.W. Lehman, G. Moody, T. Heldt, T.H. Kyaw, B. Moody, and R.G. Mark. Multiparameter intelligent monitoring in intensive care II (MIMIC-II): a public-access intensive care unit database. *Critical Care Medicine*, 39(5):952, 2011.

S. Saria and A. Goldenberg. Subtyping: What it is and its role in precision medicine. *IEEE Intelligent Systems*, 2015.

S. Saria, D. Koller, and A. Penn. Learning individual and population level traits from clinical temporal data. In *Advances in Neural Information Processing Systems (NIPS)*, pages 1–9. Citeseer, 2010.

S. Saria, A. Duchi, and D. Koller. Discovering deformable motifs in continuous time series data. In *International Joint Conference on Artificial Intelligence*, 2011.

S. Särkkä and A. Solin. Lecture notes on applied stochastic differential equations, 2014.

D. Scharfstein, A. McDermott, W. Olson, and F. Wiegand. Global sensitivity analysis for repeated measures studies with informative dropout: A fully parametric approach. *Statistics in Biopharmaceutical Research*, 6(4):338–348, 2014.

P Schulam and R Arora. Disease trajectory maps. In *Advances in Neural Information Processing Systems*, pages 4709–4717, 2016.

P. Schulam and S. Saria. A framework for individualizing predictions of disease trajectories by exploiting multi-resolution structure. In *Advances in Neural Information Processing Systems (NIPS)*, pages 748–756, 2015.

P. Schulam and S. Saria. Integrative analysis using coupled latent variable models for individualizing prognoses. *The Journal of Machine Learning Research*, 17(1):8244–8278, 2016.

P. Schulam and S. Saria. Reliable decision support using counterfactual models. In *Advances in Neural Information Processing Systems (NIPS)*, pages 1697–1708, 2017.

P. Schulam, F. Wigley, and S. Saria. Clustering longitudinal clinical marker trajectories from electronic health data: Applications to phenotyping and endotype discovery. In *Conference on Artificial Intelligence (AAAI)*, 2015.

Steven Shea and George Hripcsak. Accelerating the use of electronic health records in physician practices. *New England Journal of Medicine*, 362(3):192–195, 2010.

J.Q. Shi, R. Murray-Smith, and D.M. Titterington. Hierarchical gaussian process mixtures for regression. *Statistics and computing*, 15(1):31–41, 2005.

J.Q. Shi, B. Wang, E.J. Will, and R.M. West. Mixed-effects gaussian process functional regression models with application to dose–response curve prediction. *Stat. Med.*, 31(26):3165–3177, 2012.

R. Simon and A. Maitournam. Evaluating the efficiency of targeted designs for randomized clinical trials. *Clinical Cancer Research*, 10(20):6759–6763, 2004.

E. Snelson and Z. Ghahramani. Sparse gaussian processes using pseudo-inputs. In *NIPS*, 2005.

A. Sokol and N.R. Hansen. Causal interpretation of stochastic differential equations. *Electronic Journal of Probability*, 19(100):1–24, 2014.

H. Soleimani, A. Subbaswamy, and S. Saria. Treatment-response models for counterfactual reasoning with continuous-time, continuous-valued interventions. In *Uncertainty in Artificial Intelligence (UAI)*, 2017.

D. Sontag, K. Collins-Thompson, P.N. Bennett, R.W. White, S. Dumais, and B. Billerbeck. Probabilistic models for personalizing web search. In *Proceedings of the fifth ACM international conference on Web search and data mining*, pages 433–442. ACM, 2012.

D.J. Spiegelhalter. Probabilistic prediction in patient management and clinical trials. *Statistics in medicine*, 5(5):421–433, 1986.

M. W. State and N. Sestan. Neuroscience. the emerging biology of autism spectrum disorders. *Science (New York, N.Y.)*, 337(6100):1301–1303, Sep 14 2012.

Virginia Steen and Thomas A Medsger. Predictors of isolated pulmonary hypertension in patients with systemic sclerosis and limited cutaneous involvement. *Arthritis & Rheumatism*, 48(2):516–522, 2003.

E.W. Steyerberg. *Clinical prediction models*, volume 381. Springer, 2009.

A. Subbaswamy and S. Saria. Counterfactual normalization: Proactively addressing dataset shift using causal mechanisms. In *Uncertainty in Artificial Intelligence (UAI)*, 2018.

A. Subbaswamy, P. Schulam, and S. Saria. Preventing failures due to dataset shift: Learning predictive models that transport. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2019.

R.S. Sutton and A.G. Barto. *Reinforcement learning: An introduction*. MIT press Cambridge, 1998.

A. Swaminathan and T. Joachims. Counterfactual risk minimization. In *International Conference on Machine Learning (ICML)*, 2015.

D.P. Tashkin et al. Cyclophosphamide versus placebo in scleroderma lung disease. *New England Journal of Medicine*, 354(25):2655–2666, 2006.

S.L. Taubman, J.M. Robins, M.A. Mittleman, and M.A. Hernán. Intervening on risk factors for coronary heart disease: an application of the parametric g-formula. *International Journal of Epidemiology*, 38(6):1599–1611, 2009.

J. Taylor, W. Cumberland, and J. Sy. A stochastic model for analysis of longitudinal AIDS data. *Journal of the American Statistical Association*, 89(427):727–736, 1994.

R. Temple. Enrichment of clinical study populations. *Clinical Pharmacology & Therapeutics*, 88 (6):774–778, 2010.

M. E. Tipping and C. M. Bishop. Probabilistic principal component analysis. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 61(3):611–622, 1999.

M.K. Titsias. Variational learning of inducing variables in sparse gaussian processes. In *AISTATS*, 2009.

M.K. Titsias and N.D. Lawrence. Bayesian gaussian process latent variable model. In *AISTATS*, 2010.

N. Tomašev, X. Glorot, J.W. Rae, M. Zielinski, H. Askham, A. Saraiva, A. Mottram, C. Meyer, S. Ravuri, I. Protsyuk, et al. A clinically applicable approach to continuous prediction of future acute kidney injury. *Nature*, 572(7767):116, 2019.

A. Tsiatis. *Semiparametric theory and missing data*. Springer Science & Business Media, 2007.

B. Varadarajan et al. Unsupervised learning of acoustic sub-word units. In *Proc. ACL*, pages 165–168, 2008.

J. Varga, C.P. Denton, and F.M. Wigley. *Scleroderma: From pathogenesis to comprehensive management*. Springer Science & Business Media, 2012.

G. Verbeke and G. Molenberghs. *Linear mixed models for longitudinal data*. Springer, 2009.

H. Wang, F. Nie, H. Huang, J. Yan, S. Kim, S. Risacher, A. Saykin, and L. Shen. High-order multi-task feature learning to identify longitudinal phenotypic markers for alzheimer's disease progression prediction. In *Advances in Neural Information Processing Systems (NIPS)*, pages 1277–1285, 2012.

X. Wang, D. Sontag, and F. Wang. Unsupervised learning of disease progression models. In *International Conference on Knowledge Discovery and Data Mining (KDD)*, pages 85–94. ACM, 2014.

S. Watanabe. Karhunen-loeve expansion and factor analysis, theoretical remarks and applications. In *Proc. 4th Prague Conf. Inform. Theory*, 1965.

S.E. Wenzel, L.B. Schwartz, E.L. Langmack, J.L. Halliday, J.B. Trudeau, R.L. Gibbs, and H.W. Chu. Evidence that severe asthma can be divided pathologically into two inflammatory subtypes with distinct physiologic and clinical characteristics. *American journal of respiratory and critical care medicine*, 160(3):1001–1008, 1999.

J. Wiens, J. Guttag, and E. Horvitz. Patient risk stratification with time-varying parameters: a multitask learning approach. *Journal of Machine Learning Research (JMLR)*, 17(209):1–23, 2016.

L.D. Wiggins, D.L. Robins, L.B. Adamson, R. Bakeman, and C.C. Henrich. Support for a dimensional view of autism spectrum disorders in toddlers. *Journal of autism and developmental disorders*, 42(2):191–200, 2012.

Y. Xu, Y. Xu, and S. Saria. A Bayesian nonparametric approach for estimating individualized treatment-response curves. In *Machine Learning for Healthcare (MLHC)*, pages 282–300, 2016.

J. Zhou, L. Yuan, J. Liu, and J. Ye. A multi-task learning formulation for predicting disease progression. In *International Conference on Knowledge Discovery and Data Mining (KDD)*, pages 814–822, 2011.

Biographic Statement

Peter Schulam is a PhD candidate in the Computer Science Department at Johns Hopkins University where he is working with Professor Suchi Saria. His research interests lie at the intersection of machine learning, statistical inference, and healthcare with an emphasis on developing methods to support the personalized medicine initiative. Before coming to JHU, he received his MS from Carnegie Mellon's School of Computer Science and his BA from Princeton University. He was awarded a National Science Foundation Graduate Research Fellowship and the Dean's Centennial Fellowship within Johns Hopkins' Whiting School of Engineering.

He was born in Houston, Texas in March, 1989. He lives in the Hampden neighborhood of Baltimore, Maryland with his wife (Jenny) and 1-year-old son (Peter Harold). In his free time, he loves to study the classic texts of probability, statistics, and machine learning with Peter Harold. These include titles such as "Bayesian Probability for Babies", "Neural Networks for Babies", and "Baby Loves to Code".