Reliable and Practical Machine Learning for Dynamic Healthcare Settings

Helen Zhou

December 2023
CMU-ML-23-108

Machine Learning Department
School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213

Thesis Committee:
Zachary C. Lipton (Chair, Carnegie Mellon University)
Sivaraman Balakrishnan (Carnegie Mellon University)
Jeremy C. Weiss (National Institutes of Health)
Maggie Makar (University of Michigan)

*Submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy.*

This research was supported in part by the Paul and Daisy Soros Fellowship and the National Science Foundation Graduate Research Fellowship Program under grant numbers DGE1745016 and DGE2140739.

Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

Keywords: Machine Learning, Healthcare, Distribution Shift, Domain Adaptation, Time Series, Underreporting, Survival Analysis, Deployment, Sequential Decision-Making

Abstract

Machine learning (ML) algorithms have shown great promise on a variety of healthcare-related tasks. However, as these algorithms transition from research to deployment, they enter a constantly evolving environment rife with changes in clinical practices, record-keeping policies, patient populations, and diseases themselves. Models that performed well in the past are liable to fail in the future, and especially in such high-stakes settings as healthcare, complacency can have negative consequences. This thesis explores the application and development of machine learning algorithms for dynamic healthcare settings. First, we present case studies which characterize how common ML techniques fare on several medical datasets over time, and discuss types of distribution shifts that can occur in healthcare data. In the second part we dive into learning from underreported data and how to adapt to shifting levels of underreporting, motivated by challenges which arose when developing a model for predicting severe COVID-19. Moving from prediction over time to decision-making over time, we study two scenarios, one in which decisions are cheap, frequent, and directly tied to forecasts, and another in which the interaction dynamics are modeled off of those between a doctor and patient, where interactions have some cost. Finally, we reflect more broadly on the development of reliable machine learning algorithms in healthcare over time.

Acknowledgements

First and foremost, I would like to thank my advisor, Zachary Lipton. Zack was supportive in every aspect of my PhD, from spinning up healthcare collaborations together, to insightful research discussions, to open conversations about career and life. He pushed me to ask the right scientific questions, and gave me the freedom to explore questions I was interested in. Thank you for being a constant source of inspiration, encouragement, and guidance.

I'd also like to thank my committee members for their roles in my development as a researcher. Sivaraman Balakrishnan has been an outstanding mentor in my journey into statistics and ML theory, and I have learned a lot from him about how to tackle problems in a principled, thoughtful, and holistic way. I first met Jeremy Weiss virtually when starting a project on COVID-19. His insights from both a medical and ML perspective were vital for understanding the problem on a deeper level, and through continued collaborations he has taught me so much about how to find and tackle impactful clinical ML problems. Maggie Makar brought in fresh perspectives, gave valuable feedback on works in my thesis, and helped me think about our findings in new ways.

One of the things I most enjoyed about my PhD was getting to work with a lot of talented, genuine, and kind people. I am thankful for my academic collaborators, including Audrey Huang, Cheng Cheng, David Childers, George Chen, and Kamyar Azizzadenesheli. It was a joy bouncing around ideas and diving into research directions with each of them. I'd also like to thank my clinical collaborators at Highmark Health and Allegheny Health Network (Kelly Shields, Timothy Schreiber, Charles Li, Rishi Maheshwary, Gursimran Kochhar, Tariq Cheema, Hossein Seyed, and others) and the University of Pittsburgh Medical Center (Mohamed Adam Abdelgadir and others), who were crucial for understanding context, and without whom much of my work would not have been possible. It has been fulfilling mentoring masters students through their own research journeys over the years, and I would also like to thank Yuwen Chen, Pratheek D'Souza, Jesse Kim, and Jamin Chen, from whom I've learned a lot in turn.

I'd also like to thank all the wonderful folks in ACMI lab, including Daniel Jeong, Danish Pruthi, Divyansh Kaushik, Emily Byun, Jacob Tyo, Jennifer Hsia, Kundan Krishna, Manley Roberts, Michael Feffer, Nil-Jana Akpinar, Pranav Mani, Pratyush Maini, Saurabh Garg, Shantanu Gupta, Tanya Marwah, and many more. Thank you for all the camaraderie and adventures over the years, from runs through Schenley Park, to lab dinners, to late-night paper revising sessions, to exploring new cities at conferences over the years.

Industry internships gave me the opportunity to meet new colleagues at other institutions and dedicate some time to explore various new topics of interest. I would like to thank my research internship hosts at Google, Andrew Dai, Yuan Xue, Jingtao Wang, and Sercan Arik, the Medical Brain and Google Cloud AI Discovery teams, as well as many fellow interns.

It takes a village to raise a PhD, and I am truly grateful for the lovely friends and support network I found not only in my collaborators and labmates, but also in the broader MLD and CMU community. There are too many to list individually and comprehensively, but I am eternally thankful for all that they have done to keep me sane throughout the PhD, providing solidarity in the challenging times and celebrating the wins.

At CMU, there are several groups from which I drew strength in my PhD. My officemates have been great sources of support over the years, including Leqi Liu and Sebastian Caldas, who have been there for me from my first year of PhD to my last, along with Brandon Trabucco, Conor Igoe, Ian Char, Stefani Karp, Swaminathan Gurumurthy, Theophile Gervet, and Zixin Wang. My cohortmates early on made Pittsburgh feel like home, from late-night karaoke, to playing board games, to exploring the food scene. My housemates over the years Paul Liang, Tom Yan, Ben Eyesenbach, Leonid Keselman, and Victoria Dean have also been great companions in the PhD journey, and helped keep me inspired. I'm also grateful to the SCS Dean's Advisory Committee that made it possible for me to start the Social Connectedness Working Group and to meet awesome students in other departments including Catherine King, Emmy Liu, Rishi Veerapaneni, Shahul Alam, Tobi Duerschmid, and many others. I'm also grateful to senior students for their mentorship and kindness early on in the PhD, including Lisa Lee, Maruan Al-Shevidat, and Otilia Stretcu, as well as the MLD women group for their friendship and sense of shared experience. Finally, thank you to Diane Stidle for her warmth, kindness, and incredible amount of support that she's provided MLD students throughout the years.

Outside of MLD, I'm grateful to a few communities that have helped me keep the bigger picture in mind. The Paul and Daisy Soros Fellowship gave me the opportunity to meet many new Americans doing amazing things in their respective fields, one of whom is my good friend and fellow Pittsburgher Anna Li. The machine learning for healthcare community is one that I cherish, and I look forward to every conference to catch up with old friends, meet new ones, and engage in lively conversations about cutting edge ML and healthcare research. I'm also grateful to my fellow ML4H Symposium organizers, who have been a pleasure to work with and all are doing inspiring work as well.

Zooming out, I'd like to express my deep gratitude to mentors that guided me down the academic path. I'm grateful to David Sontag and the members of the Clinical Machine Learning group who first helped me get oriented to research at the intersection of machine learning and healthcare, including Michael Oberst, Sanjat Kanjilal, Christina Ji, Irene Chen, Rahul Krishnan, and Frederik Johansson. I'm also grateful to Soroush Vosoughi, Deb Roy, and the Laboratory for Social Machines for taking a chance on me in my second year of undergrad and opening my eyes to the world of research.

I would also like to thank Everardo Rosales, co-parent to our fluffy goldendoodle, my sounding board for big decisions, and lifter of spirits after a long day of failed experiments or perplexing proofs. Thank you for being there for me.

Finally, and most profoundly, I would like to thank my parents for their unwavering support and unconditional love throughout my entire life. Their hard work and sacrifices paved the way for me to access the best possible education and opportunities, and I would not be the person I am today without them. I'm proud to follow in my family's academic pursuits, and proud to be my parents' daughter. This thesis is dedicated to them.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Machine learning (ML) is better-positioned to transform healthcare than ever before. Over the last 15 years, a wave of digitalization and adoption of electronic medical records (Figure 1.1, left) has led to the creation of large repositories of rich healthcare data. Empowered by advancements in data and computational resources, researchers have developed ML algorithms for assisting with tasks such as diagnosis and early detection (Lipton et al., 2016a; Gulshan et al., 2016; Kanjilal et al., 2020), monitoring and forecasting trends in public health (Reinhart et al., 2021), and modeling disease progression (Wang et al., 2014; Severson et al., 2021). In addition to research publications, deployments in ML for healthcare have also been accelerating (Figure 1.1). The U.S. Food and Drug Administration has authorized hundreds of AI/ML-based software as medical devices (FDA, 2021), hospitals have deployed ML-powered sepsis alert systems (Sendak et al., 2020), and millions of smart watch consumers have atrial fibrillation detection ML algorithms running on their wrists (Perez et al., 2019).

As ML models born at the research bench become integrated into products ripe for deployment, it is critical to understand how such models might fare in the tumultuous reality where healthcare problems, practices, and systems are ever-changing. New diseases emerge and mutate, new treatments are developed, clinical standards and government policies change over time, and the way healthcare data is collected and stored is also transforming. In a world where ML models influence medical practice, reliance on fickle or unrepresentative data can pose real downstream risk. In the University of Michigan hospital system, for example, a widely used sepsis-alerting model developed by Epic Systems had to be decomissioned due to shifts in patient characteristics associated with the onset of the coronavirus disease 2019 (COVID-19) pandemic (Finlayson et al., 2021). This thesis centers around the driving question:

How can we develop machine learning systems suitable for dynamic healthcare settings?

Interest in ML for dynamic settings has grown substantially over the last several years. One popular paradigm for dynamic interaction and decision-making is reinforcement learning (RL), where an agent learns to optimize a reward function by interacting with an environment (Sutton and Barto, 2018). The distribution of collected experience (i.e. data on states visited, actions

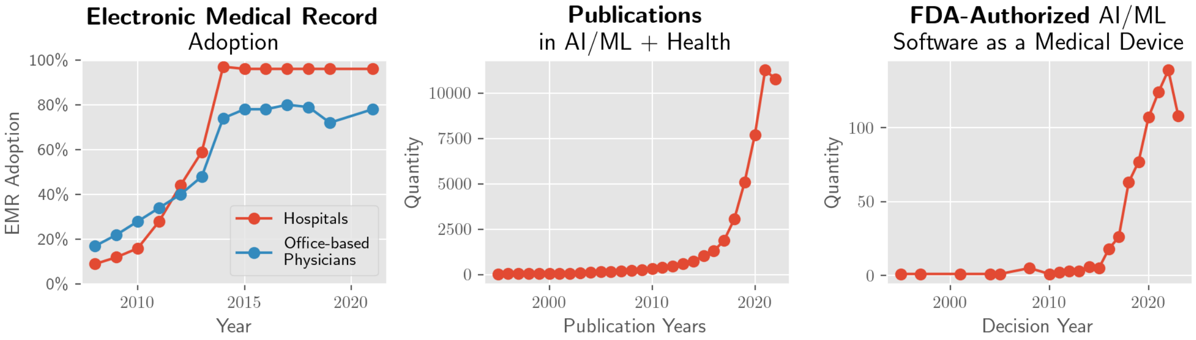


Figure 1.1: Adoption of electronic medical records in vs. year (top), and publications and FDA-approved medical devices in artificial intelligence and machine learning for healthcare over time (bottom).

taken, and rewards received) changes depending on the agent's policy, and the agent continually updates its policy based on the data it collects. While these methods have been successful in simulated or closed-loop settings (Yu et al., 2021) where actions, observations, and learning can be tightly integrated, it should be noted that in several real-world healthcare settings, experience is expensive to collect and doctors are not acting in a closed-loop environment. Researchers have proposed using RL for optimizing the choice of medications, drug dosing, and for targeting personalized laboratory values (Liu et al., 2020), however, reliable evaluation of how such algorithms would perform in the real world remains challenging (Gottesman et al., 2018; Gottesman et al., 2019), and examples of these algorithms being deployed in hospitals remain scarce.

Instead, ML applications deployed in healthcare have largely focused on prediction, where a model is trained on data to detect or diagnose some condition (Topol, 2019). However, the dynamic nature of healthcare still poses a challenge for typical prediction tasks. As companies and researchers have begun deploying ML models in the real world, concerns over robustness to distribution shift have become more prominent. D'Amour et al. (2022a) note that ML systems frequently exhibit unexpectedly poor behavior upon deployment in real-world domains, and suggest that models be selected after they are *stress-tested* along practically important dimensions, such as testing an opthamological model on images taken from a different camera, and stratifying performance of a dermatological model by skin type. In the healthcare domain, Finlayson et al. (2021) name dataset shift as a major driver of AI system malfunction, and present common causes of dataset shift including technological changes (e.g. software vendors), population and setting changes (e.g. demographics), and behavioral changes (e.g. reimbursement incentives). Researchers in ML have also formalized various notions of distribution shift, working on settings such as covariate shift (Shimodaira, 2000a; Zadrozny, 2004; Sugiyama et al., 2007a; Gretton et al., 2009), where the distribution of covariates $p(x)$ changes but $p(y|x)$ remains the same, as well as label shift (Storkey et al., 2009; Zhang et al., 2013; Lipton et al., 2018; Garg et al., 2020), where the distribution of labels $p(y)$ changes but $p(y|x)$ remains the same. However, there is limited work on understanding how these notions of shift manifest in real healthcare data over time, and to what degree these shifts affect model performance over time.

1.1 Thesis Statement and Overview

This thesis dives into the realities and challenges of applying ML to dynamic healthcare settings. We will explore case studies from experiments on open-access medical datasets, describe findings from collaborations with regional healthcare providers, and discuss idealized simplifications of healthcare settings more amenable to a formal analysis of the dynamics at play.

Thesis Statement. *Building reliable and practically useful machine learning systems suitable for deployment in dynamic healthcare settings requires us to:*

($S1$) understand the types of shifts that occur in healthcare data over time,

($S2$) develop models and algorithms that are robust to these shifts, and

($S3$) take decision-making processes into account.

From the Oxford Languages dictionary, *reliable* is defined as "consistently good in quality or performance; able to be trusted," and *dynamic* is defined as "characterized by constant change, activity, or progress." To achieve consistently good performance in the face of constant change, it is critical to understand the types of changes that might occur to thwart good performance ($S1$). Upon characterization of these types of changes, it is possible to develop models and algorithms that might proactively account for them ($S2$). Finally, since healthcare delivery happens through the translation of medical insights into practical actions taken in order to improve patient well-being, it is important to move beyond pure prediction and to also take decision-making processes and dynamics into account ($S3$).

The thesis is organized into three main parts, each exploring a facet of the thesis statement.

Part I: Real-World Distribution Shifts in Healthcare Over Time

In the first part of the thesis, we characterize some of the distribution shifts that occur in real-world healthcare data over time. We start with a case study of the emergence of COVID-19 in the United States, a time of high uncertainty and constant change (Chapter 2). At the same time, it was a period of unprecedented data collection and collaboration, testing the limits of the healthcare system and the capabilities of policy-makers to react quickly and effectively. We examine areas where ML models could be useful given the data available at various points in time, and discuss the myriad of factors that were constantly shifting and potentially confounding popularly reported statistics. Zooming out, we then perform a large scale empirical analysis on several open-access medical datasets (Chapter 3), providing a framework and code package for evaluating how models would have performed if they had been deployed at various points in the past. This type of analysis allows practitioners to detect distribution shifts that might have occured in the past, and thus gives them the opportunity to prepare for similar types of shifts in the future.

Part II: Underreporting in Healthcare Data and Missingness Shift

In the second part of the thesis, we expand upon some of the challenges that arose from predicting severe COVID-19 using real-world healthcare data provided by our clinical collaborators. In particular, we noticed that the raw medical data was very sparse, and several concepts were underreported (Chapter 4). For example, instead of having a single flag for whether a patient had received a COVID-19 vaccine, the Moderna, Pfizer, or Johnson & Johnson vaccines were all coded separately, as were the different boosters. Even after combining all such columns into a single vaccination feature, the rate of COVID-19 vaccinations recorded in the hospital database was dramatically lower than the actual rate of vaccination. Taking into account the issue of underreporting, we propose and implement a technique for learning clinical concepts relevant to the prediction task at hand, and find that this technique maintains better performance over time than directly using raw features. Motivated by the problem of underreporting, we also formulate the problem of *domain adaptation under missingness shift* (Chapter 5), where in this setup one has access to labeled data in a source domain and unlabeled data in a target domain, rates of missing data very between source and target. The goal of domain adaptation is then to a learn a predictor that performs well in the target domain. We provide a theoretical analysis of this setting, propose techniques for estimation of the optimal target predictor under various assumptions, and discuss extensions that help bridge the gap between theory and practice.

Part III: Decision-Making in Dynamic Healthcare Settings

Transitioning from prediction to decision-making, we consider a problem in healthcare operations: inventory management (Chapter 6). This is an environment where algorithms might be deployed not only to make predictions, but also make decisions based on those predictions. In this setting where decisions might be directly and straightforwardly tied forecasts of future demand, we argue that forecasting models should not necessarily optimize for generic objectives such as mean squared error, but objectives that take downstream desiderata into account (e.g. cost, customer satisfaction, etc.). At the same time, there are several settings in healthcare where this framework falls short. For example, when a doctor sees a patient, they typically must prescribe a course of treatment (e.g. 500 mg of Metformin twice a day), and decide when to next see the patient. Furthermore, each appointment has some cost, so constant observations or medical interventions may not be feasible. We formalize this interaction dynamic in a reinforcement learning setup, and study how one might learn to time their actions (Chapter 7).

1.2 Bibliographic Notes

Most of the work in thesis is based on the following papers. The asterisk * indicates authors with equal contribution.

Part I, Chapter 2 is based on:

- Helen Zhou, Cheng Cheng, Zachary C Lipton, George H Chen, and Jeremy C Weiss. "Mortality Risk Score for Critically Ill Patients with Viral or Unspecified Pneumonia: Assisting Clinicians with COVID-19 ECMO Planning". In: *International Conference on Artificial Intelligence in Medicine*. Springer. 2020, pp. 336–347
- Cheng Cheng, Helen Zhou, Jeremy C Weiss, and Zachary C Lipton. "Unpacking the Drop in COVID-19 Case Fatality Rates: A Study of National and Florida Line-Level Data". In: *AMIA Annual Symposium Proceedings*. Vol. 2021. American Medical Informatics Association. 2021, p. 285

Part I, Chapter 3 is based on:

- Helen Zhou, Yuwen Chen, and Zachary Lipton. "Evaluating Model Performance in Medical Datasets Over Time". In: *Conference on Health, Inference, and Learning*. PMLR. 2023, pp. 498–508

Part II Chapter 4 is based on:

- Helen Zhou, Cheng Cheng, Kelly J Shields, Gursimran Kochhar, Tariq Cheema, Zachary C Lipton, and Jeremy C Weiss. "Learning Clinical Concepts for Predicting Risk of Progression to Severe COVID-19". In: *AMIA Annual Symposium Proceedings*. Vol. 2022. American Medical Informatics Association. 2022, p. 1257

Part II, Chapter 5 is based on:

- Helen Zhou, Sivaraman Balakrishnan, and Zachary Lipton. "Domain adaptation under missingness shift". In: *International Conference on Artificial Intelligence and Statistics*. PMLR. 2023, pp. 9577–9606

Part III, Chapter 6 is based on:

- Helen Zhou, Sercan O Arik, and Jingtao Wang. "Business Metric-Aware Forecasting for Inventory Management". In: *arXiv preprint arXiv:2308.13118* (2023)

Part III, Chapter 7 is based on:

- Helen Zhou, Audrey Huang, Kamyar Azizzadenesheli, David Childers, and Zachary Lipton. "Timing as an Action: Learning When to Observe and Act". In: *under submission* (2023)

Part I

Characterizing Real-World Distribution Shifts in Healthcare Over Time

"Pure logical thinking cannot yield us any knowledge of the empirical world; all knowledge of reality starts from experience and ends in it."

- Albert Einstein, *Ideas and Opinions*

This part of the thesis covers empirical works that observe naturally-occurring distribution shifts over time in medical data. It begins with a case study of coronavirus disease (COVID-19) in the United States, and then remarks upon trends more broadly observed in several medical datasets.

Chapter 2

COVID-19 in the United States

The emergence of coronavirus disease COVID-19 tested the limits of medicine and the capabilities of policy-makers to react quickly and effectively. As the virus circulated throughout different pockets of the U.S. population, hospital resources were stretched thin, COVID-19 tests saw ramp-ups and shortages, new potential treatments were introduced (or debunked), and the disease itself mutated over time (Figure 2.1). As our understanding of the virus evolved, so did the systems for monitoring, treating, and recording COVID-19.

When COVID-19 first emerged the U.S., there was limited historical data reflected in hospital medical records, and reporting efforts had yet to be consolidated. Yet, there was already a rapidly growing need for data-driven tools to help doctors assess and treat patients. When a new disease emerges, how can we develop ML tools that are useful before we have to wait for the data to catch up? In this chapter, we start by delving into a case study on the development of a risk prediction tool to assist clinicians with COVID-19 extracorporeal membrane oxygenation (ECMO) planning (Section 2.1). Respiratory complications due to coronavirus claimed hundreds of thousands of lives in 2020. ECMO is a life-sustaining oxygenation and ventilation therapy that may be used when mechanical ventilation is insufficient. The ECMO machine acts as an artifical lung, and works by pumping and oxygenating blood outside of the body, before warming it to body temperature and pumping it back into the body. While early planning and surgical cannulation for ECMO can increase survival, clinicians report the lack of a risk score hinders these efforts. In this work, we develop the PEER risk score to highlight critically ill patients with viral or unspecified pneumonia at high risk of mortality in a subpopulation eligible for ECMO.

Next, we take a population-level view of COVID-19, characterizing the state of the disease's spread and severity in the U.S. population over the course of its first year (Section 2.2). To carefully decompose several of the claims put forth about the state of the pandemic by both academics and news outlets at the time, we critically analyze data from the U.S. Centers for Disease Control and Prevention (CDC) and state departments of health, characterizing and controlling for various simultaneously shifting factors over time, and obtaining adjusted estimates of a population-level quantity (Cheng et al., 2021). We conclude with broader reflections on how the changes observed in this case study motivate interesting challenges in machine learning research.

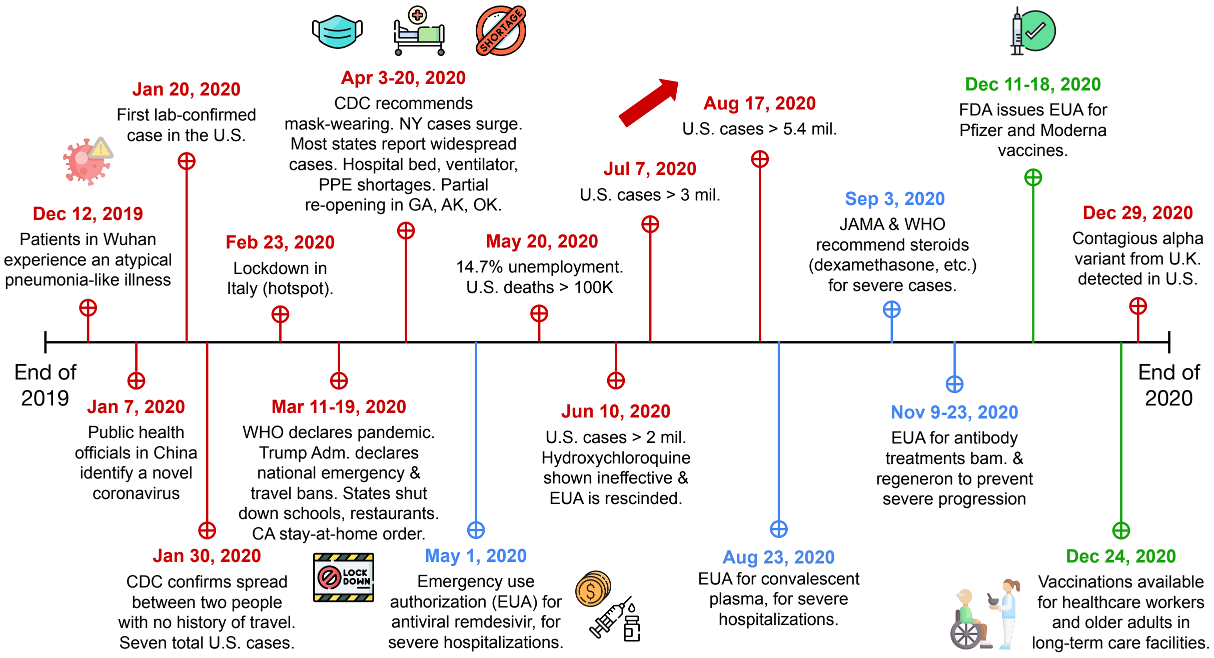

End of 2019
Dec 12, 2019
Patients in Wuhan experience an atypical pneumonia-like illness
Jan 7, 2020
Public health officials in China identify a novel coronavirus
Jan 20, 2020
First lab-confirmed case in the U.S.
Jan 30, 2020
CDC confirms spread between two people with no history of travel. Seven total U.S. cases.
Feb 23, 2020
Lockdown in Italy (hotspot).
Mar 11-19, 2020
WHO declares pandemic. Trump Adm. declares national emergency & travel bans. States shut down schools, restaurants. CA stay-at-home order.
LOCK DOWN
Apr 3-20, 2020
CDC recommends mask-wearing. NY cases surge. Most states report widespread cases. Hospital bed, ventilator, PPE shortages. Partial re-opening in GA, AK, OK.
SHORTAGE
May 1, 2020
Emergency use authorization (EUA) for antiviral remdesivir, for severe hospitalizations.
May 20, 2020
14.7% unemployment. U.S. deaths > 100K
Jun 10, 2020
U.S. cases > 2 mil. Hydroxychloroquine shown ineffective & EUA is rescinded.
Jul 7, 2020
U.S. cases > 3 mil.
Aug 17, 2020
U.S. cases > 5.4 mil.
Aug 23, 2020
EUA for convalescent plasma, for severe hospitalizations.
Sep 3, 2020
JAMA & WHO recommend steroids (dexamethasone, etc.) for severe cases.
Nov 9-23, 2020
EUA for antibody treatments bam. & regeneron to prevent severe progression
Dec 11-18, 2020
FDA issues EUA for Pfizer and Moderna vaccines.
Dec 24, 2020
Vaccinations available for healthcare workers and older adults in long-term care facilities.
Dec 29, 2020
Contagious alpha variant from U.K. detected in U.S.
End of 2020

Figure 2.1: Timeline of the first year after the emergence of COVID-19, with a focus on U.S. events, synthesized from the U.S. Centers for Disease Control and Prevention (CDC, 2023). Green corresponds to COVID-19 vaccine-related events, blue corresponds to events related to other treatments, and red corresponds to other events.

2.1 Risk Prediction for COVID-19 ECMO Planning

Introduction

Respiratory complications due to coronavirus claimed hundreds of thousands of lives in 2020. Many COVID-19 cases progress from Severe Acute Respiratory Syndrome (SARS-CoV-2) with viral pneumonia to acute respiratory distress syndrome (ARDS) to death. Extracorporeal membrane oxygenation (ECMO) can temporarily sustain patients with severe ARDS when mechanical ventilation fails to facilitate with oxygenation via lungs. However, ECMO is costly and applicable only for patients healthy enough to recover and return to a high functional status.

While ECMO is more effective when planned in advance (Combes et al., 2018), applicable risk scores remain unavailable (Liang et al., 2020a; American College of Cardiology, 2020). This work introduces the Viral or Unspecified **P**neumonia **E**CMO-**E**ligible **R**isk (PEER) Score, using measurements from the time of would-be planning—early in the critical care stay. In contrast to prior pneumonia risk scores (Fine et al., 1997; Marti et al., 2012; Charles et al., 2008; Lim et al., 2003), the PEER score targets those with viral or unspecified pneumonia in the critical care setting, for a cohort potentially eligible for ECMO. Unspecified pneumonia is included since the infectious etiology of pneumonia often cannot be determined, and it broadens the study population.

Though limited by geographic availability, ECMO usage has increased 4-fold in the last decade (Ramanathan et al., 2020). COVID-19 guidelines suggest ECMO as a late option in escalation of care for severe ARDS secondary to SARS-CoV-2 infection (Liang et al., 2020a; Alhazzani et al., 2020). Early epidemiological studies of coronavirus (Wang et al., 2020; Yang et al., 2020; Zhou et al., 2020a) had yet to establish ECMO's utility. A pooled analysis of four studies (Henry, 2020) showed mortality rates of 95% with ECMO vs. 70% without, but the number of ECMO recipients was small, and no studies described a protocol specifying indications for ECMO. A later pooled analysis of 331 cases found mortality rates of 46% with ECMO and 59-71% without ECMO (Melhuish et al., 2020).

To better understand the role of ECMO as a rescue for ventilation non-responsive, SARS-CoV-2 ARDS, we study its broader use in ARDS. Treatment guidelines suggest ECMO use in severe ARDS alongside other advanced ventilation strategies (Matthay et al., 2020; World Health Organization et al., 2020), with the World Health Organization citing effectiveness for ARDS and reducing mortality of the Middle East Respiratory Syndrome (MERS). Despite these recommendations and allocated ECMO resources (Ramanathan et al., 2020), risk scores tailored to ECMO consideration are lacking. Our study addresses this by drawing from viral and source unidentified cases of pneumonia that escalate to critical care admissions, guided by the intuition that ARDS from these pneumonia are expected to better resemble COVID-19 ARDS than all-comer ARDS.

Related Work

There are a number of pneumonia (W et al., 2003; Fine et al., 1997; Charles et al., 2008; Lim et al., 2003; Guo et al., 2019), COVID-19 (Gong et al., 2020b; Jiang et al., 2020; Gong et al., 2020a), hospitalization mortality (Zimmerman et al., 2006), and ECMO risk scores (Schmidt et al., 2015), but none center on the time of risk evaluation for ECMO candidacy. The pneumonia and COVID-19 risk scores are assessed on populations with lower acuity, while APACHE is not focused on respiratory illness. Our risk score is meant for use in ECMO planning rather than predicting outcomes among patients already receiving ECMO. Registry-based studies have also compared SARS-CoV-2 outcomes to that of other viral infections, including MERS, H1N1 flu, and seasonal flu. One MERS-related ARDS study of critically ill patients demonstrated higher mortality than those in studies on COVID-related ARDS, but may be attributed to sicker patients at enrollment (Arabi et al., 2017). A similar H1N1 study reported lower mortality (12-17%), albeit considering a younger population (average age 40) (Aokage et al., 2015).

Physiologic concerns have also been raised about the use of ECMO for SARS-CoV-2. One argues that while ECMO is primarily beneficial for respiratory recovery, a spike in all-cause death but not ARDS-related death could indicate a limited role of ECMO(Henry and Lippi, 2020). Others point out that COVID-associated lymphopenia might be exacerbated by ECMO-induced lymphopenia which could mechanistically affect a healthy immune response to infection. Inflammatory cytokines and specifically interleukin 6 elevation is associated with COVID-19 mortality and rises with the use of ECMO (Henry, 2020; Bizzarro et al., 2011). These expert voices do not argue for the avoidance of ECMO, but rather call for additional study.

Data

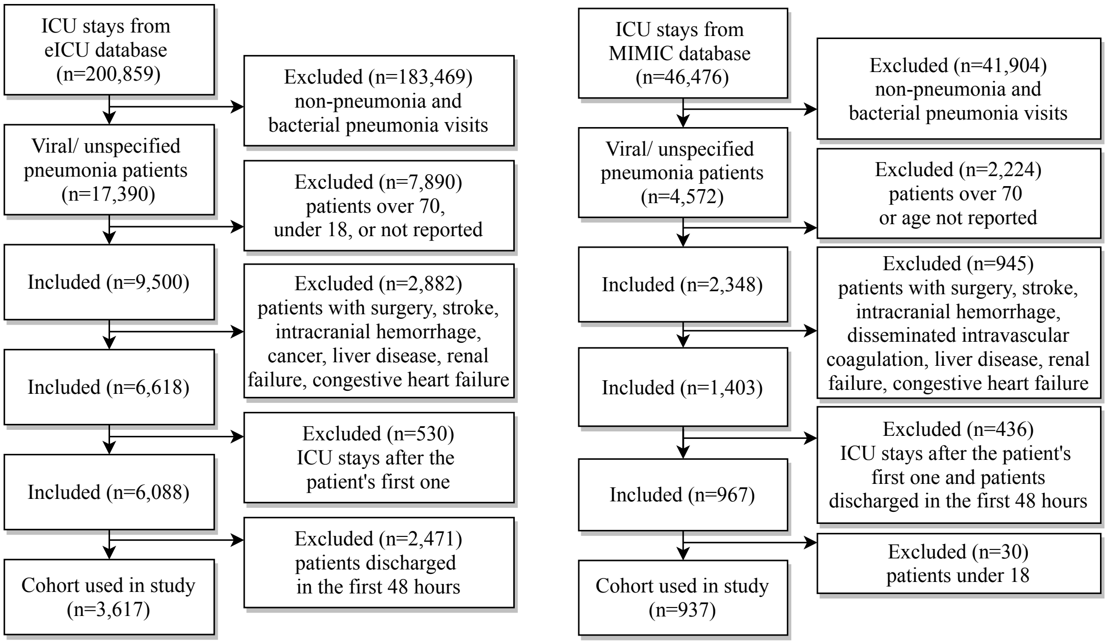
The eICU Collaborative Research Database (Pollard et al., 2018b) contains 200,859 admissions to intensive care units (ICU) across multiple centers in the United States between 2014 and 2015. The MIMIC-III clinical database (Johnson et al., 2016) consists of data from 46,476 patients who stayed in critical care units of the Beth Israel Deaconess Medical Center between 2001 and 2012. Model development and in-domain validation primarily use data from eICU, and out-of-domain validation uses MIMIC-III.

Cohort Selection Inclusion criteria for the study cohort are delineated in Figure 2.2. The population of interest is among patients with viral or otherwise unspecified non-bacterial, non-fungal, non-parasitic, and non-genetic pneumonia. While there are no absolute contraindications of ECMO, the therapy is reserved for patients likely to have functional recovery. Patients over 70 years old would not be good candidates for ECMO, and SARS-CoV-2 pneumonia progressing to hypoxic respiratory failure is exceedingly rare in patients under 18. Other relative contraindications to ECMO are also listed in Figure 2.2. We select the first ICU stay within each patient's hospital stay, and exclude patients who died or were discharged within the first 48 hours of being admitted. This is done to focus on the stage of critical care after initial entry when lower-risk oxygen supplementation strategies (*e.g.*, ventilation) are being performed, and, methodologically, to provide a richer set of features for prediction. Table 2.1 and Appendix Table A.1 summarize characteristics of the cohorts.

Data Extraction The study cohorts are extracted using string matching on diagnosis codes and subsequent clinician review. Features are merged through a process of visualization, query, and physician review. This includes harmonizing feature units, removing impossible values, and merging redundant data fields. Additional details are in Appendix A. All features are combined into a fixed-length vector, using the most recent value prior to 48 hours after ICU admission. Before imputation, approximately half of the features had missingness below 5%, and 80% of the features had missingness below 30%, however multiple variables had high missingness (Appendix A). Missing values are imputed using MissForest (Stekhoven and Bühlmann, 2011), which we find PEER is insensitive to (Appendix A).

Features Features are extracted from demographics, comorbidities, vitals, physical exams, and lab findings routinely collected in critical care settings. Numerical features are normalized, and categorical features are converted with dummy variables. All variables in Tables 2.1 and A.1 are provided to the model.

Outcomes Our primary outcome of interest is in-ICU mortality. Secondary outcomes indicating decompensation are vasopressor use and mechanical ventilation use. For each outcome, we define the time to event as the time to first outcome or censorship, where censorship corresponds to discharge from the ICU.


ICU stays from eICU database (n=200,859)
Excluded (n=183,469) non-pneumonia and bacterial pneumonia visits
Viral/ unspecified pneumonia patients (n=17,390)
Excluded (n=7,890) patients over 70, under 18, or not reported
Included (n=9,500)
Excluded (n=2,882) patients with surgery, stroke, intracranial hemorrhage, cancer, liver disease, renal failure, congestive heart failure
Included (n=6,618)
Excluded (n=530) ICU stays after the patient's first one
Included (n=6,088)
Excluded (n=2,471) patients discharged in the first 48 hours
Cohort used in study (n=3,617)
ICU stays from MIMIC database (n=46,476)
Excluded (n=41,904) non-pneumonia and bacterial pneumonia visits
Viral/ unspecified pneumonia patients (n=4,572)
Excluded (n=2,224) patients over 70 or age not reported
Included (n=2,348)
Excluded (n=945) patients with surgery, stroke, intracranial hemorrhage, disseminated intravascular coagulation, liver disease, renal failure, congestive heart failure
Included (n=1,403)
Excluded (n=436) ICU stays after the patient's first one and patients discharged in the first 48 hours
Included (n=967)
Excluded (n=30) patients under 18
Cohort used in study (n=937)

(a) eICU cohort selection

(b) MIMIC-III cohort selection

Figure 2.2: Inclusion and exclusion criteria for cohorts extracted from eICU and MIMIC. Disseminated intravascular coagulation was highly missing from eICU.

Methods

Lasso-Cox To predict patient survival, we use the Cox proportional hazards model with L1 regularization, referred to as *Lasso-Cox* (Simon et al., 2011). Lasso-Cox is chosen for its ease of interpretation and calculation, owing to its selection of sparse models.[1] For a patient with covariates $\mathbf{x} \in \mathbb{R}^d$, the predicted log hazard is $\beta^\top \mathbf{x}$, (higher hazard implies shorter survival time), where $\beta \in \mathbb{R}^d$ are coefficients that can be interpreted as log hazard ratios. L1 regularization $\lambda \sum_{j=1}^{d} |\beta_j|$ is used to encourage sparsity in β, where $\lambda > 0$ is a user-specified hyperparameter.

Evaluation Metrics To evaluate model performance, we consider concordance and calibration. *Concordance* (c-index) is a common measure of goodness-of-fit in survival models (Harrell and al., 1982), defined as the fraction of pairs of subjects whose survival times are correctly ordered by a prediction algorithm, among all pairs that can be ordered. Confidence intervals are computed using 1000 bootstrapped samples. We evaluate *calibration* by plotting the Kaplan-Meier observed survival probability versus the predicted survival probability. We construct our calibration plots (Figure 2.4) (Xiao et al., 2016) with 1000 bootstrap resamplings for internal calibration. Both internal and external calibrations use 5 groups for 7 days.[2]

[1] We also tried the Cox model with elastic-net regularization (combined L1 and L2 regularization) but found little to no gain in cross-validation concordance.

[2] We plot at day 7 instead of 30 because censorship level is too high beyond a week.

Table 2.1: Demographics and outcomes of patients with viral or unspecified pneumonia in eICU and MIMIC-III cohorts. Data are median (Q1-Q3) or count (% out of n).

	Variable	eICU (n = 3617)	MIMIC (n = 937)
Demographics	Age, years	58.0 (48.0-64.0)	54.5 (44.1-62.7)
	18-30	225 (6.2%)	83 (8.9%)
	30-39	277 (7.7%)	94 (10.0%)
	40-49	500 (13.8%)	159 (17.0%)
	50-59	1064 (29.4%)	281 (30.0%)
	60-70	1546 (42.7%)	320 (34.2%)
	Male	1949 (53.9%)	542 (57.8%)
	Female	1663 (46.0%)	395 (42.2%)
Out.	Deceased	270 (7.5%)	94 (10.0%)
	Vasopressors administered	589 (16.3%)	389 (41.5%)
	Ventilator used	1835 (50.7%)	758 (80.9%)

Experimental Setup The eICU cohort is divided into a training set (70% of the data, n=2537) and test set (30%, n=1080). The eICU training set is used for model development, whereas the eICU test set and entirety of the MIMIC cohort are used for model evaluation. Throughout our evaluation, we compare our risk score (PEER) to three pneumonia risk scores: CURB-65 (W et al., 2003), PSI/PORT (Fine et al., 1997), and SMART-COP (Charles et al., 2008); and one COVID-19 risk score: GOQ (Gong et al., 2020b).

Model selection We select λ via 10-fold cross validation and grid search on the eICU training set to maximize concordance subject to sufficient sparsity. We observe that $\lambda = 0.01$ gives the best trade-off between concordance (0.73) and number of features selected (18), as a 0.01 increase in concordance corresponds to 10 additional non-zero features. To check the stability of this hyperparameter choice, we impute our data using ten random seeds and run 10-fold cross validation on the resulting datasets. Across all runs, $\lambda = 0.01$ achieves concordance of approximately 0.73 and selects similar features and coefficients. Additional details about grid search, the concordance and sparsity tradeoff, and robust selection of coefficients can be found in Appendix A. Code for data extraction and all model results is available at https://github.com/hlzhou/peer-score.

Results

The hazard ratios from Lasso-Cox with $\lambda = 0.01$ are displayed in Table 2.2. For easy calculation of the PEER score, we also provide a nomogram (Figure 2.3)[3].

[3] To compute risk, look up a patient's values in the nomogram, match it to points listed across the top, add them up, and look up the total in the scale across the bottom.

The PEER score achieves concordance greater than or comparable to that of existing risk scores on all datasets (Table 2.3). On the eICU test set, PEER achieves the highest concordance among the risk scores, 0.77. On MIMIC, the maximum concordance degrades to 0.66, achieved by PEER and SMART-COP. The PEER calibration curves (Figure 2.4) show one high risk group separate from low risk groups. While predicted survival of the high risk group is overestimated in the training set, it is within confidence intervals in both test sets.

We define low and high risk subpopulations by thresholding our model's predicted risks on the training set at the 90th percentile. Each group's Kaplan-Meier survival curves are plotted over a 30-day period (Figure 2.5). For the first week, the low and high risk curves are clearly distinct (Figure 2.5), with respective survival proportions 0.68 and 0.95 on eICU test, and 0.75 and 0.95 on MIMIC. Beyond the first week, censorship grows quickly and there is less data, resulting in increased uncertainty. Compared to low and high risk curves derived from related risk scores, those of the PEER score are the most separated (Appendix A). Secondary indicators of decompensation (i.e. ventilator and vasopressor use) are also more common in the high risk group than the low risk group (Figure 2.6).

Table 2.2: Hazard ratios (HR) for the Lasso-Cox model, i.e. the PEER score. HR and 95% confidence intervals (CI) are reported on normalized data. Means and standard deviations used for scaling are included for reference.

Feature	**HR (95% CI)**	mean	std. dev.
Age (years)	1.22 (1.04 – 1.43)	54.5	12.5
Heart rate (beats per minute)	1.13 (0.984 – 1.3)	89.4	17.8
Systolic blood pressure (mmHg)	0.928 (0.755 – 1.14)	122	22
Diastolic blood pressure (mmHg)	0.996 (0.745 – 1.33)	67.7	15.1
Mean arterial pressure (mmHg)	0.926 (0.673 – 1.27)	83.7	17.9
Glasgow Coma Scale	0.93 (0.803 – 1.08)	11.3	3.26
White blood cells (thousands/μL)	0.984 (0.871 – 1.11)	12.9	8.91
Platelets (thousands/μL)	0.924 (0.79 – 1.08)	208	108
Red blood cell dist. width (%)	1.24 (1.08 – 1.43)	15.8	2.47
Neutrophils (%)	0.972 (0.853 – 1.11)	79.1	13
Blood urea nitrogen (mg/dL)	1.07 (0.937 – 1.23)	25.1	19.5
Aspartate aminotransferase (units/L)	1.12 (1.06 – 1.18)	143	774
Direct bilirubin (mg/L)	1.03 (0.935 – 1.13)	0.385	0.816
Albumin (g/dL)	0.954 (0.82 – 1.11)	2.65	0.636
Troponin (ng/mL)	1.06 (0.985 – 1.14)	1.07	3.85
Prothrombin time (sec)	1.05 (0.909 – 1.2)	16.6	6.75
pH	0.856 (0.75 – 0.977)	7.38	0.0713
Arterial oxygen saturation (mmHg)	0.787 (0.723 – 0.856)	95.8	4.12

Table 2.3: Concordances (and 95% confidence intervals) of the PEER score, CURB-65, PSI/PORT, SMART-COP, and GOQ.

Score	Train eICU	Test eICU	MIMIC
PEER (ours)	**0.77 (0.72 - 0.81)**	**0.77 (0.69 - 0.83)**	**0.66 (0.57 - 0.74)**
CURB-65 (W et al., 2003)	0.66 (0.61 - 0.70)	0.62 (0.55 - 0.69)	0.59 (0.52 - 0.66)
PSI/PORT (Fine et al., 1997)	0.71 (0.66 - 0.76)	0.71 (0.63 - 0.78)	0.62 (0.55 - 0.69)
SMART-COP (Charles et al., 2008)	0.69 (0.64 - 0.73)	0.73 (0.67 - 0.80)	**0.66 (0.59 - 0.72)**
GOQ (Gong et al., 2020b)	0.67 (0.63 - 0.71)	0.62 (0.54 - 0.70)	0.58 (0.50 - 0.66)

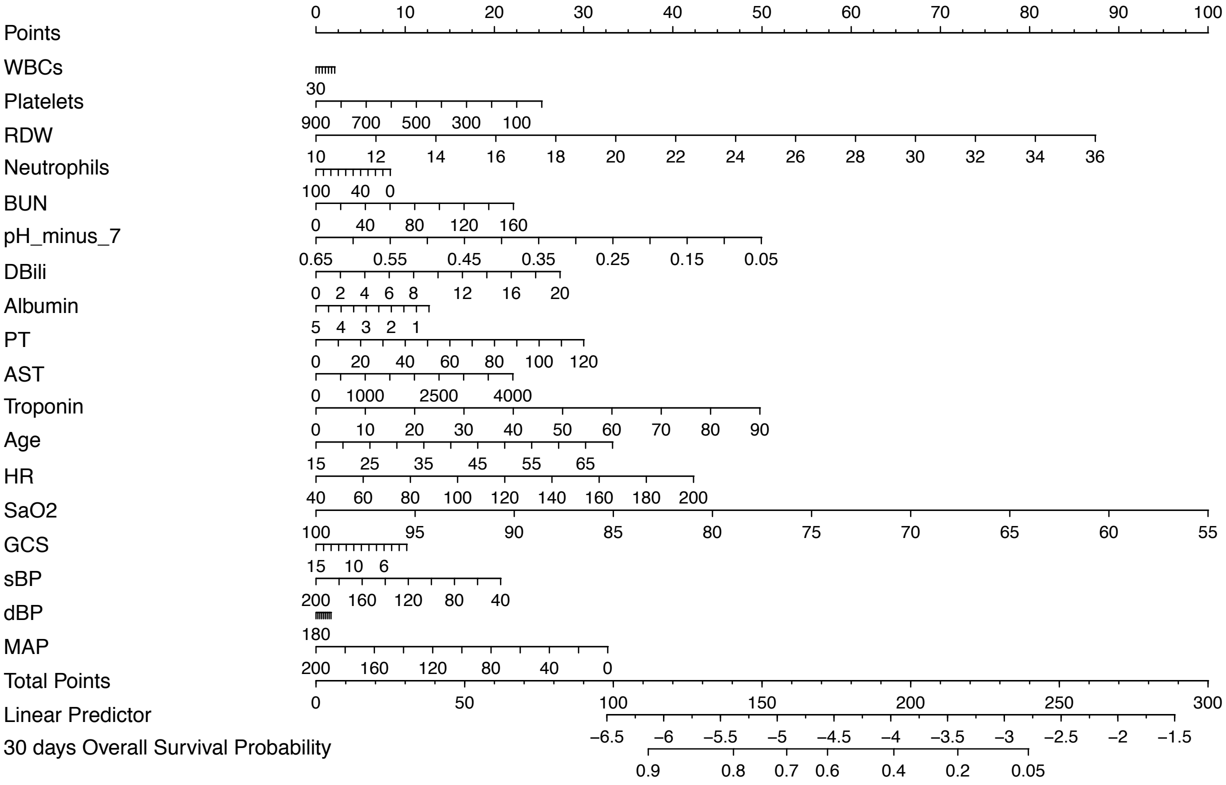
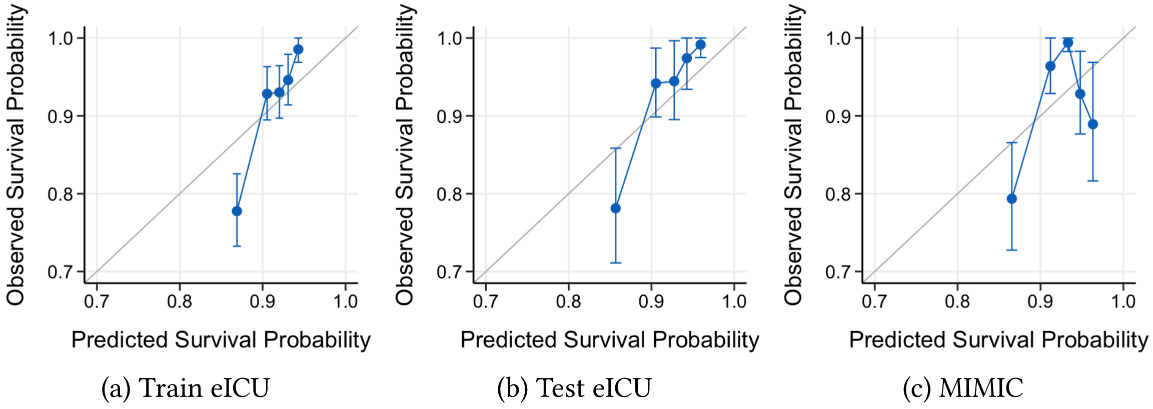


Figure 2.3: Nomogram for manual calculation of the PEER score.



(a) Train eICU

(b) Test eICU

(c) MIMIC

Figure 2.4: Calibration plots with 95% confidence intervals.

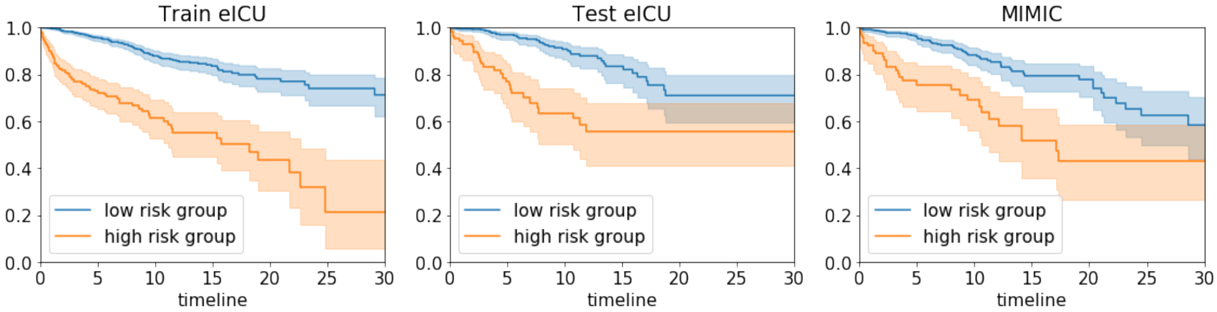
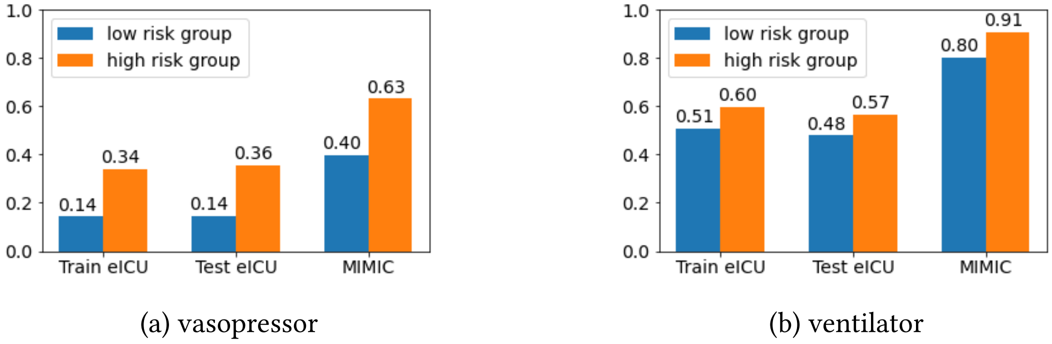


Figure 2.5: Kaplan-Meier survival curves of high vs. low risk groups in train eICU, test eICU, and MIMIC. Shaded regions are the 95% confidence intervals.



(a) vasopressor

(b) ventilator

Figure 2.6: Proportion of each subgroup that received vasopressors or ventilators.

Discussion

The PEER score achieves greater or comparable concordance to baselines on the eICU (in-domain) and MIMIC (out-of-domain) test sets. Lasso-Cox selects 18 features, making for easy computation. Qualitatively, the score is consistent with clinical intuition. SaO2, associated with poorer oxygenation status, is predictive of decompensation. Old age is predictive of death. Red blood cell distribution width, associated with expanded release of immature red blood cells in response to insufficient oxygen delivery to tissues, is also a strong risk factor for death with COVID-19 (Gong et al., 2020a). However, the hazard ratios themselves should be interpreted with caution as three variables (pH, prothrombin time, and age) violate the proportional hazards assumption, and L1 regularization shrinks coefficients towards 0.

Stratifying each cohort into high and low risk subpopulations based the PEER score, we observe a clear separation in their survival curves (Figure 2.5) across all three datasets. Additionally, secondary indicators of decompensation (e.g. vasopressor and ventilator use) are more prevalent in the high risk group (Figure 2.6). Calibration plots for PEER also show a high risk group separated from the rest (Figure 2.4). While the survival probability of the high risk group is overestimated on the eICU training set, it is within error bars on all test sets.

For ECMO allocation, practically, accurate *ranking* of risk, as measured by concordance, may be more important than the precise probabilities predicted. The PEER score outperforms other risk scores on the eICU test set, but there is a decline in performance on the MIMIC test set, and the performance of PEER becomes comparable to that of SMART-COP. One possible reason for this decline is that in MIMIC, an important feature for PEER, the arterial oxygen saturation (SaO2), has 72.6% missingness. In contrast, it has 1.5% missingness in eICU. This demonstrates the importance of thinking critically about how our risk score, which was trained on the eICU cohort and depends on 18 specific features, generalizes to the population to which the score is being applied.

Limitations and Future Work Importantly our cohort is defined not by COVID-19 positive pneumonia patients but instead by viral or unspecified pneumonia patients who are ECMO-eligible. While our risk score demonstrates good discriminative ability and is interpretable, there are several additional decision-making considerations beyond the scope of this paper. Clinicians interested in applying the risk score to COVID-19 pneumonia should consider how representative this population is of their own. Because ECMO is a constrained resource, there are also ethical questions about who should get treatment. This risk score does not attempt to address these questions, but simply provides relevant information to those making such decisions. More broadly, we hope to provide this risk score as a potential resource for future SARS-like diseases that require ECMO consideration.

2.2 Unpacking U.S. COVID-19 Fatality Rates in 2020

Introduction

In this work, we consider the tumultuous first year of the COVID-19 pandemic, focusing in particular on the time range from April 2020 to December 2020. At this time, infections were spreading rapidly throughout the United States, but COVID-19 vaccines were not yet available to the general public. This was a time of high uncertainty and constant change, with disease outbreaks expanding and contracting; social distancing mandates tightening and loosening; testing capacity (mostly) increasing (Wu, 2020); and treatments protocols evolving. Public officials, clinicians, and business leaders tried to grasp the the rapidly unfolding situation, often looking to publicly reported aggregate data to inform decisions about lockdown measures, allocation of hospital resources, and corporate policies.

Consider the two most widely reported statistics, cases and deaths. Cases peak for the first time in April 2020 (nearly $32,000$ daily cases), peak again with more than twice as many cases in a second wave in July 2020 (nearly $67,000$), and yet again in a much larger third wave in December 2020 (nearly $230,000$) (Figure 2.7, left panel). However, reported deaths appear to tell a contradictory story concerning the relative severity of the three waves (Figure 2.7, middle panel), with the second wave being the smallest. Dividing deaths by cases, *the reported case fatality rate (CFR) fell dramatically* after the first wave, from nearly 7.9% at the height of the first wave in mid-April to the 0.7%–2.3% range since July. (Figure 2.7, right panel).

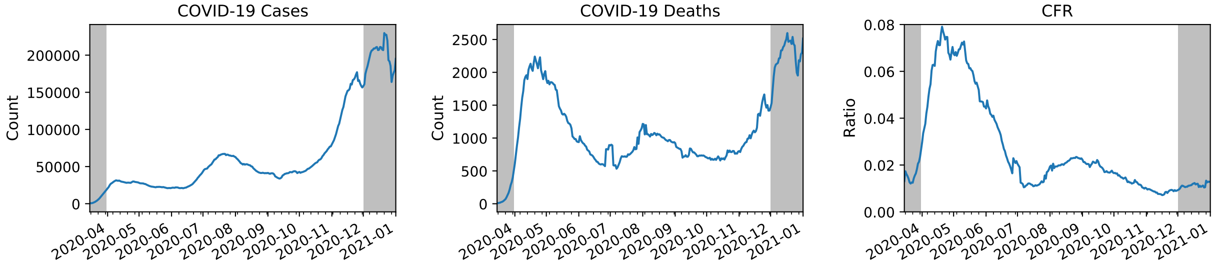


Figure 2.7: Trailing 7-day averages of cases, deaths, and case fatality rate from April 1st to December 1st, using national data available via USAFacts in the COVIDcast API (Project, 2020).

In a White House briefing on July 27th, 2020, President Trump attributed this drop in CFR to treatment improvements, stating: "Due to the medical advances we've already achieved and our increased knowledge in how to treat the virus, the mortality rate for patients over the age of 18 is 85 percent lower than it was in April" (Trump, 2020). While this would be an impressive statistic, as we show, this estimate is heavily confounded by factors unrelated to treatment improvements. So, *what explains the movement (and apparent overall decline) in case fatality rate over the course of the pandemic?* Several plausible explanations have been floated, with academic and public discourse centering around the following hypotheses:

($H1$) The *age distribution* of infected patients shifted, heavily altering CFR due to higher risk among the elderly (Thompson, 2020; Whet, 2020; Horwitz et al., 2020; Dennis et al., 2020).

($H2$) *Increases in testing capacity* have driven down the CFR due to a rising number of tests catching milder cases (Fan et al., 2020; Madrigal and Moser, 2020; Spychalski et al., 2020).

($H3$) Apparent shifts in CFR are artifacts due to the *delay between detection and fatality* (Thompson, 2020; Madrigal, 2020; Spychalski et al., 2020).

($H4$) *Treatment has improved* with growing doctor experience and new therapeutics (Levy, 2020; Horwitz et al., 2020; Beigel et al., 2020; Group, 2020; Self et al., 2020).

($H5$) The disease itself is *mutating*, leading to changes in the actual infection fatality rate over time (Pachetti et al., 2020; Fan et al., 2020).

($H6$) Social distancing has reduced the *viral load* that individuals are exposed to, resulting in milder infections (Zein et al., 2020; Pachetti et al., 2020; Piubelli et al., 2020).

Note that H1–H3 can be misleading if not sufficiently accounted for. If there are large differences in fatality between different age groups, if the age distribution shifts (H1), it is even possible to observe an *overall decrease* in CFR despite *increasing* CFRs in every age group (Simpson's paradox). Additionally, testing ramp-up (H2) and delays between detection and fatality (H3) can cause the behavior of case fatality rate to diverge substantially from the behavior of the true infection fatality rate. Thus, CFR can be a poor proxy for actual infection fatality rate.

On the other hand, the last three phenomena—improved treatments, disease mutation, and changing viral load—correspond to actual reductions in mortality and could be grounds for policy changes. This work *demonstrates how given accurate, sufficiently granular data, H1–H3 ("artifacts") can be accounted for to attempt to quantify true improvements in treatment* (H4). We note, however, that without additional data, H5 and H6 cannot be separated from H4.

In particular, we argue that complete and accurate *age-stratified, line-level hospitalization data* is pivotal for distinguishing true improvements from artifacts. Age stratification allows us to adjust for H1, and line-level data allows us to match lagged outcomes such as death with the corresponding case it originated from, thus avoiding H3. Hospitalizations should be less influenced by testing capacity than cases (allowing us to bypass H2), and compared to the general population, testing among the inpatient population was relatively thorough throughout the course of the pandemic. While there may have been changes in admitting criteria at the very worst moments (Phua et al., 2020; Cohen et al., 2020) (e.g., when New York hospital demand exceeded capacity in late March), for the most part, criteria for inpatient hospitalization is relatively consistent across time periods. Additionally, in the study time period (April 2020 to December 2020), treatment improvements mostly targeted hospitalized COVID-19 patients.

Our analysis yields several important observations: (i) large increases in testing do occur between the waves but do not explain them away; (ii) since age distributions shifted substantially between the first and second waves (and have fluctuated since), age must be accounted for in order to separate out the effects of treatment from age shift; (iii) between the first and second waves age-stratified HFRs improved substantially in the national data (with HFR decreasing by as little as 27% in the 80+ age group and as much as 37% in the 30-39 age group), but were relatively unchanged in Florida (with a slight *increase* in HFR by as little as 2.9% in the $80+$

age group and as much as 13% in the 60-69 age group); (iv) by December 1st, both Florida and national data suggest significant decreases in HFR since April 1st—at least 17% in Florida and at least 55% nationally in every age group; and (v) comprehensive age-stratified hospitalization data is of central importance to providing situational awareness during the COVID-19 pandemic. As far as we are aware, this is the largest national-scale (588,126 hospitalizations, 10.3 million cases) data-driven analysis to quantify and account for all three artifacts (age distribution shift, increased testing, and detection-to-fatality delay) when estimating treatment improvements. To allow users to apply our analyses to time ranges, states, and demographics of interest, we release an interactive web application at acmi-lab.org/unpack_cfr.

Related Work

Several COVID-19 treatments were developed over the study time range (April 1st to December 1st), each underwent randomized controlled trials testing for its individual efficacy. Dexamethosone resulted in a lower 28-day mortality among COVID-19 inpatients receiving respiratory support (Group, 2020). Remedisivir was associated with shortened recovery time among adults hospitalized with lower respiratory tract infection (Beigel et al., 2020; Madsen et al., 2020). Clinical trials for hydroxychloroquine (Self et al., 2020; Horby et al., 2020) and convalescent plasma (Agarwal et al., 2020b) found no positive results in prevention of disease progression or mortality. In November, monoclonal antibody treatments bamlanivimab and the combination therapy casirivimab and imdevimab were approved for emergency use authorization (U.S. FDA, 2020). These therapies, unlike dexamethosone and remdesivir, are not recommended for inpatients (Dyer, 2020), but were shown to have great benefits in outpatients likely to progress to severe COVID-19 (for bamlanivimab) (Chen et al., 2021), and in patients who have not yet mounted their own immune response or have high viral load (for casirivimab and imdevimab) (Regeneron, 2020). Note that monoclonal antibodies were only approved for emergency use within the last month of our study time range, before therapeutic distribution had ramped up (HHS, 2021). More recently in December (outside of our study time range), the first coronavirus vaccines from Pfizer-BioNTech (Food and (FDA), 2021b) and Moderna (Food and (FDA), 2021a) were approved for emergency use, with the Pfizer-BioNTech vaccine clinically proven to achieve 95% efficacy (Polack et al., 2020) and the Moderna vaccine, 94.1% efficacy (Baden et al., 2020). While these clinical trials have evaluated the effects of specific treatments in their identified target populations, our work studies the broader impacts of treatment improvements over time at a larger national scale.

To get a holistic sense of improvements over time several studies have examined CFRs. In a study of 53 countries, all but ten were found to have lower CFRs in the second wave compared to the first (Fan et al., 2020). However, as delineated in our introduction, confounding factors such as shifting age distribution (H1), testing capacity (H2), and detection-to-death lags (H3) can lead to misleading interpretations of the CFR (Whet, 2020; Horwitz et al., 2020; Fan et al., 2020; Madrigal and Moser, 2020; Thompson, 2020; Dennis et al., 2020; Madrigal, 2020; Spychalski et al., 2020). For example, when comparing CFR by age group in Italy and China, Onder et al. (2020) suggested variation in testing strategies as a possible explanation for discrepancies. In study of

COVID-19 cases in Germany, Stafford attributed an apparent discrepancy between cases and deaths to shifting age distribution, testing capacity, or true effectiveness of government-issued directives. While these country-level studies identify the three “artifacts” (H1-H3) as limitations of interpreting the CFR, none explicitly account for them.

To account for changes in testing capacity, we examine hospitalization data. While (as far as we are aware) no nation-wide studies in the U.S. account for all three artifacts, some hospital systems have controlled for them by conducting age-stratified cohort studies. Among 5,121 hospitalized COVID-19 patients in a single New York health system, Horwitz et. al. demonstrated that after adjusting for age, sex, ethnicity, and other clinical factors, HFR between March 1st and June 20th decreased but not as much as observed before adjusting for these factors (Horwitz et al., 2020). In another New York hospital system, Mehta et. al. demonstrated that cancer and older age were associated with increased risk of case fatality, finding no significant associations between race and mortality or gender and mortality (Mehta et al., 2020). In a study conducted among 21,082 COVID-19 patients admitted to 108 English critical care units between March 1st and June 27th, mortality risk in mid-April and May was found markedly lower than earlier in the pandemic even after adjusting for age, sex, ethnicity, comorbidities, and geographic region (Dennis et al., 2020). While these studies provide thorough estimates of mortality for their respective regions during their specific time periods, we analyze data over a longer time range and larger scale (588,126 hospitalizations, 10.3 million cases) that purportedly captures all of Florida and most of the United States.

Our data does not contain information about viral mutations and viral loads (H5 and H6). However, a few studies have begun to investigate their impact. The B.1.1.7 and B.1.351 variants of COVID-19 were first reported in the U.S. at the end of December 2020 and January 2021, respectively (CDC, 2021). While it is unclear how these variants will ultimately impact HFRs, recent studies indicate that vaccines may be more effective against B.1.1.7 than B.1.351 (Liu et al., n.d.; Wu et al., 2021). Regarding social distancing precautions reducing viral load, in a study across seven countries, declining CFR was found to be correlated with strict lockdown policies and widespread PCR testing (Pachetti et al., 2020). At a hospital system in northern Italy, Piubelli et. al. found that among patients diagnosed with COVID-19 in their emergency room, the proportion of patients requiring intensive care decreased over time, also having lower viral load (Piubelli et al., 2020). Our analysis does not attempt to separate out the effects of viral mutations (H5) and changing viral loads (H6), but we note that these are factors that can affect the true infection fatality rate, and therefore can be reflected in our estimates as well.

Methodologically, prior studies on the COVID-19 fatality have employed logistic regression (Horwitz et al., 2020), Cox proportional hazards (Dennis et al., 2020), and propensity matching (Mehta et al., 2020) to adjust for age and other comorbidities. While logistic regression and propensity matching can quantify risk of death averaged over their study time period, we are interested in the evolution of fatality risk over time. While the standard Cox proportional hazards does model risk over time, it assumes a simple linear form between the covariates and the log hazard. Without making this assumption, we leverage techniques in time series literature to reduce noise in the raw signal, and compute uncertainty around the estimates at any given time. Assuming smoothness in the true underlying trend, the moving average can obtain a

Table 2.4: Demographics and outcomes of the Florida and national cohorts. Data are provided as counts (percentage). For Hospitalized and Died, the recoded "No" category is merged from the original "No", "Unknown", and "Missing" categories. Unknown corresponds to checking an "unknown" box in the reporting form, and missing corresponds to leaving the question empty.

		Florida	National
Demographics	COVID-19 Cases	1,004,818	10,332,725
	Age		
	0-9	37,153 (3.7%)	388,867 (3.8%)
	10-19	91,887 (9.1%)	1,045,408 (10.1%)
	20-29	192,790 (19.2%)	2,021,007 (19.6%)
	30-39	172,027 (17.1%)	1,699,260 (16.4%)
	40-49	153,717 (15.3%)	1,546,212 (15.0%)
	50-59	149,530 (14.9%)	1,485,318 (14.4%)
	60-69	102,477 (10.2%)	1,043,294 (10.1%)
	70-79	61,436 (6.1%)	583,837 (5.7%)
	80+	42,696 (4.2%)	453,229 (4.4%)
	Unknown	1,105 (0.1%)	66,293 (0.6%)
	Gender		
	Female	517,640 (51.5%)	5,297,936 (51.3%)
	Male	482,227 (48.0%)	4,861,508 (47.0%)
	Missing	0 (0.0%)	17,071 (0.2%)
	Unknown	4,951 (0.5%)	80,093 (0.8%)

		Florida	National
Outcome	Hospitalized		
	Yes	56,673 (5.6%)	588,126 (5.7%)
	No (recoded)	948,145 (94.4%)	9,744,599 (94.3%)
	No	538,428 (53.6%)	3,989,850 (38.6%)
	Unknown	402,616 (40.1%)	1,593,887 (15.4%)
	Missing	7,101 (0.7%)	4,160,862 (40.3%)
	Died		
	Yes	21,028 (2.1%)	199,677 (1.9%)
	No (recoded)	983,790 (97.9%)	10,133,048 (98.1%)
	No	0 (0.0%)	4,847,491 (46.9%)
	Unknown	0 (0.0%)	1,185,926 (11.5%)
	Missing	983,790 (97.9%)	4,099,631 (39.7%)

better estimate of the trend than the raw signal (Eshel, 2012). To quantify uncertainty in time series, the moving block bootstrap technique was developed in lieu of standard bootstrapping targeting independent and identically distributed observations. In this technique, the time series is chunked into blocks to reduce dependence among them. This reduced dependence is hard to ensure, however, thereby suggesting a strategy between model-based and block resampling. To reduce much of the dependence between original observations, one can "pre-whiten" the data by fitting a model to the data, and computing the residuals. Instead of block resampling the original dependent series, the residuals can be resampled, and added back to the model estimates. This intermediate solution, termed post-blackening, has been shown to work more consistently in practice (Davison and Hinkley, 1997).

Methods

Data Description We center our analysis on (1) state-level COVID-19 Case Line Data made available by the Florida Department of Health (FDOH)(Florida Department of Health, 2020) and (2) national-level COVID-19 Case Surveillance Data made available by the United States Centers for Disease Control and Prevention (CDC) (CDC, COVID-19 Response, 2020). Both datasets are line-level, including date of detection, demographics (including age and gender), and indicators of eventual outcomes for each case (Table 2.4). In our defined cohorts, all COVID-19 cases are confirmed with a positive PCR lab result. Each FDOH case is marked with the date it was confirmed, and each CDC case is marked with the date it was reported. Overall, there are 1,004,818 confirmed cases in the FDOH data, and 10,332,725 in the CDC data.

Signal Smoothing For each date, we compute the 7-day lagged averages for COVID-19 cases, hospitalizations, and deaths. From this point in the paper on, whenever we discuss these quantities or calculate CFR and HFR based on them, unless otherwise stated, we are referring to the smoothed signal. Thus, in both the FDOH and CDC data, we collect data extending back to March 26th in order to conduct our analyses on the April 1st to December 1st time range.

Separating Artifacts from True Improvements We argue that three main phenomena fuel an "artificial" decrease in CFR: increased testing capacity (H1), shifting age distributions (H2), and delays between detection and fatality (H3).

Since testing (H1) is not included in the FDOH or CDC data, we pull in data from The COVID Tracking Project (Meyer and Madrigal, 2020) to quantify increased testing capacity in Florida and nationally. We plot 7-day lagged averages of tests administered and the positive test rates. To avoid artifacts from increased testing, we examine changes in *HFRs* rather than CFRs.

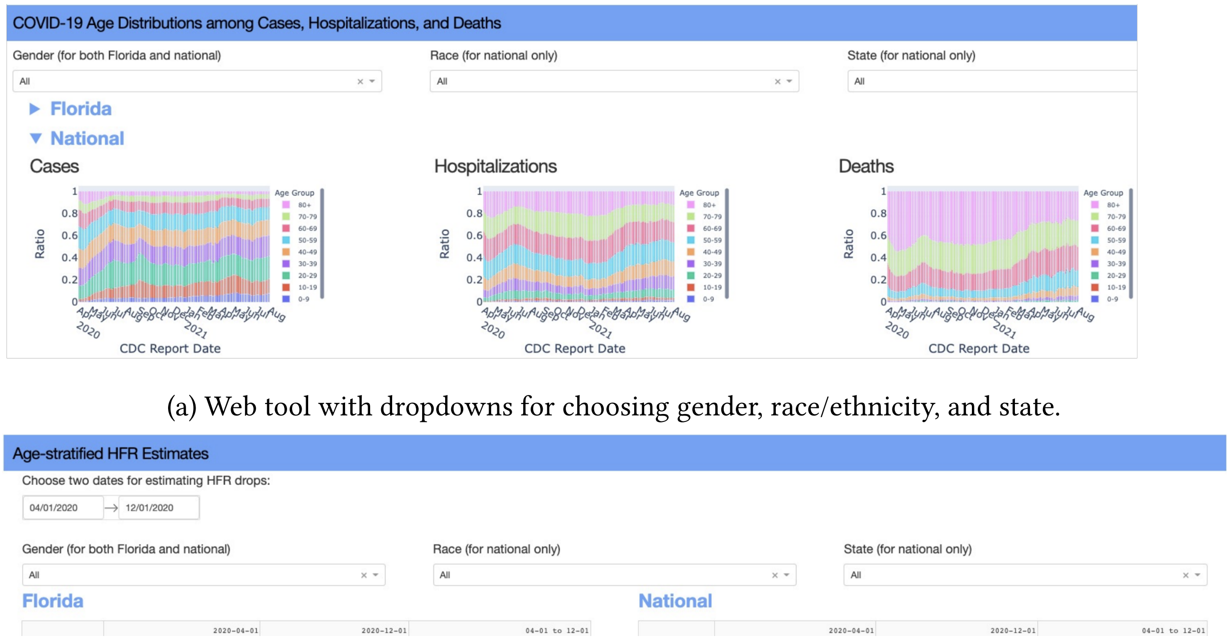
To establish and account for shifting age distributions (H2), we examine cases, hospitalizations, and deaths stratified by age groups: 0-9, 10-19, 20-29, 30-39, 40-49, 50-59, 60-69, 70-79, and 80+. Naturally, age stratification reduces the amount of data for each estimate, so we omit HFR estimates that are based on fewer than two deaths.

Finally, to account for delays between detection and fatality (H3), we take advantage of the line-level nature of the data in order to perform a cohort-based analysis. For each date, we extract the cohort of individuals confirmed positive on that date, as well as whether those individuals *eventually* died or were hospitalized. By contrast, publicly reported case fatality rates are typically not cohort-based (Thompson, 2020; Madrigal, 2020; Spychalski et al., 2020)—the patients whose deaths are reported in the numerator are not in general the same patients whose confirmed infections show up in the denominator. Because case confirmation tends to precede reported deaths, these signals tend to be misaligned and are subject to fluctuation, even if the actual case fatality rate were fixed (so long as incidence does change). Line-level data enables us to circumvent this problem.

Taking the above three adjustments into account, our primary quantity of interest for treatment improvements is the age-stratified HFR. For the rest of the paper, we define CFR and HFR at day t as follows:

$$\text{CFR}_t = \frac{\text{cases at day } t \text{ that eventually die}}{\text{cases at day } t}$$
$$\text{HFR}_t = \frac{\text{cases at day } t \text{ that eventually get hospitalized and die}}{\text{cases at day } t \text{ that eventually get hospitalized}}$$

Quantifying True Improvements Thus far, news and academic sources have highlighted three main "true improvements": improvements in treatment (H4), disease mutations (H5), and reduced viral loads due to social distancing (H6). We seek to quantify treatment improvements (H4) by computing the drop in HFR.



COVID-19 Age Distributions among Cases, Hospitalizations, and Deaths

Gender (for both Florida and national): All

Race (for national only): All

State (for national only): All

▶ Florida

▼ National

(a) Web tool with dropdowns for choosing gender, race/ethnicity, and state.

Age-stratified HFR Estimates

Choose two dates for estimating HFR drops: 04/01/2020 → 12/01/2020

Gender (for both Florida and national): All

Race (for national only): All

State (for national only): All

Florida

	2020-04-01	2020-12-01	04-01 to 12-01

National

	2020-04-01	2020-12-01	04-01 to 12-01

(b) Web tool with date range selector for estimating HFR drop, i.e. treatment improvement.

Figure 2.8: Interactive choices on the webapp for cohort selection to produce visualization and treatment improvement estimation.

Estimation using Block-Bootstrap and Cubic Splines We use a cubic spline to fit the trend of the 7-day lagged average HFRs, and use a moving block-bootstrapping technique with post-blackening (Davison and Hinkley, 1997) to estimate uncertainty around this trend. As described in the related work, 7-day lagged averages provide better estimate of trend by assuming smoothness rather than a specific functional form. Block-bootstrapping with post-blackening enables us to estimate uncertainty around this trend, with a weaker assumption than the standard i.i.d. assumption. We use a 7-day block size, based on the length of the time series (Shalizi, 2013) and our observations that reporting follows a weekly cadence. After block-bootstrap resampling the residuals, the residuals are added back to the cubic spline, creating the replicates needed for estimating HFR with uncertainty. A visual walkthrough of this procedure is in the "HFR estimation" section of our web tool.

Visualization Tool We publish an interactive web tool, available at acmilab.org/unpack_cfr, for dynamically applying our analyses to any demographic or date range of interest. On both FDOH data and CDC data, it displays plots for aggregate and age-stratified cases, hospitalizations, and deaths over time (Figure 2.10); plots for age distributions of cases, hospitalizations, and deaths over time (Figure 2.11); and estimates of age-stratified HFR as well as the change between two user-provided dates (Table 2.5 and 2.6). For a cohort of interest, the user can select gender, race/ethnicity, and state from dropdown menus in the web interface (Figure 2.8a). For HFR estimates, the user can use a date selector to obtain new estimates for their date range of interest (Figure 2.8b).

Results

In both the FDOH and the CDC data, one can discern three waves of COVID-19 cases, peaking in (1) mid-April, (2) mid-July, and (3) towards December (Figure 2.10).

Increased Testing Between April 1st and December 1st, testing increases significantly, by approximately 964% in Florida and 1080% nationally (Figure 2.9, left and middle panel). Florida observes a spike in testing near the second peak, whereas national testing rises more smoothly. However, we note that these increases in testing cannot fully account for the peaks. Despite increased testing inflating the number of cases, we still observe two peaks in positive test rates in April and July (Figure 2.9, right panel), and a surge in positive test rates towards December.

Cases Across all age strata, as measured by cases, Florida's second wave is the most severe (Figure 2.10a, left panel) out of the three waves. In aggregate, it has approximately 1153% more cases than in the first peak and 46% more cases than in the ongoing third wave (Figure 2.10a, left panel). In contrast, nationally the third wave has substantially more cases than the first two peaks—392% more than the first peak and 150% more than the second peak (Figure 2.10b, left panel). Also, note that the relative jump in cases between the first two peaks is 96%, much less than the 1153% jump seen in Florida. This could be due to a combination of the spike in Florida's testing in the second peak, as well as variation in the trajectories of different states (e.g. the populous state of New York was particularly hard-hit in the first wave).

Hospitalizations and Deaths Overall, hospitalizations and deaths corroborate the story told by positive test rate (Figure 2.10a, center and right panels). In Florida, hospitalizations and deaths indicate a more severe second peak than first peak, though the contrast in peak size is not as dramatic as in the plot of cases. By contrast, in the national data, the second peak is *smaller* than the first peak, which is opposite to the trend seen in Florida cases. Much of the discrepancies of trends seen in cases versus in hospitalizations and deaths are likely attributable to increases in testing (Figure 2.9). Towards the third wave in December, Florida hospitalizations and deaths are at similar levels to that of the first wave. Nationally, the ongoing third wave appears to be worse than in the second wave.

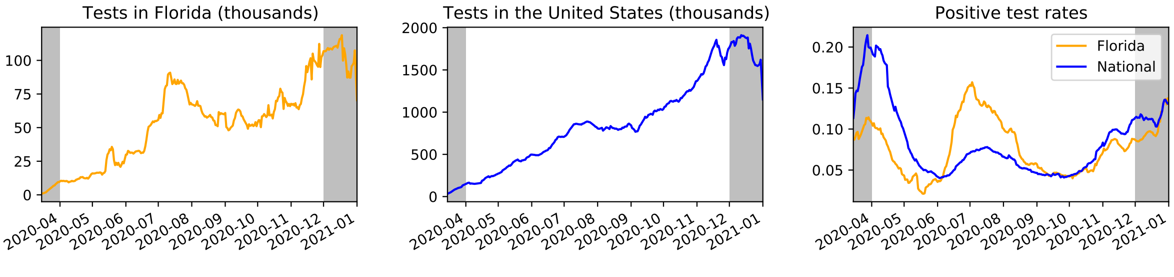
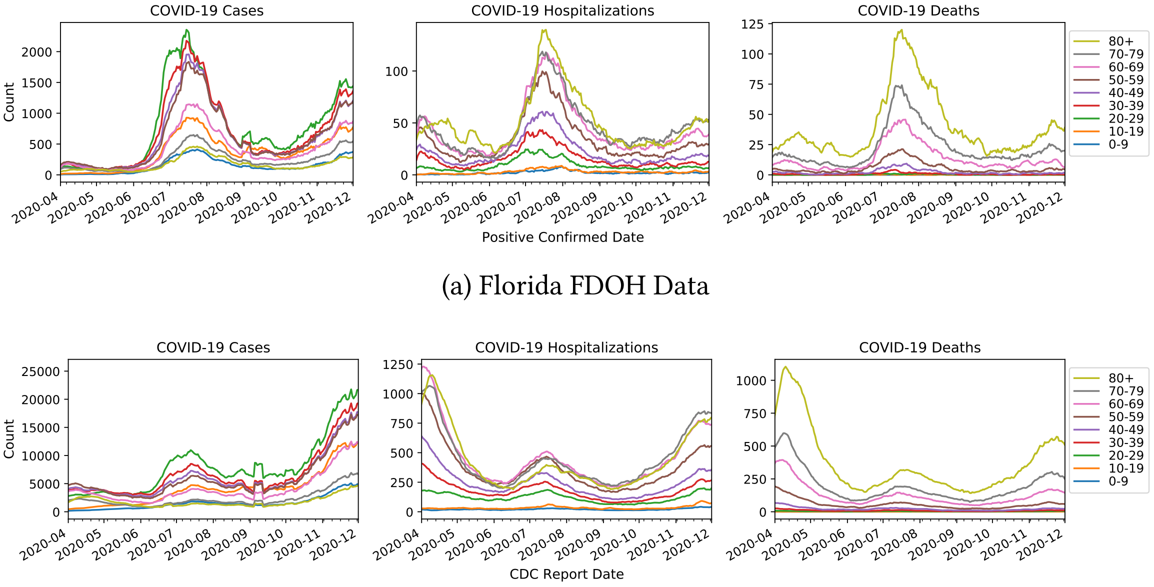


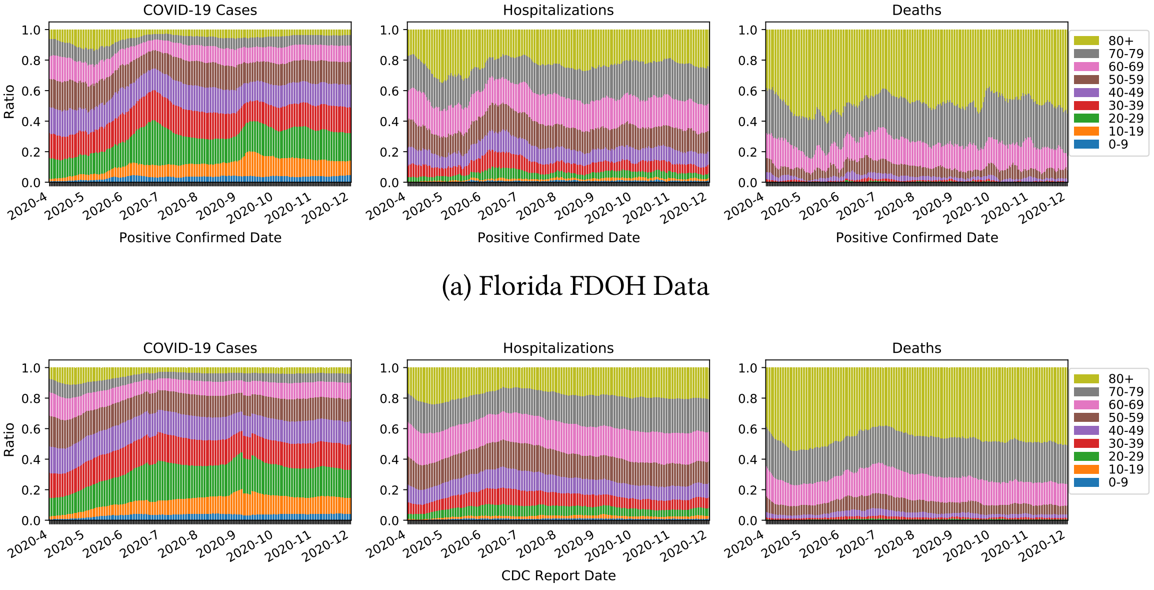
Figure 2.9: COVID-19 positive test rates (right) and tests (left and middle) for Florida and the United States, calculated using 7-day trailing averages, based on the COVID Tracking Project (Meyer and Madrigal, 2020). Positive test rate is calculated by dividing new positives by total new tests on each day. Data outside the April 1st to December 1st study period is grayed out.



(a) Florida FDOH Data

(b) United States CDC Data

Figure 2.10: Age-stratified cases, (eventual) deaths, and (eventual) hospitalizations in Florida and in the U.S., by the date of first positive test result (Florida) and date of report to the CDC (U.S.). Note that the x axis is *not* the date of death or date of hospitalization.



(a) Florida FDOH Data

(b) United States CDC Data

Figure 2.11: Age distributions among Florida and national cases, (eventual) hospitalizations, and (eventual) deaths, by the date of first positive test result (Florida) and date of report to the CDC (U.S.), respectively.

Age Between the first two peaks, the age distribution of cases shifts substantially, with the median age in Florida falling from 51 to 40, and the median national age group falling from 50-59 to 30-39. After the second peak, the age distributions of cases, hospitalizations, and deaths continue to fluctuate. In September, younger cases increase, possibly related to the start of the school year (Figure 2.11, left panel). By December 1st, the Florida median age remains at 40 but the national median age group rises to 40-49. Older individuals comprise a disproportionate share of the hospitalization and death counts (Figure 2.11, middle and right panel).

Gender The gender ratios in each age group's cases, hospitalizations, and deaths appear relatively flat over time. Thus, in this paper we choose not to stratify by gender due to the reasonably small shift in the gender distribution over time, and practically to have more support in each group. However, we do provide this option in our web tool. Consistent with prior literature (Mehta et al., 2020), we find that as the age group increases so does the corresponding HFR (Tables 2.5 and 2.6). Measuring treatment improvements by HFR drop (computed as $\frac{\mathrm{HFR}_{new} - \mathrm{HFR}_{old}}{\mathrm{HFR}_{old}}$), we observe larger treatment improvements between April and December to be correlated with younger age (Table 2.6). Note, however, that the younger groups have small HFRs to begin with, so the opposite trend might appear when considering *absolute* rather than *relative* improvements. Additionally, the confidence intervals for HFR drops are wider for younger age groups.

Between the first two peaks (Table 2.5), the national age-stratified HFR estimates from block bootstrapping decrease by as little as 27% in the 80+ age group, and as much as 37% in the 30-39 age group. On the other hand, in Florida the age-stratified HFR actually *increases* in each age group by as little as 2.9% in the 80+ age group, and as much as 13% in the 60-69 age group. Note that the HFR changes between peak dates in Florida are an example of Simpson's paradox, where in each age group the HFR increase, but the aggregate HFR actually decreases by 2.3%.

Compared to peak-to-peak changes, across the entire time range (Table 2.6) we observe a more dramatic decrease in HFR. In Florida, the HFR drops by as little as 17% in the 80+ age range, and as much as 42% in the 60-69 age range. Nationally, the HFR drops by as little as 55% in the 80+ age groups, and as much as 73% in the 20-29 age group.

While this paper presents estimates at the two peaks and the endpoints of the study time range, we can easily read off similar estimates with uncertainty for all dates between April 1st and December 1st. This type of interactive functionality is available in our web tool. In our web tool, when stratifying by gender in addition to age, the conclusions surrounding drops in HFR are similar to those when just stratifying by age.

Table 2.5: Estimates of HFR and drop in HFR on peak dates. Median and 95% confidence intervals (CI) are computed using block bootstrapping. Results with inadequate support are omitted.

	Florida			**National**		
Age group	2020-04-15	2020-07-15	04-15 to 07-15	2020-04-15	2020-07-15	04-15 to 07-15
aggregate	0.23 (0.21, 0.26)	0.23 (0.21, 0.24)	-0.023 (-0.14, 0.13)	0.29 (0.28, 0.3)	0.17 (0.17, 0.18)	-0.41 (-0.44, -0.37)
20-29	-	-	-	0.021 (0.018, 0.025)	0.014 (0.012, 0.016)	-0.34 (-0.48, -0.16)
30-39	-	-	-	0.044 (0.041, 0.047)	0.027 (0.025, 0.03)	-0.37 (-0.44, -0.3)
40-49	-	-	-	0.079 (0.074, 0.084)	0.056 (0.053, 0.059)	-0.29 (-0.35, -0.22)
50-59	0.092 (0.078, 0.11)	0.1 (0.093, 0.11)	0.12 (-0.085, 0.38)	0.15 (0.14, 0.15)	0.1 (0.096, 0.1)	-0.31 (-0.35, -0.27)
60-69	0.18 (0.15, 0.21)	0.21 (0.19, 0.22)	0.13 (-0.045, 0.38)	0.26 (0.25, 0.27)	0.18 (0.18, 0.19)	-0.3 (-0.33, -0.26)
70-79	0.31 (0.28, 0.34)	0.33 (0.31, 0.34)	0.034 (-0.078, 0.18)	0.4 (0.39, 0.42)	0.27 (0.26, 0.27)	-0.34 (-0.37, -0.31)
80+	0.46 (0.43, 0.49)	0.48 (0.46, 0.49)	0.029 (-0.055, 0.12)	0.57 (0.55, 0.59)	0.41 (0.4, 0.42)	-0.27 (-0.3, -0.24)

Table 2.6: Estimates of HFR and drop in HFR between April 1st and December 1st. Median and 95% CI are computed using block bootstrapping. Results with inadequate support are omitted.

	Florida			**National**		
Age group	2020-04-01	2020-12-01	04-01 to 12-01	2020-04-01	2020-12-01	04-01 to 12-01
aggregate	0.23 (0.2, 0.27)	0.16 (0.12, 0.19)	-0.33 (-0.52, -0.094)	0.34 (0.32, 0.35)	0.13 (0.11, 0.15)	-0.61 (-0.67, -0.56)
20-29	-	-	-	0.025 (0.019, 0.03)	0.0066 (0.0014, 0.012)	-0.73 (-0.95, -0.43)
30-39	-	-	-	0.049 (0.045, 0.054)	0.019 (0.014, 0.025)	-0.61 (-0.72, -0.48)
40-49	-	-	-	0.087 (0.08, 0.095)	0.031 (0.024, 0.038)	-0.65 (-0.74, -0.54)
50-59	-	-	-	0.16 (0.15, 0.17)	0.06 (0.05, 0.07)	-0.63 (-0.69, -0.55)
60-69	0.18 (0.14, 0.22)	0.1 (0.06, 0.14)	-0.42 (-0.69, -0.096)	0.28 (0.27, 0.3)	0.11 (0.097, 0.13)	-0.61 (-0.66, -0.55)
70-79	0.31 (0.26, 0.35)	0.19 (0.14, 0.23)	-0.38 (-0.57, -0.17)	0.45 (0.43, 0.47)	0.18 (0.16, 0.2)	-0.6 (-0.65, -0.54)
80+	0.44 (0.39, 0.49)	0.37 (0.32, 0.41)	-0.17 (-0.32, -0.0038)	0.62 (0.59, 0.64)	0.28 (0.25, 0.31)	-0.55 (-0.59, -0.49)

Discussion

We unpack the drop in CFR to quantify improvements reasonably attributable to advances in treatment, accounting for shifting age distributions (H1) by age-stratifying, increased testing capacity (H2) by focusing on the hospitalized, and the detection-to-fatality delay (H3) by conducting a cohort-based analysis. We find that increased testing does not explain away the three waves due to corresponding peaks in hospitalizations, deaths, and positive test rates. We visualize the shifting age distributions, and quantify the decrease in age-stratified HFRs between the first two peaks and across the entire study time range. Combining all these analyses, we arrive at the following narrative:

At the beginning of April, testing was relatively sparse (Figure 2.9). Cases, hospitalizations, and deaths were rising, and reached peak levels circa April 15th (Figure 2.10). Roughly one in every ten tests came back positive in Florida, and one in every five tests, nationally. In Florida, the aggregate HFR was approximately 23%, with age-stratified HFRs ranging between 9.2% for the 50-59 age group to 46% for the 80+ age group (Table 2.5). Nationally, the aggregate HFR was approximately 29%, with the age-stratified HFRs ranging between 2.1% for the 20-29 age group and 57% for the 80+ age group (Table 2.5). In each age group, the national HFR was higher than the Florida HFR, possibly due to overwhelmed hospital systems in states hit hard during the first wave. In fact, our web tool indicates that 34.5% of national CDC cases between April 1st and April 15th were recorded in New York alone.

Over the next three months, the proportion of younger individuals with COVID-19 grew steadily (Figure 2.11). Testing continued to rise nationally, and spiked in Florida as it approached a heavier second peak around July 15, with positive test rates also at an all-time high (Figure 2.9). Florida experienced record hospitalizations and deaths, and the age-stratified HFRs were at least as high as in the first wave (Table 2.5). While Bill Gates had publicly attributed "a factor-of-two improvement in hospital outcomes" to dexamethosone and remdesivir (Levy, 2020), this did not yet appear to be true in Florida. (Alternatively, treatment improvements might have been counterbalanced by strain on the hospital system.) On the other hand, cases in New York had diminished (shown in our web tool) and were starting to surge in other states, forming a smaller second peak nationally (as measured by hospitalizations and deaths). Between the first two peaks, the national HFR had dropped by 41% in aggregate, with age-stratified HFRs dropping as much as 37% in the 30-39 age group and as little as 27% in the 80+ age group. The different stories told here by Florida and the national aggregate data underscore the importance of state-level rather than national analysis.

Finally, come December 1st, a third wave is underway. Approximately 31% of all national cases since April were confirmed in the last month alone (Figure 2.10b). In terms of hospitalizations, deaths, and positive test rate, this third wave has already surpassed the nation's second wave. For Florida, the third wave is already at least as severe as the first wave. Fortunately, however, age-stratified HFRs in both Florida and the national aggregate data appear to have dropped significantly since the start of the pandemic, likely indicating treatment improvements (though possibly confounded by disease mutations (H5) and reduced viral loads (H6)). Since April 1st, the age-stratified HFR in Florida has decreased by as much as 42% in the 60-69 age group and as little as 17% in the 80+ age group. Nationally, the age-stratified HFR has decreased by as much as 73% in the 20-29 age group and as little as 55% in the 80+ group. Regarding the CFR, on July 27, former President Donald Trump stated in a press briefing that "Due to the medical advances we've already achieved and our increased knowledge in how to treat the virus, the mortality rate for patients over the age of 18 is 85 percent lower than it was in April." (Trump, 2020) Note, however, that none of our estimates of improvements attributable to treatment are as large as the 85% touted by Trump. In summary, CFR can be misleading if age distribution shift, increased testing, and delays between detection and fatality are unaccounted for.

Limitations We aim to quantify treatment improvements (H4) in Florida and the U.S. by estimating changes in the age-stratified HFR, but H4 could also be influenced by disease mutation (H5) and changing viral loads (H6). To distinguish their effects in future work, we need additional data. Furthermore, while we listed the six hypotheses we found in literature review, possible alternative explanations may arise in the future as pandemic evolves.

We assume that treatment improvements will be reflected in the HFR because over our study's time range, major treatment improvements (e.g., dexamethosone and remdesivir) targeted hospitalized patients. While the first U.S. vaccination was administered outside of our study time range, vaccines take effects before hospitalization, and so their treatment improvements may not be reflected in the HFR for future studies. While our method could still quantify post-hospitalization treatment improvements, we note that vaccination roll-out criteria (e.g. occupation, age) and other characteristics (e.g. socioeconomic background) could influence who gets hospitalized in the first place.

Other limitations arise from data quality issues. In both the FDOH data and CDC data, missingness for hospitalization and death are high (Table 2.4), potentially introducing bias in the estimates for HFR if the data are not missing at random. Stratifying the national data by state, it appears that that each state may have different patterns of reporting their data to the CDC (as can be seen in our web tool, when any state is filtered for). First, the reported CDC cases appear to be incomplete for several states. For instance, in the subset of CDC data reported from Florida, the cases only account for 67% of the cases provided by the FDOH, they appear to be reported sporadically even after 7-day smoothing, and no death data is reported since October. Cross-referencing with the COVIDcast API, we find that in the subset of CDC data from Texas, only 4.9% of the cases and only 0.01% of the deaths are accounted for (Project, 2020), and only 14 hospitalizations were recorded across the entire studied time range. Thus, in our national analysis, we are making the assumption that in aggregate the signal will outweigh the noise. Despite the data limitations, the CDC data appears to be the best available source of line-level cases needed for cohort-based analysis across the United States. We note that in the Florida FDOH data, on the other hand, we use the positive test confirmed date which is not missing at all in this data, making the Florida HFR estimates more reliable than those from the national data.

Chapter 3

Evaluating Models on Medical Datasets Over Time (EMDOT)

Machine learning (ML) models deployed in healthcare systems must face data drawn from continually evolving environments. However, researchers proposing such models typically evaluate them in a time-agnostic manner, splitting datasets according to patients sampled randomly throughout the entire study time period. This work proposes the Evaluation on Medical Datasets Over Time (EMDOT) framework, which evaluates the performance of a model class across time. Inspired by the concept of backtesting, EMDOT simulates possible training procedures that practitioners might have been able to execute at each point in time and evaluates the resulting models on all future time points. Evaluating both linear and more complex models on six distinct medical data sources (tabular and imaging), we show how depending on the dataset, using all historical data may be ideal in many cases, whereas using a window of the most recent data could be advantageous in others. In datasets where models suffer from sudden degradations in performance, we investigate plausible explanations for these shocks. We release the EMDOT package to help facilitate further works in deployment-oriented evaluation over time.

We use the following data six sources of data: (1) the Surveillance, Epidemiology, and End Results (SEER) cancer dataset (National Cancer Institute, 2020), (2) the COVID-19 Case Surveillance Detailed Data provided by the CDC (CDC, COVID-19 Response, 2020), (3) the Southwestern Pennsylvania (SWPA) COVID-19 dataset, (4) the MIMIC-IV intensive care database (Johnson et al., 2021), (5) the Organ Procurement and Transplantation Network (OPTN) database for liver transplant candidates (Organ Procurement and Transplantation Network, 2020), and (6) the MIMIC-CXR-JPG database of chest radiographs (Johnson et al., 2019a; Johnson et al., 2019b). MIMIC-IV and MIMIC-CXR-JPG (referred to as MIMIC-CXR in this paper) are available on the PhysioNet repository (Goldberger et al., 2000). Except for the SWPA dataset, all are publicly accessible (after accepting a data usage agreement). Details for accessing each dataset are in Appendices C.3–C.7. The code is publicly available on GitHub.

3.1 Introduction

As medical practices, healthcare systems, and community environments evolve over time, so does the distribution of collected data. Features are deprecated as new ones are introduced, data collection may fluctuate along with hospital policies, and the underlying patient and disease populations may shift. Amidst this ever-changing environment, models that perform well on one time period cannot be assumed to perform well in perpetuity. In the MIMIC-III critical care dataset, Nestor et al. (2019) found that a change to the electronic health record (EHR) system in 2008 coincided with sudden degradations in AUROC for mortality prediction. In COVID-19 data from the Centers for Disease Control and Prevention (CDC), Cheng et al. (2021) noted that the age distribution among cases shifted continually throughout the pandemic, and that these continual shifts confounded estimates of improvements in mortality rate.

We propose an evaluation framework to characterize model performance over time by simulating training procedures that practitioners could have executed up to each time point, and subsequently deployed in future time points. We argue that standard time-agnostic evaluation is insufficient for selecting deployment-ready models, showing across several datasets that it over-estimates deployment performance. Instead, we propose EMDOT as a worthwhile pre-deployment step to help practitioners gain confidence in the robustness of their models to distribution shifts that have occurred in the past and may to some extent repeat in the future.

There is a large body of work that addresses adaptation under various structured forms of distribution shift, including covariate shift (Shimodaira, 2000b; Zadrozny, 2004; Huang et al., 2006; Sugiyama et al., 2007b; Gretton et al., 2009), label shift (Saerens et al., 2002; Storkey et al., 2009; Zhang et al., 2013; Lipton et al., 2018; Garg et al., 2020), missingness shift (Zhou et al., 2022a), and concept drift (Tsymbal, 2004; Gama et al., 2014). However, in the real-world medical datasets we analyze, none of these structural assumptions can be guaranteed, and distributional changes in covariates, labels, missingness, etc. could even occur simultaneously. This motivates our empirical work, as it is unclear across a variety of model classes and medical datasets, how existing models might degrade due to naturally occurring changes over time, and whether different training practices might impact on robustness over time.

However intuitive it might seem, evaluation of models over time remains uncommon in standard machine learning for healthcare (ML4H) research. In the proceedings of the Conference on Health, Inference, and Learning (CHIL) 2022, for example, none of the 23 papers performed evaluations which took time into account (see Appendix C.1 for similar statistics from CHIL 2021 and the Radiology medical journal). One possible reason for this is lack of access—as noted by Nestor et al. (2019), it is common practice to remove timestamps when de-identifying medical datasets for public use. In this work, we identify six sources of medical data containing varying granularities of temporal information per-record, five of which are *publicly available*. We profile the performance of various training strategies and model classes across time, and identify possible sources of distribution shifts within each dataset. Finally, we release the Evaluation on Medical Datasets Over Time (EMDOT) Python package (details in Appendix C.2) to allow researchers to apply EMDOT to their own datasets and test techniques for handling distribution shifts over time.

3.2 Related work

The promise of ML for improving healthcare has been explored in several domains, including cancer survival prediction (Hegselmann et al., 2018), diabetic retinopathy detection (Gulshan et al., 2016), antimicrobial stewardship (Kanjilal et al., 2020; Boominathan et al., 2020), recognizing diagnoses from electronic health record data (Lipton et al., 2016a), and mortality prediction in liver transplant candidates (Bertsimas et al., 2019; Byrd et al., 2021). Typically, these ML models are evaluated on randomly held out patients, and sometimes externally validated on other hospitals or newly collected data. Even with cross-site validations, we cannot be sure how models will perform in the future.

For decades, the medical community has had a history of utilizing (mostly) fixed, simple risk scores to inform patient care (Hermansson and Kahan, 2018; Kamath et al., 2001; Wilson et al., 1998; Wells et al., 1995). Risk scores often prioritize ease-of-use, are computed from few variables, verified by domain experts for clear causal connections to outcomes of interest, and validated through use over time and across hospitals. Together, these factors give clinicians confidence that the model will perform reliably for years to come. With increasingly complex models, however, trust and adoption may be hindered by a lack of confidence in robustness to changing environments.

As noted by D'Amour et al. (2022b), ML models often exhibit unexpectedly poor behavior when deployed in real-world domains. A key reason for these failures, they argue, is *under-specification*, where ML pipelines yield many predictors with equivalently strong held-out performance in the training domain, but such predictors can behave very differently in deployment. By testing performance across a variety of distribution shifts that have previously occurred over time, EMDOT could serve as a stress test to help combat under-specification.

Although evaluation over time is far from standard in ML4H literature, changes in performance over time have been noted in prior work. To predict wound-healing, Jung and Shah (2015) found that when data were split by cutoff time instead of patients, benefits of model averaging and stacking disappeared. Pianykh et al. (2020) found degradation in performance of a model for wait times dependent on how much historical data was trained on. To predict severe COVID-19, Zhou et al. (2022c) found that learned clinical concept features performed more robustly over time than raw features. Closest to our work is Nestor et al. (2019), which evaluated AUROC in MIMIC-III critical care data from 2003–2012, comparing training on just 2001–2002; the prior year; and the full history. Using the full history and curated clinical concepts, they bridged a big drop in performance due to changing EHR systems. Whereas Nestor et al. (2019) considers three models per test year, EMDOT simulates model deployment every year and evaluates across *all future years*.

While we do not consider time series models in this work (instead considering those which treat data as i.i.d.), there are similarities between how training sets are defined in EMDOT and in techniques for evaluating time-series forecasts (Bergmeir and Benítez, 2012; Cerqueira et al., 2020). These techniques often roll forward in time, taking either a window of recent data or all historical data as training sets, and evaluate test performance on the next time point.

Table 3.1: Summary of datasets used for analysis. For more details, see Appendices C.3–C.7.

Dataset name	Outcome	Time Range (time point unit)	# samples	# positives
SEER (Breast)	5-year Survival	1975–2013 (year)	462,023	378,758
SEER (Colon)	5-year Survival	1975–2013 (year)	254,112	135,065
SEER (Lung)	5-year Survival	1975–2013 (year)	457,695	49,997
CDC COVID-19	Mortality	Mar 2020–May 2022 (month)	941,140	190,786
SWPA COVID-19	90-day Mortality	Mar 2020–Feb 2022 (month)	35,293	1,516
MIMIC-IV	In-ICU Mortality	2009–2020 (year)	53,050	3,334
OPTN (Liver)	180-day Mortality	2005–2017 (year)	143,709	4,635
MIMIC-CXR	14 diagnostic labels	2010–2018 (year)	376,204	209,088

Performance from each time point is then averaged to summarize performance. This type of back-testing technique is common in rapidly evolving, non-stationary applications like finance (Chauhan et al., 2020; Alberg and Lipton, 2017), where time series models are constantly updated. In the healthcare domain, however, models may not be so easily updated, with risk scores developed several years ago still being used to this day (Six et al., 2008; Kamath et al., 2001; Wilson et al., 1998; Wells et al., 1995). Thus, we track performance not only the immediate year after the training set, but all subsequent years in the dataset. Additionally, instead of collapsing performance from models trained at different time points into summary statistics, which could conceal distribution shifts over time, our framework tracks these granular fluctuations over time, and creates tools to help provide insight into the nature and potential causes of such changes.

3.3 Data

We sought medical datasets that had: (1) a timestamp for each record, (2) interesting prediction task(s), and (3) enough distinct time points to evaluate over. Six data sources satisfied these criteria: SEER cancer data, national CDC COVID-19 data, COVID-19 data from a healthcare provider in Southwestern Pennsylvania (SWPA), MIMIC-IV critical care data, OPTN data from liver transplant candidates, and MIMIC-CXR chest radiographs. All datasets are tabular except for MIMIC-CXR (medical imaging data). All but SWPA are publicly accessible.

Table 3.1 summarizes the dataset outcomes, time ranges, and number of samples. Figure 3.1 visualizes data quantity over time. Appendices C.3–C.8 include cohort selection diagrams, cohort characteristics, features, heat maps of missingness, preprocessing steps, and additional details. Categorical variables are converged to dummies, and numerical variables are normalized and centered at 0. Missing values in categorical variables are treated as another category, and in numerical variables they are imputed with the mean. In all datasets except MIMIC-CXR (where each sample is a distinct radiograph), each sample corresponds to a distinct patient.

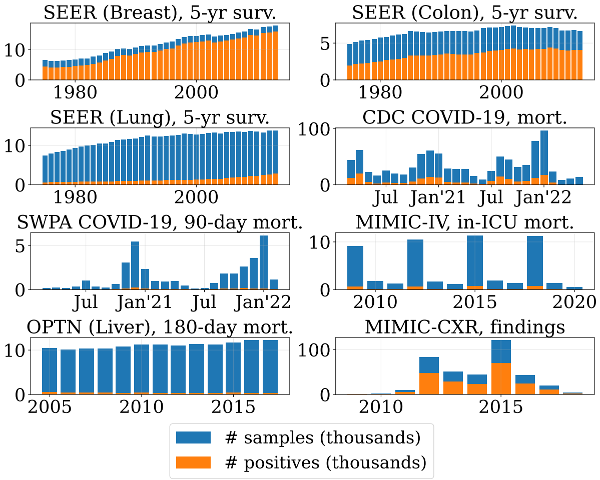


Figure 3.1: Number of samples and positive[1] outcomes per time point.

3.3.1 SEER Cancer Data

The Surveillance, Epidemiology, and End Results (SEER) Program collects cancer incidence data from registries throughout the U.S. Each case includes demographics, primary tumor site, tumor morphology, stage, diagnosis, first course of treatment, and survival outcomes (collected with follow-up) (National Cancer Institute, 2020). We use the SEER*Stat software (Program, 2015) to define three cohorts of interest: (1) breast cancer, (2) colon cancer, and (3) lung cancer. The outcome is 5-year survival, i.e. whether the patient was confirmed alive five years after the year of diagnosis. The amount of data has mostly increased each year (Figure 3.1). Performance over time is evaluated *yearly*. See Appendix C.3 for more details.

3.3.2 National CDC COVID-19 Data

The COVID-19 Case Surveillance Detailed Data (CDC, COVID-19 Response, 2020) is a national dataset provided by the CDC. It has the largest number of samples among the datasets considered, and contains 33 elements, with patient-level data including symptoms, demographics, and state of residence. The cohort consists of all lab-confirmed positive COVID-19 cases that were hospitalized, so the quantity of samples over time has a seasonality reflecting surges in COVID-19 (Figure 3.1). The outcome of interest is mortality, defined by "death yn" = "Yes" in the dataset. Performance over time is evaluated on a *monthly* basis. See Appendix C.4 for more details.

3.3.3 SWPA COVID-19 Data

The Southwestern Pennsylvania (SWPA) COVID-19 dataset consists of EHR data from patients tested for COVID-19. It is the smallest dataset considered in this work, and was collected by

[1] In MIMIC-CXR, all labels except "No Finding" are considered positive in Figure 3.1 and Table 3.1.

a major healthcare provider in SWPA. Features include patient demographics, labs, problem histories, medications, inpatient vs. outpatient status, and other information collected in the patient encounter. The cohort consists of COVID-19 patients testing positive for the first time, and not already in the ICU or mechanically ventilated. Similar to the CDC COVID-19 dataset, there is a seasonality to the monthly number of samples that reflects surges in COVID-19 (Figure 3.1). The outcome of interest is 90-day mortality, derived by comparing the death date and test date. Performance over time is evaluated on a *monthly* basis. See Appendix C.5 for more details.

3.3.4 MIMIC-IV Critical Care Data

The Medical Information Mart for Intensive Care (MIMIC)-IV (Johnson et al., 2021) database contains EHR data from patients admitted to critical care units from 2008–2019. MIMIC-IV is an update to MIMIC-III, adding time annotations placing each sample into a three-year time range, and removing elements from the old CareVue EHR system (before 2008). We approximate the year of each sample by taking the midpoint of its time range, but note that this causes certain years (2009, 2012, 2015, 2018) to have substantially more samples than others (Figure 3.1). The cohort is selected by taking the first encounter of all patients in the "icustays" table, and the outcome of interest is in-ICU mortality. Performance over time is evaluated on a *yearly* basis. See Appendix C.6 for more details.

3.3.5 OPTN Liver Transplant Data

The Organ Procurement and Transplantation Network (OPTN) database tracks organ donation and transplant events in the U.S. The selected cohort consists of liver transplant candidates on the waiting list. The same pipeline as Byrd et al. (2021) is used to extract the data, except that the first record is selected for each patient. The outcome of interest is 180-day mortality from when the patient was added to the list. The performance over time is evaluated on a *yearly* basis. More details are in Appendix C.7.

3.3.6 MIMIC-CXR

The MIMIC Chest X-ray (MIMIC-CXR) JPG dataset (Johnson et al., 2019b) contains chest radiographs in JPG format. Similar to MIMIC-IV, we approximate the year by taking the midpoint of its three-year time range. The selected cohort consists of all radiographs from 2010 to 2018. The outcomes of interest are 14 diagnostic labels: Atelectasis, Cardiomegaly, Consolidation, Edema, Enlarged Cardiomediastinum, Fracture, Lung Lesion, Lung Opacity, Pleural Effusion, Pneumonia, Pneumothorax, Pleural Other, Support Devices, and No Finding. Performance over time is evaluated on a *yearly* basis. More details are in Appendix C.8.

3.4 Methods

We tackle the following guiding questions:

1. On each dataset, what would the reported performance of a model be if it were trained using standard time-agnostic splits (**all-period**)?
2. **Simulating** how a practitioner might have trained and deployed models in the past, how would performance have varied **over time**?
3. When might it be better to train on a **recent window** of data versus **all historical** data?
4. What is the comparative performance of different **classes of models** over time?
5. To what extent might we be able to diagnose possible **reasons** for changes in model performance?

3.4.1 All-period Training

We mimic common practice in evaluation by using time-agnostic data splits which randomly place patients from the entire study time range into train, validation, and test sets (details in Appendix C.12), and reporting the test set performance. We refer to training with this type of split as *all-period* training.

3.4.2 EMDOT Evaluation

For more realistic simulation of how practitioners train models and subsequently deploy them on future data, we define the *Evaluation on Medical Datasets Over Time* (EMDOT) framework. At each time point t (termed *simulated deployment date*), an *in-period* subset of data from times $\leq t$ is available for model development. After training a model on this in-period data, one might be interested in both recent in-period performance (at time t) and future *out-of-period* performance (at times $> t$).

In-period data is split into train, validation, and test sets (split ratios in Appendix C.12). For MIMIC-CXR, where one patient could have multiple radiographs, the data is split such that there are no overlapping patients between splits. Recent in-period performance is evaluated on held-out test data from the most recent time point. Out-of-period performance is evaluated on all data from each future time point. For example, a model trained up to time 6 is tested on data from 6, 7, 8, etc. (Figure 3.2). At time 8, the model is considered two time points *stale*. Although this procedure can take $O(T)$ times more computation than all-period training for T time points, we argue that this procedure yields a more realistic view of the type of performance that one might expect models to have over time.

Additionally, practitioners face a tradeoff between using recent data perhaps most reflective of the present and using all available historical data for a larger sample size. Intuitively, the former may be appealing in modern applications with massive datasets, whereas the latter may

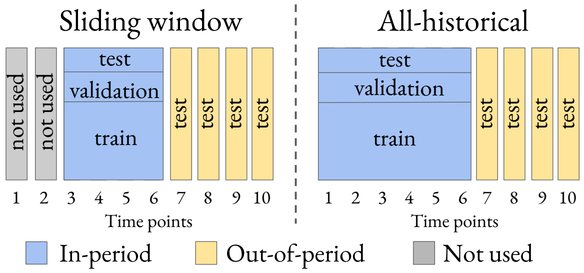


Figure 3.2: EMDOT training regimes, with a simulated deployment date of $t = 6$.

be necessary in data-scarce applications. We explore these two training regimes, with different definitions of in-period data (Figure 3.2):

1. **Sliding window**: The last W time points are considered in-period. In this paper, we use window size W = 4 for sufficient positive examples.
2. **All-historical**: Any data prior to the current time point is considered in-period.

To decouple the effect of sample size from that of shifts in the data distribution, comparisons are also performed with all-historical data that is **sub-sampled** to be the same size as the corresponding training set under the sliding window training regime.

To summarize more formally, let D_t refer to the set of all data points occurring at time $t \in \{1, ..., T\}$, where T is the number of time points that the dataset spans. Each D_t can be partitioned by splitting patients at random into disjoint train, validation, and test sets: $D_t = D_t^{\text{train}} \cup D_t^{\text{val}} \cup D_t^{\text{test}}$. For simulated deployment dates $t^* \in \{W, W + 1, ..., T\}$, training, validation, and test sets are defined for the *sliding window* training regime as follows:

- training: $\bigcup_{k=t^*-W+1}^{t^*} D_k^{\text{train}}$
- validation: $\bigcup_{k=t^*-W+1}^{t^*} D_k^{\text{val}}$
- in-period test: $D_{t^*}^{\text{test}}$
- out-of-period test: D_k for $k = t^* + 1, ..., T$

Training, validation, and test sets are defined for the *all-historical* training regime as follows:

- training: $\bigcup_{k=1}^{t^*} D_k^{\text{train}}$
- validation: $\bigcup_{k=1}^{t^*} D_k^{\text{val}}$
- in-period test: $D_{t^*}^{\text{test}}$
- out-of-period test: D_k for $k = t^* + 1, ..., T$

At each simulated deployment date t^*, models are trained using the training set, validated using the validation set, and tested on the in-period test set as well as all out-of-period test sets. If a model with simulated deployment date t^* is being evaluated on an out of period test set D_{t^*+j}, then the model is j time points *stale.*

3.4.3 Evaluation Metrics

All binary classification tasks are evaluated by AUROC. For multi-label prediction in MIMIC-CXR, each of the 14 diagnostic labels is treated as a separate binary classification task, and a weighted sum of AUROCs is computed, where the weight for a particular label is given by the proportional prevalence of that label among all positive labels. That is, for some class a, its weight is $p_a / \sum_x p_x$, where p_x is the number of positives with label x. Samples are treated in an i.i.d. manner for training.

3.4.4 Models

Logistic regression (LR), gradient boosted decision trees (GBDT) and feedforward neural networks (MLP) are trained on the tabular datasets. DenseNet-121 is trained on the MIMIC-CXR imaging dataset. Hyperparameters are selected based on in-period validation performance, and the hyperparameter grids are in Appendix C.13.

3.4.5 Detecting Sources of Change

To better understand possible reasons for changing performance, we create *diagnostic plots* to track model performance alongside changes in the data distribution over time. In tabular datasets, we plot feature importances and average values of the most important features over time. Generating these plots for logistic regression, we define feature importance by the magnitudes of the coefficients, but note that other feature importance techniques could be used for more complex model classes. To avoid overcrowding the plots, we take the union of the top k most important features from each time point is taken, where k is tuned depending on the dataset. We additionally highlight (using a thicker line) categorical features with consistently high prevalence or which experience a large change in prevalence across one time point, and numerical features with high average rank (see Appendix C.10 for thresholds for each dataset). For the imaging dataset, where feature importance is less straightforward, we plot the distribution of pixel intensities over time, along with proportions of each of the 14 diagnostic labels. By highlighting sudden changes in model performance and the corresponding time periods in all other plots, diagnostic plots can help bring attention to shifts in the distribution of data that coincide with changing model performance.

3.4.6 EMDOT Python Package

We release the EMDOT python package[2] to help practitioners move from standard model evaluation to EMDOT evaluation. See Appendix C.2 for a schematic of the EMDOT workflow, and see the GitHub repository for a step-by-step tutorial.

[2] https://github.com/acmi-lab/EvaluationOverTime

Table 3.2: Test AUROC from all-period training and time-agnostic evaluation.

Model	SEER (Breast)	SEER (Colon)	SEER (Lung)	CDC COVID-19	SWPA COVID-19	MIMIC-IV	OPTN (Liver)	MIMIC-CXR
LR	0.888	0.863	0.894	0.837	0.928	**0.935**	0.846	-
GBDT	**0.891**	0.868	0.894	0.851	**0.930**	0.931	**0.854**	-
MLP	**0.891**	**0.869**	**0.898**	**0.852**	0.928	0.898	0.847	-
DensetNet	-	-	-	-	-	-	-	**0.860**

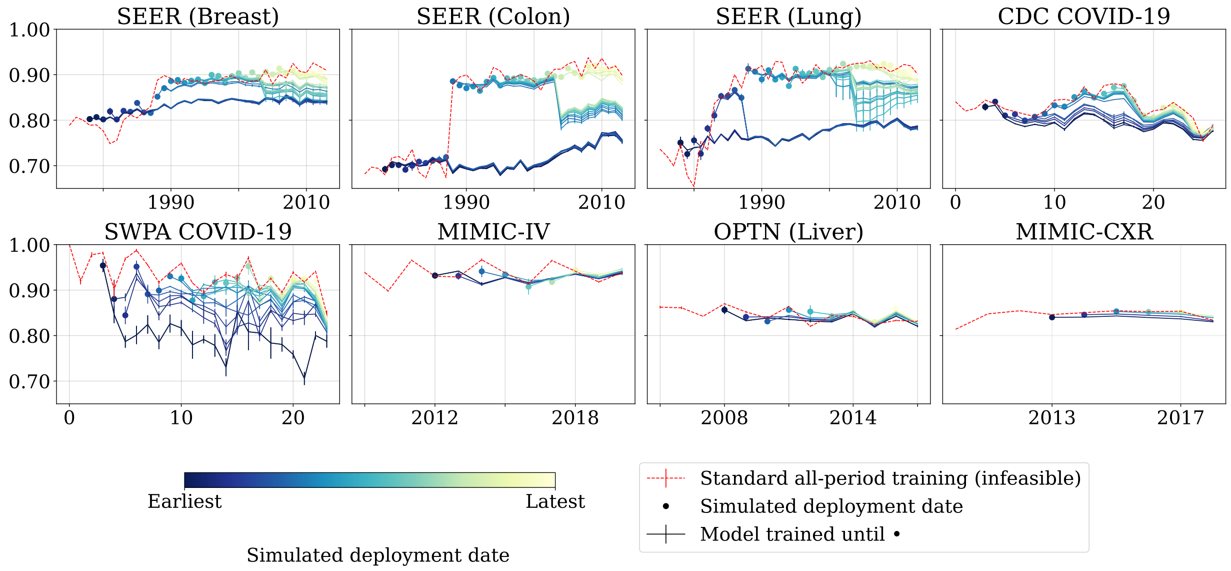


Figure 3.3: Average test AUROC of logistic regression vs. time. Each solid line gives the performance of a model trained up to a simulated deployment time (marked by a dot), evaluated across future time points. Error bars are ± standard deviation computed over 5 random splits. Red dotted line gives per-timepoint test performance of a model from all-period training (infeasible in reality, as it would involve training on data after the simulated deployment date).

3.5 Results

3.5.1 All-period Training

In standard time-agnostic evaluation, GBDT and MLP achieve the highest average test AUROC on all tabular datasets except MIMIC-IV (Table 3.2). Note however that LR often has comparable or only slightly lower AUROC than the more complex models. The top 10 coefficients of each LR with all-period training are in Appendices C.3–C.7, and the per-label AUROC of MIMIC-CXR is in Appendix Table C.9. To form a baseline for comparison across time, we also evaluate the all-period models on subsets of the all-period test data that belong to each year (red dotted line in Figure 3.3), but note that this type of training (on future data) is not feasible in deployment.

3.5.2 EMDOT Evaluation

Figure 3.3 plots the AUROC of LR for all tabular datasets (and DenseNet-121 for MIMIC-CXR) over time when using the all-historical training regime. Plots for GBDT and MLP are in Appendix C.11, along with plots for AUPRC. We mainly discuss AUROC, but note that AUPRC observes similar trends as in AUROC. One difference however is that the baseline AUPRC performance is given by the label prevalence (rather than a constant 0.5, as in AUROC), and so observed trends in label prevalence over time appear to influence trends in AUPRC (Appendix Figure C.39).

For both AUROC and AUPRC, the reported test performance of a model from standard all-period training (red dotted line) mostly sits above the performance of any model that could have realistically been deployed by that date. Thus, all-period training tends to provide an over-optimistic estimate of performance upon deployment.

Across the datasets, a variety of trajectories of model performance are observed over time. In the SEER datasets, the AUROC of freshly trained models increases dramatically near 1988, but several of these models experience a large drop in AUROC around 2003 (Figure 3.3). Additionally, in-period test AUROCs tend to increase over time. By contrast, in CDC data, in-sample test AUROCs fluctuate up and down, and model performance over time varies more smoothly, appearing to loosely follow the in-sample performance. Models trained after December 2020 have a slight boost in AUROC, coinciding with a surge in cases (and hence sample size, Figure 3.1), however by January 2022 the in-sample AUROC decreases. In SWPA COVID-19, there is more variation and uncertainty in AUROC early in the pandemic, where sample sizes are small. In December 2020, sample sizes increase, and models seem to become more robust to changes over time. Finally, in the MIMIC-IV, MIMIC-CXR, and OPTN datasets, AUROC appears relatively stable across time.

3.5.3 Training Regime Comparison

As the staleness of training data increases (i.e. as test date strays from simulated deployment date), the training regimes fare differently depending on the dataset (Figure 3.4, left).

In SEER (Breast) and SEER (Lung), sliding window is initially comparable to all-historical on fresh (low-staleness) data, but significantly underperforms both all-historical and all-historical (subsampled) when data are 8 to 22 years stale. At larger stalenesses, all training regimes start to become comparable. In CDC COVID-19, sliding window outperforms all-historical regardless of how stale the data is. By contrast, in SWPA COVID-19, which has the least amount of data (Table 3.1), both sliding window and all-historical (subsampled) underperform all-historical. In SEER (Colon), performance is relatively stable regardless of training regime. In MIMIC-IV, OPTN (Liver), and MIMIC-CXR, sliding window is on average comparable or slightly outperforms all-historical when staleness is 0, but at nonzero stalenesses all-historical outperforms both sliding window and all-historical subsampled.

[3] Note: at the largest stalenesses, there are fewer simulated deployment dates being averaged over, and they must be early in the dataset. Here, the sliding window and all-historical can be expected to perform similarly

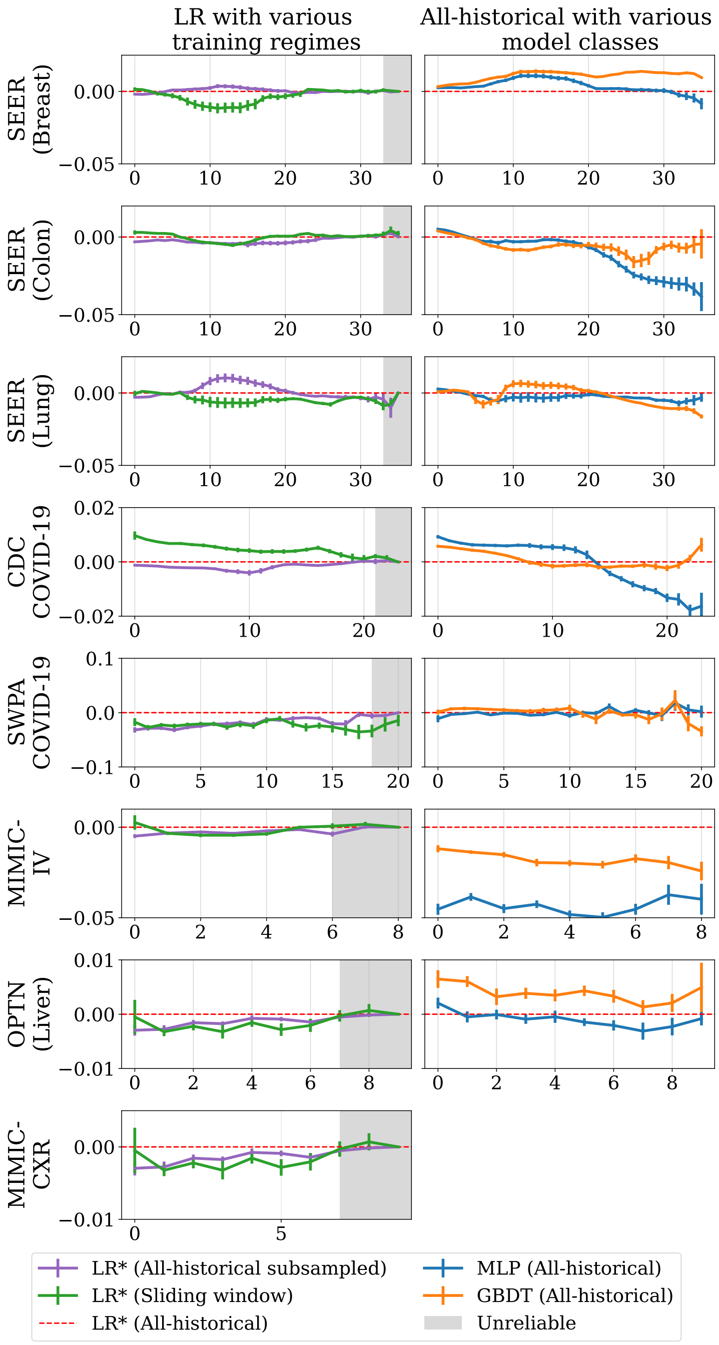


Figure 3.4: AUROC − AUROC$_{\text{LR* all-historical}}$ vs. staleness. i.e., AUROC difference relative to a LR* all-historical baseline across varying stalenesses of data,[3] for different training regimes (left) and model classes (right). Error bars are ± std. dev. (*in MIMIC-CXR, DenseNet-121 is used instead of LR)

(especially when the sliding window is not much larger than or even matches the history). Since this is an artifact of finite time ranges, we gray out stalenesses where at least half of the all-historical data is the first sliding window

3.5.4 Model Comparison

In SEER (Breast) and OPTN, GBDT outperforms both LR and MLP across the entire time range (Figure 3.4, right). In SEER (Colon), SEER (Lung), and CDC COVID-19, both GBDT and MLP initially outperform LR when staleness of the training data is less than 4 years, 4 years, and 7 months, respectively, however both eventually underperform LR as staleness increases further. While there is an uptick in GBDT performance on CDC COVID-19 towards 21-month staleness, we note this data point is derived from less data than other points on the line because the data time range is finite. In the SWPA COVID-19 dataset, LR, MLP, and GBDT appear to perform comparably over time. In the MIMIC-IV dataset, LR performed best to begin with and remained the best.

3.5.5 Detecting Possible Sources of Change

Diagnostic plots for all datasets are in Appendix C.10. Here, we discuss SEER (Lung) (Figure 3.5) in detail as it has several interesting changes in model performance over time. In 1983, as `EOD 4` features from the extent of disease coding schema are introduced (Figure 3.5, bottom right), a sudden jump in AUROC occurs (Figure 3.5, top and middle left). However, models trained at this time later experience a large AUROC drop (Figure 3.5, bottom left). By 1988, `EOD 4` is phased out, and `EOD 10` features are introduced. This coincides with another jump in AUROC, sustained until 2003 when the `EOD 10` features are removed. In this dataset, the all-historical training regime seems more robust to changes over time, as all-historical models trained after 1988 avoid the drop that sliding window models undergo once their window excludes pre-1988 data (Figure 3.5, bottom left).

3.6 Discussion

Reported model performance from standard all-period training tends to be over-optimistic (Figure 3.3) as models are evaluated on time points already seen in their training set (unrealistic in deployment settings). Thus, AUROCs reported from all-period training do not capture degradation that would have occurred in deployment.

Comparing model classes, in all datasets except MIMIC-IV, GBDT and MLP slightly outperform LR under standard time-agnostic evaluation (Appendix Table 3.2). However, evaluated across time, LR is often comparable and even outperforms more complex models once enough time passes after the simulated deployment date. For example, MLP achieves the best AUROCs in SEER Breast, Colon, and Lung in standard time-agnostic evaluation (Table 3.2). However, in evaluation over time, LR had superior performance once some amount of time (30, 5, 4 years respectively) had passed (Figure 3.4, right). In most datasets GBDT appears more robust over time than MLP, however as the training data becomes more stale it tends to become comparable

of data.

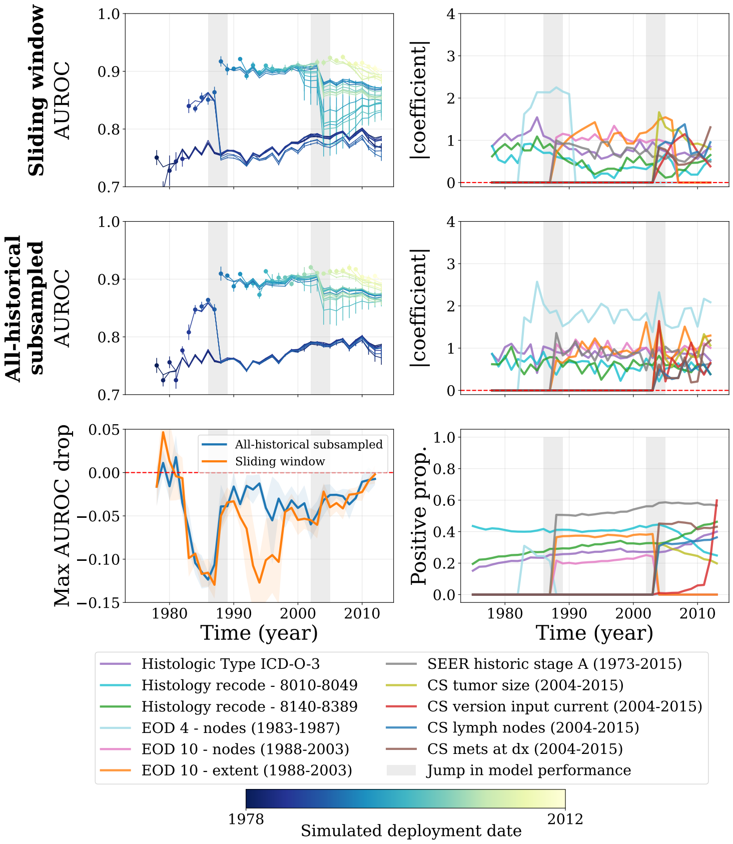


Figure 3.5: SEER (Lung) diagnostic plots. AUROC vs. time for sliding window (top-left) and all-historical subsampled (mid-left), max. drop in AUROC for each simulated deployment time (low-left), absolute feature coefficients for LR models from sliding window (top-right) and all-historical subsampled (mid-right) and prevalences of important features over time (low-right).

to LR (in all datasets except OPTN Liver and SEER Breast, GBDT dipped below the performance of LR for several stalenesses). Thus, although complex model classes may appear to outperform simpler linear model classes in standard time-agnostic evaluation, one should consider performance over time when selecting a model class for deployment. As demonstrated by the different relative performances of model classes when evaluated over time versus in a time-agnostic manner, EMDOT can serve as a helpful stress-test to combat under-specification.

Regarding training regimes, we find that with increasing stalenesses, all-historical appears more reliable than sliding window across all datasets except for CDC COVID-19 (Figure 3.4, left). In SWPA COVID-19, MIMIC-IV, OPTN (Liver), and MIMIC-CXR, the benefit of all-historical data likely comes from the increased sample size, as subsampling all-historical data to be the same size as the corresponding sliding window resulted in comparable performance to sliding window. In the SEER datasets, the effect of sample size is less pronounced, as sliding window and subsampled all-historical are frequently comparable to all-historical. There are certain stalenesses for which sliding window underperforms all-historical, which may be due to the addition and removal of features. If the sliding window model learns to rely on recently added features which are later removed, this could result in drops in performance whereas an all-historical model which had learned to predict without the presence of such features would be more robust to such changes. On the other hand, in CDC COVID-19 (the setting with the most data and fewest features), subsampled all-historical performs comparably to all-historical, and sliding window outperforms both across all stalenesses (Figure 3.4, left). This suggests that the performance of LR may have been saturated even when a sub-sample of all-historical data was used, and the benefit of using more recent data outweighs the larger sample size afforded by all-historical. More broadly, in rapidly evolving environments with simple models, few features, and large quantities of data, the sliding window training regime could be advantageous.

The SEER datasets had dramatic changes in data distribution in both 1988 and 2003, when important features were added and/or removed (Figure 3.5). One possible reason for the robustness of all-historical models in this dataset is that after 2003, when features like EOD 10 were removed, the model could still rely on features that were introduced prior to the use of EOD 10 in 1988. More broadly, we hypothesize that if a model was trained on a mixture of distributions that occurred throughout the past, it may be better equipped to handle shifts to settings similar to those distributions in the future.

While the SEER datasets and COVID-19 datasets displayed several changes in model performance over time, the OPTN and MIMIC datasets had relatively stable behavior. One possible reason for this is that the outcomes or diseases of interest were relatively stable in nature, we did not observe any substantial changes in the distribution of data. Another is that in the MIMIC datasets, a three-year range was given for each sample rather than a specific date. This uncertainty around the date, along with the limited number of date ranges, could result in a smoothing effect on the resulting estimates of performance.

In conclusion, EMDOT yields insights into the suitability of different model classes or training regimes for deployment, and also helps one detect distribution shifts that occurred in the past. Understanding such shifts may help practitioners be prepared for shifts of a similar nature in the future. Although the EMDOT framework requires more computation time than standard time-agnostic evaluation, we argue that the insights gained from this procedure can be vital before deployment in high-stakes settings.

Limitations and Future Work One possible reservation that users might have about using EMDOT is that it could involve training up to T times as many models as would normally be required (where T is number of timepoints). To help alleviate this concern, in future work we plan to implement parallelization in EMDOT. For noisier estimates of model performance in less time, one could also subsample the dataset. Another interesting extension is exploring performance over time in other data modalities (e.g. time series, natural language, etc.). Depending on the complexity of models used in these modalities, this may require additional computational resources. More broadly, we hope that others may also build upon EMDOT to shine new light on how models and methodologies fare when evaluated with an eye towards deployment.

Part II

Underreporting in Healthcare Data and Missingness Shift

"...all models are wrong, but some are useful."

- George E. P. Box, *Empirical Model-Building and Response Surfaces*

Missing data was a common problem in the datasets studied in Part I. As states updated their COVID-19 reporting policies over time, we saw artificial jumps in the number of cases and deaths reported. Other features recorded in medical records data were also at lower rates than expected. In this part of the thesis, we start with an empirical work which models and adjusts for underreporting in electronic medical records data from COVID-19 patients. Then, we formalize the intuition from fluctuating reporting policies by defining the problem of domain adaptation under missingness shift.

Chapter 4

Learning Clinical Concepts for Predicting Progression to Severe COVID-19

4.1 Introduction

As COVID-19 becomes endemic, communities are learning what it means for them to "live with" COVID-19. An important component of living with COVID-19 is understanding when individuals who contract the disease are likely to progress to a severe condition. Our work studies the risk of severe COVID-19 progression, using data collected by a major healthcare provider in Southwestern Pennsylvania from January 2020 to January 2022. We define *severe COVID-19* as a COVID-19 case involving mechanical ventilation, admission to an intensive care unit (ICU), or death.

Healthcare providers often have different systems for collecting and storing patient data. To utilize this data for prediction, researchers usually leverage domain expertise to manually extract a large initial set of potentially relevant features, and subsequently use automatic feature selection techniques to eliminate all but the most significant. Clinicians can then consider these features when determining a patient's care plan, and hospitals could potentially extract these features to calculate risk. While expert-guided curation of features can help reduce the model search space, it can also limit performance due to imperfect feature extraction and inadvertent removal of informative features. Automatic feature selection may yield features that are predictive of the outcome, but these features may actually be serving as *proxies* for higher-level concepts that cause the outcome (e.g. insulin medication may be predictive, but diabetes is the underlying risk factor), *especially when the higher-level concepts are underreported.* While reliably recorded proxies can be effective predictors, they can also yield misleading interpretations. Additionally, it is unclear whether these proxies will be equally effective when applied to new settings such as different time periods or different hospitals. As a result, doctors may favor smaller models with features whose relevance is intuitive even if these models suffer some loss in performance owing, in part, to underreporting of the features.

To strike a balance between incorporating domain knowledge, model simplicity, transparency, and performance, we propose to learn clinical concepts anchored to intuitive expert-selected features, and to use these concepts to predict severe COVID-19 progression. Motivated by high levels of missingness in our data, clinical concepts are learned by treating the presence of an expert-selected feature (e.g. diabetes ICD code) as a positive label, treating its absence as unlabeled, and applying positive and unlabeled learning algorithms to learn the probability of the concept given the other covariates. We find that learned concepts (LC) for an expert-selected subset of features provide a boost in performance over the features (C-index 0.858 vs. 0.844), and that this boost places the LC model approximately halfway between the selected features model (C-index 0.844) and the model trained on all available features (All Features) or LC + All Features combined (both C-index 0.872). While there is some loss of performance going from the All Features model to the LC model, this gap seems to close quickly on subsequent time periods, suggesting that there may be some reason why the LC model is favored over time. Qualitatively, we find that some of the features important to the All Features model are incorporated into the learned concept classifiers, possibly indicating that they serve as proxies for the concepts. Finally, we publish an interactive web visualization tool at acmilab.org/severe_covid for users to explore the learned concepts, original features, and how both are utilized in our models.

4.2 Related Work

Several works have identified predictive factors for severe COVID-19, where the population studied and the definition of severity vary. Docherty et al. (2020) performed a prospective observational study on COVID-19 hospitalizations in the UK and identified risk factors for mortality including old age, male, and chronic comorbidities such as obesity. Henry et al. (2020) performed a meta-analysis of 21 studies and identified white blood cell count, lymphocytes, platelets, IL-6 and serum ferritin as inpatient biomarkers for progression to severe or fatal illness. The VACO Index (King Jr et al., 2020) uses three pre-COVID-19 health status variables, demographics, pre-existing medical conditions, and Charlson Comorbidity Index, in a mortality score. To identify severe COVID-19 patients in need of limited ventilation resources, some works(Xu et al., 2021; Zhou et al., 2020c) have predicted patient risk of developing acute respiratory distress syndrome (ARDS) using labs, demographics, and other clinical data.

Covichem(Bats et al., 2021) is an admission risk score predicting severity as defined by a composite of lab values, ARDS, or ICU admission. After stepwise model selection on the Akaike Information Criterion, Covichem identified risk factors including: obesity, cardiovascular conditions, plasma sodium, albumin, ferritin, lactate, and creatinine. COVID-GRAM(Liang et al., 2020b) predicts risk of ICU admission, invasive ventilation, or mortality for inpatients using ten predictors: chest radiography abnormality, age, hemoptysis, dyspnea, unconsciousness, comorbidity count, cancer history, neutrophil-to-lymphocyte ratio, lactate dehydrogenase, and direct bilirubin, chosen via LASSO regression. Galloway et. al.(Galloway et al., 2020) created a simple count-based risk score for predicting ICU admission or mortality, using twelve features: age, male, ethnicity, oxygen saturation, radiological severity score, neutrophils, C-reactive protein, albumin, creatinine, diabetes mellitus, hypertension, and chronic lung disease. Other works(Salaffi et al., 2020; Li et al., 2020) have used chest CTs to score severity.

Several works have used deep learning to extract embeddings of medical concepts from EHRs(Choi et al., 2018; Rasmy et al., 2021). While useful for various downstream tasks, these embeddings usually suffer a lack of transparency. As an alternative, Halpern et. al. proposed an “anchor-and-learn” framework in which expert-defined binary medical concepts are learned by treating certain informative features as positive labels for those concepts, and applying algorithms from positive and unlabeled learning (Elkan and Noto, 2008a; Halpern et al., 2016; Bekker and Davis, 2020a). An advantage of this method is the interpretable coefficients of classifiers used for learning the concepts.

4.3 Data

Cohort Description. We use retrospective observational data collected by a major healthcare provider in Southwestern Pennsylvania from January 1st, 2020 to January 12th, 2022. Out of 171,009 patients who were tested for COVID-19, we extract the 40,190 who tested positive. Of those, we remove individuals who were already mechanically ventilated or admitted to the ICU within 30 days prior to the time t_0 of testing positive for the first time. This leaves a cohort of 31,336 individuals (Table 4.1). Note that this study seeks to predict the risk of progressing to severe COVID-19 *upon testing positive for the first time*, and so features and outcomes are defined relative to time t_0.

Features. Features are extracted no later than the date of each patient’s first covid positive test. These include testing location (inpatient/outpatient), demographics, labs, medications, vaccines, symptoms, and problem history. The most recent value of each feature is extracted, and symptoms are limited to a one-day window around t_0. Since there are tens of thousands of distinct medications, labs, diagnoses, vaccines, etc. in our data, the feature pool is limited to the top 20 of each data type except for labs (top 50). Upon clinician review, 45 more features are extracted. After removing low-variance features, converting categorical values to indicators, and normalizing continuous values, this yields a fixed-length 139-dimensional feature vector (see acmilab.org/severe_covid) for patient information known at t_0.

Outcome. Since patients are right-censored upon leaving the hospital system, the outcome of interest is a time-to-event, where the time is computed as the time elapsed between t_0 and severe COVID-19 (mechanical ventilation, ICU admission, or death) or censorship (when the patient was last seen in hospital records), whichever is first.

Table 4.1: Cohort characteristics (n = 31,336). Demographics, inpatient vs. outpatient status, outcomes.

Characteristic	Count (%)
Gender	
Female	17,874 (57.0%)
Male	13,455 (42.9%)
Age	
Under 20	2,836 (9.1%)
20 – 30	3,987 (12.7%)
30 – 40	4,134 (13.2%)
40 – 50	4,155 (13.3%)
50 – 60	5,444 (17.4%)
60 – 70	5,017 (16.0%)
70 or above	5,763 (18.4%)

Characteristic	Count (%)
Location of Test	
Inpatient	13,246 (42.3%)
Outpatient	15,868 (50.6%)
Unknown	2,222 (7.1%)
Outcomes	
Severe COVID-19	5,272 (16.8%)
ICU Admission	4,811 (15.4%)
Death	1,554 (5.0%)
Mechanical ventilation	1,096 (3.5%)

4.4 Learning Clinical Concepts

Different types of data often provide partial information about higher-level concepts. For example, a saline IV bolus is typically administered inpatient, and is highly predictive of inpatient status even if inpatient status is unavailable. Certain labs could further confirm inpatient status. While one could methodically create rules for every concept of interest, it is difficult to do so comprehensively. As a result, learned models may end up using proxies that indirectly encode important risk factors (e.g. IV bolus encoding inpatient status), possibly leading to misinterpretation. Thus, we learn clinical concepts corresponding to major risk factors, and use these for downstream risk prediction.

PU Algorithm for Learning Concepts. To learn these concepts, we use the "anchor-and-learn" framework (Halpern et al., 2016). For each concept of interest, we identify some key informative observations ("positive anchors") relating to that concept. In this work, we only consider binary-valued concepts (present vs. not present). An observation is an anchor for a concept if it is conditionally independent of all other observations conditioned on the concept. When the presence of an anchor almost certainly implies the presence of the concept, this is known as a *positive anchor*.

Consider a patient with covariates $x \in \mathbb{R}^d$. Suppose we want to extract a concept c with positive anchor $x_c \in \{0, 1\}$ (e.g. extracting a diabetes concept with a diabetes diagnosis code as a positive anchor). Let $y_c \in \{0, 1\}$ be the true binary label for whether concept c is present. Note that in most observational health data, we observe the presence of a clinical condition, but not the absence of it. For example, when extracting the diabetes concept, we can be fairly confident that a patient marked as diabetic does indeed have diabetes, but patients unmarked do not necessarily *not have* diabetes. Said differently, we have positive and unlabeled (PU) data rather than positive and negative data. Since only positive examples are labeled, $y_c = 1$ is certain when $x_c = 1$, but when $x_c = 0$, then y_c could be either 0 or 1.

Thus, we leverage algorithms designed to learn from positive and unlabeled data, or "PU learning" algorithms. Let $x_{\bar{c}}$ refer to all covariates except for x_c. Since anchors are conditionally independent of all other observations conditioned on the concept, we have that $p(x_c|y_c = 1) = p(x_c|y_c = 1, x_{\bar{c}})$. Now, consider $p(x_c = 1|x_{\bar{c}})$. We have that:

$$\begin{aligned} p(x_c = 1|x_{\bar{c}}) &= p(x_c = 1 \wedge y_c = 1|x_{\bar{c}}) \\ &= p(y_c = 1|x_{\bar{c}})p(x_c = 1|y_c = 1, x_{\bar{c}}) \\ &= p(y_c = 1|x_{\bar{c}})p(x_c = 1|y_c = 1) \\ \implies p(y_c = 1|x_{\bar{c}}) &= p(x_c = 1|x_{\bar{c}})/\delta_c \end{aligned}$$

where $\delta_c = p(x_c = 1|y_c = 1)$. The first equality follows from the fact that $y_c = 1$ is certain when $x_c = 1$, and the second equality follows from Bayes rule. In words, the expression indicates that *true probability of the concept* being present is *proportional to the probability of the positive anchor being present* by a factor of δ_c. Thus, if we can train a PU classifier $g(x_{\bar{c}}) = p(x_c = 1|x_{\bar{c}})$ that learns the probability that a positive anchor is present given the remaining covariates, we need only scale the probability by δ_c in order to get the probability of the underlying concept being present. As noted in Elkan and Noto (Elkan and Noto, 2008a), for the set P of positive labeled examples, one can construct an empirical estimate of the constant δ_c as $\hat{\delta}_c = \frac{1}{n}\sum_{x_{\bar{c}} \in P} g(x_{\bar{c}})$, due to the observation that $g(x_{\bar{c}}) = \delta_c$ for $x_{\bar{c}} \in P$:

$$\begin{aligned} g(x_{\bar{c}}) &= p(x_c = 1|x_{\bar{c}}) \\ &= p(x_c = 1|x_{\bar{c}}, y_c = 1)p(y_c = 1|x_{\bar{c}}) + p(x_c = 1|x_{\bar{c}}, y_c = 0)p(y_c = 0|x_{\bar{c}}) \\ &= p(x_c = 1|x_{\bar{c}}, y_c = 1) \cdot 1 + 0 \cdot 0 \quad \text{since } x_{\bar{c}} \in P \\ &= p(x_c = 1|y_c = 1) \\ &= \delta_c. \end{aligned}$$

Finally, this yields the following procedure for learning clinical concepts:

1. **Identify clinical concepts of interest**, and corresponding positive anchors.
2. **Learn a positive vs. unlabeled classifier.** Use logistic regression to learn a classifier $g(x_{\bar{c}})$ that outputs the probability of the positive anchor given the other covariates.
3. **Estimate the scaling constant.** On a validation set, estimate $\hat{\delta}_c$ by averaging the output of $g(x_{\bar{c}})$ on all positive labeled examples (i.e. examples with the positive anchor).
4. **Scale predictions from the PU classifier** by the estimated scaling constant to get the probability that the underlying concept is present. That is, compute $p(y_c = 1|x_{\bar{c}}) = g(x_{\bar{c}})/\hat{\delta}_c$ for all examples where the positive anchor is not present. If the positive anchor is present, leave the probability as 1.

This procedure is also used in Halpern et al. (2016), except instead of drawing the concept from a Bernoulli distribution parameterized by $p(y_c = 1|x_{\bar{c}})$, we directly use the computed probability $p(y_c = 1|x_{\bar{c}})$ since it can provide more granular information. The scikit-learn (Pedregosa et al., 2011) python package was utilized for its logistic regression implementation.

Identifying Concepts of Interest. In order to define clinical concepts of interest, we surveyed several clinicians in the healthcare provider network about the main concepts they would look for when assessing risk of severe COVID-19. The survey yielded 21 concepts: old age, inpatient, outpatient, diabetes, shortness of breath, fever, cough, fatigue, COVID-19 vaccination, flu vaccination, obesity, hypertension, immunocompromised, COPD, congestive heart failure, chronic kidney disease, hyperglycemia, transplant, cancer, lung disease, and myalgia. We identify positive anchors for these concepts (precise definitions at acmilab.org/severe_covid) and apply the PU algorithm to extract a more complete representation of the concepts. Learning Severe COVID-19 Risk Using the lifelines python package (Davidson-Pilon, 2019), a Cox proportional hazards model with L1 regularization (Lasso-Cox) is used to model risk of progression to severe COVID-19. For a patient with covariates X, their hazard h at time t is given by:

$$h(t) = h_0(t) \exp(X\beta)$$

where h_0 is a baseline hazard function, and β are learned coefficients. The regularization penalty is given by $\lambda||\beta||_1$, where regularization strength λ is selected using 5-fold cross validation and grid search over penalties between 0 and 0.2, with a step size of 0.001 between each penalty. For stability of training, features with variance < 0.01 are removed.

4.5 Experimental Setup

Feature Sets. To explore the marginal effect of incorporating learned concepts vs. the original set of 139 features, we evaluate Lasso-Cox models learned from five different feature sets:

1. **Raw positive anchors**: only the positive anchors identified in the data, without learning the corresponding clinical concepts (e.g. mention of ”diabetes” in a note, ICD code, etc.)
2. **Learned concepts (LC)**: only the learned clinical concepts (e.g. the diabetes concept)
3. **LC + Numeric**: the learned concepts and numerical features (e.g. diabetes concept, labs)
4. **LC + All Features**: the learned concepts, as well as all of the original 139 features
5. **All Features**: all 139 original features, no learned concepts.

Back-testing and Data Splits. In real-world settings, hospital systems may want to use updated data to revise their models. To emulate this process, we re-train models (including PU concept classifiers) up to the end of each 3-month season (spring, summer, fall, winter), and evaluate their performance on subsequent seasons. Spring is March 20th until June 21st, followed by summer until September 22nd, followed by fall until December 21st, followed by winter until March 20th of the following year. For each 3-month period, a 70-30 split designates train and test sets, where test data is never included in any model training. To keep the risk score interpretation simple, for each time period a grid search on the Lasso-Cox penalty is done to choose a model with approximately ten features. We additionally train models on the entire study time range, with train and test sets that aggregate the respective 3-month datasets.

4.6 Evaluation

Clinical Concept Evaluation. Since the concepts are only positively labeled or unlabeled, it is not possible to compute precision of the concept classifiers (Bekker and Davis, 2020a). However, we can compute recall as the proportion of known positives recovered by the classifiers. Additionally, we examine the number of previously unlabeled samples predicted to be positive.

Model Interpretation. The Lasso-Cox model coefficients are in terms of original features as well as learned concepts. In addition to listing the Lasso-Cox coefficients, we create an interactive Sankey diagram to visualize how raw features translate into concepts, and how the resulting models pull from both. This gives the user a birds-eye view of how each concept is defined, the strength and sign of the coefficients, and which concepts are used in different models.

Survival Model Evaluation Metrics. The concordance, or C-index, is used to evaluate the model's discriminative ability. To evaluate calibration, both one-calibration at 14 days and D-calibration are used (Haider et al., 2020). Additionally, low, medium, and high-risk strata are defined and their 14-day Kaplan-Meier survival curves are inspected.

Baselines. We compare our model performance to that of the Covichem(Bats et al., 2021) and Galloway(Galloway et al., 2020) risk scores. For Covichem, in order to conduct a fairer comparison than directly applying their logistic regression coefficients learned on a different population, we extract the same features and re-train logistic regression on our own training data. For Galloway, we extract all but one of their twelve features (radiological severity is not available in our data), and compare performance against two versions of their model: (1) directly applying their proposed count-based risk score (Galloway count), and (2) re-training a logistic regression model using the twelve variables (Galloway reweighted).

4.7 Results

The PU learning algorithm yields concepts ranging from those with high recall, e.g. 0.974 for inpatient status, to low recall, e.g., 0.381 for immunocompromised (Table 4.2). Some concepts have substantially more new positives (obesity, with 2,157 new positives). Concept classifier coefficients are available at acmilab.org/severe_covid.

The model trained on learned concepts (LC) achieves a higher aggregate concordance than the original features corresponding to those concepts (0.858 vs. 0.844, Table 4.3). The model learned from all original features (All Features) and LCs (C-index 0.872) performs comparably to All Features alone (C-index 0.872). The addition of numerical features to the LCs does not significantly improve performance (C-index of both are 0.858). In all models except for the ones trained on All Features or All Features + LCs, the aggregate C-index is higher than the C-indices on the inpatient and outpatient subpopulations.

In the models using all features, medications such as dexamethasone, acetaminophen, and

Table 4.2: The number of new positives extracted by PU learning in the test set, the number of clinical concepts originally in the test set (determined solely by the presence of positive anchors), and the recall of the PU classifier among known positives in the test set. Concepts with prevalence $< 1.5\%$ are omitted.

Learned Concept (LC)	**New Positives (% of Test Set)**	**Original Positives (% of Test Set)**	**Recall Among Original Positives (count)**
Old age	147 (1.6%)	3,158 (33.7%)	0.956 (3,019)
Inpatient	227 (2.4%)	3,914 (41.8%)	0.974 (3,813)
Outpatient	207 (2.2%)	4,811 (51.3%)	0.961 (4,623)
Diabetes	594 (6.3%)	834 (8.9%)	0.553 (461)
Fever	4,171 (44.5%)	1,021 (10.9%)	0.826 (843)
Shortness of breath	1,518 (16.2%)	1,005 (10.7%)	0.767 (771)
COVID-19 vaccination	2,128 (22.7%)	1,884 (20.1%)	0.739 (1,392)
Flu vaccine	2,326 (24.8%)	4,191 (44.7%)	0.864 (3,619)
Obesity	2,157 (23.0%)	433 (4.6%)	0.610 (264)
Immunocompromised	909 (9.7%)	168 (1.8%)	0.381 (64)
COPD	575 (6.1%)	220 (2.3%)	0.623 (137)
Hyperglycemia	496 (5.3%)	171 (1.8%)	0.737 (126)
Cough	4,023 (42.9%)	1,862 (19.9%)	0.815 (1,517)
Fatigue	2,947 (31.4%)	602 (6.4%)	0.694 (418)

intravenous saline are selected (Table 4.4). Across all models, the inpatient status is the feature with the greatest hazard ratio. Blood urea nitrogen is used by both the LC + All Features and All Features models. Figure 4.1 is a screenshot of an interactive Sankey diagram which allows users to explore the coefficients for both the underlying clinical concepts and the classifiers built on top of the features and concepts. The interactive web tool is available at acmilab.org/severe_covid.

When evaluated over time, the models with learned concepts achieve higher concordance several months after the model was initially trained, whereas the All Features model achieves higher concordance in the immediate term. For example, in Spring 2020, the concordance of the All Features model trained up until the end of Spring 2020 is 0.842, compared to 0.797 in the LC only model. By Fall and Winter 2021, however, the All Features model degrades to 0.808 or stays around 0.845, whereas the LC only model actually increases concordance to 0.83 and 0.904. Reading the table from left to right, the performance of any model fluctuates no more than 0.121 over all seasons. Reading the table from top to bottom, several columns shown an increase in performance as models are trained on more recent data.

The Kaplan-Meier curves corresponding to the low, medium, and high risk groups derived from the LC + All Features model and LC only model predictions are shown in Figure 4.2. There is a clear separation between the survival trajectories of the different risk groups. The LC + all features model appears to slightly under-estimate the risk of the high-risk groups, whereas the LC only model appears to be better calibrated at 14 days (Figure 4.3). The LC only model also appears to have better d-calibration across all time points (Figures 4.4).

Table 4.3: Median Test C-index of models and baselines, with 95% CI are reported from bootstrapping the test set with 1000 replicates. Bold highlights the two models with highest C-index.

Model	**Aggregate**	**Inpatient**	**Outpatient**
Covichem	0.598 (0.580 – 0.616)	0.584 (0.569 – 0.600)	0.546 (0.509 – 0.581)
Galloway count	0.745 (0.734 – 0.757)	0.647 (0.633 – 0.662)	0.714 (0.677 – 0.750)
Galloway reweighted	0.810 (0.803 – 0.824)	0.699 (0.673–0.703)	0.764 (0.728–0.709)
Raw positive anchors	0.844 (0.836 – 0.851)	0.665 (0.650 – 0.680)	0.756 (0.709 – 0.796)
Learned concepts (LC) only	0.858 (0.851 – 0.865)	0.699 (0.685 – 0.713)	0.798 (0.757 – 0.834)
LC + numerical features	0.858 (0.851 – 0.865)	0.695 (0.681 – 0.71)	0.814 (0.777 – 0.849)
LC + all features	0.872 (0.865 – 0.877)	0.715 (0.702 – 0.728)	0.879 (0.858 – 0.901)
All features (no LC)	0.872 (0.866 – 0.878)	0.717 (0.703 – 0.730)	0.880 (0.860 – 0.901)

Table 4.4: Hazard ratios (HR) of LC + All Features, LC, and All Features models. Abbreviations: Med = Medication, loc. = location, Dex. = Dexamethasone sodium phosphate, APAP = acetaminophen, SOB = Shortness of breath, BUN = blood urea nitrogen, NEUT = neutrophils, Immunocomp. = immunocompromised, vax = vaccine, OP = outpatient.

All Features + LCs		**Learned Concepts (LC) Only**		**All Features Only**	
Features	HR (95% CI)	Features	HR (95% CI)	Features	HR (95% CI)
(LC) Inpatient	2.31 (2.09 – 2.56)	(LC) Inpatient	7.23 (5.43 – 9.62)	(Test Location) Inpatient	3.62 (3.31 – 3.97)
(LC) SOB	1.72 (1.58 – 1.88)	(LC) Old age	2.54 (2.33 – 2.77)	(Med) Dex. 4mg/mL injection sol.	1.91 (1.77 – 2.06)
(Med) Dex. 4mg/mL injection sol.	1.54 (1.42 – 1.68)	(LC) SOB	2.31 (2.14 – 2.49)	(Med) APAP 325mg tablet	1.67 (1.51 – 1.85)
(Med) APAP 325 mg tablet	1.47 (1.34 – 1.62)	(LC) Diabetes	1.28 (1.16 – 1.41)	Age 70+	1.60 (1.49 – 1.72)
(LC) Old age	1.34 (1.25 – 1.44)	(LC) COPD	1.22 (1.13 – 1.32)	(Med) NaCl 0.9% IV sol.	1.35 (1.24 – 1.46)
(Med) NaCl 0.9 % IV sol.	1.33 (1.23 – 1.44)	(LC) Obesity	1.22 (1.14 – 1.3)	(OP ICD) SOB	1.10 (1.01 – 1.19)
(Lab) BUN	1.05 (1.01 – 1.1)	(LC) Immuno-compromised	1.10 (1.04 – 1.16)	(Lab) BUN	1.10 (1.05 – 1.14)
		(LC) Fatigue	1.08 (1.02 – 1.15)	(Med) Pantoprazole 40 mg tablet	1.05 (0.97 – 1.12)
		(LC) Outpatient	1.06 (0.80 – 1.42)	(Lab) NEUT relative %	1.04 (0.99 – 1.09)
		(LC) Hyper-glycemia	0.90 (0.81 – 1.00)	(Med) NaCl 0.9% IV Bolus	1.04 (0.96 – 1.13)
		(LC) COVID-19 vax	0.87 (0.78 – 0.97)	(Lab) Albumin	0.98 (0.93 – 1.02)
		(LC) Fever	0.82 (0.75 – 0.90)		
		(LC) Flu vax	0.74 (0.66 – 0.83)		
		(LC) Cough	0.58 (0.53 – 0.65)		

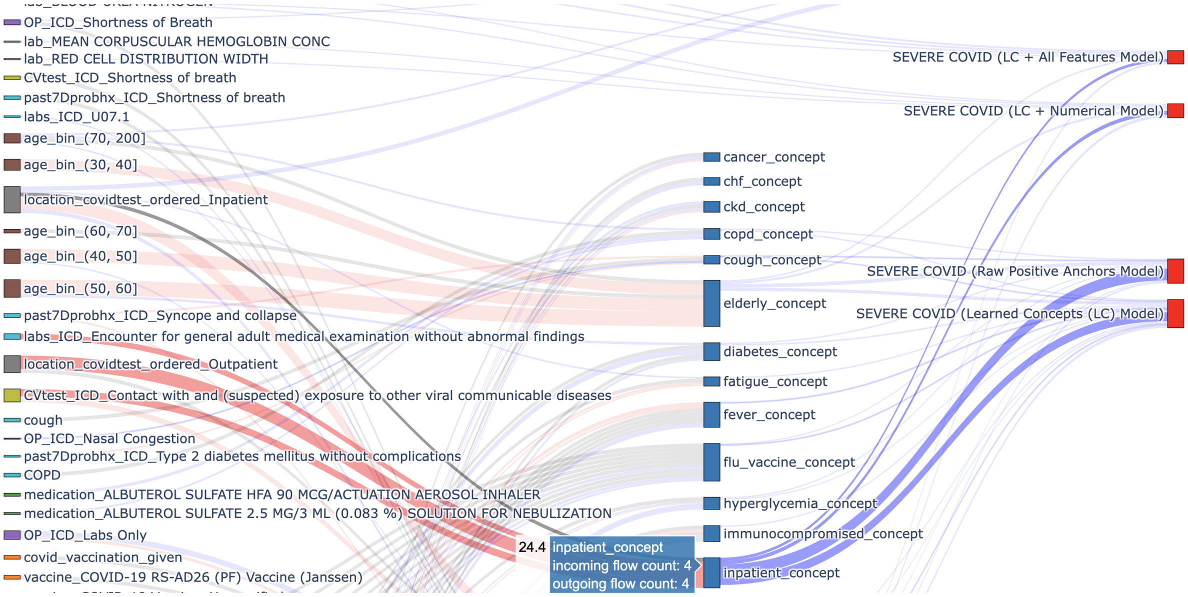
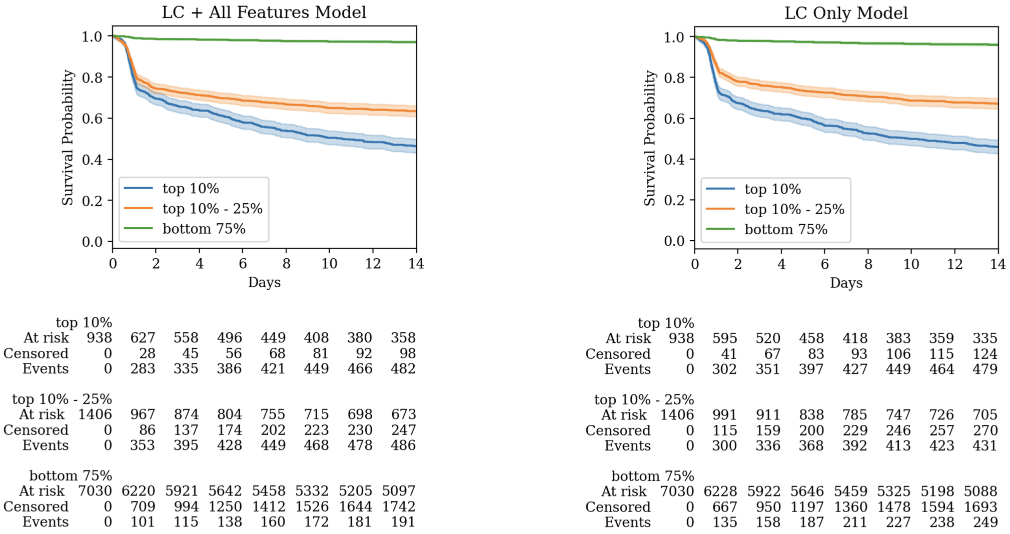

OP_ICD_Shortness of Breath
lab_MEAN CORPUSCULAR HEMOGLOBIN CONC
lab_RED CELL DISTRIBUTION WIDTH
CVtest_ICD_Shortness of breath
past7Dprobhx_ICD_Shortness of breath
labs_ICD_U07.1
age_bin_(70, 200]
age_bin_(30, 40]
location_covidtest_ordered_Inpatient
age_bin_(60, 70]
age_bin_(40, 50]
age_bin_(50, 60]
past7Dprobhx_ICD_Syncope and collapse
labs_ICD_Encounter for general adult medical examination without abnormal findings
location_covidtest_ordered_Outpatient
CVtest_ICD_Contact with and (suspected) exposure to other viral communicable diseases
cough
OP_ICD_Nasal Congestion
past7Dprobhx_ICD_Type 2 diabetes mellitus without complications
COPD
medication_ALBUTEROL SULFATE HFA 90 MCG/ACTUATION AEROSOL INHALER
medication_ALBUTEROL SULFATE 2.5 MG/3 ML (0.083 %) SOLUTION FOR NEBULIZATION
OP_ICD_Labs Only
covid_vaccination_given
vaccine_COVID-19 RS-AD26 (PF) Vaccine (Janssen)
cancer_concept
chf_concept
ckd_concept
copd_concept
cough_concept
elderly_concept
diabetes_concept
fatigue_concept
fever_concept
flu_vaccine_concept
hyperglycemia_concept
immunocompromised_concept
inpatient_concept
24.4
inpatient_concept
incoming flow count: 4
outgoing flow count: 4
SEVERE COVID (LC + All Features Model)
SEVERE COVID (LC + Numerical Model)
SEVERE COVID (Raw Positive Anchors Model)
SEVERE COVID (Learned Concepts (LC) Model)

Figure 4.1: Screenshot of interactive Sankey diagram showing how raw features (first column) translate into clinical concepts (second column), and how both are ultimately used in each model (third column). Magnitude of coefficients correspond to flow thickness, positive log HRs are blue, and negative log HRs are red. Black flows indicate positive anchors for the corresponding concept. Visit acmilab.org/severe_covid to interact with the full diagram.



LC + All Features Model

top 10%								
At risk	938	627	558	496	449	408	380	358
Censored	0	28	45	56	68	81	92	98
Events	0	283	335	386	421	449	466	482
top 10% - 25%								
At risk	1406	967	874	804	755	715	698	673
Censored	0	86	137	174	202	223	230	247
Events	0	353	395	428	449	468	478	486
bottom 75%								
At risk	7030	6220	5921	5642	5458	5332	5205	5097
Censored	0	709	994	1250	1412	1526	1644	1742
Events	0	101	115	138	160	172	181	191

LC Only Model

top 10%								
At risk	938	595	520	458	418	383	359	335
Censored	0	41	67	83	93	106	115	124
Events	0	302	351	397	427	449	464	479
top 10% - 25%								
At risk	1406	991	911	838	785	747	726	705
Censored	0	115	159	200	229	246	257	270
Events	0	300	336	368	392	413	423	431
bottom 75%								
At risk	7030	6228	5922	5646	5459	5325	5198	5088
Censored	0	667	950	1197	1360	1478	1594	1693
Events	0	135	158	187	211	227	238	249

Figure 4.2: Kaplan Meier survival curves for the high (top 10%), medium (top 10-25%), and low (bottom 75%) risk groups using predictions from the LC + All Features model (left) and the LC only model (right). Counts at the bottom show the number of individuals who are at risk, are censored, or experienced the severe COVID-19 event across time.

Table 4.5: Back-testing performance of All Features, LC + All Features, and LC only over 3-month seasons. Spring (SP) is March 20th until June 21st, followed by summer (SU) until September 22nd, followed by fall (F) until December 21st, followed by winter (W) until March 20th.

All Features only,	**Test C-index evaluated on:**							
trained up to:	SP 2020	SU 2020	F 2020	W 2020	SP 2021	SU 2021	F 2021	W 2021
End of spring 2020	0.842	0.903	0.855	0.839	0.804	0.841	0.808	0.845
End of summer 2020	-	0.713	0.694	0.697	0.622	0.699	0.667	0.711
End of fall 2020	-	-	0.882	0.868	0.813	0.855	0.84	0.907
End of winter 2020	-	-	-	0.749	0.646	0.718	0.718	0.735
End of spring 2021	-	-	-	-	0.818	0.856	0.844	0.908
End of summer 2021	-	-	-	-	-	0.859	0.847	0.91
End of fall 2021	-	-	-	-	-	-	0.850	0.911
1/12/2022 (study end)	-	-	-	-	-	-	-	0.912

All Features + LCs,	**Test C-index evaluated on:**							
trained up to:	SP 2020	SU 2020	F 2020	W 2020	SP 2021	SU 2021	F 2021	W 2021
End of spring 2020	0.791	0.840	0.852	0.837	0.774	0.814	0.805	0.876
End of summer 2020	-	0.847	0.852	0.845	0.775	0.806	0.818	0.891
End of fall 2020	-	-	0.852	0.845	0.781	0.812	0.822	0.896
End of winter 2020	-	-	-	0.846	0.778	0.811	0.823	0.896
End of spring 2021	-	-	-	-	0.778	0.813	0.827	0.899
End of summer 2021	-	-	-	-	-	0.812	0.826	0.899
End of fall 2021	-	-	-	-	-	-	0.827	0.900
1/12/2022 (study end)	-	-	-	-	-	-	-	0.900

LCs only,	**Test C-index evaluated on:**							
trained up to:	SP 2020	SU 2020	F 2020	W 2020	SP 2021	SU 2021	F 2021	W 2021
End of spring 2020	0.797	0.863	0.869	0.850	0.795	0.833	0.83	0.904
End of summer 2020	-	0.857	0.863	0.842	0.785	0.819	0.822	0.904
End of fall 2020	-	-	0.867	0.850	0.800	0.828	0.831	0.909
End of winter 2020	-	-	-	0.854	0.801	0.826	0.834	0.911
End of spring 2021	-	-	-	-	0.803	0.828	0.834	0.911
End of summer 2021	-	-	-	-	-	0.829	0.833	0.910
End of fall 2021	-	-	-	-	-	-	0.837	0.914
1/12/2022 (study end)	-	-	-	-	-	-	-	0.913

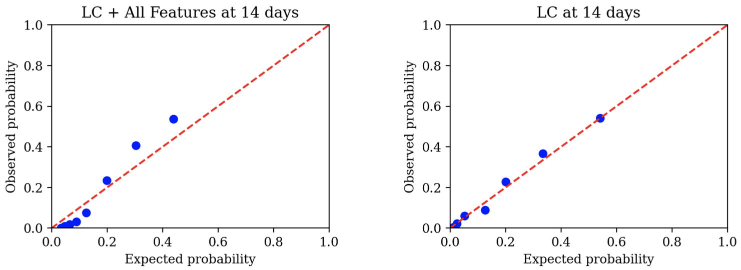


Figure 4.3: One-calibration of the LC + All Features model (left) and LC model (right) at 14 days, binned into ten groups. Red dotted line corresponds to perfect calibration.

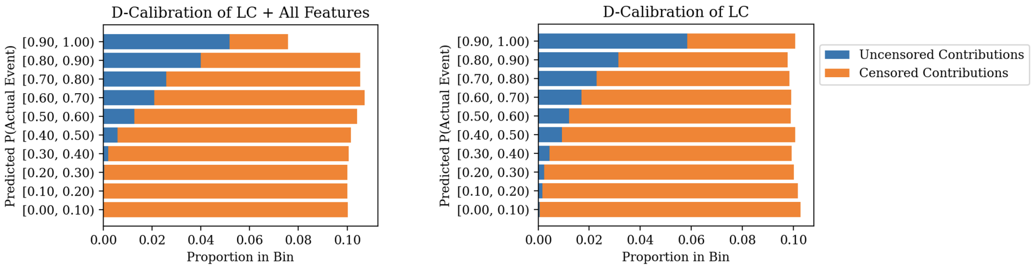


Figure 4.4: D-calibration histogram of LC + All Features model (left) and LC model (right), binned into ten groups. In perfect D-calibration (Haider et al., 2020), all bars should be at 0.10.

4.8 Discussion

Learned Concept Classifiers. The strongest coefficients for each concept classifier are often not features that immediately come to mind, but nevertheless match clinical intuition. High PF flu vaccine has a large coefficient (3.31) for the old age concept, and is a vaccine only given to patients 65 and older. Shortness of breath (SOB) during outpatient visit (3.23) is the second highest coefficient for the inpatient concept, possibly indicating that outpatients with the SOB symptom are at high risk of becoming an inpatient. The SOB concept depends on dexamethasone (0.75) which relieves inflammation, and albuterol sulfate (0.67), prescribed for lung conditions. For the obesity concept, sleep apnea (often caused by excess weight) has largest coefficient (0.96).

However, learned concepts are an imperfect representation of the underlying concept. None of the concept classifiers perfectly recover the original positives, with recall ranging from 0.381 to 0.974 (Table 4.2). This could be due to insufficient signal in the remaining covariates or underfitting of the simple model class. Additionally, some concepts learn substantially more positives than were available in the original data. For example, obesity originally has 433 positives in the data but the concept classifier marks 2,157 patients as having obesity with probability greater than 0.5. It is difficult to verify the faithfulness of the concept classifiers to the true concepts without manual review, but it is possible that the concepts classifiers may mark patients as "obesity-like" based on their other covariates rather than learning whether they truly have obesity. Additionally, while learned concepts are amenable to interpretation through the coefficients of the concept classifier, they still require domain expertise to manually define positive anchor variables. Finally, the conditional independence assumptions of the selected anchors may not hold in practice, and these assumptions are difficult to verify.

Learned Survival Models. The coefficients for the survival models trained on LCs, All Features, and LC + All Features are mostly consistent with clinical intuition. Inpatients are more likely to experience adverse outcomes than those not hospitalized, old age is well-documented to be associated with higher COVID-19 death rates (Docherty et al., 2020; Liang et al., 2020b; Galloway et al., 2020), shortness of breath indicates respiratory involvement, and medications such as dexamethasone, acetaminophen, and intravenous saline are given to hospitalized patients. Higher BUN indicates worse liver and kidney function, and COVID-19 vaccines are designed to protect against severe COVID-19. While it is surprising that some of the learned concepts for fever and cough symptoms have negative HRs, upon inspection we find that these are most reliably recorded for outpatients and may encode some additional information about outpatient status. From exploring the interactive visualization, some of the features selected by the All Features model are used by LCs in the LC + All Features model, possibly indicating that they serve as proxies for higher level concepts. For example, the saline IV bolus, present only in the All Features model, is used in the inpatient concept classifier with a positive coefficient. We also note that the coefficients are likely shrunken towards zero due to the Lasso penalty, and the non-informative or independent censoring assumption of the Cox proportional hazards model may not hold since censoring occurs upon discharge.

Our Lasso-Cox models all outperform the baselines (Covichem, Galloway count, Galloway reweighted) in terms of aggregate, inpatient, and outpatient concordance (Table 4.3). As measured by aggregate concordance, we observe that the learned concepts provide a boost in performance over the raw positive anchors (C-index 0.858 vs. 0.844). This boost in performance places the LC model approximately halfway between the performance of the raw positive anchors and the LC + All Features and All Features models, which both achieve a C-index of 0.872. For LC + All Features and All Features models, the C-index on the outpatient subpopulation (0.879 and 0.880) is higher than that on the entire cohort, whereas the C-index on the inpatient subpopulation (0.715 and 0.717) is lower. For all remaining models, the performance on the inpatient and outpatient subpopulations is lower than in aggregate, possibly indicating that it is easy to order the relative risks of inpatients versus outpatients. The LC model appears slightly better calibrated than the LC + All Features model, but when used to stratify patients into high, medium, and low-risk strata, both models yield groups with clear separation between their survival curves. Finally, while there is some loss of discriminative performance going from the All Features models to the LC only model, when tested under the back-testing framework this gap seems to close quickly on subsequent time periods and the LC model even eventually surpasses the performance of the All Features model. Thus, models with LCs might be more resilient over time than models learned only from All Features. If the set of important high-level concepts themselves change over time, however, new concepts may need to be learned accordingly.

Future Work. We plan to integrate our models with the healthcare system, and continue to monitor the performance of the models over time. It would be insightful to further study how high-level clinical concepts perform across different settings, and as COVID-19 continues to evolve over time, investigation of new concepts will be important as well.

Chapter 5

Domain Adaptation under Missingness Shift

Rates of missing data often depend on record-keeping policies and thus may change across times and locations, even when the underlying features are comparatively stable. In this paper, we introduce the problem of Domain Adaptation under Missingness Shift (DAMS). Here, (labeled) source data and (unlabeled) target data would be exchangeable but for different missing data mechanisms. We show that if missing data indicators are available, DAMS reduces to covariate shift. Addressing cases where such indicators are absent, we establish the following theoretical results for underreporting completely at random: (i) covariate shift is violated (adaptation is required); (ii) the optimal linear source predictor can perform arbitrarily worse on the target domain than always predicting the mean; (iii) the optimal target predictor can be identified, even when the missingness rates themselves are not; and (iv) for linear models, a simple analytic adjustment yields consistent estimates of the optimal target parameters. In experiments on synthetic and semi-synthetic data, we demonstrate the promise of our methods when assumptions hold. Finally, we discuss a rich family of future extensions.

5.1 Introduction

As of October 2021, following extensive awareness campaigns and mass distribution efforts promoting COVID-19 vaccines, approximately 79.2% of the U.S. population over age 18 had received at least one dose (CDC, 2022). And yet, when collaborating with a regional healthcare provider, we found only 40.5% of 121,329 adults tested for COVID-19 were tagged indicating positive vaccination status in the electronic medical record (EMR). This was not a regional anomaly—cross referencing with vaccination data from the CDC, between 75.7% and 90.3% of the adult population in the region had actually received at least one dose. A more plausible explanation is that many patients were vaccinated outside of the hospital system (e.g., at a pharmacy or football stadium) but that this information was never reported to the hospital system and thus never captured in the EMR.

Now suppose that our collaborator decided to update their intake form to include a question about vaccination status. Overnight, the rate of patients being tagged in the EMR as vaccinated would increase dramatically. Absent any shift in the actual health status of patients, the distribution of observed data would still shift, owing to this sudden change in clerical practices. In real-world healthcare settings, such changes in missingness rates are common. Furthermore, as in our vaccination example, indicators disambiguating which features are genuinely negative (vs. missing) cannot be taken for granted. Faced with data from different time periods or locations, each characterized by different patterns of missing data, how should machine learning (ML) practitioners leverage the available data to get the best possible predictor on a target domain? While missing data and formal models of distribution shift are both salient concerns of the ML community, no work to date provides guidance on how to adjust a predictor under such shocks.

In this work, we introduce *missingness shift*, where distributional shocks arise due to changes in the pattern of missingness (Figure 5.1). In this setup, all domains share a fixed underlying distribution $P(X, Y)$, and observed covariates $\widetilde{X}$ are produced by stochastically zeroing out a subset of the underlying *clean* covariates, i.e., each $\widetilde{X} = X \odot \xi$ for some $\xi \in \{0, 1\}^d$. We propose the **Domain Adaptation under Missingness Shift** problem, where the learner aspires to recover the optimal target predictor given *labeled* data from the source distribution $P^s(\widetilde{X}, Y)$, and unlabeled data from the target (deployment) distribution $P^t(\widetilde{X})$.

We focus primarily on a special DAMS setting where the components of ξ's (one per feature) are drawn from independent Bernoullis with unknown constant probabilities. First, we show that when missingness indicators $(1 - \xi)$ are available, missingness shift is an instance of covariate shift. However, absent indicators, missingness shift constitutes neither covariate shift nor label shift. Thus, adaptation is required. We demonstrate that under DAMS, the optimal source predictor may even exhibit arbitrarily higher MSE than just guessing the label mean $\mathbb{E}[Y]$. One natural strategy might be to relate the source and target distributions to the underlying clean distribution, which we show is identified when missingness rates are known. However, we show that the missingness rates are not, in general, identifiable. Fortunately, as we prove, the target distribution (and thus optimal target predictor) is nevertheless identifiable, requiring only that we estimate the (observable) relative proportions of nonzero values for each covariate across domains. Using these relative proportions, we derive a simple adjustment formula that yields the optimal linear predictor on the target domain. Additionally, we provide a non-parametric, model-agnostic procedure which attempts to transform source data into labeled data i.i.d. to the target distribution. Finally, we confirm the validity of our approach and demonstrate empirical gains in settings where our assumptions hold through synthetic and semi-synthetic experiments.

5.2 Related Work

There is a rich history of learning under various missing data mechanisms when missing data indicators are available (Rubin, 1976; Robins et al., 1994; Little and Rubin, 2019; Gelman et al., 2020). Common practices for handling missing data include discarding all samples with missingness (complete-case analysis) (Little and Rubin, 2019), imputing with mean or last value carried forward, combining inferences from multiple imputations (Rubin, 1996; Van Buuren and Groothuis-Oudshoorn, 2011), matching-based algorithms, iterative regression imputation (Stekhoven and Bühlmann, 2012; Le Morvan et al., 2021), building missingness indicators into model architecture (Le Morvan et al., 2020a), and including missingness indicators as features (Groenwold et al., 2012; Lipton et al., 2016b; Little and Rubin, 2019). However, these techniques require indicators for whether each covariate is missing in the first place.

In single cell RNA sequencing, missing data indicators are often absent in count data due to dropout, where a tiny proportion of the transcripts in each cell are sequenced, so expressed transcripts can go undetected and are instead assigned a zero value. This is often dealt with by leveraging domain-specific knowledge to inform probabilistic models, such as assuming a zero-inflated negative binomial distribution of counts (Risso et al., 2018), using a mixture model to identify likely missing values before imputing with nonnegative least squares regression (Li and Li, 2018), adopting a Bayesian approach to estimate a posterior distribution of gene expressions (Huang et al., 2018), or graph-based methods on a lower dimensional manifold derived from principal component analysis (Van Dijk et al., 2018).

In survey data, underreporting (i.e. missingness without indicators) arises in binary data when respondents give false negative responses to questions. As noted in Sechidis et al. (2017), this can be viewed as a form of misclassification bias. In its simplest form, an underreported variable has *specificity* $p(\widetilde{x} = 0|x = 0) = 1$ and *sensitivity* $p(\widetilde{x} = 1|x = 1) < 1$ (one minus the rate of missingness). If sensitivity is independent of outcome Y, this is referred to as *non-differential* misclassification, which often, but not always biases measures of association towards zero (Dosemeci et al., 1990; Brenner and Loomis, 1994). Given knowledge of the specificities and sensitivities, prior work has derived adjusted estimators for the log-odds ratio (Chu et al., 2006) and relative-risk (Rahardja and Young, 2021) under non-differential exposure misclassification. Recent work has also provided conditions under which the joint distribution $p(y, \widetilde{a}|x)$ (outcome y, single binary underreported exposure $\widetilde{a}$, and fully observed covariates x) is identifiable (Adams et al., 2019).

In our setting, for binary covariates, estimating the missingness rates takes the form of learning from positive and unlabeled data (Elkan and Noto, 2008b; Bekker and Davis, 2020b). Here, identification of the missingness rates hinges on the existence of a separable positive subdomain (Garg et al., 2021), which may not hold in problems of interest. Many canonical distribution shift problems address adaptation under different forms of structure, including covariate shift (Shimodaira, 2000b; Zadrozny, 2004; Huang et al., 2006; Sugiyama et al., 2007b; Gretton et al., 2009), label shift (Saerens et al., 2002; Storkey et al., 2009; Zhang et al., 2013; Lipton et al., 2018; Garg et al., 2020), and concept drift (Tsymbal, 2004; Gama et al., 2014). We show that missingness shift with missing data indicators can often be reinterpreted as a form of covariate shift, but to our knowledge, missingness shift without indicators does not fit neatly into any previous setting.

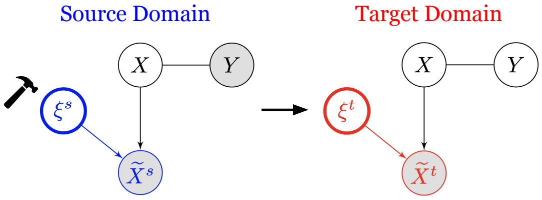


Figure 5.1: DAMS with UCAR. The source and target data are drawn from the same $P(X, Y)$, but differ in how ξ (and hence $\widetilde{X}$) takes its value. Shaded nodes are observed. Observed covariates are generated as $\widetilde{X} = X \odot \xi$. The undirected edge between X and Y indicates that they can have an arbitrary bidirectional relationship.

5.3 DAMS Problem Setup

First, we define notation for (1) missing data; (2) missingness shift; and (3) the DAMS problem. Then, motivated by the medical setting, we focus on a specific form of DAMS (Figure 5.1) for the remainder of the paper.

Let us denote *clean* covariates $X \in \mathbb{R}^d$ and labels $Y \in \mathbb{R}$. Let X_j denote the jth covariate, for $j \in \{1, 2, ..., d\}$.

Missing Data In every environment e with missing data, we do not directly observe X, but instead observe *corrupted* covariates:

$$\widetilde{X} = X \odot \xi,$$

where $\xi \in \{0, 1\}^d$ and $(X, Y, \xi) \sim P^e$ for distribution P^e. Note that mask ξ is the complement of missing data indicators $(1 - \xi)$. In this paper, we assume no missingness in Y in labeled data. An important assumption of missing data problems is how ξ takes its value, e.g. independent of other covariates, dependent on other covariates, etc. Furthermore, ξ may or may not be observed.

Definition 1 (Missingness Shift). *Consider a source domain s and target domain t in which X and Y are drawn from the same underlying distribution, i.e. $P(X, Y) = P^s(X, Y) = P^t(X, Y)$.* Missingness shift *occurs when the missing data mechanism differs between s and t, i.e. $P^s(\xi|\cdot) \neq P^t(\xi|\cdot)$.*

Domain Adaptation under Missingness Shift Suppose missingness shift occurs between source domain s and target domain t. Given observations of corrupted labeled source data $\{(\widetilde{X}^{s,i}, Y^{s,i})\}_{i=1}^{n_s}$ where $(\widetilde{X}^{s,i}, Y^{s,i}) \sim P^s(\widetilde{X}, Y)$, as well as corrupted unlabeled target data $\{\widetilde{X}^{t,i}\}_{i=1}^{n_t}$ where $\widetilde{X}^{t,i} \sim P^t(\widetilde{X})$, the goal of DAMS is to learn an optimal predictor on the corrupted target domain data. In this paper, we focus on regression-type tasks, where optimality is measured by the squared error on the corrupted target domain data, and we seek the optimal predictor $\mathbb{E}_{(\widetilde{X}^t, Y) \sim P^t}[Y|\widetilde{X}^t]$.

As we will show (in Section 5.4), DAMS is particularly challenging when missing data indicators are *not available*. This setting without observing ξ is trickiest when there are a substantial number of true 0 values that now become indistinguishable from missing values. Without knowledge of which data are missing versus true 0s, conventional techniques for imputing missing entries do not apply. To make this difficult setting tractable, we define the DAMS with underreporting completely at random (UCAR) setting, which we focus on in this paper.

DAMS with UCAR Assume that ξ (unobserved) is drawn independently of other variables, and parameterized by *constant (but unknown) missingness rates* $m^s \in [0,1]^d$ in source and $m^t \in [0,1]^d$ in target. That is, $\forall j \in \{1, 2, ..., d\}$, we have independently drawn $\xi^s_j \sim \text{Bernoulli}(1 - m^s_j)$ and $\xi^t_j \sim \text{Bernoulli}(1 - m^t_j)$, abbreviated as:

$$\xi^s \sim \text{Bernoulli}(1 - m^s)$$
$$\xi^t \sim \text{Bernoulli}(1 - m^t).$$

For binary data, this setting without missingness indicators is known as *underreporting*. We thus refer to this setting as underreporting completely at random, but note our results are not limited to binary data.

5.4 Cost of Non-Adaptivity

Here, we provide intuition on the cost of not adapting the source predictor to the target domain in DAMS with UCAR. Let us start with a simple motivating example. Define the risk of an estimator $\widehat{h}$ to be $r(\widehat{h}) = \mathbb{E}[(Y - \widehat{h}(X))^2]$.

Example 1 (Redundant Features). *Let* $m^s = [1 - \epsilon, \epsilon]$ *and* $m^t = [\epsilon, 1 - \epsilon]$. *Consider the data generating process:*

$$\begin{aligned} Z &= u_Z & & \\ X_1 &= Z & u_Z &\sim \mathcal{N}(0, \sigma_z^2) \\ X_2 &= Z & u_Y &\sim \mathcal{N}(0, \sigma_y^2) \\ Y &= Z + u_Y & & \end{aligned}$$

where σ_z is a positive constant, Z is a latent variable, X_1 and X_2 are observed, and Y is the outcome of interest.

Remark 1. *In Example 1, as $\epsilon \to 0$, the optimal linear source and target predictors have coefficients $\beta^s_* \to [0, 1]$ and $\beta^t_* \to [1, 0]$. The risk on target data $r^t(\beta^s_*) \to Var(Y)$.*

That is, failing to adapt to the target levels of missingness results in performance no better than simply guessing the label mean (proof in Appendix D.1). Now, let us consider a slightly more complex example.

Example 2 (Confounded Features)**.** *Now, suppose that* $m^s = [0, 0]$ *and* $m^t = [1, 0]$*. For some constants* a, b, c *consider the following data generating process:*

$$\begin{aligned} X_1 &= \nu_1 & \nu_1 &\sim \mathcal{N}(0,1) \\ X_2 &= aX_1 + \nu_2 & \nu_2 &\sim \mathcal{N}(0,1) \\ Y &= bX_1 + cX_2 + \nu_Y & \nu_Y &\sim \mathcal{N}(0,1) \end{aligned}$$

Remark 2. *In Example 2, the optimal source and target predictors are* $\beta^s_* = [b, c]$ *and* $\beta^t_* = [0, \frac{ab}{a^2+1} + c]$*. By setting* $a = -\frac{b}{c}$*, we can show that for any* $\tau > 1$*, there exists values of* a, b, c *such that* $r^t(\beta^s_*) > \tau\, Var(Y)$*.*

Here, failing to adapt to target levels of missingness can result in performance arbitrarily *worse* than predicting the constant label mean (proof in Appendix D.1).

Observing ξ, Reduction to Covariate Shift In DAMS with UCAR, missing data indicators are absent. By contrast, *suppose we observed missingness indicators* $(1 - \xi)$ (and hence ξ). Then, we show that missingness shift is an instance of covariate shift, where the optimal predictor does not change across domains. This result holds not only when ξ is drawn independently of other covariates, but also when it is dependent on other completely observed covariates (proof in Appendix D.2). Here, when ξ is "drawn independently of other covariates," as described in the DAMS with UCAR setup (Section 5.3), we have that $\xi \sim$ Bernoulli$(1 - m)$ for some constant vector of missingness rates $m \in [0, 1]^d$. When ξ is drawn depending only on other completely observed covariates, we have that some subset of covariates $X_c \subseteq X$ is completely observed (i.e. no missingness), and the missingness of the other covariates $X_m = X \setminus X_c$ depends on X_c. That is, $\xi \sim$ Bernoulli$(f(X_c))$ for some function $f : \mathbb{R}^{|X_c|} \to [0, 1]^{|X_m|}$. Mohan and Pearl (2021) classifies these missingness mechanisms as MCAR (missing completely at random) and v-MAR (variant of the missingness at random described by Rubin (1976)), respectively.

Proposition 1 (Reduction to Covariate Shift)**.** *Assume we observe* ξ*. Consider augmented covariates* $\tilde{x}' = (\tilde{x}, \xi)$*. When* ξ *is drawn independently of other covariates or depending only on other completely observed covariates, missingness shift satisfies the covariate shift assumption, i.e,* $P^s(Y|\widetilde{X}' = \tilde{x}') = P^t(Y|\widetilde{X}' = \tilde{x}')$*.*

Covariate shift problems are well-studied (Shimodaira, 2000b; Zadrozny, 2004; Huang et al., 2006; Sugiyama et al., 2007b; Gretton et al., 2009). When source and target distributions have shared support, covariate shift only requires adaptation under model misspecification (Shimodaira, 2000b), where the most common approach is to re-weight examples according to $p^t(x)/p^s(x)$, rendering the (re-weighted) training and target data exchangeable. However, even given missingness indicators, DAMS may still require some care. For example, in the augmented covariate space (with missing data indicators), one might need more complex models than in the original covariate space. When re-weighting is necessary, the structure of the DAMS problem might be leveraged to estimate importance weights more efficiently, or to identify the optimal target predictor in certain cases where missingness introduces non-overlapping support. However, because our work is primarily motivated by underreporting in the medical setting, we focus our attention on the case where missingness indicators are absent.

UCAR as Regularization While the optimal predictor does not change across domains when ξ are observed (as the covariate shift assumption holds), it is less obvious *how missingness without indicators impacts the optimal predictor*. To build intuition on the effect of underreporting completely at random, we note that applying mask ξ, which zeros out covariates with some probability, resembles the mechanism of dropout in neural networks. Using similar theoretical arguments as in how dropout acts as a form of regularization (Wager et al., 2013), we show that for linear models, the phenomenon of UCAR in data with constant missingness rate m translates into a form of regularization on the resulting predictor (proof in Appendix D.3). First, we show that for generalized linear models, UCAR results in a regularization effect. Here, generalized linear models are defined as $p_\beta(y|x) = h(y)\exp\{yx \cdot \beta - A(x \cdot \beta)\}$, where $h(y)$ is a quantity independent of x and β, and $A(\cdot)$ is the log partition function, and the negative log likelihood objective is $l_{x,y}(\beta) = -\log p_\beta(y|x)$. Then, considering linear regression, we show that the regularization penalty can be viewed as a form of L2 regularization.

Theorem 5.4.1. *Under UCAR with missingness rates $m \in [0,1)^d$, the minimizer $\widehat{\beta}$ of the negative log likelihood of the corrupted training data $\widetilde{X}$ scaled by $\frac{1}{1-m}$ is given by:*

$$\begin{aligned}\widehat{\beta} &= \arg\min_{\beta \in \mathbb{R}^d} \sum_{i=1}^{n} \mathbb{E}_\xi[l_{\widetilde{x}^{(i)},y^{(i)}}(\beta)] \\ &= \arg\min_{\beta \in \mathbb{R}^d} \sum_{i=1}^{n} l_{x^{(i)},y^{(i)}}(\beta) + R(\beta),\end{aligned}$$

where $l_{\widetilde{x}^{(i)},y^{(i)}}(\beta)$ and $l_{x^{(i)},y^{(i)}}(\beta)$ are the negative log likelihoods of a corrupted sample and the corresponding clean sample (respectively). For linear regression, the regularization term $R(\beta)$ is given by:

$$R(\beta) = \frac{1}{2}\left(\beta\widetilde{\Delta}_{diag}\right)^\top \left(\beta\widetilde{\Delta}_{diag}\right),$$

where we define $\widetilde{\Delta}_{diag} = diag\left(\sqrt{\frac{m}{1-m}}\right) diag(I)^{1/2}$, where $diag\left(\sqrt{\frac{m}{1-m}}\right)$ refers to a diagonal matrix with $\sqrt{\frac{m_j}{1-m_j}}$ on the diagonal, and $diag(I)^{1/2}$ refers to the square root of the diagonal of the Fisher information matrix.

Thus, for linear regression, applying missingness rates m to data scaled by $\frac{1}{1-m}$ can be viewed as a form of L2 regularization of β scaled by $\widetilde{\Delta}_{\text{diag}}$.

5.5 Identification Results

This section shows that in DAMS with UCAR, the clean joint distribution p is identifiable from the corrupted joint distribution $\widetilde{p}$ with missingness rates $m \in [0,1)^d$ when m is known (Lemma 5.5.1). However, m is not in general identifiable directly from the observed corrupted data (Remark 4). Instead, we identify *relative* rates of non-missingness from the corrupted data across domains (Remark 5), which can in turn be used to identify the labeled target distribution $\widetilde{p}^t$ from the labeled source distribution $\widetilde{p}^s$ (Theorem 5.5.2).

First, we define some notation useful for our identification results. Consider vectors $a \in \mathbb{R}^d$ and $b \in \mathbb{R}^d$. Let $a \prec b$ denote that $\forall j \in \{1, 2, ..., d\}$, we have $a_j < b_j$. Similarly, let $a \succeq b$ denote that $\forall j \in \{1, 2, ..., d\}$, $a_j \geq b_j$.

To help clarify the relationship between corrupted and clean distributions, we define the notion of m-reachability.

Definition 2 (m-reachable)**.** *We say b is m-reachable from a (denoted $a \rightsquigarrow b$) if $\exists \xi \in \{0, 1\}^d$ such that $b = a \odot \xi$.*

Remark 3 (Characteristics of m-reachability)**.** *If $a \rightsquigarrow b$, then the dimensions of a that are 0 must be a subset of the ones that are 0 in b. Additionally, any dimensions that are nonzero in both a and b must match in value.*

For example, if we observe a data point $b = [1, 1, 1]$, the only data point a for which $a \rightsquigarrow b$ is $a = [1, 1, 1]$. If $b = [1, 1, 0]$, then possible values of a are $a = [1, 1, c]$ for any value of $c \in \mathbb{R}$. In binary data, $a \rightsquigarrow b \iff a \succeq b$.

Let $p_{x,y} = P(X = x, Y = y)$ denote the probability of some set of covariates $x \in \mathbb{R}^d$ and label $y \in \mathbb{R}$ in the clean distribution, and let $\widetilde{p}_{x,y} = P(\widetilde{X} = x, Y = y)$ denote the same in the corrupted distribution. Throughout the paper we use notation for discrete X, but note that it is straightforward to extend the results to continuous X (e.g. by replacing summations with integrals, etc.). Summing over all possible values of $z \in \mathbb{R}^d$ from which x is m-reachable, $\widetilde{p}$ can be expressed in terms of p and m:

$$\widetilde{p}_{x,y} = \sum_{z:z \rightsquigarrow x} p_{z,y} \cdot \prod_{j=1}^{d} (1 - m_j)^{[x_j]_{\neq 0}} m_j^{[z_j]_{\neq 0} - [x_j]_{\neq 0}} \tag{5.1}$$

where $[x]_{\neq 0} \triangleq \mathbb{1}[x \neq 0]$ is an indicator function for nonzero values. While it is obvious that one can obtain $\widetilde{p}$ from p, we show, surprisingly, that the above system is in fact invertible.

Lemma 5.5.1. *Given m, where $m \prec 1$, the clean distribution p is identifiable from the corrupted distribution $\widetilde{p}$.*

Roughly, the proof of Lemma 5.5.1 (in Appendix D.4) rearranges equation (5.1) and uses Remark 3 to observe that any entry $p_{(x,y)}$ can be expressed in terms of $\widetilde{p}$, m, and entries of p with fewer zeros. Using proof by induction on the number of zeros (0 to d), one can show that p is identifiable from $\widetilde{p}$.

Returning to the DAMS problem, *given m^s and m^t*, one could in theory identify p from $\widetilde{p}^s$ thru Lemma 5.5.1, and then use equation (5.1) to derive $\widetilde{p}^t$. Unfortunately, however, *missingness rates are not in general identifiable* from the observed corrupted data.

Remark 4. *Missingness rates are not in general identifiable directly from corrupted data. To see this, consider the following simple counterexample. Consider two distinct possible source distributions $A \sim$ Bernoulli(0.5) and $B \sim$ Bernoulli(0.25). Application of missingness with rates $m_A = 0.5$ to A and $m_B = 0$ to B yields identical corrupted distributions $\widetilde{A} \sim$ Bernoulli(0.25) and $\widetilde{B} \sim$ Bernoulli(0.25). Thus, the rates are not identifiable.*

While missingness rates are not in general identified given corrupted data from a single domain, one might hope to nevertheless *relate* the missingness rates between source and target domains. For this, we leverage nonzero values. Whereas observed zeros are a mixture of zeroed-out values and true zeros, all observed nonzeros were nonzero in the clean data. Thus, the relative proportions of nonzeros are informative of relative *non-missingness rates* $1 - m$. For a covariate X_j, where $j \in \{1, ..., d\}$, denote the true proportion of nonzeros in the underlying data as $q_j = P(X_j \neq 0)$. Then, the proportion of observed nonzeros in the corrupted data is $P(\widetilde{X}_j \neq 0) = (1 - m_j)q_j$. Vectorized, $P(\widetilde{X} \neq 0) = (1 - m) \odot q$.

Remark 5. *The ratio between non-missingness rates* $1 - m^t$ *and* $1 - m^s$ *is given by:*

$$\frac{1 - m^t}{1 - m^s} = \frac{(1 - m^t) \odot q}{(1 - m^s) \odot q} = \frac{P^t(\widetilde{X} \neq 0)}{P^s(\widetilde{X} \neq 0)} \triangleq 1 - r^{s \to t}, \tag{5.2}$$

where the divisions are element-wise. Note that the second-to-last expression is estimable from observed data.

We refer to $r^{s \to t} = 1 - \frac{1-m^t}{1-m^s} = \frac{m^t - m^s}{1-m^s}$ as the *relative missingness rates* between s and t. Interestingly, while identification of the *clean* distribution from a corrupted distribution (Lemma 5.5.1) may be difficult due to unidentifiability of m^s and m^t in general (Remark 4), we leverage identifiability of $r^{s \to t}$ to show that *adapting* from one corrupted distribution to another corrupted distribution does not require identification of the clean distribution.

Theorem 5.5.2. *For source and target distributions* $\widetilde{p}^s$ *and* $\widetilde{p}^t$ *with unknown missingness rates* m^s *and* m^t *(respectively), where* $m^s \prec 1$*,* $\widetilde{p}^t$ *is identifiable from* $\widetilde{p}^s$ *given* $r^{s \to t}$*:*

$$\widetilde{p}^t_{x,y} = \sum_{z : z \rightsquigarrow x} \widetilde{p}^s_{z,y} \cdot \prod_{j=1}^{d} (1 - r_j^{s \to t})^{[x_j]_{\neq 0}} (r_j^{s \to t})^{[z_j]_{\neq 0} - [x_j]_{\neq 0}}. \tag{5.3}$$

That is, while the precise missingness rates m^s and m^t may be unidentifiable in general from corrupted data, one can identify relative missingness rates $r^{s \to t}$ (Remark 5) and use them to directly identify $\widetilde{p}^t$ from $\widetilde{p}^s$ (proof in Appendix D.5), rather than explicitly using the clean distribution as an intermediate step. Note that the form of (5.3) matches that of (5.1), with missingness rates set to $m = r^{s \to t}$.

5.6 Estimation Results

We discuss estimation of optimal target predictors from labeled source data $\{(\widetilde{X}^{s,i}, Y^{s,i})\}_{i=1}^{n_s}$, drawn from $P^s(\widetilde{X}, Y)$ and unlabeled target data $\{\widetilde{X}^{t,i}\}_{i=1}^{n_t}$, drawn from $P^t(\widetilde{X})$.

Non-parametric adjustment procedure for nonnegative relative missingness The parallels between equations (5.3) and (5.1) suggest an intuitive non-parametric procedure when $m^s \preceq m^t$, so that $r^{s\to t} \succeq 0$ (Algorithm 1). To obtain data distributed identically to $\widetilde{X}^t$, one can sample masks $\xi^{s\to t}$ with missingness rates $r^{s\to t}$ and apply them to $\widetilde{X}^s$. Let us define a *missingness filter* applied to each datapoint $x \in \mathbb{R}^d$ as $\nu_{s\to t}(x) = x \odot \xi^{s\to t}$, where $\xi^{s\to t} \sim \text{Bernoulli}(1 - r^{s\to t})$. When a missingness filter is applied to a dataset, $\xi^{s\to t}$ is independently drawn for every data point. A proof showing that labeled data $\{(\nu_{s\to t}(\widetilde{X}^{s,i}), Y^{s,i})\}_{i=1}^{n_s}$ are drawn i.i.d. to $P^t(\widetilde{X}, Y)$ is in Appendix D.7. For any desired model class, we can now train a predictor on this labeled data. When $m^s \preceq m^t$, we call this adjustment a *proper* adjustment as it yields a predictor trained on data i.i.d. to labeled target data.

When $m^s \npreceq m^t$, i.e. $r^{s\to t} \nsucceq 0$, it is less obvious what the proper non-parametric adjustment procedure implied by Theorem 5.5.2 might be. As a stopgap measure, we experiment with using a missingness filter of rate $\max\{r^{s\to t}, 0\}$ (Algorithm 1), but call this an improper adjustment as it does not create data i.i.d to the target distribution.

Algorithm 1 Non-parametric adjustment procedure
(proper adjustment when $m^s \preceq m^t$)

```
1: Compute $\widehat{q}_j^t = \frac{\text{count}(\widetilde{x}_j^t \neq 0)}{n_t}$, $\widehat{q}_j^s = \frac{\text{count}(\widetilde{x}_j^s \neq 0)}{n_s}$, and $\widehat{r}^{s\to t} = 1 - \frac{\widehat{q}^t}{\widehat{q}^s}$.
2: Compute $\widetilde{r}^{s\to t} = \max\{\widehat{r}^{s\to t}, 0\}$ (element-wise max). Note that if $\widehat{r}^{s\to t} \succeq 0$, then $\widehat{r}^{s\to t} = \widetilde{r}^{s\to t}$.
3: Apply a missingness filter with rate $\widetilde{r}^{s\to t}$ to source data to get $\{(\widetilde{\nu}_{s\to t}(\widetilde{X}^{s,i}), Y^{s,i})\}_{i=1}^{n_s}$.
4: Fit a predictor on data $\{(\widetilde{\nu}_{s\to t}(\widetilde{X}^{s,i}), Y^{s,i})\}_{i=1}^{n_s}$.
```

Step 1 of Algorithm 1 estimates the relative missingness $r^{s\to t}$ from data. Using Hoeffding's inequality, we show that with high probability, the estimated $\widehat{r}^{s\to t}$ is close to $r^{s\to t}$ (proof in Appendix D.6).

Theorem 5.6.1. *With probability at least* $1 - \delta$*,*

$$\left|\widehat{r}^{s\to t} - r^{s\to t}\right| \leq \frac{1}{\widehat{q}^s}\left(\sqrt{\frac{\log(4/\delta)}{2n_t}} + (1 - r^{s\to t})\sqrt{\frac{\log(4/\delta)}{2n_s}}\right).$$

A proper non-parametric adjustment requires $r^{s\to t} \succeq 0$. Next, we derive a closed-form expression for the optimal linear target predictor for any given relative missingness.

Closed-Form Solution for Optimal Linear Predictor Define the optimal predictor as the one that minimizes mean squared error. Given observations of source covariates $\widetilde{X}^s$ and their corresponding labels Y^s, as well unlabeled target covariates $\widetilde{X}^t$, we seek the optimal linear predictor $f^t_*(x^t) = \beta^t_* x^t$ for the target domain. Indeed, β^t_* can be expressed in terms of quantities estimable from data (proof in Appendix D.8.1).

Proposition 2. *The optimal linear target predictor is given by:*

$$\beta^t_* = \mathbb{E}[\widetilde{X}^{t\top}\widetilde{X}^t]^{-1}\left((1 - r^{s\to t}) \odot \mathbb{E}[\widetilde{X}^{s\top} Y^s]\right). \tag{5.4}$$

Thus, *without knowing the levels of missingness*, as long as $m^s \prec 1$, the optimal linear predictor for the target domain is nevertheless estimable, using target unlabeled data to derive the covariance $\mathbb{E}[\widetilde{X}^{t\top}\widetilde{X}^t]$. As we show in Appendix D.8, it is also possible to compute the entries of $\mathbb{E}[\widetilde{X}^{t\top}\widetilde{X}^t]$ using only source data and relative missingness.

Proposition 3. *For $i \neq j$, where $i \in \{1, 2, .., d\}$, $j \in \{1, 2, .., d\}$, we have*

$$\mathbb{E}[\widetilde{X}^{t\top}\widetilde{X}^t]_{ij} = (1 - r^{s\to t}_i)(1 - r^{s\to t}_j)\mathbb{E}[\widetilde{X}^{s\top}\widetilde{X}^s]_{ij} \tag{5.5}$$

$$\mathbb{E}[\widetilde{X}^{t\top}\widetilde{X}^t]_{ii} = (1 - r^{s\to t}_i)\mathbb{E}[\widetilde{X}^{s\top}\widetilde{X}^s]_{ii}. \tag{5.6}$$

Although $\mathbb{E}[\widetilde{X}^{t\top}\widetilde{X}^t]$ could be estimated from either source or target covariates, in practice with finite samples it might be beneficial to utilize both. For example, to adjust for sample size of the source and target datasets, one could take a weighted average of the estimates of $\mathbb{E}[\widetilde{X}^{t\top}\widetilde{X}^t]$, where the weights of the source-derived and target-derived estimates are $\alpha_s = \frac{n_s}{n_s+n_t}$ and $\alpha_t = \frac{n_t}{n_s+n_t}$, respectively. This attempts to adjust for the variance of estimation error due to the different sample sizes, however it does not account for estimation error in the relative missingness rate. We leave further exploration of these weightings to future work. Algorithm 2 describes the estimation procedure for linear models adjusted for the target domain.

Algorithm 2 Adjusted linear model learning procedure

```
Compute $\widehat{q}^t_j = \frac{\text{count}(\widetilde{x}^t_j \neq 0)}{n_t}$, $\widehat{q}^s_j = \frac{\text{count}(\widetilde{x}^s_j \neq 0)}{n_s}$, and $\widehat{r}^{s\to t} = 1 - \frac{\widehat{q}^t}{\widehat{q}^s}$ for all $j \in \{1, 2, .., d\}$.
Estimate target-based $\widehat{M}^t = \widehat{\mathbb{E}}[\widetilde{X}^{t\top}\widetilde{X}^t]$ from unlabeled target samples.
Estimate source-based $\widehat{M}^s = \widehat{\mathbb{E}}[\widetilde{X}^{t\top}\widetilde{X}^t]$ by computing for all $i \neq j$, where $i \in \{1, 2, .., d\}$, $j \in \{1, 2, .., d\}$:
    $\widehat{M}^s_{ij} = (1 - \widehat{r}^{s\to t}_i)(1 - \widehat{r}^{s\to t}_j)\widehat{\mathbb{E}}[\widetilde{X}^{s\top}\widetilde{X}^s]_{ij}$
    $\widehat{M}^s_{ii} = (1 - \widehat{r}^{s\to t}_i)\widehat{\mathbb{E}}[\widetilde{X}^{s\top}\widetilde{X}^s]_{ii}$
Construct a combined weighted estimate of $\widehat{\mathbb{E}}[\widetilde{X}^{t\top}\widetilde{X}^t]$: $\widehat{M} = \alpha_s\widehat{M}^s + \alpha_t\widehat{M}^t$
Estimate $\widehat{\mathbb{E}}[\widetilde{X}^{s\top}Y^s]$ from source samples, and compute
    $\widehat{\beta}^t = \widehat{M}^{-1}\left((1 - \widehat{r}^{s\to t}) \odot \widehat{\mathbb{E}}[\widetilde{X}^{s\top}Y^s]\right).$
```

5.7 Experiments

We apply Algorithms 1 and 2 to synthetic, semi-synthetic, and real data settings. We compare the performance of four variations of predictors: (1) the oracle predictor (Oracle), trained with target labeled data and tested on a held-out target test set; (2) the source predictor (Source), trained on source labeled data without any adjustments; (3) the closed-form adjustment (Closed-form Adj.) for linear predictors, given by Algorithm 2; and (4) the non-parametric adjustment (Non-param. Adj.), given by Algorithm 1. We also do MissForest imputation of both source and target data, treating all zeros as missing values, and train a source predictor to evaluate on target (Imputed).

In synthetic and semi-synthetic experiments, the data is split 4:1:4:1 to create source training, source test, target training, and target test sets. Different levels of missingness are applied completely at random to source and target datasets. Code is provided at https://github.com/acmi-lab/Missingness-Shift.

Synthetic data experiments We draw 10,000 samples from two simple data-generating processes:

Scenario 1: "Redundant Features"

$$\begin{aligned} u_y &\sim \mathcal{N}(0,1) \\ Z &\sim \text{Bernoulli}(0.5) \\ X_1 &= Z \\ X_2 &= Z \\ Y &= Z + u_y \end{aligned}$$

Scenario 2: "Confounded Features"

$$\begin{aligned} u_{x_2} &\sim \mathcal{N}(0,1) \\ u_y &\sim \mathcal{N}(0,1) \\ X_1 &\sim \text{Bernoulli}(0.5) \\ X_2 &= \text{expit}(2X_1 + u_{x_2}) \\ Y &= X_1 - X_2 + u_y \end{aligned}$$

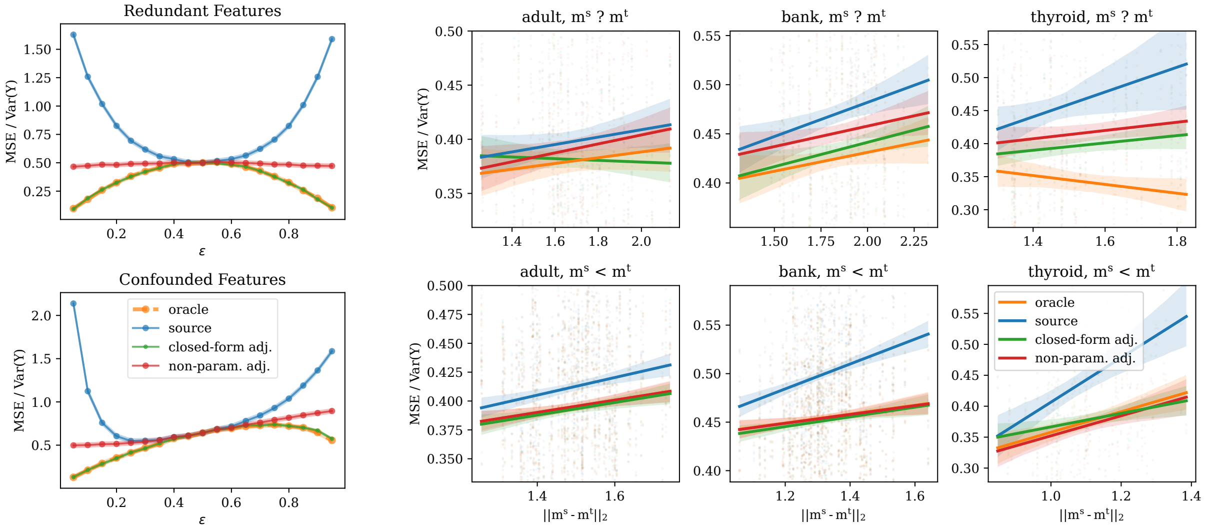
In both, we apply missingness with rates $m^s = [1-\epsilon, \epsilon]$ and $m^t = [\epsilon, 1-\epsilon]$ for varying ϵ between 0.05 and 0.95 in increments of 0.05, with 20 runs for each ϵ, and evaluate the performance of linear predictors (Figure 5.2a). At $\epsilon = 0.5$, the source and target domains are identically distributed, so Oracle, Source, Closed-form Adj., and Non-param. Adj. all attain the same mean squared error scaled by variance of the label (MSE/Var(Y)). As ϵ approaches 0 or 1, however, the error in the Source predictor grows rapidly whereas the Oracle and Closed-form Adj. errors decrease. Since $m^s \not\preceq m^t$, as expected, Non-param. Adj. cannot fully match the target distribution, and has intermediate performance.

For $\epsilon = 0.1$, we compare linear regression, XGBoost, and MLP (Table 5.1). In both Scenario 1 and 2, the linear closed-form adjustment dramatically outperforms the source linear predictor. However, in Scenario 1, source XGBoost and MLP almost match the performance of their respective oracles, and source XGBoost outperforms the linear oracle. On the other hand, in Scenario 2, the linear closed-form adjustment outperforms source XGBoost and MLP.

Semi-synthetic data experiments Using the adult ($n = 48842$), bank ($n = 48188$), and thyroid binding protein ($n = 2800$) UCI datasets (Dua and Graff, 2017), which contain a mixture of categorical and numerical variables, we construct semi-synthetic datasets by borrowing the covariates, but replacing the labels with synthetically generated labels that are linear functions of the clean covariates. That is, we train using new labels $y_{new} = \beta X$, for randomly sampled $\beta_j \sim$

Table 5.1: Target domain MSE/Var(Y), averaged across various missingness levels on synthetic and semi-synthetic data. Confidence intervals are provided in Appendix D.9. The first two columns are synthetic datasets (Redundant Features and Confounded Features), and the last three columns are semi-synthetic UCI datasets.

	Rednd.	**Confnd.**	**Adult**		**Bank**		**Thyroid**	
	$m^s ? m^t$	$m^s ? m^t$	$m^s \preceq m^t$	$m^s ? m^t$	$m^s \preceq m^t$	$m^s ? m^t$	$m^s \preceq m^t$	$m^s ? m^t$
Linear Regression Models								
Oracle	0.178	0.206	0.420	0.362	0.338	0.433	0.298	0.251
Source	1.259	1.103	0.437	0.380	0.371	0.480	0.350	0.320
Imputed	1.002	0.918	0.490	0.483	0.501	0.592	0.306	0.358
Closed-form	**0.186**	**0.209**	0.422	**0.363**	0.339	**0.442**	0.316	**0.291**
Non-param.	0.473	0.492	**0.420**	0.373	**0.338**	0.459	**0.293**	**0.291**
XGBoost Models								
Oracle	0.166	0.200	0.398	0.354	0.287	0.453	0.316	0.274
Source	**0.166**	0.475	**0.399**	**0.379**	0.305	**0.500**	**0.310**	**0.352**
Imputed	1.002	1.157	0.512	0.521	0.492	0.708	0.355	0.441
Non-param.	0.425	**0.473**	**0.399**	0.392	**0.287**	0.503	**0.310**	0.381
MLP Models								
Oracle	0.166	0.201	0.389	0.343	0.295	0.458	0.279	0.230
Source	**0.184**	**0.321**	0.399	0.357	0.322	0.499	0.320	0.303
Imputed	1.003	0.924	0.480	0.468	0.484	0.668	0.304	0.345
Non-param.	0.436	0.470	**0.389**	**0.355**	**0.294**	**0.487**	**0.278**	**0.272**



(a) Target domain error of linear models vs. ϵ. Oracle & closed-form overlap.

(b) Target domain error of linear models as the L2-norm between m^s and m^t varies. Best-fit line with 95% confidence intervals from bootstrapping.

Figure 5.2: MSE/Var(Y) of linear models on (a) synthetic and (b) semisynthetic data across varying m^s and m^t.

Table 5.2: Target domain performance of linear models on eICU 48-hour mortality prediction, where source s and target t can be Hospital 1 (H1) or Hospital 2 (H2). Here, underreporting occurs naturally in the data. Since all features are binary, imputation of all zeros behaves poorly, leading to baseline performance. AUPRC refers to average precision.

Model Class	s	t	MSE	AUROC	AUPRC
Oracle	H1	H1	0.103 (0.088 – 0.117)	0.713 (0.652 – 0.775)	0.236 (0.156 – 0.317)
Source	H2	H1	0.143 (0.135 – 0.151)	**0.593 (0.563 – 0.623)**	**0.146 (0.122 – 0.170)**
Imputed	H2	H1	**0.089 (0.081 – 0.097)**	0.500 (0.500 – 0.500)	0.097 (0.088 – 0.106))
Closed-form Adj.	H2	H1	0.439 (0.223 – 0.655)	0.540 (0.509 – 0.571)	0.123 (0.103 – 0.143)
Non-param. Adj.	H2	H1	0.142 (0.133 – 0.150)	0.555 (0.537 – 0.573)	0.126 (0.108 – 0.144)
Oracle	H2	H2	0.121 (0.100 – 0.142)	0.601 (0.528 – 0.675)	0.167 (0.103 – 0.230)
Source	H1	H2	0.122 (0.113 – 0.131)	**0.576 (0.545 – 0.608)**	**0.144 (0.120 – 0.169)**
Imputed	H1	H2	**0.090 (0.082 – 0.098)**	0.500 (0.500 – 0.500)	0.099 (0.089 – 0.109)
Closed-form Adj.	H1	H2	0.373 (0.327 – 0.420)	0.556 (0.523 – 0.588)	0.122 (0.104 – 0.141)
Non-param. Adj.	H1	H2	0.196 (0.182 – 0.210)	0.511 (0.503 – 0.520)	0.109 (0.095 – 0.123)

Uniform$(0, 10), \forall j \in \{1, 2, ..., d\}$, and original covariates X. Source and target missingness rates are sampled under two regimes: (1) To test the proper non-parametric adjustment, where $m^s \preceq m^t$, we sample $m^s_j \sim$ Uniform$(0, 0.5)$ and $m^t_j \sim m^s_j + (1 - m^s_j)\epsilon$, where $\epsilon \sim$ Uniform$(0, 0.5)$. (2) To simulate a more general form of missingness shift, we sample $m^s_j, m^t_j \sim$ Uniform$(0, 0.9)$, abbreviated as m^s ? m^t. Additional experiment and data preprocessing details in Appendix D.9.

Overall, where adjusted models are applicable/proper, they perform at least as well as (and often better than) source unadjusted models when compared within each model class (Table 5.1). Among linear models, the closed-form and non-parametric adjustments consistently outperform the source predictors. In nonlinear models, only the non-parametric adjustment applies, and this adjustment is only proper if $m^s \preceq m^t$. Among nonlinear models, if $m^s \preceq m^t$, either Non-param. and Source tie, or Non-param. performs best. When m^s ? m^t, Non-param. (improper adjustment) often has the second-best or best performance (especially when no other adjustments apply). Ignoring model class, the best-performing model for each semi-synthetic dataset is an adjusted model. Plotting the line of best fit for MSE/Var(Y) of the linear models versus the L2 distance between m^s and m^t, we note that the Source predictor tends to have the stronger positive slope than the Oracle, Closed-form Adj., or Non-parametric Adj. models (Figure 5.2b).

Real data experiments To explore the applicability of our methods to naturally-occurring missingness shifts, we use the FIDDLE data pre-processing pipeline (Tang et al., 2020) on the eICU Collaborative Research Database (Pollard et al., 2018a), which contains data from critical care units across several hospitals. FIDDLE extracts binary feature vectors capturing several patient characteristics, including demographics, physiological measurements, labs, medications, etc. We extract the binary 48-hour mortality outcome for patients in two of the hospitals with the most data ($n_1 = 3006$, $n_2 = 2663$), and verify that the prevalences of the covariates are different across these two hospitals. Additional data and experiment details are provided in Appendix D.9.

We train linear models to predict mortality, and evaluate MSE, AUROC, and AUPRC. Since the preprocessed data only contains binary features, MissForest imputation of all zeros results in a dataset consisting entirely of ones, and the linear model learns to simply predict the label mean and only achieves baseline performance. Estimated relative missingness indicates that $m^s \not\preceq m^t$ (Appendix D.9), so the non-parametric estimation procedure is not expected to produce labeled data i.i.d. to the target distribution. The source predictor achieves highest AUROC and AUPRC.

Note, however, that beyond missingness levels, there are also several other aspects of the data distribution that likely differ between these two hospitals. Different hospitals likely have different underlying $P(X, Y)$, and in practice, missingness could be dependent on other covariates (e.g. a doctor may choose not to perform a test based on patient state). Thus, fundamental assumptions of our adaptation methods are likely violated in this dataset.

5.8 Discussion

This work introduces the domain adaptation under missingness shift (DAMS) problem, and explores DAMS under the underreporting completely at random (UCAR) assumption. Our synthetic and semi-synthetic experiments demonstrate that when assumptions hold, the proposed methods (when applicable/proper), tend to outperform or perform at least as well as unadjusted source predictors in the same model class (Table 5.1). In linear models, our proposed adjustments (linear closed-form and non-param. adj.) consistently outperform the source predictors, and sometimes, the benefits of adaptation can even outweigh the bias incurred by restricting to linear models. For example, in the Confounded Features, Bank m^s ? m^t, and Thyroid datasets, linear adjusted models outperform all Source models, regardless of model class. Note that even if the underlying relationship between clean unobserved covariates X and label Y is linear, after X is corrupted by missingness to create observed corrupted covariates $\widetilde{X}$, the new relationship between $\widetilde{X}$ and Y is often nonlinear (a phenomenon which has also been noted by Le Morvan et al. (2020b)). Correspondingly, the best MLP and XGBoost models tend to outperform the best linear models (Table 5.1).

The best-performing model(s) in each of the synthetic and semi-synthetic datasets, except for the synthetic Redundant Features dataset, use a proposed adjustment (Table 5.1). Although the adjustments perform best in the synthetic Redundant Features dataset when restricted to the linear model class, the best-performing model in this dataset overall is a source XGBoost model, which matches the performance of the oracle. In addition to the flexibility of the XGBoost model, which improves the oracle XGBoost over the oracle linear model, a likely reason for improvement of Source XGBoost over Non-param. Adj. can be found in the particular setup of this scenario. Here, $X_1 = X_2 = Z$, and $Y = Z + u_y$, where $u_y \sim \mathcal{N}(0, 1)$, and so given knowledge of either X_1 or X_2, prediction of Y is straightforward. The only applicable adjustment, Non-param. Adj. (improper, since $m^s \npreceq m^t$), would zero out much of the data to bring the missingness rate in X_1 from 0.9 to 0.1, thus making prediction harder. There are also multiple settings in which Source XGBoost performs similarly to Non-param. Adj. XGBoost (Confounded Features, Adult $m^s \preceq m^t$, Bank m^s ? m^t, and Thyroid m^s ? m^t). On the other hand, for the MLP model class, the non-parametric adjustment outperforms all source predictors in the semi-synthetic datasets. Thus, depending on the model class, non-parametric adjustment may not always have a consistent effect on performance.

The generally worse performance of imputation in synthetic and semi-synthetic experiments (Table 5.1) helps highlight the difficulty of not having missing data indicators. Learning without missing data indicators is fundamentally more difficult than learning with them, and methods which might make sense when missing data indicators are present (e.g. imputation) can be ill-defined when the indicators are absent. In the eICU dataset, for example, all covariates were binary, and so imputing all 0's only left 1's to train on. As a result, MissForest learned to predict 1 for everything, rendering these binary features useless. Nevertheless, we included a comparison with imputation of all zeros in the other datasets, as it could still be useful for continuous variables.

The experiments with real eICU data also help demonstrate that it is important to clarify assumptions on whether one is truly in a DAMS with UCAR setting, as failure to do so could result in predictors that perform worse than if no adaptation had been done in the first place (Table 5.2). Ideally, in real-world data, DAMS with UCAR might be useful around a sudden change in clerical practices where the underlying $P(X, Y)$ is similar before and after the change, and underreporting is completely at random (e.g. determined based a blanket policy independent of covariates). In the absence of such data, however, we instead included synthetic and semisynthetic data where the missingness shift with UCAR assumptions hold, and also included a real critical care (eICU) dataset containing multiple hospitals for thoroughness. While our proposed techniques for DAMS with UCAR do not work particularly well on real eICU data, we also note that we have no particular reason to believe that missingness shift is especially prominent between the hospitals compared to factors such as selection bias (very different cohort), label shift, or changes in prevalences of disease, among others. Finding appropriate real world empirical testbeds and analyzing sensitivity to assumption violations are important directions for future work.

Beyond the UCAR setting, there are several open avenues for further research in domain adaptation under missingness shift. Allowing underreporting to depend on other covariates would significantly broaden the set of applicable real-world cases, as doctors often take certain measurements as needed in their diagnostic process. Moreover, future works could explore other variations of graphical model structures (Figure 5.1) for expressing models of missingness shift.

Part III

Decision-Making in Dynamic Healthcare Settings

> *"We look for medicine to be an orderly field of knowledge and procedure. But it is not. It is an imperfect science, an enterprise of constantly changing knowledge, uncertain information, fallible individuals, and at the same time lives on the line. There is science in what we do, yes, but also habit, intuition, and sometimes plain old guessing."*
>
> - Atul Gawande, *Complications*

Now, we move beyond prediction and into decision-making. Doctors, nurses, administrators, and other healthcare professionals make countless decisions every day, all of which add up to an operation where patients are ideally able to receive treatment when needed and check up on when necessary. In this part, we study two scenarios where decisions are made over time. In the first, we consider a setting where decisions are cheap, frequent, and directly tied to forecasts. In the second, we consider a common scenario where there is a cost to seeing a patient in order to prescribe a treatment, and observations are not made in the interim.

Chapter 6

Business Metric-Aware Forecasting: A Case Study in Inventory

Suppose that a hospital would like to order supplies on some recurring basis. Inventory management relies on forecasts to decide how many orders to place at any given time. There are several simple ordering policies, for example ordering enough to maintain an inventory such that with 95% probability, there will be no stockouts. Typically, forecasts optimize for business-agnostic metrics such as mean squared error or mean absolute percentage error. In this work, we create an end-to-end system that adjusts forecasts at each time step so that given a simple ordering policy, the forecasts optimize for downstream business objectives.

Time-series forecasts play a critical role in business planning. However, forecasters typically optimize objectives that are agnostic to downstream business goals and thus can produce forecasts misaligned with business preferences. In this work, we demonstrate that optimization of conventional forecasting metrics can often lead to sub-optimal downstream business performance. Focusing on the inventory management setting, we derive an efficient procedure for computing and optimizing proxies of common downstream business metrics in an end-to-end differentiable manner. We explore a wide range of plausible cost trade-off scenarios, and empirically demonstrate that end-to-end optimization often outperforms optimization of standard business-agnostic forecasting metrics (by up to 45.7% for a simple scaling model, and up to 54.0% for an LSTM encoder-decoder model). Finally, we discuss how our findings could benefit other business contexts.

6.1 Introduction

Time-series forecasting is an essential component of decision-making and planning. In industries ranging from healthcare (Jones et al., 2009; Reich et al., 2019; Cheng et al., 2021), to finance (Thomas, 2000; Elliott and Timmermann, 2016), to energy (Ahmed et al., 2020; Donti and Kolter, 2021), businesses leverage forecasts of future demand in order to adjust their behavior accordingly.

One common business problem reliant on time-series forecasts is *inventory management* (Chopra et al., 2007; Syntetos et al., 2009). Here, businesses must decide how much inventory to order on a recurring basis, balancing considerations such as customer satisfaction, costs of holding surplus inventory (*holding cost*), opportunity costs of out-of-stock items (*stockout cost*), and keeping the supply chain running smoothly. For example, grocery stores track their inventory, anticipating demand and placing orders such that when customers demand e.g. toilet paper, they have enough stock to avoid lost sales and keep customers happy, while not having too much stock such that stale items are taking valuable shelf or warehouse space.

Since forecasts are used to decide how much inventory to order, the quality of forecasts can greatly influence downstream measures of business performance. Typically, forecasters optimize and evaluate generic metrics agnostic to the downstream application, such as mean squared error (MSE) or mean absolute percentage error (MAPE). Assuming that these upstream forecasts are accurate, downstream decisions are subsequently treated as a separate step (Figure 6.1).

However, these generic metrics can be misaligned with downstream business performance. For example, the business costs of over-forecasting and under-forecasting are often imbalanced (e.g. opportunity cost of lost sales could outweigh cost of holding extra inventory). Additionally, conventional forecasting metrics typically aim for a mean, median, or quantile of the distribution, without regard to the magnitude of fluctuations in predictions. Fluctuations in predictions can translate into fluctuations in orders, and as orders are passed upstream through the supply chain, uncertainties in forecasts can compound to create the bullwhip effect, an unstable and wildly oscillating demand (Lee et al., 1997; Wang and Disney, 2016).

One reason for the widespread use of generic accuracy metrics such as MSE and MAPE is that downstream business metrics may be difficult to quantify or attribute to specific parts of the supply chain. Customer satisfaction, for example, might have a convoluted data generating process that is difficult to optimize directly. However, as we show, optimization of generic metrics does not necessarily translate into improvements on downstream performance indicators.

In this work, we propose a novel method for business metric-aware forecasting for inventory management systems. Our contributions include:

1. Demonstrating that optimizing conventional metrics often translates into sub-optimal downstream performance.
2. Deriving an efficient end-to-end differentiable procedure for optimizing forecasts for downstream inventory performance, compatible with any differentiable forecaster.
3. Noting that downstream metrics are often at odds with one another, and proposing alternative combined objectives which trade off these metrics in different ways.
4. Empirically demonstrating the benefit of business metric-aware forecasting in univariate and multivariate datasets, under a variety of plausible downstream scenarios.
5. Since time series datasets measuring demand are popular in the forecasting community,

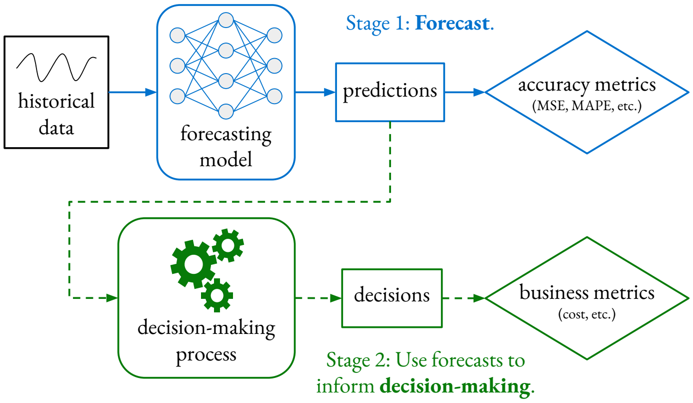


Figure 6.1: Typical separated estimation and optimization approach, where the learned forecaster (blue) is agnostic to business decision-making (green). Solid lines represent direct optimization. This work solidifies the dotted green lines.

we release code[1] for others to evaluate the downstream utility of their forecasts.

6.2 Related Work

Forecasting + Inventory Optimization Inventory management involves estimating future demand (*forecasting*) and deciding how many orders to place to minimize costs and meet customer needs (*inventory optimization*). Some works simply forecast the mean demand and treat it as the inventory optimization solution (Yu et al., 2013; Ali and Yaman, 2013). However, this approach fails to account for imbalanced costs of over- and under-forecasting. Thus, in practice, it is common to take a *separated estimation and optimization* approach (Turken et al., 2012; Oroojlooyjadid et al., 2020), which involves first (1) forecasting demand and then (2) plugging the estimates into inventory optimization (Figure 6.1). However, in this approach, errors in forecasting and inventory optimization can compound.

One widely studied inventory model is the newsvendor problem, which involves determining the optimal order quantity for perishable or seasonal products to balance holding cost and stockout cost. In this setup, the optimal solution to the inventory optimization problem is a particular quantile of the demand distribution (Petruzzi and Dada, 1999). Thus, some works forecast various quantiles of the demand distribution (Böse et al., 2017; Bertsimas and Thiele, 2005; Taylor, 2000). Others use feed-forward neural networks, kernel regression, and linear models to directly optimize these two costs (Ban and Rudin, 2019; Oroojlooyjadid et al., 2017). We also optimize downstream inventory metrics directly, but allow more general cost objectives to be computed over the inventory system variables through use of differentiable simulation, making our approach applicable beyond newsvendor.

[1] link excluded for anonymity

Demand Forecasting Demand forecasting is of interest in several businesses, including retail (Fildes et al., 2022), power grids (Ghalehkhondabi et al., 2017; Suganthi and Samuel, 2012), emergency care (Jones et al., 2009), and municipal water (Donkor et al., 2014). Techniques for forecasting include both classical statistical and modern deep learning approaches. Traditional time-series forecasting methods include autoregressive (AR) models (Box et al., 2015; Makridakis and Hibon, 1997), exponential smoothing (Gardner Jr, 1985; Winters, 1960), and the Theta model (Assimakopoulos and Nikolopoulos, 2000; Hyndman and Billah, 2003). Deep learning architectures for time-series forecasting include convolutional neural networks (Bai et al., 2018; Oord et al., 2016), recurrent neural networks (Hochreiter and Schmidhuber, 1997; Salinas et al., 2020; Rangapuram et al., 2018), and attention-based methods (Fan et al., 2019; Li et al., 2019; Lim et al., 2021), among others (Oreshkin et al., 2019; Challu et al., 2022). However, these works all optimize business-agnostic metrics.

Inventory Optimization Given demand forecasts, the decision of how many orders to place could be treated as a constrained optimization problem (Dai et al., 2021), a supervised deep learning problem (Qi et al., 2023), or a reinforcement learning problem (Oroojlooyjadid et al., 2017). There are also several common practices for placing orders (Eilon and Elmaleh, 1968). For example, the (T, S) policy places orders every T days, and orders up to an inventory level S. Petropoulos et al. (2019) explored the inventory performance of several traditional forecasting models when a fixed periodic order-up-to (T, S) policy is used, finding that methods based on combinations had superior inventory performance. We use the same order-up-to policy in this work, but instead of taking a separated estimation and optimization approach, we use differentiable simulation to optimize downstream business performance end-to-end.

Forecasting Competitions Forecasting competitions such as the M-Competitions (Makridakis and Hibon, 2000a; Makridakis et al., 2020; Makridakis et al., 2022) and the Favorita Competition (Favorita, 2017) have become popular benchmarks for development of modern time-series forecasting methods. While a substantial portion of this data is industry time-series, evaluation of model performance is largely conducted using generic error metrics which ignore downstream business performance. For example, the M3 competition measured performance using versions of symmetric mean/ median absolute percentage error (sMAPE) and median relative absolute error. Submissions to the Favorita competition were evaluated using the normalized weighted root mean squared logarithmic error. By releasing inventory performance code, would like to further challenge researchers to make high-*utility* predictions on this data.

Inventory Performance Metrics Across several inventory optimization applications, ranging from auto parts suppliers (Qi et al., 2023), to online fashion retailers (Ferreira et al., 2016), to drug inventories (Dhond et al., 2000), the common objectives of interest are typically a function of stockout cost and holding cost. These costs may be computed across various lead times, different parts of the supply chain, or simply based on historical data. High variance of orders has also been identified as an undesirable phenomenon due to the bullwhip effect in supply chains (Lee et al., 1997; Petropoulos et al., 2019), in which fluctuations in downstream demand can cause exaggerated order swings upstream in the supply chain that result in customer-upsetting stockouts and wasteful excesses.

6.3 Inventory Management

This section formalizes the quantities used and tracked in an inventory management system (Figure 6.2), and defines business-aware and business-agnostic metrics of interest.

6.3.1 Formulation

For each time-series, we apply a rolling simulation approach in order to simulate an inventory system as it steps through each time point. Consider a time-series of the true demand d_t at every time point $t = 1, 2, ..., T$. At each t, orders o_t are placed with the expectation that they will take lead-time L to come in. Using the order-up-to policy for inventory replenishment (Gilbert, 2005; Petropoulos et al., 2019), orders are given by:

$$o_t = \widehat{D}_t^L + ss_t - ip_t \tag{6.1}$$

where $\widehat{D}_t^L$ is the forecasted lead-time demand over the next L timesteps, ss_t is safety stock that adds a buffer to ensure that the orders placed cover the demand, and ip_t is the inventory position. The *true lead-time demand* D_t^L and the *forecasted lead-time demand* $\widehat{D}_t^L$ are given by:

$$D_t^L = \sum_{k=1}^{L} d_{t+k}, \qquad \widehat{D}_t^L = \sum_{k=1}^{L} \widehat{d}_{t,t+k} \tag{6.2}$$

where $\widehat{d}_{t,t+k}$ is the forecast of demand for time $t + k$ given data up to time t. *Safety stock* is computed as follows:

$$ss_t = \Phi^{-1}(\alpha_s)\sigma_e, \tag{6.3}$$

where σ_e is the standard deviation of the forecast errors, and $\Phi^{-1}(\alpha_s)$ is the inverse CDF of the normal distribution evaluated at some target service level α_s. Assuming normally-distributed errors, with α_s probability, the safety stock plus lead time demand forecast should cover the actual demand.

Inventory position ip_t is obtained by taking the previous inventory position, adding the orders o_{t-1} from the previous timestep, and subtracting the current demand d_t:

$$ip_t = ip_{t-1} + o_{t-1} - d_t. \tag{6.4}$$

We assume $ip_0 = 0$ and $o_0 = 0$. Since orders take lead time L to arrive, the inventory position can be decomposed into a sum of (a) how much inventory is actually on hand, termed *net inventory level* i_t, and (b) how much inventory is on the way, termed *work-in-progress level* w_t:

- Inventory position: $ip_t = i_t + w_t$
- Net inventory: $i_t = i_{t-1} + o_{t-L} - d_t$
- Work-in-progress: $w_t = w_{t-1} + o_{t-1} - o_{t-L}$

In summary, at each time t, orders o_t are placed based on forecasted lead-time demand $\widehat{D}_t^L$ and the current inventory position ip_t. The orders and current demand then adjust the inventory position ip_{t+1}, and this process repeats for the entire length T of the time-series.

6.3.2 Evaluation Metrics

We evaluate and optimize both downstream business performance and conventional generic forecasting metrics.

Downstream Inventory Performance

One straightforward way to balance excess inventory, lost sales, and stability of orders is to frame everything in terms of cost. Thus we introduce a total cost metric, measured in units of money. We also introduce a unitless metric, relative root-mean-square, which compares the performance versus a simple baseline.

Total cost (TC) is defined as a combination of the cost of holding excess inventory (holding cost C_h), the opportunity cost of running out of stock (stockout cost C_s), and the cost of fluctuations in the supply chain (order variance cost C_v):

$$\begin{aligned} C_h &= c_h \cdot \mathbb{E}_t[\max(0, i_t)] \\ C_s &= c_s \cdot \mathbb{E}_t[\max(0, -i_t)] \\ C_v &= c_v \cdot \mathrm{Var}_t(o_t) \\ TC &= C_h + C_s + C_v, \end{aligned}$$

where expectations are taken over all time points t, and $c_h, c_s, c_v \geq 0$ are constants. Specifically, c_h is the unit holding cost, c_s is the unit stockout cost, and c_v is the unit order variance cost. If this information is available in a given problem setting, one can directly plug it in. Otherwise, practitioners can choose how to balance these different factors based on their domain expertise. For example, one might have the intuition that sales lost are more expensive per unit than the cost of holding an extra unit of inventory. If the different components of a supply chain are well-integrated so that the compounded uncertainty is not a major concern, the unit order variance cost may not need to be large.

For settings in which the cost tradeoffs may be unknown, we introduce the *relative root-mean-square* (RRMS) metric:

$$RRMS = \sqrt{\mathrm{rel}(C_h)^2 + \mathrm{rel}(C_s)^2 + \mathrm{rel}(C_v)^2},$$

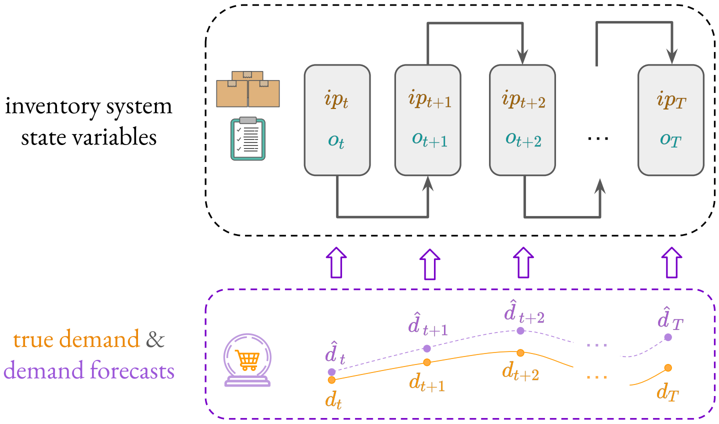


Figure 6.2: Inventory management systems keep track of state variables such as the current inventory position ip and orders placed o. Inventory position is decreased based on observed demand d, and replenished by orders o which are placed based on demand forecasts $\widehat{d}$ and the current ip.

where relative performance to a naive baseline is defined as

$$\text{rel}(x) = \sigma\left(\frac{x - x_{naive}}{x_{naive}}\right),$$

where σ is a sigmoid function. The naive baseline we use in our experiments is a model which simply outputs the previous observation from one period ago. Note that the $rel(x)$ is a unitless quantity, as the unit costs cancel out in the numerator and denominator.

Generic Forecasting Metrics

We also evaluate and optimize generic forecasting metrics to understand the extent to which they might indirectly optimize for downstream performance. *Mean squared error* (MSE) is given by averaging the squared error over time points 1 to T and forecasting horizons 1 to H:

$$\text{MSE} = \frac{1}{TH}\sum_{t=1}^{T}\sum_{k=1}^{H}\left(d_{t+k} - \widehat{d}_{t,t+k}\right)^2.$$

Symmetric mean absolute percentage error (sMAPE) is:

$$\text{sMAPE} = \frac{1}{TH}\sum_{t=1}^{T}\sum_{k=1}^{H}\frac{|d_{t+k} - \widehat{d}_{t,t+k}|}{|d_{t+k}| + |\widehat{d}_{t,t+k}|} \times 2.$$

6.4 Methods

Several challenges arise in optimization of downstream business metrics. Here, we describe how persistent state variables can be computed differentiably, how to optimize for objectives that are computed over the entire time-series of these state variables rather than point-wise, how to provide additional supervision for univariate time series, and how to simulate how models would be updated over time as new data points are observed. Additionally, we describe the models, datasets, and experiment setup.

6.4.1 Differentiable Computation of Metrics

For typical forecasting metrics (e.g. MSE, sMAPE, etc.), differentiable computation is relatively straightforward. These metrics can usually be decomposed such that at each time point, some differentiable quantity (e.g. squared difference of prediction and actual value) is computed, and an average over time points is taken. Computing inventory performance (e.g. total cost, RRMS), however, is less straightforward as there are persistent inventory state variables (e.g. net inventory level, orders) that must be tracked over time. Although inventory performance is not in general a differentiable quantity, we derive a series of computations which can simulate the inventory management system and order-up-to policy described by (6.1)–(6.4) in a differentiable manner. Composing this system with the outputs of a differentiable forecaster, we create an end-to-end differentiable system.

Naively, one could iterate over each time point, and apply the recursive inventory state update equations (6.1)–(6.4). However, for long time-series this would be computationally infeasible, due to the instability of backpropagation through a long chain of dependent states (Pascanu et al., 2013). Instead, assuming all quantities at time $t < 0$ are 0, we show by expanding out the recursion (derivations in Appendix) that the orders at any time t can be written in closed form:

$$o_t = (\widehat{D}_t^L - \widehat{D}_{t-1}^L) + \Phi^{-1}(\alpha_s) \cdot (\sigma_{e,t} - \sigma_{e,t-1}) + d_t \tag{6.5}$$

and the net inventory at time t can be written as:

$$i_t = \widehat{D}_{t-L}^L + \Phi^{-1}(\alpha_s) \cdot \sigma_{e,t-L} - \sum_{a=t-L+1}^{t} d_a. \tag{6.6}$$

These closed form equations are much more efficient to implement in terms of tensor operations than the original recursive equations, and they allow us to simultaneously compute the net inventory levels and orders at all times given a tensor of demand forecasts at all times (detailed walkthrough in the Appendix).

Given the inventory state variables o_t and i_t for all t, it is now feasible to compute metrics on top of these variables. Holding and stockout costs can be computed by applying a ReLU activation over i_t and $-i_t$, and computing the average over timepoints. Variance of orders can be computed by averaging the squared difference between orders and the average number of orders. Finally, combinations or simple differentiable functions of these quantities can straightforwardly computed to yield both total cost (TC) and the relative root-mean-square metric (RRMS) (details in Appendix).

6.4.2 Double-Rollout Supervision

Another challenge of optimizing downstream inventory performance is that some aspects must be computed holistically across multiple time points (e.g. order variance). For univariate time-series where a local model is trained on only one time-series, this is especially challenging due to limited supervision. To provide more supervision, we use a custom training method where at each time point, an inventory system simulation is rolled out over the next H time points, where forecasting horizon $H > L$. By simulating the inventory system several times using different starting points in the univariate time-series, one can obtain several evaluations of TC and RRMS to serve as supervision from just one time-series (see Appendix for diagram). For multivariate time-series, instead of forecasting for a horizon $H > L$ and then unrolling lead-time demands across that horizon, only forecasts of the requisite lead time L are made since the other time-series can provide supervision, and double-rollouts are more computationally expensive.

6.4.3 Roll-Forward Evaluation

In real-world settings, as new data are collected, forecasting models are updated and decisions are made accordingly. To simulate this process, we employ a training procedure which rolls forward in time. For each time point from $t = 1$ to $t = T$, the model is trained with data up to t using double-rollout supervision for univariate time-series, and single-rollout supervision for multivariate time-series. Then, the model forecasts the next L timesteps after t, i.e. $\widehat{d}_{t,t+k}$ for all $k \in \{1, 2, ..., L\}$. After all T timesteps have been trained on and forecasted from, giving a $N \times T \times L$ tensor, inventory performance is computed over the T timepoints. For each dataset we designate training, validation, and test time ranges, where validation data is used for hyperparameter tuning, and test data is used for reporting final performance.

6.4.4 Models

We explore two differentiable models for forecasting: (1) a seasonal scaling model, and (2) an LSTM encoder-decoder model. For univariate time-series, one local model is trained per time-series, and for multivariate time-series, one global model is trained across all time-series. Hyperparameter and model training details are in the Appendix.

Naive Seasonal Scaling Model This model has one learnable parameter $\beta \in \mathbb{R}$, the amount to scale observations from one period P ago. That is, $\widehat{d}_{t,\text{seasonal scaler}} = \beta \cdot d_{t-P}$. This model is valuable from an interpretability perspective, as $\beta > 1$ could indicate a preference towards over-forecasting versus the previous period of data, and $\beta < 1$ could indicate a preference towards under-forecasting.

LSTM Encoder-Decoder This model has an LSTM encoder which sequentially encodes a window of inputs, and an LSTM decoder which sequentially decodes to yield predictions across a forecasting horizon. For multivariate time-series, the covariates are embedded before being fed into the encoder, and a linear layer is used on top of the outputs of the decoder to yield the forecasts. See the Appendix for a diagram of the model architecture.

6.4.5 Data

M3 Monthly Industry Subset (Univariate)

The monthly industry subset of the M3 competition data Makridakis and Hibon, 2000b consists of 334 univariate time-series with up to 144 time points, where time points occur on a monthly basis. As described by Petropoulos et al. (2019), this subset can serve as a proxy for demand on a monthly basis. These time-series are not aligned by start date, have varying lengths, and are not directly related to each other. Hence, each time-series in this dataset is treated separately as a univariate time-series for modeling purposes.

Favorita Grocery Sales (Multivariate)

The Corporación Favorita Grocery Sales Forecasting dataset Favorita, 2017 consists of sales data across several stores and products. The dataset includes covariates such as oil prices, location, day of week, month, and holidays. We use a similar preprocessing pipeline as in Lim et al. (2021) to yield 90,193 distinct time-series with up to 396 time points, where time points occur on a daily basis from 2015 to 2016. As grocery replenishment often occurs on a daily basis, the inventory system is updated daily. These time-series are aligned to start at the same time in the real world, missing values are imputed with zeros, and the time-series are likely correlated with each other since they are all associated with sales in Corporación Favorita. Thus, this dataset is treated as a multivariate time-series dataset, and one global model is learned.

6.4.6 Experiment Setup

The models are optimized using the mean squared error (MSE), relative root mean square (RRMS), and total cost (TC) objectives across several settings of unit costs.

For M3, a separate local model is trained with double-rollout supervision and roll-forward evaluation for each of the 334 univariate time-series. Since each time point corresponds to one month, a periodicity of $P = 12$ is used for seasonal models. An encoding window of 24 months is used as input to the model, allowing the model to learn use the previous two periods of history for its predictions. Predictions are made for a forecasting horizon of 12 months, so that the double-rollout can compute inventory performance over multiple time points. A lead time of $L = 6$ months is used. Out of 144 months, forecasting models are initially trained with 72 months, then validated until 108 months, and then tested until 144 months. Since the safety stock discouraged forecasting errors, whereas a high or low unit holding cost could encourage over- or under-forecasting, we choose to have $\alpha_s = 0.5$, so $ss_t = 0$ for all t, to avoid unstable interactions between unit costs and safety stocks. For the TC objective, models are trained on every combination of unit holding costs $c_h \in \{1, 2, 10\}$, unit stockout costs $c_s \in \{1, 2, 10\}$, and unit order variance costs $c_v \in \{1\text{e}{-6}, 1\text{e}{-5}\}$ (chosen based on the order of magnitude of demand).

For Favorita, one global model is trained with single-rollout supervision and roll-forward evaluation for all 90,193 multivariate time-series. A global model is used because all time-series are aligned and correlated with one another, and training a separate model for each series would be computationally expensive. Each time point corresponds to one day, so a periodicity of $P = 7$ is used. An encoding window of 90 days is input to the model, which forecasts the next 30 days. A lead time of $L = 7$ days is used. Out of 396 days, the training cutoff is at day 334, the validation cutoff is day 364, and the remainder is used for testing. Again, we set $\alpha_s = 0.5$. For the TC objective, due to more expensive training, a subset of $N = 10,000$ samples are extracted to test all combinations of unit holding costs $c_h \in \{1, 2, 10\}$, unit stockout costs $c_s \in \{1, 2, 10\}$, and unit order variance costs $c_v \in \{1\text{e}{-3}, 1\text{e}{-2}\}$.

6.5 Results

Unit Cost-Agnostic Performance Tables 6.1 and 6.2 characterize the performance of several forecasters trained on the full M3 and Favorita datasets when evaluated on typical forecasting metrics, MSE and sMAPE, and an inventory performance metric, RRMS. All of the models in these tables are trained and evaluated on objectives that are agnostic to unit costs c_h, c_s, and c_v.

In both M3 and Favorita, the Seasonal Scaler and LSTM models trained with MSE objective perform competitively with classical models on MSE and sMAPE, either performing better than or within the range of performance spanned by the ARIMA, Exponential Smoothing, and Theta models. In the M3 dataset, the model with best RRMS is the seasonal scaler trained on RRMS—abbreviated as Seasonal Scaler (RRMS). However, it achieves worse MSE (22.10$\times 10^5$) than the Seasonal Scaler (MSE), LSTM (MSE), Exponential Smoothing, and Theta models (MSEs ranging 13.82$\times 10^5$ to 20.89$\times 10^5$). In the Favorita dataset, the Seasonal Scaler (MSE) and the Seasonal Scaler (RRMS) outperform all other models on RRMS, despite having worse MSE (1.23$\times 10^2$ and 1.25$\times 10^2$) than the LSTM (MSE), ARIMA, and Theta models (0.88$\times 10^2$ to 1.12$\times 10^2$). On the M3 dataset, the LSTM (MSE) achieves the best MSE (13.82$\times 10^5$), yet has the second to worst RRMS (1.23). On Favorita, the LSTM (MSE) objective again achieves the best MSE (0.88$\times 10^2$), yet has the worst RRMS (1.17). Overall, performance on typical forecasting metrics (MSE and sMAPE) appears misaligned with relative inventory performance (RRMS), and optimizing for one does

Table 6.1: M3 test performance of models that are agnostic to unit costs. Note that the best-performing models on MSE are misaligned with the best-performing models on RRMS.

Model (Objective)	MSE ($\times 10^{-5}$)	sMAPE	RRMS
Seasonal Scaler (MSE)	20.89	0.27	1.27
Seasonal Scaler (RRMS)	22.10	0.28	**0.74**
LSTM (MSE)	**13.82**	0.24	1.23
LSTM (RRMS)	226.49	1.25	1.02
ARIMA	25.78	0.34	1.12
Exponential Smoothing	15.21	0.24	1.12
Theta	14.00	**0.22**	1.09

Table 6.2: Favorita test performance of models that are agnostic to unit costs. Note the best-performing models on MSE are misaligned with the best-performing models on RRMS.

Model (Objective)	MSE ($\times 10^{-2}$)	sMAPE	RRMS
Seasonal Scaler (MSE)	1.23	**1.51**	**0.85**
Seasonal Scaler (RRMS)	1.25	1.73	0.94
LSTM (MSE)	**0.88**	1.77	1.17
LSTM (RRMS)	3.42	2.84	1.10
ARIMA	1.12	1.76	1.06
Exponential Smoothing	1.36	1.80	1.10
Theta	1.08	1.81	1.09

not inherently seem to optimize for the other.

Performance Across Several Unit Cost Tradeoffs Tables 6.3 and 6.4 contain the test total cost across various unit cost settings. While Seasonal Scaler observes some benefit from training using the RRMS objective, it appears to cause unstable performance for the LSTM. On the other hand, *using the TC objective almost always improves the total cost* of the Seasonal Scaler and LSTM models, except for the LSTM on the M3 dataset, where a more consistent benefit is observed for imbalanced c_h and c_s (Figure 6.3 and Appendix Figure E.7). The greater the imbalance in c_h and c_s, the greater the improvement from using TC objective. For example, on Favorita, the Seasonal Scaler trained on TC achieves a 45.7% improvement over that trained by MSE when $(c_h, c_s, c_v) = (1, 10, 10^{-3})$, and the LSTM encoder-decoder trained on TC achieves a 54.0% improvement over that trained by MSE when $(c_h, c_s, c_v) = (10, 1, 10^{-3})$.

Leveraging the interpretability of the Seasonal Scaler model, we graph the relationship between the learned βs and the tradeoffs between c_h and c_s (Figure 6.4). In both M3 and Favorita, with larger c_h/c_s ratios and increasing c_v, the learned β scaling factor decreases.

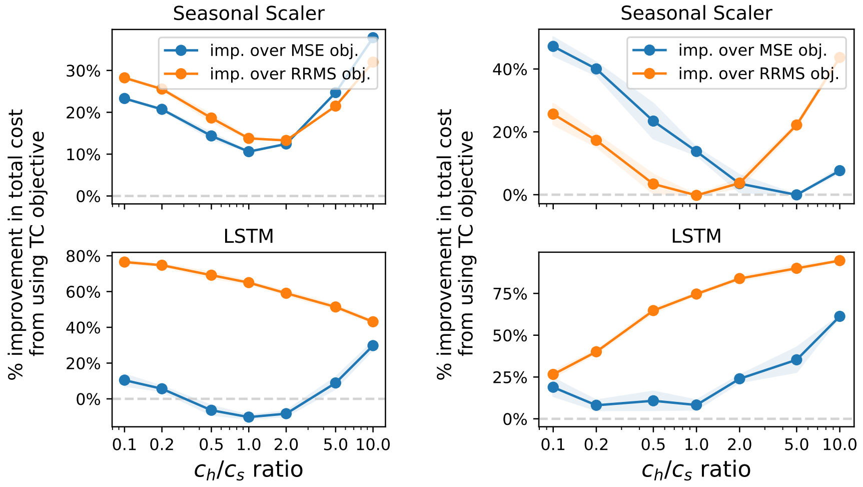
Another benefit of end-to-end optimization of forecasts is interpretability that the forecasts

Table 6.3: M3 test total cost across several unit cost settings (c_h, c_s, c_v).

Model (Objective)	(1, 1, 1e-5)	(1, 1, 1e-6)	(1, 10, 1e-5)	(1, 10, 1e-6)	(10, 1, 1e-5)	(10, 1, 1e-6)
LSTM (MSE)	5,826	5,435	39,519	39,128	20,660	20,269
LSTM (RRMS)	17,188	17,102	170,738	170,652	17,474	17,388
LSTM (TC)	6,390	5,997	36,086	35,805	**10,146**	**10,400**
Seasonal Scaler (MSE)	5,680	5,372	43,791	43,483	15,610	15,302
Seasonal Scaler (RRMS)	5,771	5,476	45,216	44,920	15,314	15,018
Seasonal Scaler (TC)	**5,185**	**4,884**	**35,268**	**34,918**	12,178	11,996

Table 6.4: Favorita test total cost across several unit cost settings (c_h, c_s, c_v).

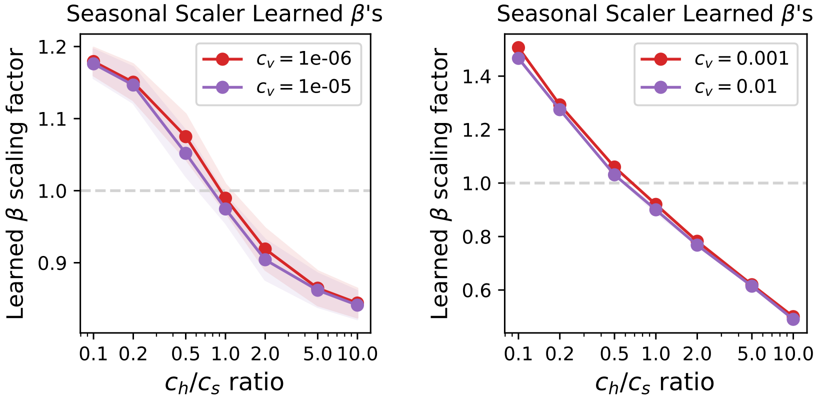
Model (Objective)	(1, 1, 1e-2)	(1, 1, 1e-3)	(1, 10, 1e-2)	(1, 10, 1e-3)	(10, 1, 1e-2)	(10, 1, 1e-3)
LSTM (MSE)	26.94	20.30	158.65	152.01	71.25	64.61
LSTM (RRMS)	94.51	72.08	162.55	140.12	652.76	630.33
LSTM (TC)	24.69	**18.39**	**115.72**	116.40	35.96	**29.72**
Seasonal Scaler (MSE)	24.52	21.22	200.51	197.22	36.25	32.96
Seasonal Scaler (RRMS)	**23.40**	18.69	158.51	153.80	51.84	47.13
Seasonal Scaler (TC)	23.52	18.51	118.48	**107.03**	**32.68**	30.18



(a) M3

(b) Favorita

Figure 6.3: Average relative percentage improvement in test total cost from using the TC objective over the MSE objective and RRMS objective across various c_h/c_s ratios. 95% CI are computed across different c_v values.



(a) M3

(b) Favorita

Figure 6.4: Learned scaling factors for the naive seasonal scaler on the M3 (left) and Favorita (right) datasets, across several unit cost tradeoffs. Dotted line corresponds to $\beta = 1$.

themselves provide. Appendix Figure E.6 plots the forecasted and true lead demands, averaged over all series in each dataset, for various cost objectives. When trained on MSE, the LSTM model tends to slightly over-forecast the true demand in aggregate, whereas the Seasonal Scaler model appears to match or slightly under-forecast. When trained on the TC objective, both models tend to under-forecast when the c_h/c_s ratio is high, and over-forecast when the ratio is low. The LSTM predictions on M3 are smoother than that of the Seasonal Scaler, perhaps because the LSTM is more flexible and able to reduce the variance of its predictions in order to help reduce order variance, whereas the Seasonal Scaler can only scale the previous period by some constant (but note that due to roll-forward evaluation, where the model is updated each time point, this constant can change over time). The LSTM predictions on Favorita are more variable, perhaps due to the small order variance penalty.

6.6 Discussion

We demonstrate the limitations of using standard forecasting metrics that are agnostic to downstream business metrics, and propose a method for augmenting models with business metric-aware objectives. Common forecasting metrics such as MSE and sMAPE can be misaligned with downstream inventory performance, and optimizing for such metrics does not inherently optimize for inventory performance metrics (Tables 6.1 and 6.2). We derive a differentiable procedure for computing inventory performance, and demonstrate that especially when costs are imbalanced, utilizing a business metric-aware total cost objective often significantly improves downstream costs (Tables 6.3 and 6.4, Figures 6.3 and E.7).

When deployed in a roll-forward evaluation framework, we observe that the Seasonal Scaler can be surprisingly effective (Table 6.1 and 6.2) in some cases despite only having one learned parameter. One possible explanation is that the learned constant can vary over time and adapt to new data each timestep as it is observed. In contrast to standard evaluation in which one assumes that the model is learned on data from a fixed time period and evaluated on a fixed test time period, this form of evaluation could be more realistic for the inventory management setting, where forecasts are constantly updated to inform daily, weekly, or monthly decisions. At the same time, the Seasonal Scaler must have an output proportional to the previous period which restricts the flexibility of this model class even if the proportion can change over time. The more flexible LSTM model performs best in MSE and in several of the cost tradeoff settings.

There are also some practical benefits of end-to-end optimization with business metric-aware objectives. When demand forecasting and inventory optimization are treated separately (as is typical), errors in each component are likely to compound. While some have proposed searching over conventional forecasting methods for models which happen to achieve better downstream inventory performance (Petropoulos et al., 2019), this is computationally expensive as several models must be trained, each of which have their own hyperparameters to be tuned. By directly optimizing end-to-end, this can save computation. Additionally, our methods for optimizing inventory performance are compatible with any differentiable forecaster.

More broadly, business metric-aware forecasting could be useful beyond inventory management, for other business problems that rely on forecasts. Through our case study in inventory, we have shown how to tackle some common challenges that might arise from attempting to simulate downstream decision-making processes and systems, including: (1) differentiable computation of persistent state variables, (2) optimization of objectives that must be computed over the entire time-series rather than point-wise, (3) providing additional supervision with limited time series, and (4) simulating how models would be updated over time as new data points are observed. Business decisions that rely on forecasts may inherently prefer error distributions that are biased in certain ways, and we encourage others to explore business metric-aware forecasting in their own business problems.

Limitations and Future Work While TC outperforms MSE when costs are imbalanced, when costs balance each other out, the MSE objective can perform comparably or even slightly better than the TC objective (Figure 6.3a, bottom). The TC objective, while differentiable, is more complex than MSE, and can be sensitive to hyperparameter tuning. Similarly, while RRMS is a convenient unitless objective, it can also be difficult to optimize. While in this work we decided to purely compare inventory vs. generic objectives, future work might explore pre-training with MSE and fine-tuning with TC or RRMS.

Finally, there are several possible avenues for further exploration. Future work could use the lens of business metric-aware forecasting to consider other differentiable model architectures, time series datasets, downstream objectives, and downstream business problems.

Chapter 7

Timing as an Action: Learning when to Observe and Act

In standard reinforcement learning setups, the agent receives observations and performs actions at evenly spaced intervals. However, in many real-world settings, observations are expensive, forcing agents to commit to courses of action for designated periods of time. Consider that doctors, after each visit, typically set not only a treatment plan but also a follow-up date at which that plan might be revised. In this work, we formalize the setup of *timing-as-an-action*. Through theoretical analysis in the tabular setting, we show that while the choice of delay intervals could be naively folded in as part of a composite action, these actions have a special structure and handling them intelligently yields statistical advantages. Taking a model-based perspective, these gains owe to the fact that delay actions do not add any parameters to the underlying model. For model estimation, we provide provable sample-efficiency improvements, and our experiments demonstrate empirical improvements in both healthcare simulators and classical reinforcement learning environments.

7.1 Introduction

In the real-world, decisions are often spread across irregular intervals of time. After each visit, doctors must commit to not only a course of treatment but also to a follow up plan. Each office visit offers an opportunity to gain fresh information and course correct if the current treatment regime is unsuccessful. On the other hand, excessive visits are expensive, consuming hospital resources and consuming time that could be spent on patients in greater need. Thus doctors must trade off the value of information gained the cost of more frequent opportunities to observe and intervene. Similarly, research advisors must decide not only how to advise students in each meeting, but also how frequently to schedule these touchpoints. Economists have considered related scenarios where firms incur a cost for observing market state and must set pricing policies that will hold in between observations (Mankiw and Reis, 2002; Stokey, 2008). When the state is fully unobserved between actions, agents must anticipate both the state and the speed at which their information of the state will go stale in order to choose both action and time to next observation.

Several works in disease progression modeling have applied multi-state models, which assign probabilities or intensities to transitions between different discrete states, to capture state transitions across periods of non-observation (Jackson, 2011; Young et al., 2020; Lorenzi et al., 2019; Cheung et al., 2022). For cardiovascular disease, Lindbohm et al. (2019) utilized multi-state Markov models to estimate rates of progression, for different risk groups, demonstrating how different screening intervals can lead to different tradeoffs between cost and quality-adjusted life years. Breast cancer screening has been the subject of substantial controversy (Esserman, 2017; Marmot et al., 2013), with different organizations recommending different screening policies (Ren et al., 2022). However, the prior literature leaves open the question of how a reinforcement learner ought to go about learning a joint policy over actions and observation intervals.

In this chapter, we explore how one might *learn* policies in an action space augmented by the choice of when to observe and take the next action. We show that this setting is amenable to standard model-free and model-based reinforcement learning algorithms in this augmented action space, but also propose a new *timing-aware* model-based approach which can leverage the temporal nature of the timing action. We prove theoretically that the timing-aware algorithm has improved sample efficiency compared to the aforementioned standard approaches, which arises from more efficient model estimation, and empirically characterize the estimation error rates of timing-aware, timing-naive, and model-free strategies under various quantities of samples and exploration policies, showing that the timing-aware strategy is able to consistently achieve the lowest estimation error with fewer samples. In the disease progression, windy gridworld, and glucose reinforcement learning environments, we demonstrate empirically that timing-aware learning consistently achieves the lowest estimation error the quickest, and is also able to achieve the highest average cumulative reward. At the same time, we empirically find that low estimation error is not always necessary for good performance as measured by average cumulative reward. Finally, we release our timing-as-an-action simulators to encourage further model and algorithmic development in this setting.

7.2 Timing-as-an-Action

Consider the motivating setting where a patient with a chronic illness visits a doctor, who prescribes them a daily medication and schedules a follow-up appointment. We design the timing-as-an-action problem setting to mimic this interaction dynamic, where importantly, (1) the doctor must choose not only which treatment (*action*) to recommend but also how long (*delay*) to recommend it for, (2) the doctor *does not observe* the patient's intermediate state or the benefit of the medication until the next appointment (no observations of state or reward until after the delay), (3) there is some *cost* to each appointment (observation and action cost). With these characteristics in mind, we define the timing-as-an-action Markov decision process (MDP) and reinforcement learning (RL) setup.

Timing-as-an-action MDP The *timing-as-an-action Markov decision process* is an infinite-horizon MDP defined by the tuple $M = (\mathcal{S}, \mathcal{A}, \mathcal{K}, P, R, \gamma, C, s_0)$, with state space $\mathcal{S}$, action space $\mathcal{A}$, and *delay space* $\mathcal{K} = \{1, 2, ..., K\}$, where $K \in \mathbb{N}$. The delay space $\mathcal{K}$ represents the set of numbers of timesteps for which an action can be repeated, and a policy in the timing-as-an-action MDP must make decisions over both actions and how long to take them for (the delay), i.e., $\pi : \mathcal{S} \to \Delta(\mathcal{A} \times \mathcal{K})$, where Δ indicates the probability simplex. Note that with different choices of k, the resulting sequence of observations will be unevenly spaced in terms of the underlying timestep, which we will refer to as the *primitive* timestep. The underlying one-step reward function, or *primitive reward function* $R : \mathcal{S} \times \mathcal{A} \to [0, 1]$, is assumed to be deterministic and in a bounded nonnegative interval. Additionally, there is a discount factor $\gamma \in [0, 1)$; a fixed, known interaction cost $C \in \mathbb{R}_{\geq 0}$; and a deterministic starting state s_0. The k-step transition probabilities are given by $P : \mathcal{S} \times \mathcal{A} \times \mathcal{K} \to \Delta(\mathcal{S})$, where Δ is the probability simplex, and $P(s'|s, a, k)$ denotes the probability of transitioning to a next state s' after taking action a for k timesteps from a state s.

Importantly, the true transition probabilities P have the structure that the k-step transitions are induced by the 1-step transitions. Before formalizing this property we introduce some additional notation. For any valid transition P', let $P'_{a,k}(s'|s) := P'(s'|s, a, k)$ for short, and let the bolded version $\mathbf{P}'$ denote the corresponding $A \times K \times S \times S$ matrix, where indexing into the matrix is denoted as $\mathbf{P}'[a, k, s, s'] := P'(s'|s, a, k)$, and we also denote $\mathbf{P}'_{a,k} := \mathbf{P}'[a, k, :, :]$ and $\mathbf{P}'_{a,k}(s'|s) := \mathbf{P}'_{a,k}[s, s']$. In the timing-as-an-action MDP, we have $\mathbf{P}_{a,k} = \mathbf{P}_{a,1}^k$ for all $(a, k) \in \mathcal{A} \times \mathcal{K}$, which refers to the one-step transition probability matrix multiplied by itself k times.

Timing-as-an-action RL setup In the *timing-as-an-action RL setup*, the agent alternately observes a state s, chooses an action a to commit to, as well as a delay k, that corresponds to the number of timesteps the action a is played for. The agent then observes state s' as well as the aggregated k-step reward g,

$$g = -C + \textstyle\sum_{j=0}^{k-1} \gamma^j r_j, \tag{7.1}$$

that is the discounted sum of the (unobserved) one-step rewards encountered along the k steps of taking action a, from which C, the cost of interaction, is subtracted. For clarity, one step of an agent's interaction with the environment $\texttt{env} = M$, i.e. calling `s', g = env.step(a,k)`, is summarized below (ignoring termination conditions and `done`'s for simplicity):

Timing-as-an-action `env.step(a,k)`

Given: $\texttt{env} = M$, current state s.

Initialize $s_0 = s$. For $j = 0, \ldots, k-1$:

1. Sample $r_j \sim R(s_j, a)$
2. Transition to $s_{j+1} \sim P(\cdot|s_j, a, 1)$

Out: aggregate reward $g = -C + \sum_{j=0}^{k-1} \gamma^j r_j$; next state $s' = s_k$

Crucially, the intermediate states $(s_1, \ldots, s_{k-1})$ and intermediate one-step rewards $(r_1, \ldots, r_{k-1})$ *are not* observed—only s_k and the aggregate rewards g (defined above) are. This captures the challenges of learning when to observe, core to healthcare applications as discussed previously, and distinguishes our problem setting from the "standard" RL setup.

Value functions We call $G : \mathcal{S} \times \mathcal{A} \times \mathcal{K} \to \Delta([-C, \frac{1}{1-\gamma} - C])$ the distribution over aggregated rewards induced by R and P, i.e., $g \sim G(s, a, k)$, with expected value

$$\mathbb{E}[G(s, a, k)] = -C + \textstyle\sum_{j=0}^{k-1} \gamma^j \mathbb{E}[R(s_j, a) | s, a].$$

When a policy π interacts continuously with M, it observes a trajectory $(s_0, a_0, g_0, s_1, a_1, g_1, \ldots)$, and its state-action value function of policy π is the expectation of its total discounted return over the infinite horizon of interaction:

$$Q^\pi(s, a, k) = \mathbb{E}\left[\sum_{\tau=0}^{\infty} \gamma^{\left(\sum_{\tau'=0}^{\tau-1} k_{\tau'}\right)} g_\tau | \pi, s_0 = s, a_0 = a, k_0 = k\right]$$

and its state value function is $V^\pi(s) = \mathbb{E}_{a,k \sim \pi(\cdot|s)}[Q^\pi(s, a)]$. Lastly, the goal of the *timing-as-an-action reinforcement learning problem* is to find the optimal policy $\pi^* = \text{argmax}_{\pi : \mathcal{S} \to \Delta(\mathcal{A} \times \mathcal{K})} V^\pi(s_0)$ that learns which actions *and* delays to take in order to maximize its total discounted return. We define the optimal value function $Q^* := Q^{\pi^*}$, and same for V^*.

7.3 Related Work

The timing-as-an-action framework is most closely related to the options and hierarchical RL literatures, but there are key differences, described shortly, that make them very different learning problems. Broadly, an option is a pre-defined, temporally extended sequence of actions. An MDP endowed with a set of options is called a semi-MDP, and the agent's policy chooses options to take. The options framework has been commonly used for reasoning at different levels of temporal abstraction (Sutton et al., 1998; Sutton et al., 1999; Bacon et al., 2017; Machado et al., 2023). Options belong to a class of reinforcement learning (RL) approaches called hierarchical RL, which involves decomposing a task into subtasks at varying levels of granularity (Barto and Mahadevan, 2003; Dietterich, 2000; Vezhnevets et al., 2017; Co-Reyes et al., 2018; Eysenbach et al., 2019; Hafner et al., 2022).

The key difference between the timing-as-an-action framework and semi-MDPs or hierarchical RL is that the latter frameworks generally assume that per-step observations and rewards are available to the learner, and subtasks are often accomplished with a combination of different granular actions. In contrast, this work utilizes repeated actions (to reflect, e.g., a patient following a treatment plan), and does not assume access to per-step observations or rewards, but rather the aggregated reward and final observation after the chosen duration for the action has passed (e.g. when the patient comes back for a follow-up visit). Semi-MDP methods which rely on per-step rewards and observations are thus not applicable.

Model-based RL has offered a sample-efficient approach for settings where interactions may be expensive to collect (Kaelbling et al., 1996; Deisenroth et al., 2011; Sutton and Barto, 2018). Motivated by human cognition, Ha and Schmidhuber (2018) proposed a model-based framework that learns an autoencoder vision network and recurrent neural memory network to represent the environment dynamics. In a "dream" world simulated by these learned networks, a small controller network is trained. For continuous-time domains with irregularly observed data, Du et al. (2020) use neural ordinary differential equations for model-based reinforcement learning in semi-Markov decision processes. However, as far as we are aware, none of these setups assign a cost to observing and acting, and proactively jointly optimize for the next choice of delay and action.

Repetition of actions has been found to be useful for improving exploration in simple classical RL environments (Dabney et al., 2020) as well as gaming environments such as Atari (Braylan et al., 2015) and VizDoom (Khan et al., 2019), where skipping frames can lead to improvements in learning speed and final performance. Prior work on learning action repetitions has used a Q-network with multiple output heads per action for different repetition lengths (Lakshminarayanan et al., 2017), a framework that jointly learns an action policy and a second policy that decides how often to repeat (Sharma et al., 2017), and using all pairs of intermediate observations to learn the values of multi-step actions (Biedenkapp et al., 2021). However, these works assume access to intermediate observations and rewards.

7.4 Methods

7.4.1 Timing-as-an-action Bellman Backup

To facilitate planning in the timing-as-an-action MDP, we begin with defining the following timing-as-an-action Bellman optimality equation, that recursively relates the value function to itself. For any (s, a, k),

$$Q(s, a, k) = \mathbb{E}[G(s, a, k)] + \gamma^k \mathbb{E}_{s' \sim P(\cdot|s,a,k)}[\max_{a',k'} Q(s', a', k')] \tag{7.2}$$

Such recursive equations are the backbone of value-based RL methods (Agarwal et al., 2019), that optimize policies from learned value functions. (see Appendix F.1.1 for proof). In particular, finding a value function that satisfies (7.2) for all (s, a, k) implies that we have obtained the optimal value function Q^*.

Lemma 7.4.1. *The timing-as-an-action Bellman optimality equation* (7.2) *has a unique fixed point for* Q^*.

As Lemma 7.4.1 is analogous to well-established results for value-based learning in standard MDPs, the immediate implication is that one could solve the timing-as-an-action RL problem by applying standard value-based RL algorithms (e.g., Q-learning), with an expanded action space equal to the cross product of actions and delays $\mathcal{A}' = \mathcal{A} \times \mathcal{K}$. Indeed, we will show that this is the case in Section 7.4.2.

However, it should also be immediately clear that such methods will be sample-inefficient because they do not leverage the structure of the timing-as-an-action MDP, namely, that observing k-step transitions also provides information about the dynamics for $k' \neq k$. In general, the sample complexity of RL algorithms depends on the size of the action space (Azar et al., 2012; Agarwal et al., 2019), here $|\mathcal{A}'| = |\mathcal{A}||\mathcal{K}|$. This can grow rapidly depending on the choice of delay space $\mathcal{K}$, which, in general, we expect to be relatively large as it represents discretized time. For example, if one chose delays up to one day with one-minute intervals between them, the action space would be 1,440 times as large as the single-timestep action space. Ideally, leveraging the temporal nature of the timing action should result in more efficient learning.

7.4.2 Learning Algorithms

We define three value-based RL algorithms based on the timing-as-an-action Bellman backup (7.2). Two approaches, one model-free and one model-based, give a naive treatment of the delay by treating it as any other action, and can be viewed as standard RL algorithms translated directly to the timing-as-an-action setup. The third approach is also model-based, but leverages the temporal nature of the delay, i.e, that $\mathbf{P}_{a,k} = \mathbf{P}_{a,1}^{k}$, to share information between different values of delays. As an extreme example, obtaining perfect 1-step transitions automatically translates to perfect estimation of $P_{a,k}$ for all $k \in K$; more generally, observations of any delay allows for reasoning about the transitions for other delays. Because the delay structure is embedded in the transitions, model-based methods are a natural choice for leveraging this structure (which is not encoded in the Q-values).

Model-free

After taking action a for k steps from state s, the environment returns an aggregated reward g and the next state s'. The *model-free* approach updates the action-values using the standard Q-learning update (Watkins and Dayan, 1992):

$$\widehat{Q}(s, a, k) \leftarrow g + \gamma^k \max_{a',k'} \widehat{Q}(s', a', k'). \tag{7.3}$$

Here the action space is simply the cross product of all actions and delays $(a, k) \in \mathcal{A} \times \mathcal{K}$, and a sample of experience with delay k does not inform the values associated with $k-1$, $k+1$, etc. To help improve sample efficiency, we add experience replay, iterating through tuples (s, a, k, g, s') and updating using (7.3) (details in Algorithm 3). To further improve sample efficiency, we consider model-based methods.

Model-based

In the model-based approaches, we learn models of the transition probabilities $\widehat{P}$ and one-step rewards $\widehat{R}$ using a dataset of the form $\{(s_i, a_i, k_i, g_i, s'_i)\}_{i=1}^{N}$, which are then used to obtain the Q-value estimates $\widehat{Q}$.

Algorithm 3 Model-free learning procedure

```
Input: MDP $M$, environment env to interact with $M$, policy transformation $\pi_Q : Q \to \Delta(\mathcal{A} \times \mathcal{K})^S$
Initialize replay buffer $B = \emptyset$ and Q-values $\widehat{Q}(s,a,k) = 0 \; \forall s,a,k \in \mathcal{S} \times \mathcal{A} \times \mathcal{K}$.
for each episode do
    $s$, done $=$ env.reset$()$, False
    while not done do
        $a, k \sim \pi_{\widehat{Q}}(\cdot, \cdot | s)$
        $s', g,$ done $=$ env.step$(a, k)$
        Append $B \leftarrow B \cup \{(s, a, k, s', g)\}$
        for $(s, a, k, g, s') \in B$ do
            Update $\widehat{Q}$ via (7.3):
                $\widehat{Q}(s,a,k) \leftarrow g + \gamma^k \max_{a',k'} \widehat{Q}(s', a', k')$.
        end for
    end while
end for
```

For the *timing-naive model-based* approach, the transitions are learned through maximum likelihood estimation from the function class $\mathcal{P}$:

$$\widehat{P} = \underset{p \in \mathcal{P}}{\operatorname{argmax}} \sum_{i=1}^{N} \log p_{a_i,k_i}(s_i' | s_i), \tag{7.4}$$

where $\mathcal{P} = \{P : \mathcal{S} \times \mathcal{A} \times \mathcal{K} \to \Delta(\mathcal{S})\}$ is the set of all valid transitions (involving actions and delays). For the *timing-aware model-based* approach,

$$\widehat{P} = \underset{p \in \mathcal{P}_1}{\operatorname{argmax}} \sum_{i=1}^{n} \log \left(p_{a_i,1}^{k_i}(s_i' | s_i) \right). \tag{7.5}$$

where $\mathcal{P}_1 = \{p : p_{a,k}(\cdot|s) = [\mathbf{p}_{a,1}]^k(\cdot|s), \forall (s,a,k) \in \mathcal{S} \times \mathcal{A} \times \mathcal{K}\}$, recalling that $\mathbf{p}$ is the tensor version of p thus $[\mathbf{p}_{a,1}]^k$ is the $S \times S$ one-step transition probability matrix multiplied by itself k times. Note that $\mathcal{P}_1 \subseteq \mathcal{P}$ from (7.4), and the true transitions $P \in \mathcal{P}_1$ given the structure the environment.

Then, given an estimate of the transition probabilities $\widehat{P}$ (from either (7.4) or (7.5)), estimates of the one-step reward function $\widehat{R}$ are obtained as follows:

$$\widehat{R} = \underset{R' \in \mathcal{R}}{\operatorname{argmin}} \frac{1}{N} \sum_{i=1}^{N} (\mathcal{G}_{R',\widehat{P}}(s_i, a_i, k_i) - g_i)^2, \tag{7.6}$$

where $\mathcal{R}$ is a one-step reward function class and $\mathcal{G}$ is a deterministic mapping from a one-step reward function R' and transition P' to the corresponding expected aggregated multi-step

Algorithm 4 Model-based learning procedure

```
Given: MDP $M$, policy transformation $\pi_Q : Q \to \Delta(\mathcal{A} \times \mathcal{K})^S$, $\mathcal{P}' = \mathcal{P}$ in the timing-naive
    approach and $\mathcal{P}' = \mathcal{P}_1\}$ in the timing-naive approach.
Initialize replay buffer $B = \emptyset$, Q-values $\widehat{Q} = 0$, transitions $\widehat{P}$, and one-step rewards $\widehat{r}$.
for each episode do
    $s$, done $=$ env.reset(), False
    while not done do
        $a, k \sim \pi_Q(\cdot, \cdot | s)$
        $s', g$, done $=$ env.step$(a, k)$
        Append $B \leftarrow B \cup \{(s, a, k, s', g)\}$
        for $(s, a, k, g, s') \in B$ do
            Update $\widehat{P}$ using (7.4) (timing-naive):
                $\widehat{P} = \operatorname*{argmax}_{p \in \mathcal{P}} \sum_{i=1}^{n} \log p_{a_i, k_i}(s_i' | s_i)$
            or (7.5) (timing-aware):
                $\widehat{P} = \operatorname*{argmax}_{p \in \mathcal{P}} \sum_{i=1}^{n} \log \left( p_{a_i, 1}^{k_i}(s_i' | s_i) \right).$
            Update $\widehat{R}$ using (7.6). $\mathcal{G}_{R', P'}$ is defined in (7.7):
                $\widehat{R} = \operatorname*{argmin}_{R' \in \mathcal{R}} \frac{1}{N} \sum_{i=1}^{N} (\mathcal{G}_{R', \widehat{P}}(s_i, a_i, k_i) - g_i)^2.$
            Update $\widehat{Q} \leftarrow$ value_iteration$(\widehat{P}, \mathcal{G}_{\widehat{R}, \widehat{P}})$.
        end for
    end while
end for
```

reward,

$$\mathcal{G}_{R', P'}(s, a, k) = -C + \sum_{\tau=0}^{k-1} \gamma^{\tau} \mathbb{E}_{s' \sim P'_{a,k}(\cdot | s)}[R'(s', a)]. \tag{7.7}$$

Note that by plugging in the true R and P, we have $\mathcal{G}_{R,P}(s, a, k) = \mathbb{E}[G(s, a, k)]$. The $\widehat{P}$ and corresponding $\widehat{R}$ are then used to learn the Q-value functions via value iteration, i.e., by finding $\widehat{Q}$ that is the fixed point of the Bellman equation involving the estimated $\mathcal{G}_{\widehat{R}, \widehat{P}}$ and $\widehat{P}$ below:

$$\widehat{Q}(s, a, k) = \mathcal{G}_{\widehat{R}, \widehat{P}}(s, a, k) + \gamma^k \mathbb{E}_{s' \sim \widehat{P}_{a,k}(\cdot | s)}[\max_{a', k'} \widehat{Q}(s', a', k')]. \tag{7.8}$$

7.4.3 Analysis

To highlight the sample complexity improvements in model learning achieved by (7.5), we provide our guarantees in the generative setting, which isolates the estimation problem from the challenges of exploration in RL (Azar et al., 2012; Agarwal et al., 2019):

Definition 3 (Generative Setting). *A generative model takes as input (s, a, k) and outputs $s' \sim P(\cdot|s, a, k)$. In the generative setting, we obtain n samples of s' from each $(s, a, k) \in \mathcal{S} \times \mathcal{A} \times K$ using the generative model, i.e. $\mathcal{D}_{s,a,k} = \{(s, a, k, s_i')\}_{i=1}^n$.*

Proofs for the below are provided in Appendix F.1.2, in addition to the more general versions for the non-generative setting:

Lemma 7.4.2. *In the generative setting (Definition 3), for $\widehat{P}$ from (7.4), with probability $\geq 1 - \delta$,*

$$\max_{s,a,k} \|\widehat{P}_{a,k}(\cdot|s) - P_{a,k}(\cdot|s)\|_1 \lesssim S\sqrt{\frac{AK\log(1/\delta)}{n}}.$$

Lemma 7.4.3. *In the generative setting (Definition 3), for $\widehat{P}$ from (7.5), with probability $\geq 1 - \delta$,*

$$\max_{s,a,k} \|\widehat{P}_{a,k}(\cdot|s) - P_{a,k}(\cdot|s)\|_1 \lesssim S\sqrt{\frac{A\log(K/\delta)}{n}}.$$

Comparison of the above transition estimation results reveals the sample complexity gains from (7.5) are obtained by leveraging the delay structure, as evidenced by their respective dependencies on K. While the timing-naive estimate has $\sqrt{K}$ in its upper bound (Lemma 7.4.2), the timing-smart estimation obtains $\log K$ (Lemma 7.4.3), and this is because learning just the 1-step transitions from all samples is more efficient, while still being sufficient to express all k-step transitions.

Lemma 7.4.4. *Fix $\widehat{P}$ and let $\varepsilon_{\widehat{P}} = \max_{s,a,k} \left\|\widehat{P}(\cdot|s, a, k) - P(\cdot|s, a, k)\right\|_1^2$. Then in the generative setting (Definition 3), with probability $\geq 1 - \delta$ we have*

$$\left\|\mathcal{G}_{\widehat{R},\widehat{P}} - \mathcal{G}_{R,P}\right\|_\infty \lesssim \frac{1}{(1-\gamma)} \left(SAK\varepsilon_{\widehat{P}}\right)^{1/2} + \left(\frac{G_{\max}^2 S^2 A^2 K}{n}\right)^{1/2} + \left(\frac{1}{(1-\gamma)^2} \frac{G_{\max}^2 S^2 A^2 K}{n} \varepsilon_{\widehat{P}}\right)^{1/4},$$

where $G_{\max} = \max\{C, |\frac{1}{1-\gamma} - C|\}$.

Lemma 7.4.4 demonstrates that the error of aggregated reward estimation is directly related to the error of transition estimation through $\varepsilon_{\widehat{P}}$; better transition estimation (smaller $\varepsilon_{\widehat{P}}$) translates to faster reward convergence. For $\widehat{P}$ used in the timing-aware or timing-naive model updates, $\varepsilon_{\widehat{P}}$ is given by the bounds in Lemma 7.4.3 and Lemma 7.4.2, respectively, which is n^{-1}, giving the RHS of the bound in Lemma 7.4.4 a fast $n^{-1/2}$ rate of estimation, with $\mathcal{G}_{\widehat{R},\widehat{P}} \to \mathcal{G}_{R,P}$ as $n \to \infty$. Thus, the sample complexity gains in timing-aware model estimation translate to reward estimation as well.

As $\widehat{Q}$ is formed directly from $\widehat{P}$ and $\mathcal{G}_{\widehat{R},\widehat{P}}$ in the model-based update (7.8), the Q-value estimate directly inherits the quality of the $\widehat{P}$ and $\mathcal{G}_{\widehat{R},\widehat{P}}$ estimates, which is a version of the classic simulation lemma below:

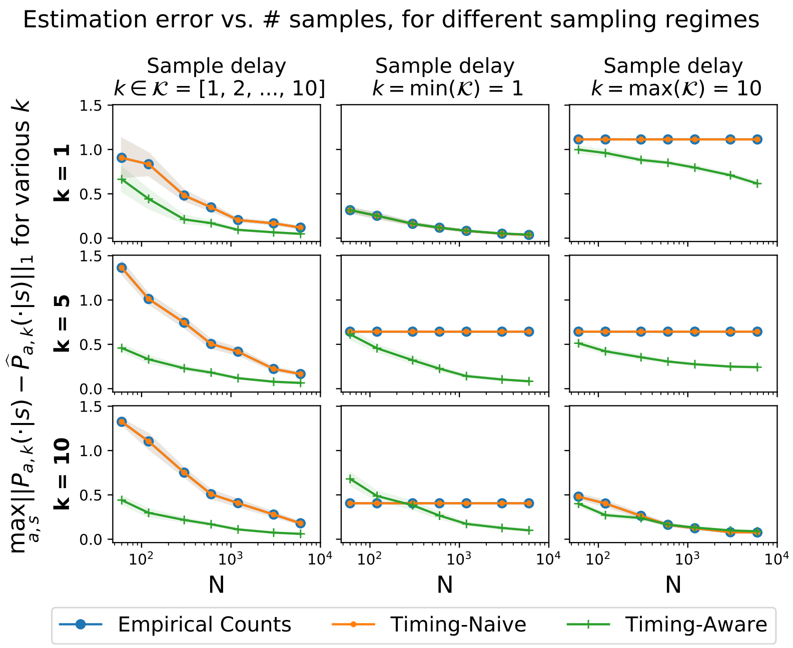


Figure 7.1: Est. error $\max_{a,s} \|P_{a,k}(\cdot|s) - \widehat{P}_{a,k}(\cdot|s)\|_1$ (with 95% CI) for $\widehat{P}_{a,1}$, $\widehat{P}_{a,5}$, and $\widehat{P}_{a,10}$ vs. N, the # of samples generated from three sampling regimes: (a) the generative setting of Definition 3, (b) only sampling $k = \min(\mathcal{K})$, and (c) only sampling $k = \max(\mathcal{K})$.

Proposition 4. *For any $\widehat{P}$ and $\mathcal{G}_{\widehat{R},\widehat{P}}$, let $\widehat{Q}$ satisfy (7.8). Then*

$$\left\|Q^* - \widehat{Q}\right\|_\infty \leq \frac{1}{1-\gamma}\left\|\mathcal{G}_{R,P} - \mathcal{G}_{\widehat{R},\widehat{P}}\right\|_\infty + \frac{\gamma^K}{2(1-\gamma)^2}\max_{s,a,k}\left\|P(\cdot|s,a,k) - \widehat{P}(\cdot|s,a,k)\right\|_1 .$$

7.5 Experiments

Our experiments investigate the model estimation problem (Section 7.5.1) separately from the policy learning problem (Section 7.5.2).

Implementation Details All models are implemented using PyTorch, with transition probabilities $\widehat{P}$ initialized uniformly, and single-step reward estimates $\widehat{R}$ initialized to -1. For the timing-aware model, the estimate of the one-step transition matrix for action a is $\widehat{P}_{a,1} = \text{softmax}(\mathbf{T}[a,:,:])$, where $\mathbf{T}$ is an unconstrained $A \times S \times S$ parameter tensor initialized with ones, and the softmax is over the last dimension. For the timing-naive model, the estimate of the k-step transition matrix for action a is $\widehat{P}_{a,k} = \text{softmax}(\mathbf{T}'[a,k,:,:])$, where $\mathbf{T}'$ is an unconstrained $A \times K \times S \times S$ parameter tensor initialized with ones, and the softmax is over the last dimension. The single-step rewards $\widehat{R}$ are initialized as an $A \times S$ tensor, and estimates of the expected aggregate reward are computed using $\widehat{g} = \mathcal{G}_{\widehat{R},\widehat{P}}(s,a,k)$, defined in (7.7). Additional optimization details are in Appendix F.2.

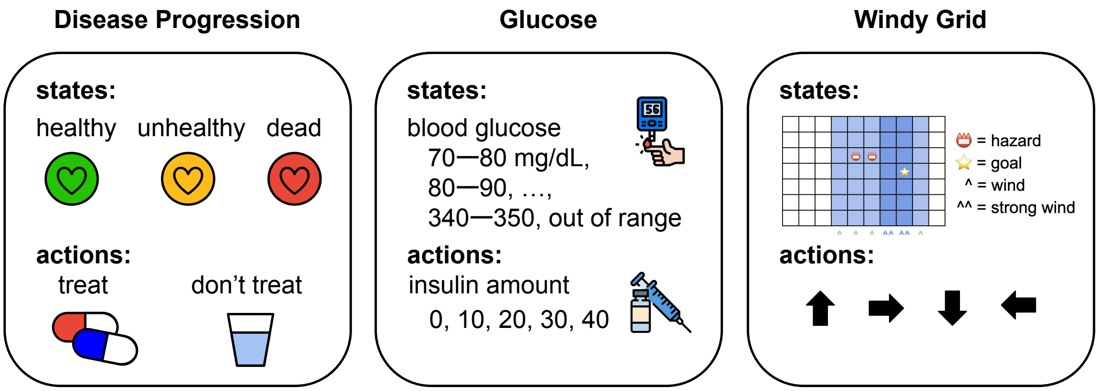


Figure 7.2: Summary of disease, glucose, and windy grid environments. Details in Appendix F.3.

Table 7.1: Final average cumulative reward after 200 episodes. Values are written in the hundreds.

	Disease Progression	Windy Grid	Glucose
Timing-Aware	**4.26 (4.00–4.53)**	**81.5 (80.3–82.6)**	**0.420 (0.287–0.554)**
Timing-Naive	4.24 (4.00–4.47)	76.9 (75.0–78.8)	0.334 (0.224–0.443)
Model-Free	3.47 (3.28–3.65)	3.96 (1.83–6.10)	0.270 (0.183–0.356)

7.5.1 Transition Model Estimation

As the likelihood objective in the timing-aware model-based approach (7.5) may be non-convex in the one-step parameters $p_{a,1}$, we first verify empirically that standard gradient-based optimization methods can learn $\widehat{P} \approx P$ in Figure 7.3. To better compare the rates of learning the transition probabilities in the timing-aware and timing-naive approaches, we examine the L1 error curves for three sampling regimes: (a) drawing samples in the generative setting (Definition 3), (b) drawing an equivalent number of samples selecting actions uniformly with just the minimum delay, and (c) drawing an equivalent number of samples selecting actions uniformly with just the maximum delay. For (a), we draw $n = [1, 2, 5, 10, 20, 50, 100]$ per-(s, a, k) samples $\forall(s, a, k) \in \mathcal{S} \times \mathcal{A} \times \mathcal{K}$, giving $N = nSAK = [60, 120, \ldots, 6000]$ samples to estimate $\widehat{P}$. For (b) and (c) which sample just one value of k, we draw ten times as many per-(s, a, k) samples to obtain the same number of samples N. True transition probabilities P come from the disease progression environment, where $S = 3$, $A = 2$, and $K = 10$. We also sanity-check against the estimate of P from empirically counting transitions to each state given each state and action. Results are averaged over 30 trials.

When all delays are sampled, the timing-aware model-based approach achieves lower estimation error faster than the timing-naive model-based approach and empirical counts. In the generative setting (Definition 3), for example, it only takes the timing-aware approach $n = 20$ draws of all $(s, a, k) \in \mathcal{S} \times \mathcal{A} \times \mathcal{K}$ to achieve a maximum L1 error less than 0.1 in the estimate of the one-step transition probabilities, whereas it takes timing-naive approach more than $n = 100$ per-(s, a, k) draws to achieve the same. While in the timing-naive approach each sample only contributes information towards the corresponding entry with the same delay, in the timing-aware approach each sample contributes to the estimates of transition probabilities of all other delays. This is further demonstrated in the second and third columns of Figure 7.3, where when only one delay is sampled, the timing-naive approach only improves its estimate with samples of the same delay, and the estimates for the other delays remain unchanged.

7.5.2 Reinforcement Learning Setting

For $\pi_{\widehat{Q}}$, we use an ϵ-greedy policy with $\epsilon = 0.1$, where with probability $1 - \epsilon$ the delay and action are $\arg\max_{a',k'} \widehat{Q}(s, a', k')$, and otherwise the delay is $k = 1$ and the action is drawn uniformly from $\mathcal{A}$. Each experiment has 200 episodes (limited to mimic real-world situations with limited data), and 50 trials of each experiment are run. We explore three environments of increasing complexity: (1) a three-state disease progression simulator, (2) a glucose simulator, and (3) a windygrid environment (Figure 7.2). We implement an augmented version of all three simulators which accepts both the action a and delay k, and returns the aggregated rewards g and state s' after having taken k steps with action a (see box in Section 7.2). Code for these simulators is in the supplement.

Progression Environment This simulator is an environment with three states (healthy, unhealthy, dead) and two actions (treat and do not treat). We consider delays of up to ten timesteps, $k \in \mathcal{K} = [1, 2, ..., 10]$. The simulator is based on models commonly used in disease progression modeling, such as multi-state Markov models used in breast cancer progression modeling (Yen et al., 2003; Olsen et al., 2006; Chen et al., 1996; Duffy et al., 1995). The true transition probabilities P are included in Appendix F.3. The healthy state has a reward of 25, the unhealthy state has a reward of 5, and the dead state terminates the episode and has as reward of 0. The action cost is $C = 5$, and $\gamma = 0.99$.

Glucose Environment We implement an augmented version of the SimGlucose simulator (Xie, 2018), with 29 states corresponding to ranges of blood glucose measurements, and five actions corresponding to different insulin amounts. We consider delays $\mathcal{K} = [1, 2, 3, 4]$. The reward and transition probabilities are not defined explicitly, but rather according to dynamics in Clarke and Kovatchev (2009), Xie (2018), and Man et al. (2014). The action cost is $C = 0.5$, and $\gamma = 0.99$.

Windy Grid Environment The windy grid simulator is a classic RL environment (Sutton and Barto, 2018), consisting of a 7×10 grid with 70 states, and four actions (up, down, left, right). We consider a delay space $\mathcal{K} = [1, 2, ..., 10]$. The agent starts at (3, 0), and the goal state is at (3, 7). With probability 0.5, wind (columns 4–6 and 9) pushes the agent up one space, and strong wind (columns 7 and 8) pushes the agent up two spaces. Except for states with wind, the actions produce the expected transition to adjacent states with probability 1. In row 3, columns 5 and 6 have hazards which have a negative reward, -5. The goal state has a reward of 10,000, and the remaining states have a reward of -1. Upon reaching the goal state, the episode terminates. The action cost is $C = 1$, and $\gamma = 0.99$.

RL Results Across episodes, the timing-aware approach achieves the highest cumulative

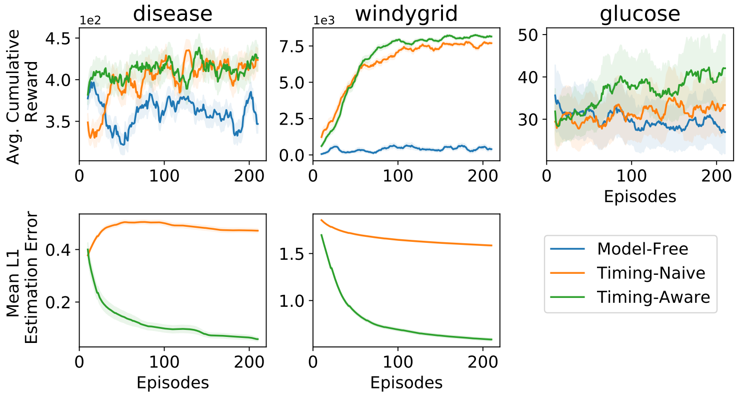


Figure 7.3: Average cumulative reward and mean L1 error ($\|\widehat{P}_{a,k}(\cdot|s) - P_{a,k}(\cdot|s)\|_1$ averaged over all s, a, k) across 50 trials, smoothed with a running average over 20 episodes. Shaded regions are the standard errors.

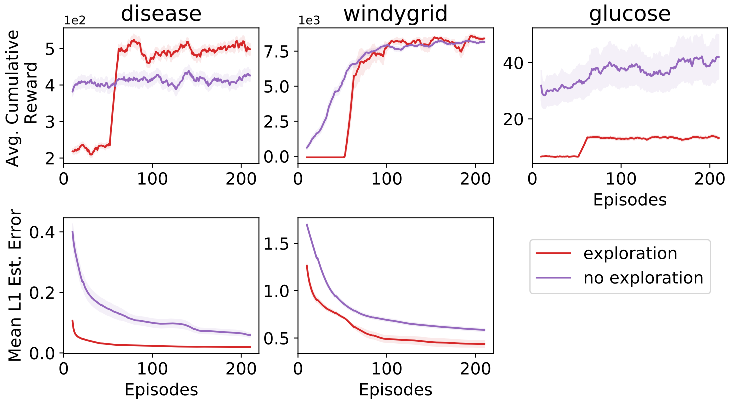


Figure 7.4: Avg. cumulative reward and mean L1 error for timing-aware with/without exploration phase.

reward in all environments (Table 7.1 and Figure 7.2, top). In the disease progression environment it only achieves slightly higher cumulative reward than the timing-naive approach, but in the windy grid and glucose environments it significantly outperforms the timing-naive approach. In all cases, the model-based approaches outperform the model-free approach. In the disease progression and windy grid environments (where we have access to the true P for evaluation purposes), the timing-aware approach is able to obtain significantly lower estimation error $\max_{s,a,k} \|P_{a,k}(\cdot|s) - \widehat{P}_{a,k}(\cdot|s)\|_1$ than the timing-naive approach (Figure 7.2, bottom).

We also experiment with adding an exploration phase of 50 episodes, where actions are taken uniformly at random in the exploration phase before reverting to the ϵ-greedy policy. This approach does decrease the estimation error more quickly (Figure 7.4), however depending on the environment, it has an inconsistent effect on the resulting cumulative reward.

In the disease simulator, all methods execute the "don't treat" action more frequently (assigns higher probabilities to staying in the same state) rather than the "treat" action (encourages switching between states) (Appendix Figure F.2). Once the agent is in a healthy state, it is incentivized to remain there as long as possible. The timing-aware method often takes the largest delay (10 timesteps), whereas the timing-naive method executes intermediate delays more frequently, and the model-free methods execute much shorter delays.

In the glucose simulator, timing-aware most frequently administers the second lowest amount of insulin for the largest delay (4 timesteps) (Appendix Figure F.2). By contrast, the timing-naive method administers a greater variety of quantities of insulin, and does so for an intermediate number of timesteps, most frequently administering for two timesteps. The model-free method utilizes all actions and delays more uniformly.

In the windy grid simulator, all methods tend to utilize shorter delays closer to the goal state, where wind pushes the agent up one or two squares with probability 0.5 (Appendix Figure F.3). Along the first and last rows (rows 0 and 9) of the grid, the timing-aware policy learns to repeat the move right action just long enough to get in the vicinity of the goal. Since there is wind pushing the agent upward in columns 7 and 8, along the top half of the grid the agent learns to go to column 9 first (where there is no wind), and then walk downward to the same row as the goal state before walking left. The optimal policy more closely resembles the timing-aware approach than the timing-naive approach.

7.6 Discussion

The *timing-as-an-action* problem setting poses interesting theoretical and practical challenges for bringing reinforcement learning into real-world settings where opportunities to observe and act can be costly. We demonstrate that the timing-aware model-based method leverages the structure of the timing-as-an-action environment to obtain sample complexity advantages over either model-free (corresponding to standard value-based RL methods translated to our setting) or the timing-naive model-based method. This aligns with intuition, as the timing-naive method must learn SAK parameters for $\widehat{P}$ whereas the timing-aware method only needs to learn SA parameters for $\widehat{P}$.

We demonstrate empirically that estimation using the timing-aware approach is more sample-efficient than the timing-naive approach (Figure 7.3). Additionally, the timing-aware approach updates its estimates for transition probabilities of delay actions other than those which were sampled (middle and right columns of Figure 7.3), whereas the timing-naive approach does not.

In all RL experiments, the timing-aware approach achieves the highest or ties for the highest average cumulative reward (Table 7.1). We note that these results are after 200 episodes, and it is likely that with more episodes the other methods would eventually do as well as the timing-aware approach. In the disease progression and windy grid RL settings, timing-naive has substantially higher estimation error than timing-aware, however the resulting policy is still able substantially outperform the model-free approach and obtain performance comparable or almost comparable to timing-aware, indicating that the learned policy can still perform well even if $\widehat{P}$ is inaccurate (Figure 7.2). Similarly, although an exploration phase helps improve the estimation error (Figure

7.4), the cumulative reward may not necessarily improve. Although our results use the same exploration strategy across all settings, future works may find it beneficial in some settings to have an exploration phase for the first few episodes in order to quickly learn $\widehat{P}$.

Part IV

Conclusion

Chapter 8

Conclusion

Rapid progress in machine learning has led to a proliferation of ML models being deployed in healthcare applications. As machine learning prototypes transition from research labs into real-world healthcare systems, they enter dynamic environments different from those iterated upon in in ML model development. This thesis is driven by the question:

How can we develop machine learning systems suitable for dynamic healthcare settings?

We posit that building reliable and practical ML systems for healthcare requires us to $(S1)$ understand the types of shifts that occur in healthcare data over time, $(S2)$ develop models and algorithms that are robust to these shifts, and $(S3)$ take decision-making processes into account. Chapters 2 and 3 highlight several axes along which changes in the real-world environment can thwart model performance. One such issue in healthcare data is the problem of underreporting or missing data. In Chapters 4 and 5, we discuss possible techniques for learning despite underreporting, and formalize the problem domain adaptation under missingness shift. Finally, in Chapters 6 and 7, we consider the broader decision-making context in which ML models are utilized, and propose a framework for decision-making in the context of doctor-patient interaction dynamics.

To draw an analogy betewen ML model development and drug development and testing, many works at the intersection of ML and healthcare are in a *pre-clinical* laboratory study phase, where researchers are developing and testing new ideas in a controlled environment before testing on humans in clinical trials. Pre-clinical studies include *in-vitro* studies, which look for effects of the new treatment on cells grown in a lab dish or test tube, as well as *in-vivo* studies, which test the promising treatments on live animals in order to test their safety for living creatures ACS, 2020. In the context of ML models for healthcare applications, in-vitro studies might correspond to using synthetic, semi-synthetic, or carefully curated datasets where challenging realities of healthcare data (e.g. missing data, label imbalance, noisy features, latent confounding, distribution shifts over time, etc.) are meticulously cleaned or simplified in some manner. In-vivo studies might correspond to more realistic datasets, where there are greater uncertainties in the data generating process, but the set of data has already been collected and the model is evaluated on a fixed test set. Some of the “in-vitro” works in this thesis are

Chapter 5, Chapter 6, and Chapter 7, which propose and explore new problem formulations and algorithms motivated by dynamics and challenges present in healthcare settings, but where experiments are not conducted on true retrospective healthcare data. The translation of these works from "in-vitro" to "in-vivo" would be fruitful directions for future work. Chapter 2, Chapter 3, and Chapter 4 might be considered "in-vivo" works, as they develop and evaluate on actual retrospective healthcare data. These works elucidate some of the challenges that arise when applying ML methods to real-world healthcare settings, provide recommendations for how to evaluate ML models before deployment, and also motivate interesting new directions where existing ML methods fall short. For example, while model efficacy is typically evaluated on a single held out test set, a prudent pre-deployment stress test would be to evaluate using the framework described in Chapter 3. This type of evaluation retrospectively simulates would-be performance upon deployment, and could help practitioners understand the volatility of the environment in which the model will be deployed. Knowledge of possible distribution shifts could then help with proactive preparation for such shifts in the future.

Future Open Directions There are several open directions for future work in ML for dynamic healthcare settings. In addition to the future work mentioned at the end of each chapter, here we draw some broader connections between the chapters and highlight some open questions.

As seen in Chapters 2 and 3, missing data is a common limitation in healthcare data that can greatly impact performance of ML models. Underreporting may be driven by factors relating to inequity of access to healthcare, for example, if a patient is unable to afford a test, or if there is a language barrier resulting in inaccurate reporting of symptoms. While our methods in 4 and 5 are able to learn despite underreporting, it would be valuable to approach the issue of underreporting from a fairness perspective, characterizing how the proposed approaches might impact different subgroups, and whether there might be approaches that one could take to remedy disparate treatments or impacts.

How to regulate ML models in healthcare is a timely and important open challenge. If an ML model performs "well" in a trial period, does it give us enough confidence to approve it for use on patients, and for how long should this approval be extended? What axes of performance are most important to consider? How do we assess the relative benefits and harms of the ML model compared to the status quo? Are there any longer-lasting effects of reliance on ML models to clinical decision-making? What would give us confidence that all patients are getting the best care possible, and that no avoidable harm is being done? Are benefits and harms distributed equitably across different subgroups of patients? Given the wide variety of possible ways to integrate ML into healthcare processes and decisions, numerous subjective tradeoffs of potential benefit and potential risk, and varying degrees of volatility in different deployment environments, the question of how to regulate such models often may not be a one size fits all solution, but require careful weighing of several factors.

Due to the reliance of ML models on historical data, and the dynamic nature of our world, it is also important to have numerous sanity checks constantly running on the model so that model failures do not go undetected. In addition to the proactive modeling approaches described in this thesis, it is also important to have reactive approaches that can detect when a model is failing, and to have a plan for how to respond to such failures. After taking any immediate steps to mitigate harm, it is important to understand the root cause of the failure, and to once again take proactive steps to prevent such failures in the future. Depending on the downstream application of the model and feasibility given limited resources, it may be valuable to have a human in the loop as well.

Appendix A

Risk Score for COVID-19 ECMO Planning

Extended Version Additional details can be found at https://arxiv.org/abs/2006.01898.

Table A.1: Summary characteristics per cohort, with median (Q1-Q3) or count (% of n).

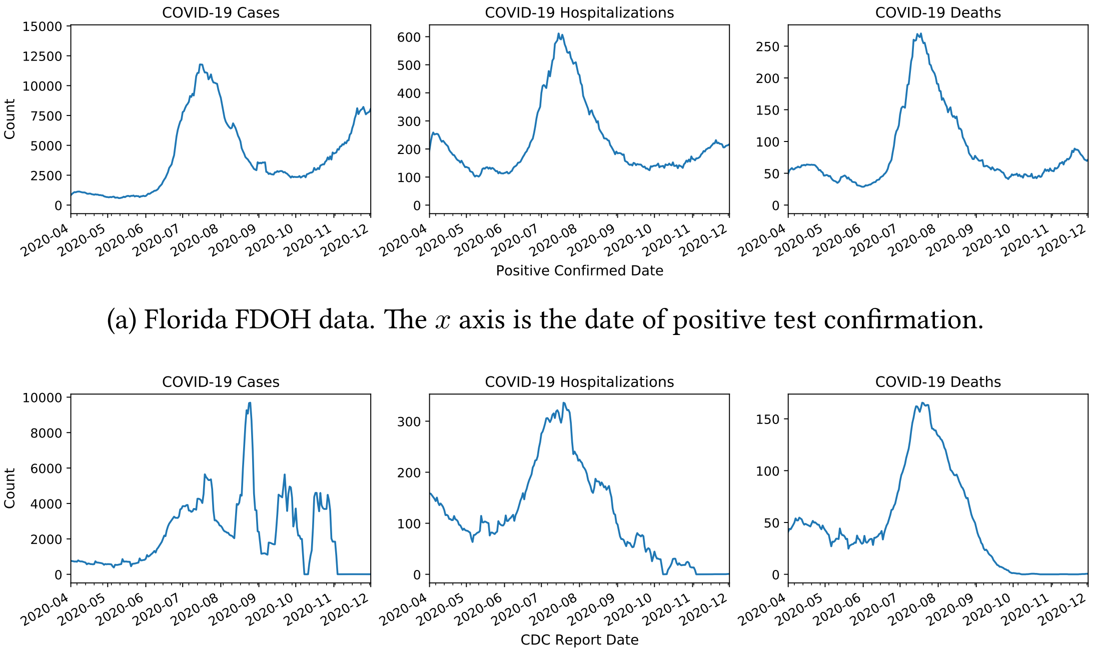
		Variable	eICU (n = 3617)	MIMIC (n = 937)
Physical exam findings		Orientation: oriented	1121 (31.0%)	411 (43.9%)
		Orientation: confused	1287 (35.6%)	76 (8.1%)
		Temperature (°C)	36.9 (36.6-37.3)	37.2 (36.6-37.7)
		Heart rate (beats per minute)	89.0 (77.0-101.0)	90.0 (78.0-104.0)
		Respiratory rate (breaths per minute)	20.0 (17.0-25.0)	20.0 (16.0-25.0)
		Systolic blood pressure (mmHg)	120.0 (106.0-136.0)	118.0 (104.0-134.0)
		Diastolic blood pressure (mmHg)	66.0 (57.0-76.0)	63.0 (54.0-72.0)
		Mean arterial pressure (mmHg)	81.0 (72.0-93.0)	79.0 (71.0-90.0)
		Glasgow Coma Scale	14.0 (10.0-15.0)	14.0 (9.0-15.0)
Laboratory findings (Abbrevations: Coagulation as Coag. and Blood Gas as B.G.)	Hemotology	Red blood cells (millions/μL)	3.5 (3.0-4.0)	3.4 (3.0-3.8)
		White blood cells (thousands/μL)	11.0 (7.9-15.6)	11.0 (8.0-15.1)
		Platelets (thousands/μL)	193.0 (136.0-261.0)	199.0 (128.8-276.0)
		Hematocrit (%)	31.1 (27.2-35.6)	30.2 (27.0-33.6)
		Red blood cell dist. width (%)	15.2 (14.0-16.8)	14.8 (13.8-16.4)
		Mean corpuscular volume (fL)	90.4 (86.0-95.0)	89.0 (85.0-93.0)
		Mean corpuscular hemoglobin/ MCH (pg)	29.7 (27.9-31.2)	30.2 (28.7-31.6)
		MCH concentration (g/dL)	32.7 (31.7-33.6)	33.8 (32.8-34.8)
		Neutrophils (%)	82.0 (73.3-89.0)	82.3 (73.8-88.5)
		Lymphocytes (%)	8.4 (5.0-14.0)	9.5 (5.8-15.7)
		Monocytes (%)	6.0 (3.7-8.6)	4.0 (2.7-5.9)
		Eosinophils (%)	0.1 (0.0-1.0)	0.4 (0.0-1.2)
		Basophils (%)	0.0 (0.0-0.3)	0.1 (0.0-0.3)
		Band cells (%)	8.0 (3.0-17.0)	0.0 (0.0-5.0)
	Chemistry	Sodium (mmol/L)	139.0 (136.0-142.0)	139.0 (136.0-142.0)
		Potassium (mmol/L)	3.9 (3.6-4.3)	3.9 (3.6-4.3)
		Chloride (mmol/L)	105.0 (101.0-109.0)	105.0 (101.0-109.0)
		Bicarbonate (mmol/L)	25.0 (22.0-28.0)	26.0 (23.0-29.0)
		Blood urea nitrogen (mg/dL)	19.0 (12.0-33.0)	17.0 (11.0-28.0)
		Creatinine (mg/dL)	0.8 (0.6-1.4)	0.8 (0.6-1.3)
		Glucose (mg/dL)	131.0 (105.0-165.0)	124.0 (104.5-151.5)
		Aspartate aminotransferase (units/L)	30.0 (19.0-57.0)	37.0 (22.0-70.0)
		Alanine aminotransferase (units/L)	27.0 (16.0-47.0)	28.0 (18.0-52.0)
		Alkaline phosphatase (units/L)	84.0 (62.0-117.0)	85.0 (62.0-121.0)
		Direct bilirubin (mg/L)	0.2 (0.1-0.5)	0.6 (0.2-2.2)
		Total bilirubin (mg/L)	0.5 (0.3-0.8)	0.6 (0.4-1.1)
		Total protein (g/dL)	6.0 (5.3-6.7)	6.1 (5.3-7.0)
		Calcium (mg/dL)	8.2 (7.7-8.6)	8.2 (7.8-8.6)
		Albumin (g/dL)	2.6 (2.2-3.1)	3.0 (2.6-3.5)
		Troponin (ng/mL)	0.1 (0.0-0.2)	0.0 (0.0-0.3)
	Coag.	Prothrombin time (sec)	14.5 (12.7-16.7)	13.9 (13.0-15.3)
		Partial thromboplastin time (sec)	33.0 (28.5-41.0)	30.2 (26.6-36.9)
	B.G.	pH	7.39 (7.33-7.43)	7.41 (7.36-7.45)
		Partial pressure of oxygen (mmHg)	83.0 (68.0-111.0)	97.0 (73.5-127.5)
		Arterial oxygen saturation (mmHg)	96.0 (94.0-99.0)	97.0 (95.0-98.0)

Appendix B

Unpacking the Case Fatality Rate

B.1 Aggregate plots

The Florida data from FDOH (Figure B.1a) is more complete than subset of the United States CDC data from Florida (Figure B.1b). The latter has fewer counts of cases, hospitalization, and deaths, spikes in cases on a few days, and almost no deaths after October.



(a) Florida FDOH data. The x axis is the date of positive test confirmation.

(b) Subset of United States CDC data in which state identifier is Florida. The positive specimen date for Florida rows is 99.99% missing, so we must compare using the CDC report date.

Figure B.1: Aggregate cases, (eventual) hospitalizations, and (eventual) deaths.

In both the FDOH and the CDC datasets, one can discern three waves of COVID-19 cases. The first wave peaks around mid-April, the second wave peaks around mid-July, and the surge of cases leading up to December indicates an ongoing third wave (Figures B.1a and B.2).

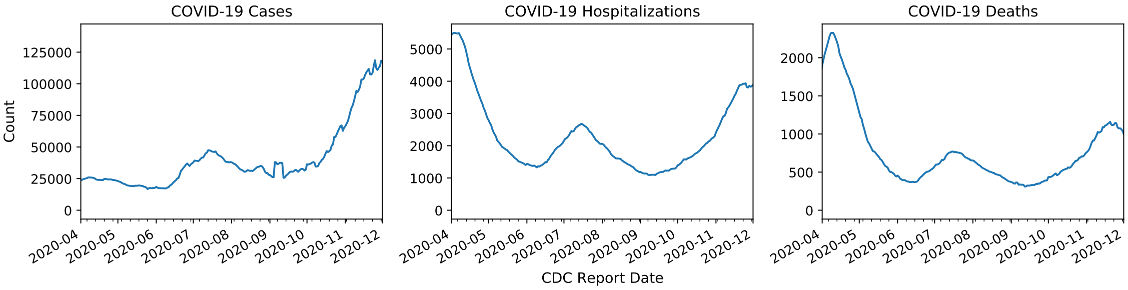
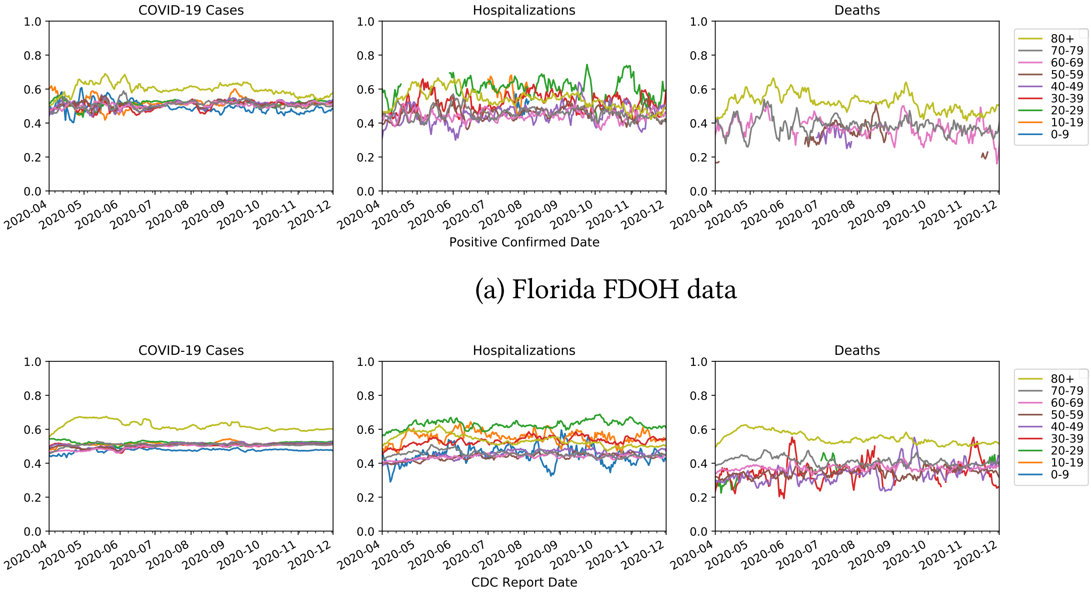


Figure B.2: Aggregate cases, (eventual) hospitalizations, and (eventual) deaths in country by the date of report to the CDC (U.S.).

B.2 Additional age-stratified plots

The gender ratios in each age group's cases, hospitalizations, and deaths appear relatively flat over time (Figure B.3).



(a) Florida FDOH data

(b) United States CDC data

Figure B.3: Female fraction of Florida and national cases, (eventual) hospitalizations, and (eventual) deaths vs. date of first positive test result (Florida) and date of report to the CDC (U.S.). Dates with fewer than five cases, hospitalizations, or deaths in the denominator are excluded.

B.3 State-wise plots

The daily cases based on CDC report date have spikes at certain days for several states (e.g. FL, GA, etc.), which might be indicative of the reporting agency submitting several cases to CDC on the same day rather than reporting daily (Figure B.4).

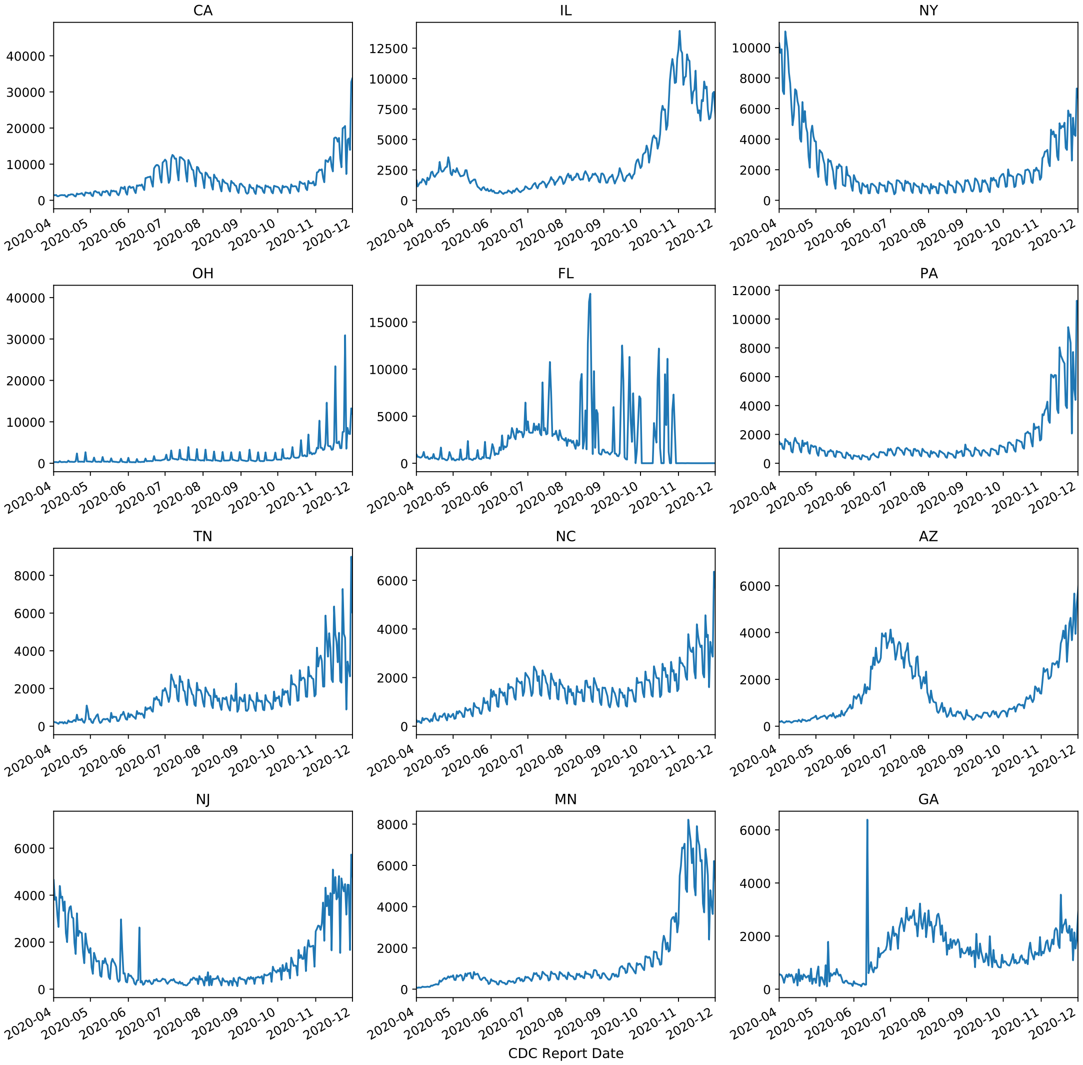


Figure B.4: Daily lab-confirmed COVID-19 cases in top twelve states with most COVID-19 cases from United States CDC data, by date of report to the CDC. Note we graph daily cases instead of the 7-day average over cases in order to demonstrate the nature of the raw data.

Between April 1st and April 15th, 34.5% of national CDC cases were recorded in New York alone (Figure B.5). From April 15th to the second peak, cases in New York had diminished (Figure B.5) and were starting to surge in other states.

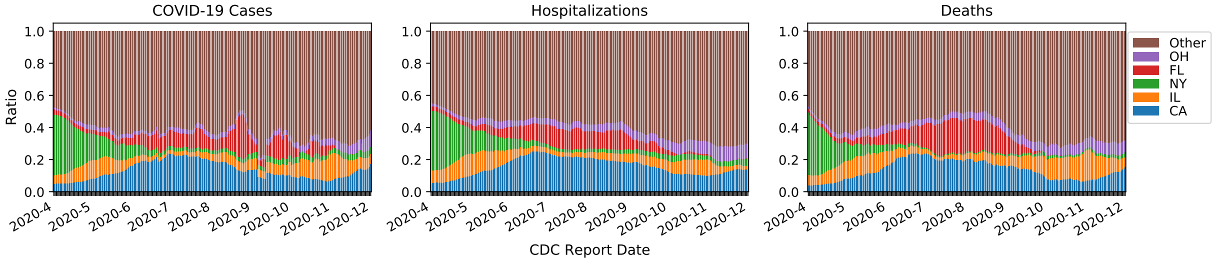


Figure B.5: State-wise distribution of cases, (eventual) hospitalizations, and (eventual) deaths for top five states CA, IL, NY, FL and OH (and the rest) with most COVID-19 cases from United States CDC data, by date of report to the CDC (U.S.). Note that in the CDC data the high missingness for data from FL and TX causes the order of the top five states here different from the order of those from the USAFacts data (CA, TX, FL, IL, NY).

B.4 Age-stratified drops in HFR by gender

Between the first two peaks, the national age-stratified HFR estimates for the female and the male populations *decreased* whereas for Florida both the female and the male HFRs slightly *increased* (Table B.1 and Table B.2). Between April 1st and December 1st, both the national and Florida estimates for the female and male populations *decreased* (Table B.3 and Table B.4). For both genders, in the age groups where we have enough support to get an estimate here (Table B.1, Table B.2, Table B.3, and Table B.4), the increased and decreased values are similar to those for the whole population in the main paper (Table 2.5 and Table 2.6).

Table B.1: Estimates of HFR and its drop between peak dates and among females. Median and 95% confidence intervals are computed using block bootstrapping.

	Florida			**National**		
Age group	2020-04-15	2020-07-15	04-15 to 07-15	2020-04-15	2020-07-15	04-15 to 07-15
aggregate	0.21 (0.19, 0.24)	0.21 (0.19, 0.22)	-0.027 (-0.16, 0.13)	0.28 (0.26, 0.29)	0.15 (0.14, 0.16)	-0.46 (-0.49, -0.42)
20-29	-	-	-	-	-	-
30-39	-	-	-	0.027 (0.022, 0.03)	0.02 (0.017, 0.022)	-0.25 (-0.39, -0.054)
40-49	-	-	-	0.061 (0.056, 0.066)	0.042 (0.038, 0.045)	-0.32 (-0.41, -0.22)
50-59	-	-	-	0.12 (0.11, 0.12)	0.08 (0.076, 0.083)	-0.32 (-0.37, -0.26)
60-69	0.17 (0.13, 0.2)	0.17 (0.15, 0.19)	0.055 (-0.17, 0.36)	0.23 (0.22, 0.23)	0.15 (0.15, 0.16)	-0.32 (-0.35, -0.28)
70-79	0.27 (0.23, 0.3)	0.28 (0.26, 0.3)	0.051 (-0.11, 0.26)	0.37 (0.35, 0.38)	0.23 (0.23, 0.24)	-0.36 (-0.39, -0.33)
80+	0.41 (0.38, 0.44)	0.44 (0.42, 0.46)	0.062 (-0.036, 0.19)	0.54 (0.52, 0.55)	0.38 (0.37, 0.39)	-0.3 (-0.33, -0.26)

Table B.2: Estimates of HFR and its drop between peak dates and among males. Median and 95% confidence intervals are computed using block bootstrapping.

	Florida			National		
Age group	2020-04-15	2020-07-15	04-15 to 07-15	2020-04-15	2020-07-15	04-15 to 07-15
aggregate	0.25 (0.23, 0.28)	0.25 (0.23, 0.27)	-0.01 (-0.13, 0.14)	0.3 (0.29, 0.32)	0.19 (0.19, 0.2)	-0.36 (-0.4, -0.33)
20-29	-	-	-	-	-	-
30-39	-	-	-	0.06 (0.054, 0.066)	0.036 (0.032, 0.039)	-0.4 (-0.48, -0.31)
40-49	-	-	-	0.092 (0.086, 0.099)	0.069 (0.065, 0.073)	-0.25 (-0.32, -0.17)
50-59	0.12 (0.099, 0.14)	0.13 (0.11, 0.14)	0.064 (-0.13, 0.3)	0.17 (0.16, 0.17)	0.12 (0.11, 0.12)	-0.3 (-0.34, -0.25)
60-69	0.2 (0.17, 0.22)	0.23 (0.21, 0.25)	0.18 (0.014, 0.44)	0.28 (0.27, 0.3)	0.21 (0.2, 0.21)	-0.28 (-0.32, -0.23)
70-79	0.35 (0.32, 0.39)	0.37 (0.35, 0.38)	0.035 (-0.071, 0.16)	0.44 (0.42, 0.45)	0.3 (0.29, 0.31)	-0.32 (-0.35, -0.28)
80+	0.52 (0.49, 0.56)	0.52 (0.5, 0.55)	0.0047 (-0.082, 0.095)	0.62 (0.6, 0.63)	0.46 (0.45, 0.47)	-0.25 (-0.28, -0.22)

Table B.3: Estimates of HFR and its drop between April 1st and December 1st and among females. Median and 95% confidence intervals are computed using block bootstrapping.

	Florida			National		
Age group	2020-04-01	2020-12-01	04-01 to 12-01	2020-04-01	2020-12-01	04-01 to 12-01
aggregate	0.21 (0.17, 0.24)	0.13 (0.097, 0.17)	-0.36 (-0.57, -0.097)	0.32 (0.3, 0.34)	0.11 (0.093, 0.12)	-0.66 (-0.72, -0.6)
20-29	-	-	-	-	-	-
30-39	-	-	-	-	-	-
40-49	-	-	-	0.069 (0.06, 0.077)	0.029 (0.021, 0.037)	-0.58 (-0.71, -0.44)
50-59	-	-	-	0.13 (0.12, 0.14)	0.05 (0.04, 0.059)	-0.62 (-0.7, -0.53)
60-69	-	-	-	0.25 (0.24, 0.26)	0.094 (0.082, 0.11)	-0.63 (-0.68, -0.56)
70-79	0.26 (0.2, 0.31)	0.15 (0.097, 0.2)	-0.43 (-0.65, -0.13)	0.4 (0.39, 0.42)	0.16 (0.14, 0.17)	-0.62 (-0.66, -0.57)
80+	0.37 (0.32, 0.42)	0.33 (0.27, 0.38)	-0.12 (-0.31, 0.11)	0.59 (0.56, 0.61)	0.24 (0.21, 0.27)	-0.59 (-0.64, -0.54)

Table B.4: Estimates of HFR and its drop between April 1st and December 1st and among males. Median and 95% confidence intervals are computed using block bootstrapping.

	Florida			National		
Age group	2020-04-01	2020-12-01	04-01 to 12-01	2020-04-01	2020-12-01	04-01 to 12-01
aggregate	0.26 (0.22, 0.3)	0.18 (0.14, 0.22)	-0.32 (-0.5, -0.074)	0.35 (0.33, 0.37)	0.15 (0.13, 0.16)	-0.58 (-0.63, -0.52)
20-29	-	-	-	-	-	-
30-39	-	-	-	0.069 (0.06, 0.078)	0.024 (0.014, 0.033)	-0.66 (-0.81, -0.5)
40-49	-	-	-	0.099 (0.089, 0.11)	0.033 (0.023, 0.043)	-0.67 (-0.77, -0.54)
50-59	-	-	-	0.18 (0.17, 0.19)	0.067 (0.056, 0.08)	-0.63 (-0.7, -0.55)
60-69	0.19 (0.15, 0.23)	0.12 (0.069, 0.16)	-0.39 (-0.67, -0.032)	0.31 (0.29, 0.33)	0.13 (0.11, 0.15)	-0.59 (-0.66, -0.51)
70-79	0.35 (0.3, 0.4)	0.22 (0.17, 0.27)	-0.37 (-0.54, -0.17)	0.48 (0.45, 0.51)	0.2 (0.17, 0.23)	-0.58 (-0.65, -0.52)
80+	0.51 (0.46, 0.57)	0.39 (0.34, 0.45)	-0.24 (-0.37, -0.065)	0.66 (0.63, 0.68)	0.32 (0.3, 0.35)	-0.51 (-0.55, -0.46)

Appendix C

Evaluation on Medical Datasets over Time

C.1 Snapshot into the State of ML4H Model Evaluation

To get a snapshot of the current standards for model evaluation in machine learning for healthcare research, we manually reviewed all of the papers from the CHIL 2022 proceedings, the first 20 papers in the CHIL 2021 proceedings, and the first 20 papers that came up in the Radiology medical journal when searching for the keyword “machine learning” and filtering for papers from 2022 to 2023 (see README.md in https://github.com/acmi-lab/EvaluationOverTime). Out of 23 papers in the CHIL 2022 proceedings, 21 did not take time into account in their data split, and two were unclear about how they split data, but it is unlikely that they split by time. Out of the 20 papers reviewed at CHIL 2021, only one paper split by time. Out of the 20 papers reviewed from Radiology, 6 did not train or evaluate any machine learning models, but out of the remaining 14 papers, 13 did not take time into account in their data split, and one did not specify how data was split.

C.2 EMDOT Python Package

Figure C.1 illustrates the workflow of the EMDOT Python package.

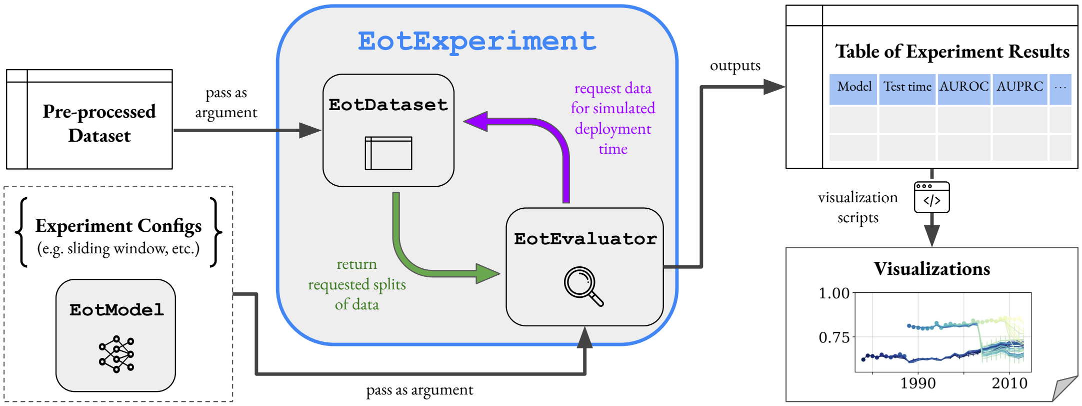


Figure C.1: EMDOT Python package workflow diagram. The primary touchpoint of the EMDOT package is the EotExperiment object. Users provide a dataframe for their (mostly) preprocessed dataset (EMDOT takes care of normalization based on the relevant training set), their desired experiment configuration (e.g. sliding window), and model class (which should subclass the simple EotModel abstract class) in order to create an EotExperiment object. Running the run_experiment() function of the EotExperiment returns a dataframe of experiment results that can then be visualized. The diagram also provides insight into some of the internals of the EotExperiment object – there is an EotDataset object that handles data splits, and an EotEvaluator object that executes the main evaluation loop.

C.3 Additional SEER Data Details

The Surveillance, Epidemiology, and End Results (SEER) Program collects cancer incidence data from registries throughout the U.S. This data has been used to study survival in several forms of cancer (Choi et al., 2008; Fuller et al., 2007; Taioli et al., 2015; Hegselmann et al., 2018). Each case includes demographics, primary tumor site, tumor morphology, stage and diagnosis, first course of treatment, and survival outcomes (collected with follow-up) (National Cancer Institute, 2020). The performance over time is evaluated on a *yearly* basis. We use the November 2020 version of the SEER database with nine registries (SEER 9), which covers about 9.4% of the U.S. population. While there are SEER databases that aggregate over more registries and hence cover a greater proportion of the U.S. population, we choose SEER 9 due to the large time range it covers (1975–2018).

- Data access: After filling out a Data Use Agreement and Best Practices Agreement, individuals can easily request access to the SEER dataset.

- Cohort selection: Using the SEER*Stat software (Program, 2015), we define three cohorts of interest: (1) breast cancer, (2) colon cancer, and (3) lung cancer. We primarily follow the cohort selection procedure from (Hegselmann et al., 2018), but we use SEER 9 instead of SEER 18, and use data from all available years instead of limiting to 2004–2009. Cohort selection diagrams are given in Figures C.2, C.3, and C.4. If there are multiple samples per patient, we filter to the first entry per patient, which corresponds to when a patient first enters the dataset. This corresponds to a particular interpretation of the prediction: when a patient is first added to a cancer registry, given what we know about that patient, what is their estimated 5-year survival probability?
- Cohort characteristics: Summaries of the SEER (Breast), SEER (Colon), and SEER (Lung) cohort characteristics are in Tables C.1, C.2, and C.3.
- Outcome definition: 5-year survival is defined by a confirmation that the patient is alive five years after the year of diagnosis.
- Features: We list the features used in the SEER breast, colon, and lung cancer datasets in Section C.3.2. For all datasets, categorical variables are converted into dummy features, and numerical variables are standard scaled (subtract mean and divide by standard deviation).
- Missingness heat maps: are given in Figures C.5, C.6, C.7, C.8, C.9, and C.10.

C.3.1 Cohort Selection and Cohort Characteristics

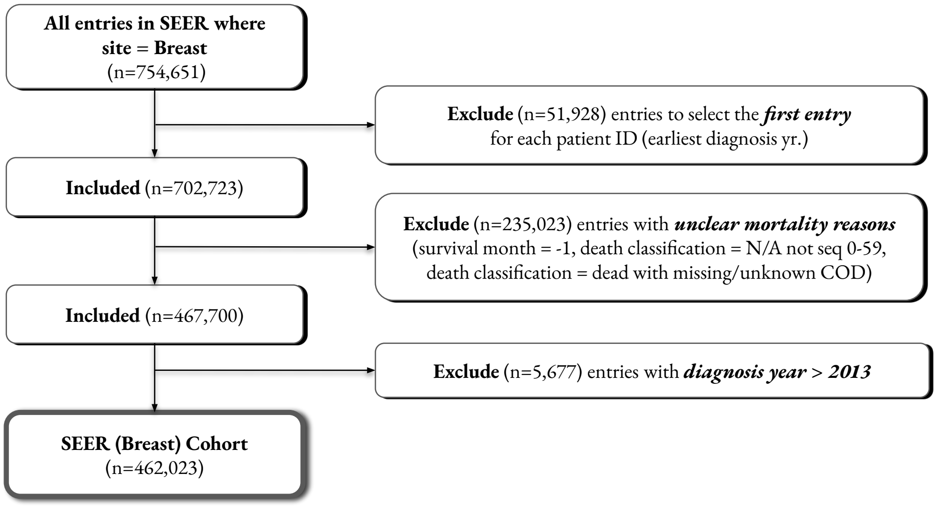

All entries in SEER where site = Breast (n=754,651)
Exclude (n=51,928) entries to select the first entry for each patient ID (earliest diagnosis yr.)
Included (n=702,723)
Exclude (n=235,023) entries with unclear mortality reasons (survival month = -1, death classification = N/A not seq 0-59, death classification = dead with missing/unknown COD)
Included (n=467,700)
Exclude (n=5,677) entries with diagnosis year > 2013
SEER (Breast) Cohort (n=462,023)

Figure C.2: Cohort selection diagram - SEER (Breast)

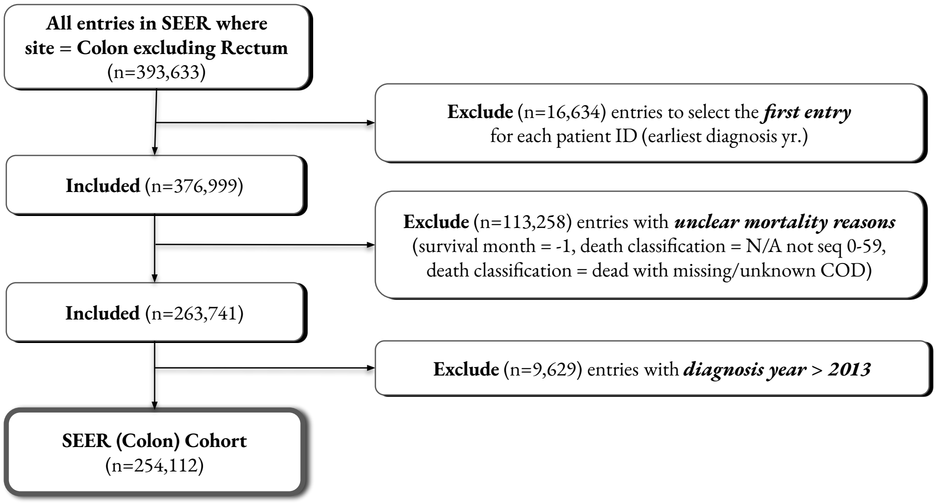

All entries in SEER where site = Colon excluding Rectum (n=393,633)
Exclude (n=16,634) entries to select the first entry for each patient ID (earliest diagnosis yr.)
Included (n=376,999)
Exclude (n=113,258) entries with unclear mortality reasons (survival month = -1, death classification = N/A not seq 0-59, death classification = dead with missing/unknown COD)
Included (n=263,741)
Exclude (n=9,629) entries with diagnosis year > 2013
SEER (Colon) Cohort (n=254,112)

Figure C.3: Cohort selection diagram - SEER (Colon)

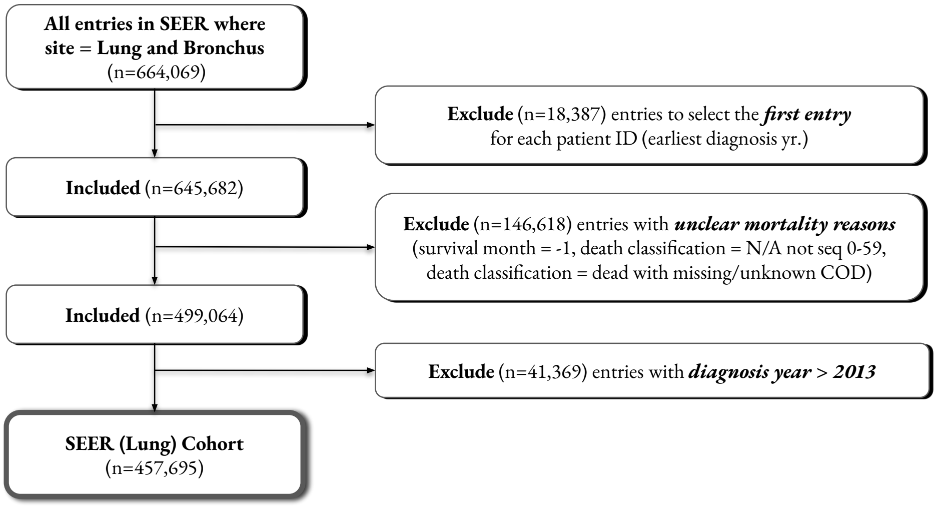

All entries in SEER where site = Lung and Bronchus (n=664,069)
Exclude (n=18,387) entries to select the first entry for each patient ID (earliest diagnosis yr.)
Included (n=645,682)
Exclude (n=146,618) entries with unclear mortality reasons (survival month = -1, death classification = N/A not seq 0-59, death classification = dead with missing/unknown COD)
Included (n=499,064)
Exclude (n=41,369) entries with diagnosis year > 2013
SEER (Lung) Cohort (n=457,695)

Figure C.4: Cohort selection diagram - SEER (Lung)

Table C.1: SEER (Breast) cohort characteristics, with count (%) or median (Q1 – Q3).

Characteristic		Missingness	Type
Sex			
Female	459,184 (99.4%)	–	categorical
Male	2,839 (0.6%)	–	categorical
Age recode with single ages and 85+	60 (50-71)	0.0%	continuous
Race/ethnicity			
White	387,247 (83.8%)	–	categorical
Black	40,217 (8.7%)	–	categorical
Other	34,559 (7.5%)	–	categorical
Laterality			
Right - origin of primary	224,777 (48.7%)	–	categorical
Left - origin of primary	233,549 (50.5%)	–	categorical
Other	3,697 (0.8%)	–	categorical
Regional nodes positive (1988+)	0 (0-3)	21.0%	continuous
T value - based on AJCC 3rd (1988-2003)	10 (10-20)	56.2%	categorical
Derived AJCC T, 7th ed (2010-2015)	13 (13-20)	85.3%	categorical
CS site-specific factor 3 (2004-2017 varying by schema)	0 (0-2)	64.8%	categorical
Regional nodes examined (1988+)	8 (2-15)	21.0%	continuous
Coding system-EOD (1973-2003)			
Four-digit EOD (1983-1987)	44,066 (9.5%)	–	categorical
Ten-digit EOD (1988-2003)	202,450 (43.8%)	–	categorical
Thirteen-digit (expanded) site specific EOD (1973-1982)	52,742 (11.4%)	–	categorical
Blank(s)	162,765 (35.2%)	–	categorical
CS version input original (2004-2015)	10,401 (10,300-20,302)	64.8%	categorical
CS version input current (2004-2015)	20,520 (20,510-20,540)	64.8%	categorical
EOD 10 - extent (1988-2003)	10 (10-13)	56.2%	categorical
Grade (thru 2017)			
Unknown	130,713 (28.3%)	–	categorical
Moderately differentiated; Grade II	135,970 (29.4%)	–	categorical
Poorly differentiated; Grade III	119,900 (26.0%)	–	categorical
Undifferentiated; anaplastic; Grade IV	8,081 (1.7%)	–	categorical
Well differentiated; Grade I	67,359 (14.6%)	–	categorical
SEER historic stage A (1973-2015)			
Regional	136,207 (29.5%)	–	categorical
Localized	286,927 (62.1%)	–	categorical
Unstaged	9,242 (2.0%)	–	categorical
Distant	29,647 (6.4%)	–	categorical
IHS Link			
Record sent for linkage, no IHS match	409,058 (88.5%)	–	categorical
Record sent for linkage, IHS match	1,505 (0.3%)	–	categorical
Blank(s)	51,460 (11.1%)	–	categorical
Histologic Type ICD-O-3	8,500 (8,500-8,500)	0.0%	categorical
EOD 10 - size (1988-2003)	18 (10-30)	56.2%	categorical
Type of Reporting Source			
Hospital inpatient/outpatient or clinic	450,801 (97.6%)	–	categorical
Other	11,222 (2.4%)	–	categorical
SEER cause-specific death classification			
Alive or dead of other cause	378,758 (82.0%)	–	categorical
Dead (attributable to this cancer dx)	83,265 (18.0%)	–	categorical
Survival months	135 (74-220)	0.0%	categorical
5-year survival			
1	378,758 (82.0%)	–	categorical
0	83,265 (18.0%)	–	categorical

Table C.2: SEER (Colon) cohort characteristics, with count (%) or median (Q1–Q3).

Characteristic		Missingness	Type
Sex			
Female	133,661 (52.6%)	–	categorical
Male	120,451 (47.4%)	–	categorical
Age recode with single ages and 85+	70 (61-79)	0.0%	continuous
Race recode (White, Black, Other)			
White	212,265 (83.5%)	–	categorical
Black	24,041 (9.5%)	–	categorical
Other	17,806 (7.0%)	–	categorical
CS version input current (2004-2015)	20,510 (20,510-20,540)	72.8%	categorical
Derived AJCC T, 6th ed (2004-2015)	30 (20-40)	73.3%	categorical
Histology ICD-O-2	8,140 (8,140-8,210)	0.0%	categorical
IHS Link			
Record sent for linkage, no IHS match	208,802 (82.2%)	–	categorical
Record sent for linkage, IHS match	744 (0.3%)	–	categorical
Blank(s)	44,566 (17.5%)	–	categorical
Histology recode - broad groupings			
8140-8389: adenomas and adenocarcinomas	213,193 (83.9%)	–	categorical
8440-8499: cystic, mucinous and serous neoplasms	28,257 (11.1%)	–	categorical
8010-8049: epithelial neoplasms, NOS	8,797 (3.5%)	–	categorical
Other	3,865 (1.5%)	–	categorical
Regional nodes positive (1988+)	1 (0-10)	29.8%	continuous
CS mets at dx (2004-2015)	0 (0-22)	72.8%	continuous
Reason no cancer-directed surgery			
Surgery performed	223,929 (88.1%)	–	categorical
Not recommended	13,003 (5.1%)	–	categorical
Other	17,180 (6.8%)	–	categorical
Derived AJCC T, 6th ed (2004-2015)	30 (20-40)	73.3%	categorical
CS version input original (2004-2015)	10,401 (10,300-20,302)	72.8%	categorical
Primary Site	184 (182-187)	0.0%	categorical
Diagnostic Confirmation			
Positive histology	244,616 (96.3%)	–	categorical
Radiography without microscopic confirm	4,822 (1.9%)	–	categorical
Other	4,674 (1.8%)	–	categorical
EOD 10 - extent (1988-2003)	45 (40-85)	57.0%	categorical
Histologic Type ICD-O-3	8,140 (8,140-8,210)	0.0%	categorical
EOD 10 - size (1988-2003)	55 (35-999)	57.0%	categorical
CS lymph nodes (2004-2015)	0 (0-210)	72.8%	categorical
SEER cause-specific death classification			
Dead (attributable to this cancer dx)	119,047 (46.8%)	–	categorical
Alive or dead of other cause	135,065 (53.2%)	–	categorical
Survival months	68 (12-151)	0.0%	categorical
5-year survival			
1	135,065 (53.2%)	–	categorical
0	119,047 (46.8%)	–	categorical

Table C.3: SEER (Lung) cohort characteristics, with count (%) or median (Q1 – Q3).

Characteristic		Missingness	Type
Sex			
Female	187,967 (41.1%)	–	categorical
Male	269,728 (58.9%)	–	categorical
Age recode with single ages and 85+	68 (60-76)	0.0%	continuous
Race recode (White, Black, Other)			
White	384,184 (83.9%)	–	categorical
Black	47,237 (10.3%)	–	categorical
Other	26,274 (5.7%)	–	categorical
Histologic Type ICD-O-3	8,070 (8,041-8,140)	0.0%	categorical
Laterality			
Left - origin of primary	178,661 (39.0%)	–	categorical
Right - origin of primary	245,321 (53.6%)	–	categorical
Paired site, but no information concerning laterality	23,196 (5.1%)	–	categorical
Other	10,517 (2.3%)	–	categorical
EOD 10 - nodes (1988-2003)	2 (1-9)	56.3%	categorical
EOD 4 - nodes (1983-1987)	3 (0-9)	88.4%	categorical
Type of Reporting Source			
Hospital inpatient/outpatient or clinic	445,606 (97.4%)	–	categorical
Other	12,089 (2.6%)	–	categorical
SEER historic stage A (1973-2015)			
Regional	79,409 (17.3%)	–	categorical
Distant	182,467 (39.9%)	–	categorical
Blank(s)	123,161 (26.9%)	–	categorical
Localized	50,375 (11.0%)	–	categorical
Unstaged	22,283 (4.9%)	–	categorical
CS version input current (2004-2015)	20,520 (20,510-20,540)	70.6%	categorical
CS mets at dx (2004-2015)	23 (0-40)	70.6%	continuous
CS version input original (2004-2015)	10,401 (10,300-20,302)	70.6%	categorical
CS tumor size (2004-2015)	50 (29-999)	70.6%	categorical
EOD 10 - size (1988-2003)	80 (35-999)	56.3%	categorical
CS lymph nodes (2004-2015)	200 (0-200)	70.6%	categorical
Histology recode - broad groupings			
8140-8389: adenomas and adenocarcinomas	147,127 (32.1%)	–	categorical
8010-8049: epithelial neoplasms, NOS	179,848 (39.3%)	–	categorical
8440-8499: cystic, mucinous and serous neoplasms	6,266 (1.4%)	–	categorical
Other	124,454 (27.2%)	–	categorical
EOD 10 - extent (1988-2003)	78 (40-85)	56.3%	categorical
SEER cause-specific death classification			
Alive or dead of other cause	49,997 (10.9%)	–	categorical
Dead (attributable to this cancer dx)	407,698 (89.1%)	–	categorical
Survival months	7 (2-19)	0.0%	categorical
5-year survival			
1	49,997 (10.9%)	–	categorical
0	407,698 (89.1%)	–	categorical

C.3.2 Features

SEER (Breast): AJCC stage 3rd edition (1988-2003), AYA site recode/WHO 2008, Age recode with single ages and 85+, Behavior code ICD-O-2, Behavior code ICD-O-3, Behavior recode for analysis, Breast - Adjusted AJCC 6th M (1988-2015), Breast - Adjusted AJCC 6th N (1988-2015), Breast - Adjusted AJCC 6th Stage (1988-2015), Breast - Adjusted AJCC 6th T (1988-2015), Breast Subtype (2010+), CS Schema - AJCC 6th Edition, CS extension (2004-2015), CS lymph nodes (2004-2015), CS mets at dx (2004-2015), CS site-specific factor 1 (2004-2017 varying by schema), CS site-specific factor 15 (2004-2017 varying by schema), CS site-specific factor 2 (2004-2017 varying by schema), CS site-specific factor 25 (2004-2017 varying by schema), CS site-specific factor 3 (2004-2017 varying by schema), CS site-specific factor 4 (2004-2017 varying by schema), CS site-specific factor 5 (2004-2017 varying by schema), CS site-specific factor 6 (2004-2017 varying by schema), CS site-specific factor 7 (2004-2017 varying by schema), CS tumor size (2004-2015), CS version derived (2004-2015), CS version input current (2004-2015), CS version input original (2004-2015), Coding system-EOD (1973-2003), Derived AJCC M, 6th ed (2004-2015), Derived AJCC M, 7th ed (2010-2015), Derived AJCC N, 6th ed (2004-2015), Derived AJCC N, 7th ed (2010-2015), Derived AJCC Stage Group, 6th ed (2004-2015), Derived AJCC Stage Group, 7th ed (2010-2015), Derived AJCC T, 6th ed (2004-2015), Derived AJCC T, 7th ed (2010-2015), Derived HER2 Recode (2010+), EOD 10 - extent (1988-2003), EOD 10 - nodes (1988-2003), EOD 10 - size (1988-2003), ER Status Recode Breast Cancer (1990+), First malignant primary indicator, Grade (thru 2017), Histologic Type ICD-O-3, Histology recode - Brain groupings, Histology recode - broad groupings, ICCC site rec extended ICD-O-3/WHO 2008, IHS Link, Laterality, Lymphoma subtype recode/WHO 2008 (thru 2017), M value - based on AJCC 3rd (1988-2003), N value - based on AJCC 3rd (1988-2003), Origin recode NHIA (Hispanic, Non-Hisp), PR Status Recode Breast Cancer (1990+), Primary Site, Primary by international rules, Race recode (W, B, AI, API), Race recode (White, Black, Other), Race/ethnicity, Regional nodes examined (1988+), Regional nodes positive (1988+), SEER historic stage A (1973-2015), SEER modified AJCC stage 3rd (1988-2003), Sex, Site recode ICD-O-3/WHO 2008, T value - based on AJCC 3rd (1988-2003), Tumor marker 1 (1990-2003), Tumor marker 2 (1990-2003), Tumor marker 3 (1998-2003), Type of Reporting Source

SEER (Colon): Age recode with less than 1 year olds, Age recode with single ages and 85+, Behavior code ICD-O-2, Behavior code ICD-O-3, CS extension (2004-2015), CS lymph nodes (2004-2015), CS mets at dx (2004-2015), CS site-specific factor 1 (2004-2017 varying by schema), CS tumor size (2004-2015), CS version input current (2004-2015), CS version input original (2004-2015), Derived AJCC M, 6th ed (2004-2015), Derived AJCC M, 7th ed (2010-2015), Derived AJCC N, 6th ed (2004-2015), Derived AJCC N, 7th ed (2010-2015), Derived AJCC Stage Group, 6th ed (2004-2015), Derived AJCC Stage Group, 7th ed (2010-2015), Derived AJCC T, 6th ed (2004-2015), Derived AJCC T, 7th ed (2010-2015), Diagnostic Confirmation, EOD 10 - extent (1988-2003), EOD 10 - nodes (1988-2003), EOD 10 - size (1988-2003), Histologic Type ICD-O-3, Histology ICD-O-2, Histology recode - broad groupings, IHS Link, Origin recode NHIA (Hispanic, Non-Hisp), Primary Site, Primary by international rules, RX Summ–Surg Prim Site (1998+), Race recode (White, Black, Other), Reason no cancer-directed surgery, Regional nodes positive (1988+), SEER modified AJCC stage 3rd (1988-2003), Sex

SEER (Lung): AYA site recode/WHO 2008, Age recode with less than 1 year olds, Age recode with single ages and 85+, Behavior code ICD-O-2, Behavior code ICD-O-3, CS extension (2004-2015), CS lymph nodes (2004-2015), CS mets at dx (2004-2015), CS site-specific factor 1 (2004-2017 varying by schema), CS tumor size (2004-2015), CS version input current (2004-2015), CS version input original (2004-2015), Derived AJCC M, 6th ed (2004-2015), Derived AJCC M, 7th ed (2010-2015), Derived AJCC N, 6th ed (2004-2015), Derived AJCC N, 7th ed (2010-2015), Derived AJCC Stage Group, 6th ed (2004-2015), Derived AJCC T, 6th ed (2004-2015), Derived AJCC T, 7th ed (2010-2015), EOD 10 - extent (1988-2003), EOD 10 - nodes (1988-2003), EOD 10 - size (1988-2003), EOD 4 - nodes (1983-1987), First malignant primary indicator, Grade (thru 2017), Histologic Type ICD-O-3, Histology recode - broad groupings, ICCC site recode 3rd edition/IARC 2017, ICCC site recode extended 3rd edition/IARC 2017, IHS Link, Laterality, Origin recode NHIA (Hispanic, Non-Hisp), Primary by international rules, Race recode (White, Black, Other), SEER historic stage A (1973-2015), Sex, Type of Reporting Source

C.3.3 Missingness heatmaps

Below are missingness heatmaps of categorical and numerical features in each SEER dataset over time. Darker color means larger proportion of missing data.

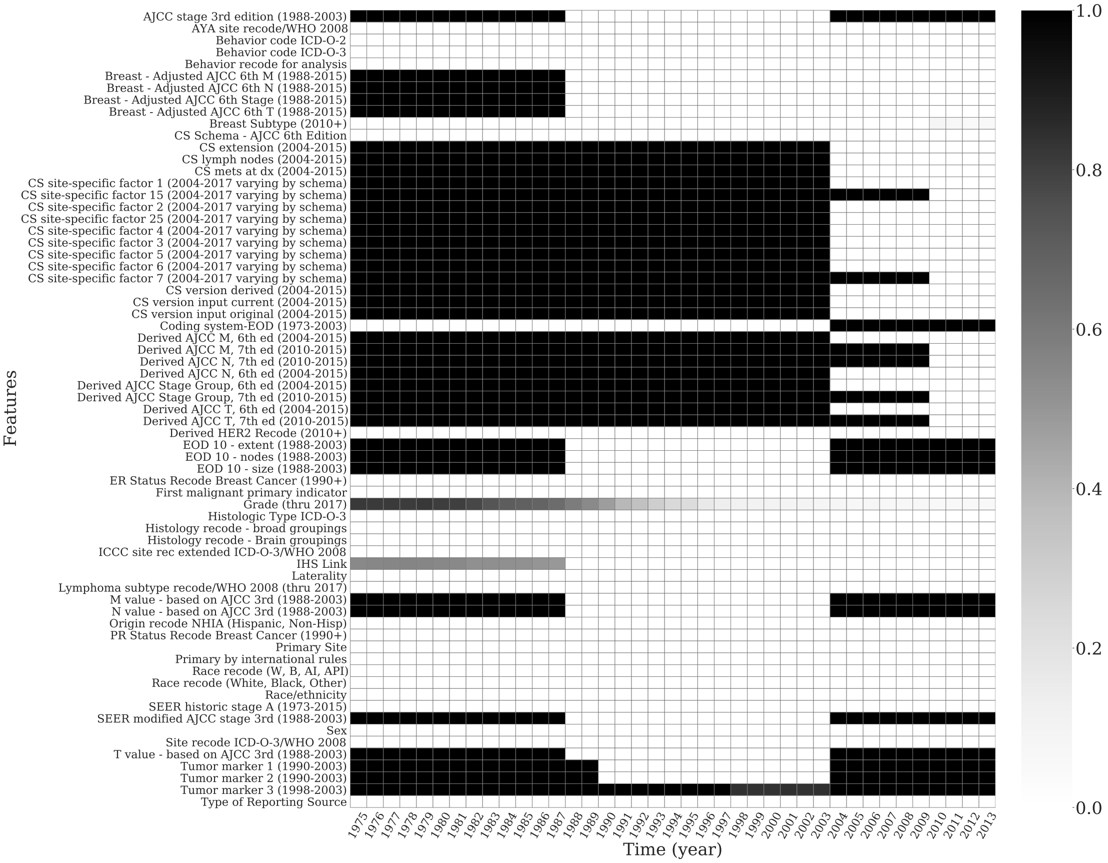


Figure C.5: Missingness of categorical features in SEER (Breast).

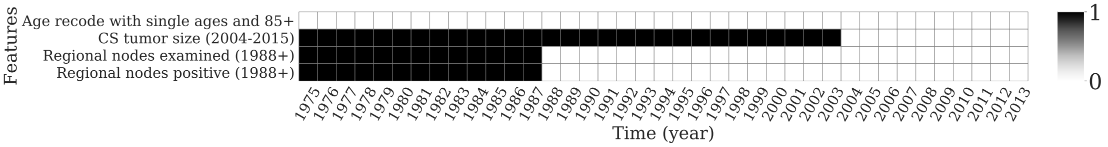


Figure C.6: Missingness of numerical features in SEER (Breast).

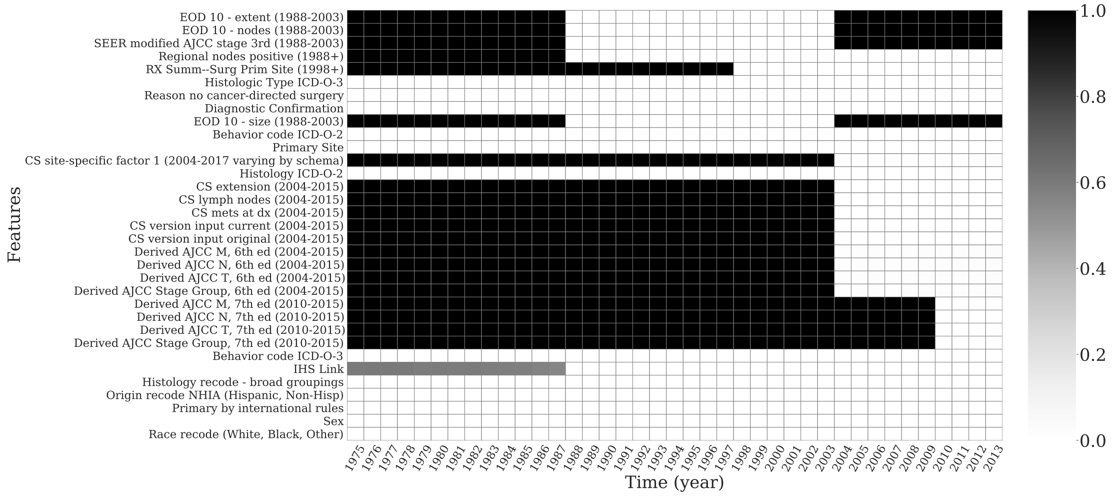


Figure C.7: Missingness of categorical features in SEER (Colon).

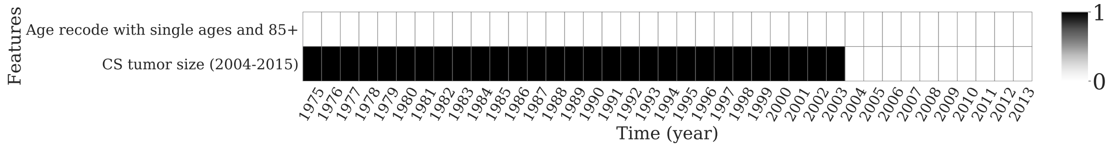


Figure C.8: Missingness of numerical features in SEER (Colon).

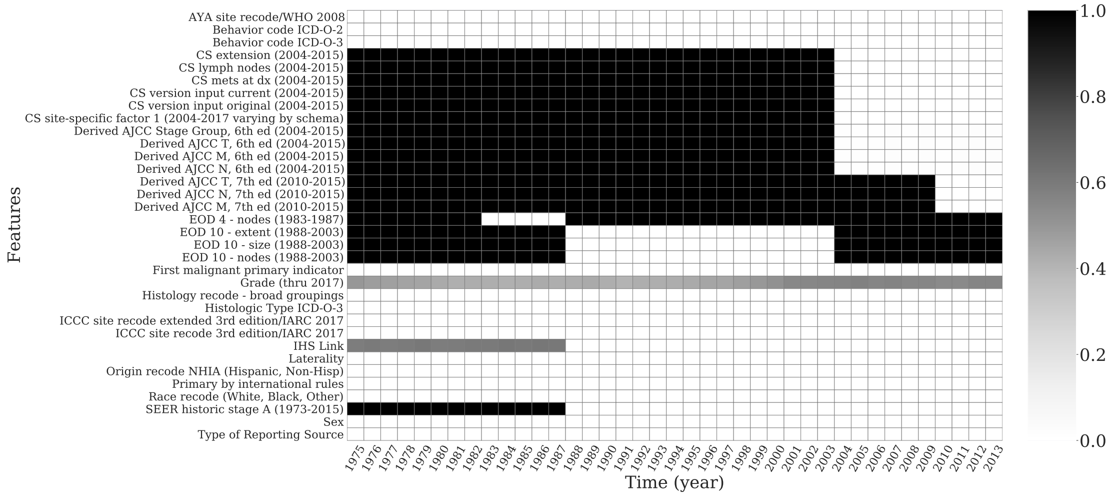


Figure C.9: Missingness of categorical features in SEER (Lung).

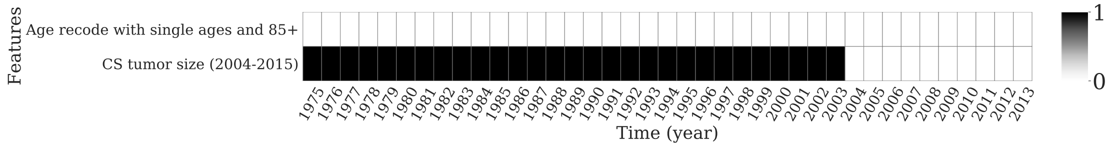


Figure C.10: Missingness of numerical features in SEER (Lung).

C.4 Additional CDC COVID-19 Data Details

The COVID-19 Case Surveillance Detailed Data (CDC, COVID-19 Response, 2020) is a national, publicly available dataset provided by the CDC. It contains 33 elements, with patient-level data including symptoms, demographics, and state of residence. The performance over time is evaluated on a *monthly* basis. We use the version the released on June 6th, 2022. Disclaimer: "The CDC does not take responsibility for the scientific validity or accuracy of methodology, results, statistical analyses, or conclusions presented."

- Data access: To access the data, users must complete a registration information and data use restrictions agreement (RIDURA).
- Cohort selection: The cohort consists of all patients who were lab-confirmed positive for COVID-19, had a non-null positive specimen date, and were hospitalized ("hosp yn" = "Yes"). Cohort selection diagrams are given in Figures C.11
- Cohort characteristics: Cohort characteristics are given in Table C.4.
- Outcome definition: mortality, defined by "death yn" = "Yes"
- Features: We list the features used in the CDC COVID-19 datasets in Section C.4.2. Categorical variables are converted into dummy features, and numerical are standard scaled.
- Missingness heat map: is given in Figure C.12.
- Figure C.13a and C.13b show how the distribution of ages and states shifts over time.

C.4.1 Cohort Selection and Cohort Characteristics

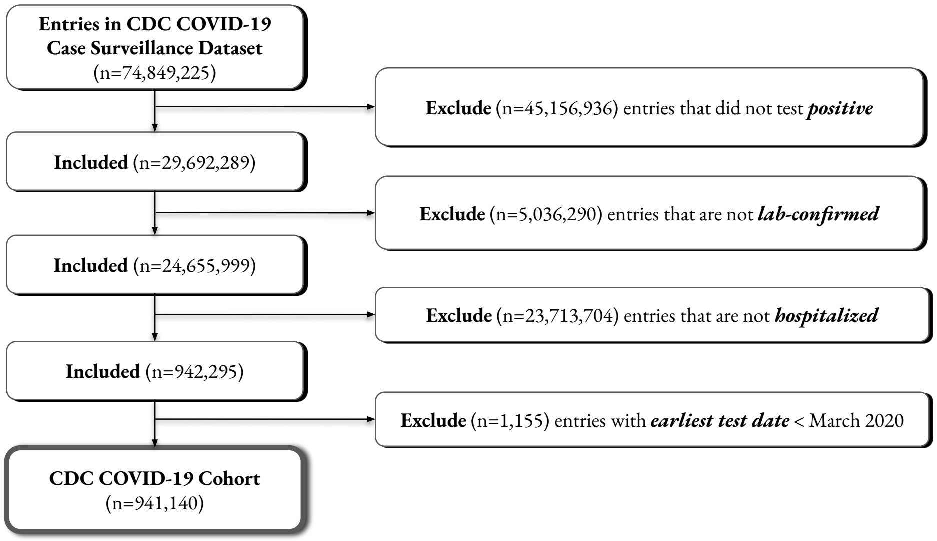

Entries in CDC COVID-19 Case Surveillance Dataset (n=74,849,225)
Exclude (n=45,156,936) entries that did not test ***positive***
Included (n=29,692,289)
Exclude (n=5,036,290) entries that are not ***lab-confirmed***
Included (n=24,655,999)
Exclude (n=23,713,704) entries that are not ***hospitalized***
Included (n=942,295)
Exclude (n=1,155) entries with ***earliest test date*** < March 2020
CDC COVID-19 Cohort (n=941,140)

Figure C.11: Cohort selection diagram - CDC COVID-19

Table C.4: CDC COVID-19 cohort characteristics, with count (%) or median (Q1–Q3).

Characteristic		Missingness	Type
Sex			
Female	455,376 (48.4%)	–	categorical
Male	475,223 (50.5%)	–	categorical
Unknown/Missing	10,541 (1.1%)	–	categorical
Age Group			
0 - 9	16,373 (1.7%)	–	categorical
10 - 19	17,252 (1.8%)	–	categorical
20 - 29	48,505 (5.2%)	–	categorical
30 - 39	71,776 (7.6%)	–	categorical
40 - 49	88,531 (9.4%)	–	categorical
50 - 59	141,805 (15.1%)	–	categorical
60 - 69	189,354 (20.1%)	–	categorical
70 - 79	189,018 (20.1%)	–	categorical
80+	177,765 (18.9%)	–	categorical
Missing	761 (0.1%)	–	categorical
Race			
White	544,199 (57.8%)	–	categorical
Black	173,847 (18.5%)	–	categorical
Other	205,547 (21.8%)	–	categorical
State of Residence			
NY	189,684 (20.2%)	–	categorical
OH	70,097 (7.4%)	–	categorical
FL	35,679 (3.8%)	–	categorical
WA	58,854 (6.3%)	–	categorical
MA	31,441 (3.3%)	–	categorical
Other	555,353 (59.0%)	–	categorical
Mechanical Ventilation			
Yes	38,009 (4.0%)	–	categorical
No	138,331 (14.7%)	–	categorical
Unknown/Missing	764,800 (81.2%)	–	categorical
Mortality			
1	190,786 (20.3%)	–	categorical
0	750,354 (79.7%)	–	categorical

C.4.2 Features

abdom yn, abxchest yn, acuterespdistress yn, age group, chills yn, cough yn, diarrhea yn, ethnicity, fever yn, hc work yn, headache yn, hosp yn, icu yn, mechvent yn, medcond yn, month, myalgia yn, nauseavomit yn, pna yn, race, relative month, res county, res state, runnose yn, sex, sfever yn, sob yn, sthroat yn

C.4.3 Missingness heatmaps

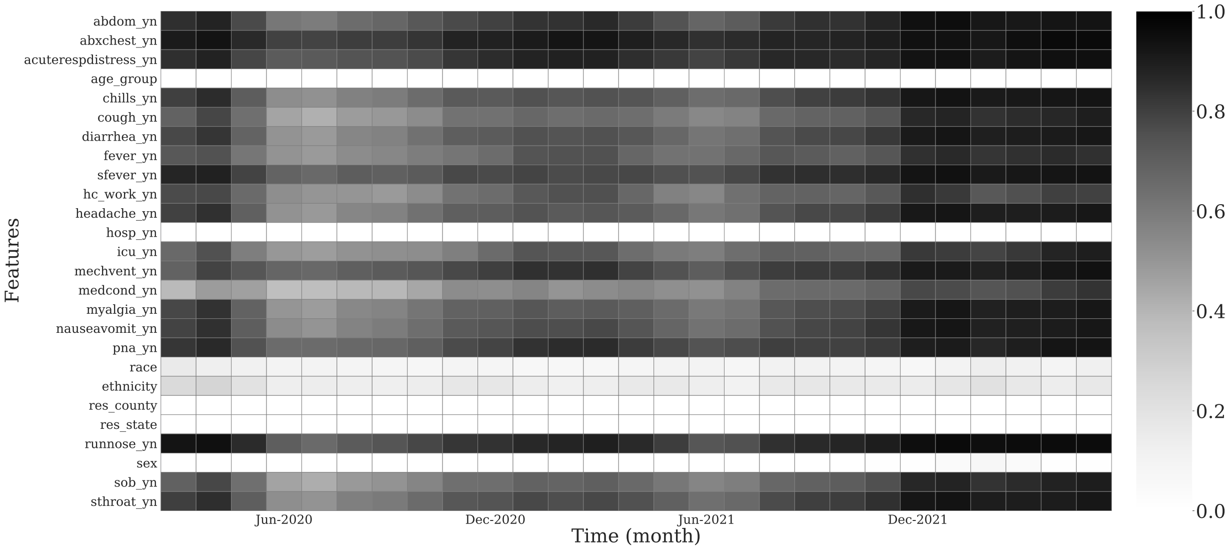
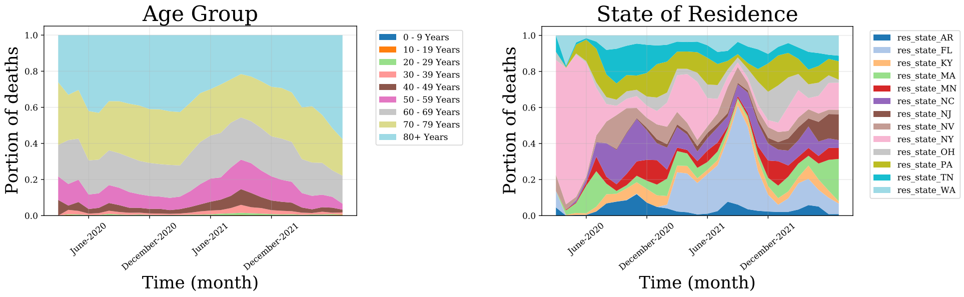


Figure C.12: Missingness over time for features in CDC COVID-19 dataset after cohort selection. The darker the color, the larger the proportion of missing data.

C.4.4 Additional Figures



(a) By age group

(b) United States CDC Data

Figure C.13: Proportion of deaths over time for each age group and state of residence.

C.5 Additional SWPA COVID-19 Data Details

The Southwestern Pennsylvania (SWPA) COVID-19 dataset consists of EHR data from patients tested for COVID-19. It was collected by a major healthcare provider in SWPA, and includes patient demographics, labs, problem histories, medications, inpatient vs. outpatient status, and other encounter information. The performance over time is evaluated on a *monthly* basis.

- Data access: This is a private dataset.
- Cohort selection: The cohort consists of patients who tested positive for COVID-19 and were not already in the ICU or mechanically ventilated. We filter for the first positive test, and define features and outcomes relative to that time. Cohort selection diagrams are given in Figures C.14. If there are multiple samples per patient, we filter to the first entry per patient, which corresponds to when a patient first enters the dataset. This corresponds to a particular interpretation of the prediction: when a patient is first tests positive, given what we know about that patient, what is their estimated risk of 90-day mortality?
- Cohort characteristics: Cohort characteristics are given in Table C.5.
- Outcome definition: 90-day mortality by comparing the death date and test date
- Features: We list the features in Section C.5.2. We convert all categorical variables into dummy features, and apply standard scaling to numerical variables (subtract mean and divide by standard deviation). To create a fixed length feature vector, where applicable we take the most recent value of each feature (e.g. most recent lab values).
- Missingness heat maps: are given in Figures C.15, C.16, C.17, and C.18,

C.5.1 Cohort Selection and Cohort Characteristics

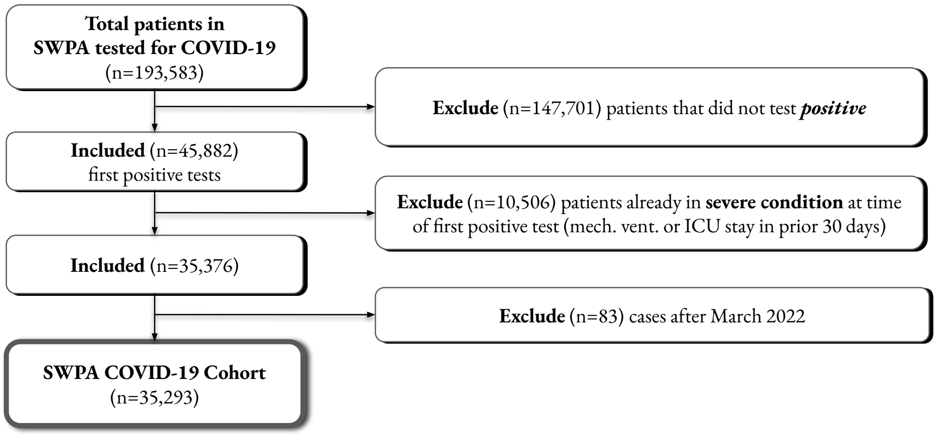

Total patients in SWPA tested for COVID-19 (n=193,583)
Exclude (n=147,701) patients that did not test ***positive***
Included (n=45,882) first positive tests
Exclude (n=10,506) patients already in **severe condition** at time of first positive test (mech. vent. or ICU stay in prior 30 days)
Included (n=35,376)
Exclude (n=83) cases after March 2022
SWPA COVID-19 Cohort (n=35,293)

Figure C.14: Cohort selection diagram for SWPA COVID-19.

Table C.5: SWPA COVID-19 cohort characteristics, with count (%) or median (Q1–Q3).

Characteristic		Missingness	Type
Gender			
Female	20,283 (57.5%)	–	categorical
Male	15,003 (42.5%)	–	categorical
Unknown	7 (0.0%)	–	categorical
Age			
Under 20	3,210 (9.1%)	–	categorical
20 – 30	4,349 (12.3%)	–	categorical
30 – 40	4,667 (13.2%)	–	categorical
40 – 50	4,653 (13.2%)	–	categorical
50 – 60	6,111 (17.3%)	–	categorical
60 – 70	5,700 (16.2%)	–	categorical
70+	6,603 (18.7%)	–	categorical
Location of test			
Inpatient	14,911 (42.2%)	–	categorical
Outpatient	17,661 (50.0%)	–	categorical
Unknown	2,721 (7.7%)	–	categorical
90-day mortality			
True	1,516 (4.3%)	–	categorical
False	33,777 (95.7%)	–	categorical

C.5.2 Features

Asthma, CAD, CHF, CKD, COPD, CRP, CVtest ICD Acute pharyngitis, unspecified, CVtest ICD Acute upper respiratory infection, unspecified, CVtest ICD Anosmia, CVtest ICD Contact with and (suspected) exposure to other viral communicable diseases, CVtest ICD Encounter for general adult medical examination without abnormal findings, CVtest ICD Encounter for screening for other viral diseases, CVtest ICD Encounter for screening for respiratory disorder NEC, CVtest ICD Nasal congestion, CVtest ICD Other general symptoms and signs, CVtest ICD Other specified symptoms and signs involving the circulatory and respiratory systems, CVtest ICD Pain, unspecified, CVtest ICD Parageusia, CVtest ICD R05.9, CVtest ICD R51.9, CVtest ICD U07.1, CVtest ICD Viral infection, unspecified, CVtest ICD Z20.822, ESLD, Hypertension, IP ICD z20.828, Immunocompromised, Interstitial Lung disease, OP ICD Abdominal Pain, OP ICD Chest Pain, OP ICD Chills, OP ICD Coronavirus Concerns, OP ICD Covid Infection, OP ICD Exposure To Covid-19, OP ICD Generalized Body Aches, OP ICD Headache, OP ICD Labs Only, OP ICD Medication Refill, OP ICD Nasal Congestion, OP ICD Nausea, OP ICD Other, OP ICD Results, OP ICD Shortness of Breath, OP ICD Sore Throat, OP ICD URI, age bin (20, 30], age bin (30, 40], age bin (40, 50], age bin (50, 60], age bin (60, 70], age bin (70, 200], bmi, cancer, cough, covid vaccination given, diabetes, fatigue, fever, gender, hyperglycemia, lab ANION GAP, lab ATRIAL RATE, lab BASOPHILS ABSOLUTE COUNT, lab BASOPHILS RELATIVE PERCENT, lab BLOOD UREA NITROGEN, lab CALCIUM, lab CALCUALTED T AXIS, lab CALCULATED R AXIS, lab CHLORIDE, lab CO2, lab CREATININE, lab EOSINOPHILS ABSOLUTE COUNT, lab EOSINOPHILS RELATIVE PERCENT, lab GFR MDRD AF AMER, lab GFR MDRD

NON AF AMER, lab GLUCOSE, lab IMMATURE GRANULOCYTES RELATIVE PERCENT, lab LYMPHOCYTES ABSOLUTE COUNT, lab LYMPHOCYTES RELATIVE PERCENT, lab MEAN CORPUSCULAR HEMOGLOBIN, lab MEAN CORPUSCULAR HEMOGLOBIN CONC, lab MEAN PLATELET VOLUME, lab MONOCYTES ABSOLUTE COUNT, lab MONOCYTES RELATIVE PERCENT, lab NEUTROPHILS RELATIVE PERCENT, lab NUCLEATED RED BLOOD CELLS, lab POTASSIUM, lab PROTEIN TOTAL, lab Q-T INTERVAL, lab QRS DURATION, lab QTC CALCULATION, lab RED CELL DISTRIBUTION WIDTH, lab SODIUM, lab VENTRICULAR RATE, lab merged CRP, lab merged albumin, lab merged alkalinePhosphatase, lab merged alt, lab merged ast, lab merged bnp, lab merged ddimer, lab merged directBilirubin, lab merged ggt, lab merged hct, lab merged hgb, lab merged indirectBilirubin, lab merged lactate, lab merged ldh, lab merged mcv, lab merged neutrophil, lab merged platelets, lab merged pt, lab merged rbc, lab merged sao2, lab merged totalBilirubin, lab merged totalProtein, lab merged troponin, lab merged wbc, labs ICD Acute pharyngitis, unspecified, labs ICD Acute upper respiratory infection, unspecified, labs ICD Chest pain, unspecified, labs ICD Contact with and (suspected) exposure to other viral communicable diseases, labs ICD Dyspnea, unspecified, labs ICD Encounter for other preprocedural examination, labs ICD Essential (primary) hypertension, labs ICD Fever, unspecified, labs ICD Heart failure, unspecified, labs ICD Other general symptoms and signs, labs ICD Other pulmonary embolism without acute cor pulmonale, labs ICD Other specified abnormalities of plasma proteins, labs ICD R05.9, labs ICD Shortness of breath, labs ICD Syncope and collapse, labs ICD U07.1, labs ICD Unspecified atrial fibrillation, labs ICD Viral infection, unspecified, labs ICD Z20.822, liver disease, location covidtest ordered Inpatient, location covidtest ordered Outpatient, lung disease, med dx Acquired hypothyroidism, med dx Anxiety, med dx COVID-19, med dx Encounter for antineoplastic chemotherapy, med dx Encounter for antineoplastic chemotherapy and immunotherapy, med dx Encounter for antineoplastic immunotherapy, med dx Encounter for immunization, med dx Gastroesophageal reflux disease without esophagitis, med dx Gastroesophageal reflux disease, esophagitis presence not specified, med dx Generalized anxiety disorder, med dx Hyperlipidemia, unspecified hyperlipidemia type, med dx Hypomagnesemia, med dx Hypothyroidism, unspecified type, med dx Iron deficiency anemia, unspecified iron deficiency anemia type, med dx Mixed hyperlipidemia, med dx Primary osteoarthritis of right knee, medication ACETAMINOPHEN 325 MG TABLET, medication ALBUTEROL SULFATE 2.5 MG/3 ML (0.083 %) SOLUTION FOR NEBULIZATION, medication ALBUTEROL SULFATE HFA 90 MCG/ACTUATION AEROSOL INHALER, medication ASPIRIN 81 MG TABLET,DELAYED RELEASE, medication DEXAMETHASONE SODIUM PHOSPHATE 4 MG/ML INJECTION SOLUTION, medication DIPHENHYDRAMINE 50 MG/ML INJECTION (WRAPPER), medication EPINEPHRINE 0.3 MG/0.3 ML INJECTION, AUTO-INJECTOR, medication FENTANYL (PF) 50 MCG/ML INJECTION SOLUTION, medication HYDROCODONE 5 MG-ACETAMINOPHEN 325 MG TABLET, medication HYDROCORTISONE SOD SUCCINATE (PF) 100 MG/2 ML SOLUTION FOR INJECTION, medication IOPAMIDOL 76 % INTRAVENOUS SOLUTION, medication LACTATED RINGERS INTRAVENOUS SOLUTION, medication MIDAZOLAM 1 MG/ML INJECTION SOLUTION, medication NALOXONE 0.4 MG/ML INJECTION SOLUTION, medication ONDANSETRON HCL (PF) 4 MG/2 ML INJECTION SOLUTION, medication OXYCODONE 5 MG TABLET, medication PANTOPRAZOLE 40 MG TABLET,DELAYED RELEASE, medication PROPOFOL 10 MG/ML INTRAVENOUS BOLUS (20 ML), medication SODIUM CHLORIDE 0.9 % INTRAVENOUS SOLUTION, medication SODIUM CHLORIDE 0.9 % IV BOLUS, myalgia, obesity, past7Dprobhx ICD Acute kidney failure, unspecified, past7Dprobhx ICD Anemia, unspecified, past7Dprobhx ICD Anxiety disorder, unspecified, past7Dprobhx ICD Chest pain, unspecified, past7Dprobhx ICD Dizziness and giddiness, past7Dprobhx ICD Encounter for general adult medical examination without abnormal findings, past7Dprobhx ICD Encounter for immunization, past7Dprobhx ICD Encounter for screening for malignant neoplasm of colon, past7Dprobhx ICD F32.A, past7Dprobhx ICD Gastro-esophageal reflux disease without esophagitis, past7Dprobhx ICD Hyperlipidemia, unspecified, past7Dprobhx ICD Hypokalemia, past7Dprobhx ICD Hypothyroidism, unspecified, past7Dprobhx ICD Mixed hyperlipidemia, past7Dprobhx ICD Obstructive sleep apnea (adult) (pediatric), past7Dprobhx ICD Syncope and collapse, past7Dprobhx ICD Type 2 diabetes mellitus without complications, past7Dprobhx ICD Unspecified atrial fibrillation, probhx ICD Acute kidney failure, unspecified, probhx ICD Anemia, unspecified, probhx ICD Anxiety disorder, unspecified, probhx ICD Chest pain, unspecified, probhx ICD Dizziness and giddiness, probhx ICD Encounter for general adult medical examination without abnormal findings, probhx ICD Encounter for immunization, probhx ICD Encounter for screening for malignant neoplasm of colon, probhx ICD F32.A, probhx ICD Gastro-esophageal reflux disease without esophagitis, probhx ICD Hyperlipidemia, unspecified, probhx ICD Hypokalemia, probhx ICD Hypothyroidism, unspecified, probhx ICD Mixed hyperlipidemia, probhx ICD Obstructive sleep apnea (adult) (pediatric), probhx ICD Syncope and collapse, probhx ICD Type 2 diabetes mellitus without complications, probhx ICD Unspecified atrial fibrillation, transplant, troponin, vaccine COVID-19 RS-AD26 (PF) Vaccine (Janssen), vaccine COVID-19 Vaccine, Unspecified, vaccine COVID-19 mRNA (PF) Vaccine (Moderna), vaccine COVID-19 mRNA (PF) Vaccine (Pfizer), vaccine Flu Whole, vaccine INFLUENZA, CCIV4, vaccine Influenza, vaccine Influenza High PF, vaccine Influenza ID PF, vaccine Influenza PF, vaccine Influenza Vaccine, Quadrivalent, Adjuvanted, vaccine Influenza, High-dose, Quadrivalent, vaccine Influenza, Quadrivalent, vaccine Influenza, Recombinant (RIV4), vaccine Influenza, Recombinant (Riv3), vaccine Influenza, Trivalent, Adjuvanted, vaccine LAIV3, vaccine Pneumococcal, vaccine Pneumococcal Conjugate 13-valent, vaccine Pneumococcal Polysaccharide, vaccine TIVA

C.5.3 Missingness heatmaps

This section plots missingness heatmaps of categorical and numerical features over time. Darker color means larger proportion of missing data.

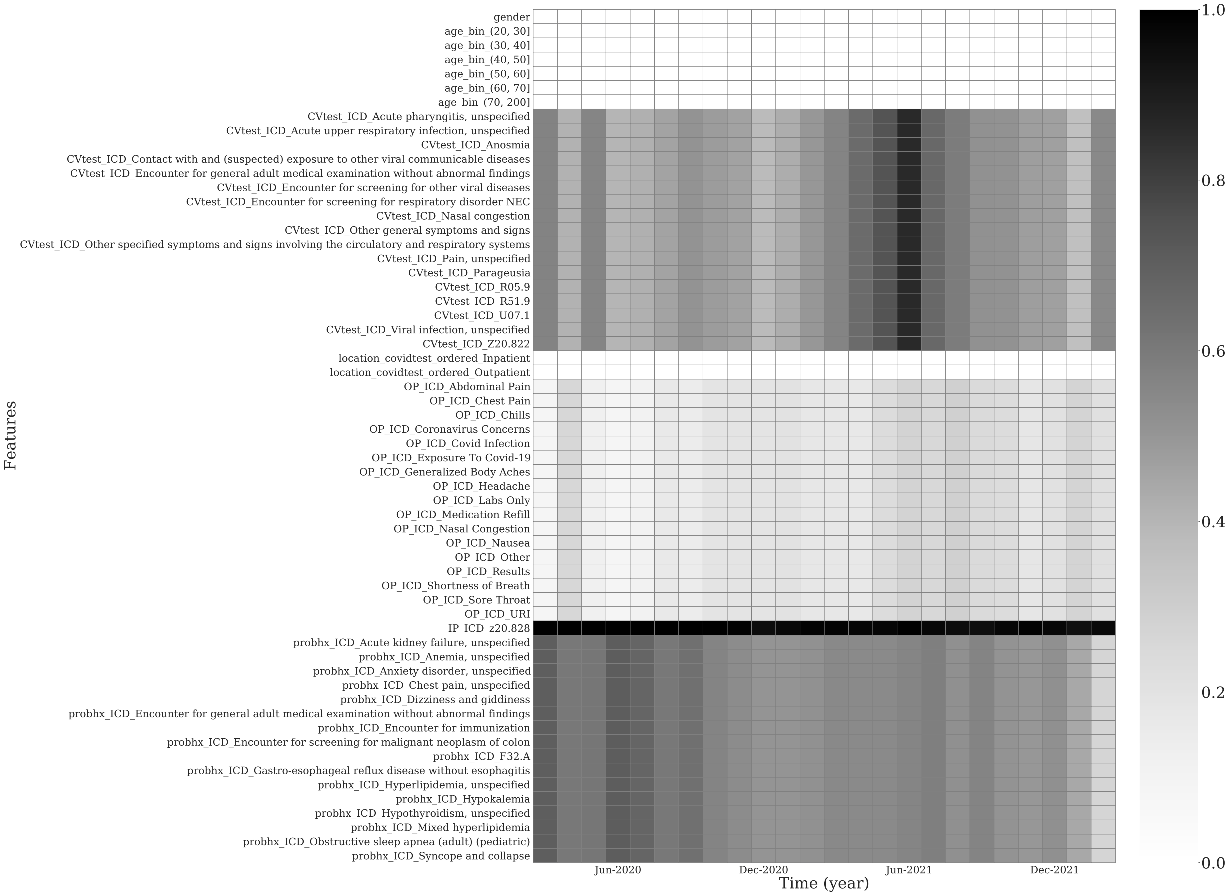


Figure C.15: Missingness of categorical features in SWPA COVID-19 dataset (part 1).

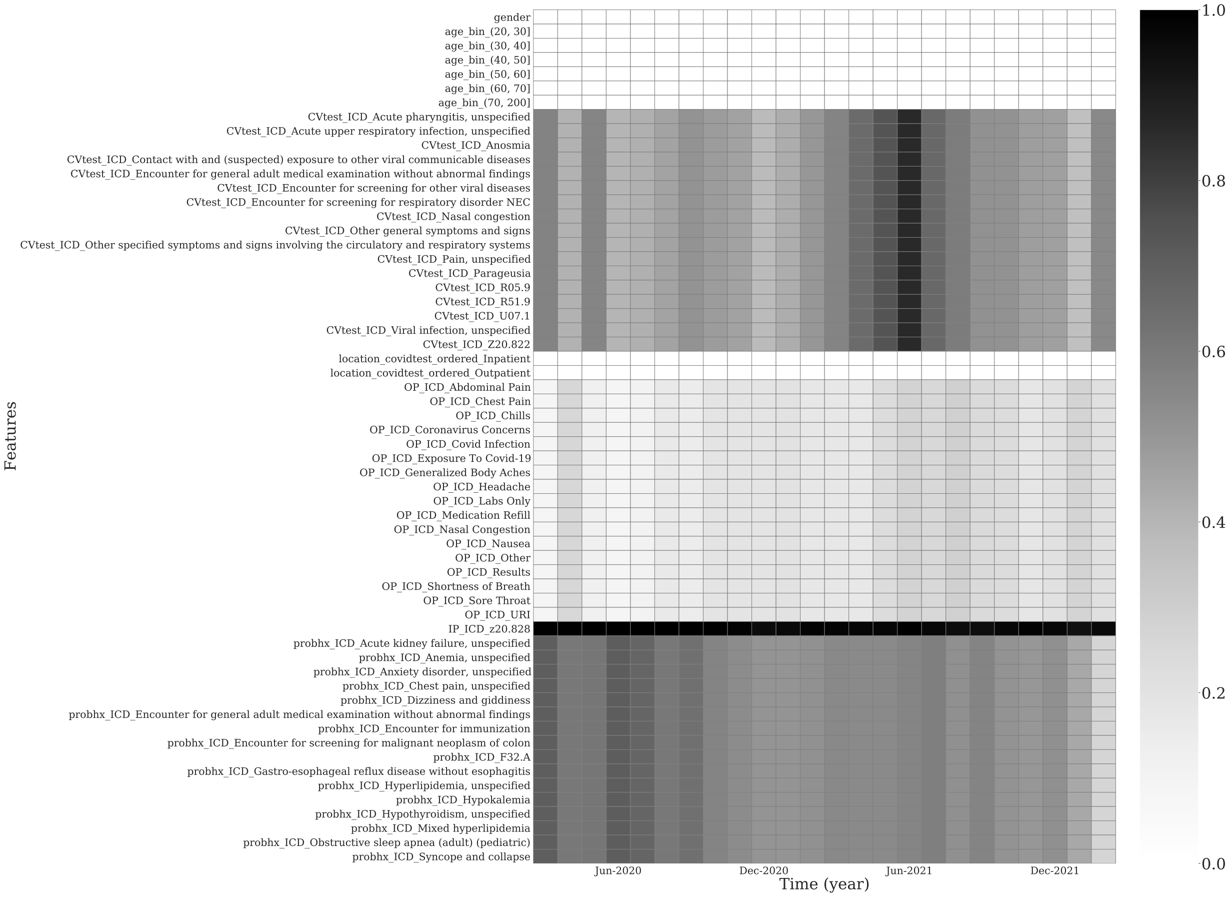


Figure C.16: Missingness of categorical features in SWPA COVID-19 dataset (part 2).

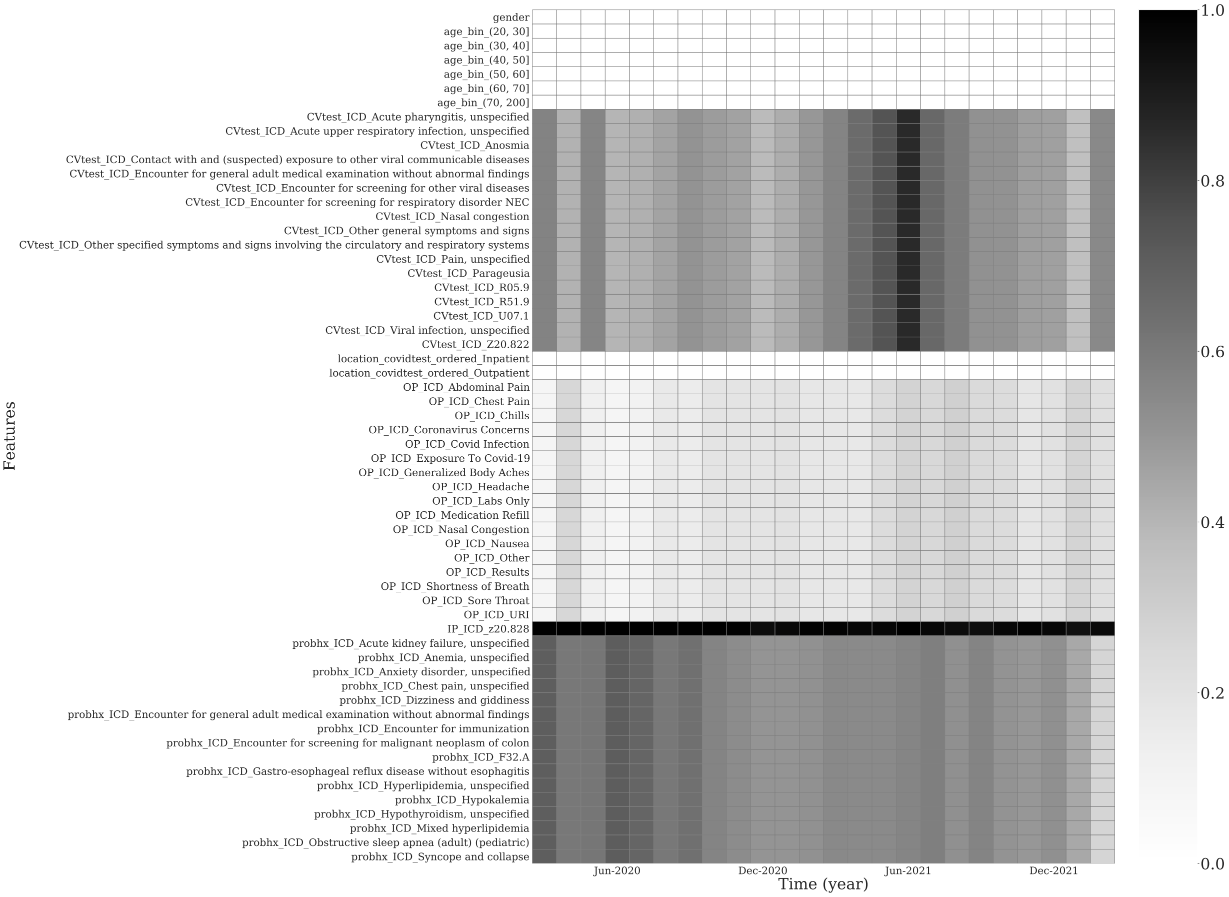


Figure C.17: Missingness of categorical features in SWPA COVID-19 dataset (part 3).

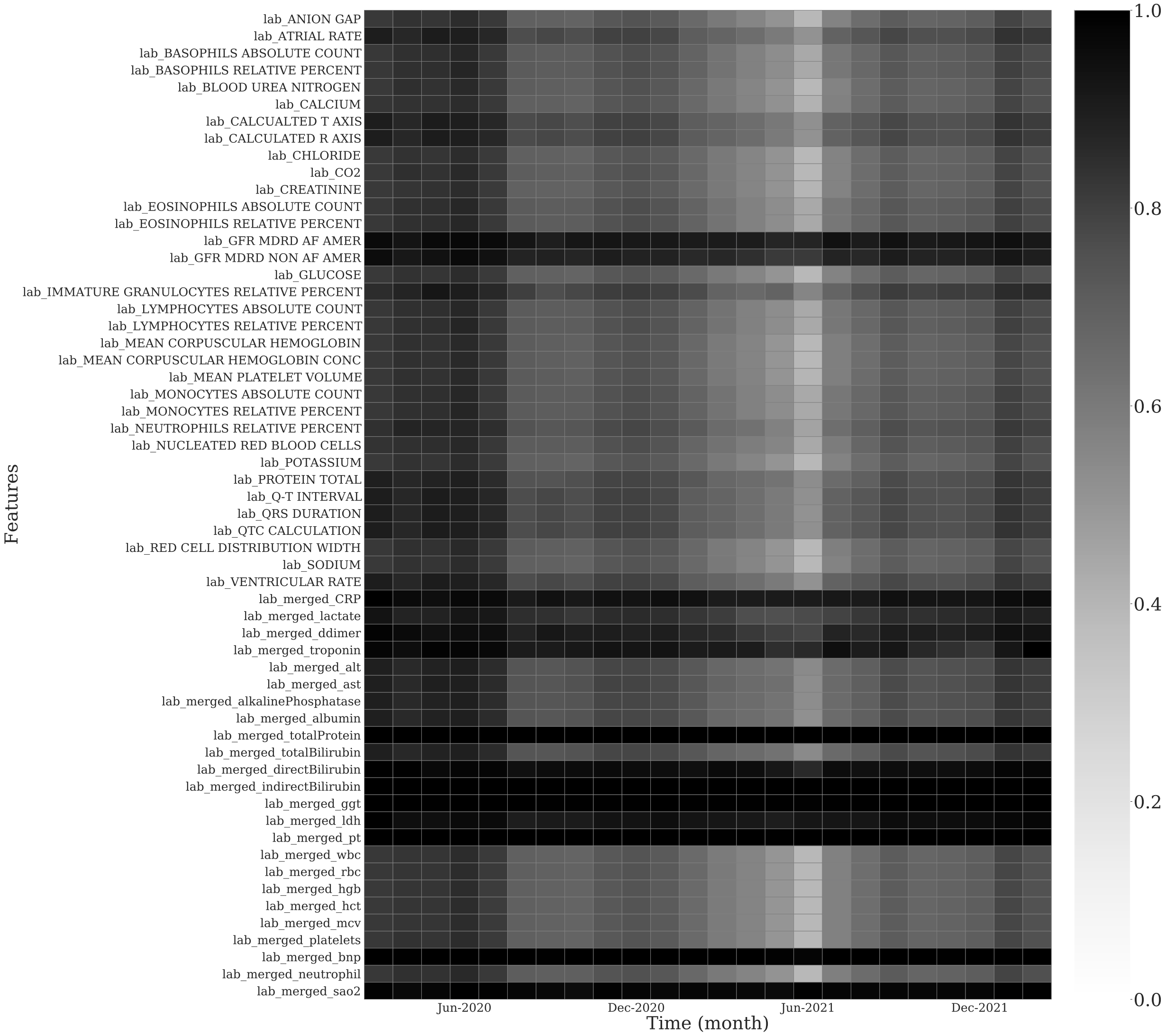


Figure C.18: Missingness of numerical features in SWPA COVID-19.

C.6 Additional MIMIC-IV Data Details

The Medical Information Mart for Intensive Care (MIMIC)-IV (Johnson et al., 2021) database contains EHR data from patients admitted to critical care units from 2008–2019. MIMIC-IV is an update to MIMIC-III, adding time annotations placing each sample into a three-year time range, and removing elements from the old CareVue EHR system (before 2008). Each patient has an `anchor_year_group`, `anchor_year` and `intime`. For each patient, we first calculated an offset as the difference between `intime` and `anchor_year`. Then, we approximated the admit time as the midpoint of `anchor_year_group` after applying the computed offset.

The performance over time is evaluated on a *yearly* basis. Our study uses MIMIC-IV-1.0.

- Data access: Credentialed Physionet account and signing a data use agreement.
- Cohort selection: From the `icustays` table, a feature vector is defined for each patient only using information available in the first 24 hrs of their first encounter. (Selection diagram in Figure C.19). If there are multiple samples per patient, we filter to the first entry per patient, which corresponds to when a patient first enters the dataset. When a patient first visits the ICU, given what we know about that patient, we would like to predict their risk of in-ICU mortality?
- Outcome definition: The outcome of interest is in-ICU mortality, defined by comparing the `outtime` of the patient's ICU visit with the patient's `dod` (date of death, in the `patients` table). Out-of-hospital mortality is not recorded.
- Cohort characteristics: Cohort characteristics are given in Table C.6.
- Features: MIMIC-IV dataset features are listed in Section C.6.2. Categorical variables are converted to dummies and numerical variables are standard scaled. To create a fixed length feature vector, we take the most recent value of any patient history data available.
- Missingness heat maps: are given in Figures C.20, C.21, C.22, C.23.

C.6.1 Cohort Selection and Cohort Characteristics

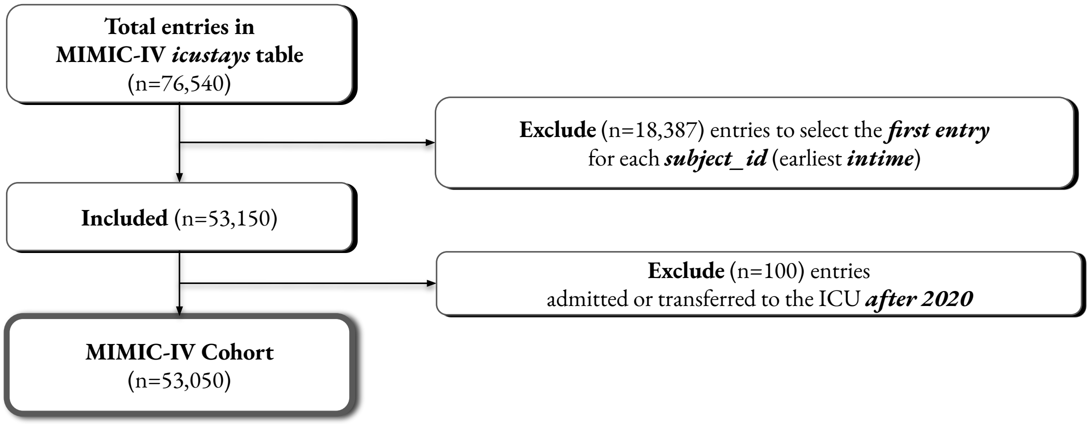

Total entries in MIMIC-IV *icustays* table (n=76,540)
Exclude (n=18,387) entries to select the ***first entry*** for each ***subject_id*** (earliest ***intime***)
Included (n=53,150)
Exclude (n=100) entries admitted or transferred to the ICU ***after 2020***
MIMIC-IV Cohort (n=53,050)

Figure C.19: Cohort selection diagram - MIMIC-IV

Table C.6: MIMIC-IV cohort characteristics, with count (%) or median (Q1–Q3).

Characteristic		Missingness	Type
Gender			
Female	23,313 (43.9%)	–	categorical
Male	29,737 (56.1%)	–	categorical
Age at Admission	66 (54-78)	0.0%	continuous
O2 Delivery Device(s)			
Use device	33,359 (62.9%)	–	categorical
None	18,549 (35.0%)	–	categorical
Missing	1,142 (2.2%)	–	categorical
Pupil Response R			
Brisk	39,708 (74.9%)	–	categorical
Sluggish	4,603 (8.7%)	–	categorical
Non-reactive	1,812 (3.4%)	–	categorical
Missing	6,927 (13.1%)	–	categorical
first careunit			
Medical Intensive Care Unit (MICU)	10,213 (19.3%)	–	categorical
Surgical Intensive Care Unit (SICU)	8,241 (15.5%)	–	categorical
Medical/Surgical Intensive Care Unit (MICU/S...	8,808 (16.6%)	–	categorical
Cardiac Vascular Intensive Care Unit (CVICU)	9,437 (17.8%)	–	categorical
Coronary Care Unit (CCU)	6,098 (11.5%)	–	categorical
Trauma SICU (TSICU)	6,947 (13.1%)	–	categorical
Other	3,306 (6.2%)	–	categorical
Anion Gap	13 (11-16)	0.5%	continuous
Heart Rhythm			
SR (Sinus Rhythm)	34,004 (64.1%)	–	categorical
Abnormal heart rhythm	18,657 (35.2%)	–	categorical
Missing	389 (0.7%)	–	categorical
Glucose FS (range 70 -100)	131 (110-164)	32.7%	continuous
Eye Opening			
Spontaneously	39,216 (73.9%)	–	categorical
To Speech	7,387 (13.9%)	–	categorical
None	4,538 (8.6%)	–	categorical
To Pain	1,702 (3.2%)	–	categorical
Missing	207 (0.4%)	–	categorical
Lactate	2 (1-2)	22.0%	continuous
Motor Response			
Obeys Commands	44,409 (83.7%)	–	categorical
Localizes Pain	3,419 (6.4%)	–	categorical
Flex-withdraws	1,673 (3.2%)	–	categorical
No response	2,930 (5.5%)	–	categorical
Abnormal extension	157 (0.3%)	–	categorical
Abnormal Flexion	238 (0.4%)	–	categorical
Missing	224 (0.4%)	–	categorical
Respiratory Pattern			
Regular	29,373 (55.4%)	–	categorical
Not regular	1,739 (3.3%)	–	categorical
Missing	21,938 (41.4%)	–	categorical
Richmond-RAS Scale	0 (-1-0)	15.4%	categorical
in-icu mortality			
0	49,716 (93.7%)	–	categorical
1	3,334 (6.3%)	–	categorical

C.6.2 Features

18 Gauge Dressing Occlusive, 18 Gauge placed in outside facility, 20 Gauge Dressing Occlusive, 20 Gauge placed in outside facility, 20 Gauge placed in the field, Abdominal Assessment, Activity, Activity Tolerance, Admission Weight (Kg), Admission Weight (lbs.), Alanine Aminotransferase (ALT), Alarms On, Albumin, Alkaline Phosphatase, All Medications Tolerated, Ambulatory aid, Anion Gap, Anion gap, Anti Embolic Device, Anti Embolic Device Status, Asparate Aminotransferase (AST), Assistance, BUN, Balance, Base Excess, Basophils, Bath, Bicarbonate, Bilirubin, Total, Bowel Sounds, Braden Activity, Braden Friction/Shear, Braden Mobility, Braden Moisture, Braden Nutrition, Braden Sensory Perception, CAM-ICU MS Change, Calcium non-ionized, Calcium, Total, Calculated Total CO2, Capillary Refill L, Capillary Refill R, Chloride, Chloride (serum), Commands, Commands Response, Cough Effort, Cough Type, Creatinine, Creatinine (serum), Currently experiencing pain, Daily Wake Up, Delirium assessment, Dialysis patient, Diet Type, Difficulty swallowing, Dorsal PedPulse L, Dorsal PedPulse R, ETOH, Ectopy Type 1, Edema Amount, Edema Location, Education Barrier, Education Existing Knowledge, Education Learner, Education Method, Education Readiness/Motivation, Education Response, Education Topic, Eosinophils, Epithelial Cells, Eye Opening, Family Communication, Flatus, GU Catheter Size, Gait/Transferring, Glucose (serum), Glucose FS (range 70 -100), Goal Richmond-RAS Scale, HCO3 (serum), HOB, HR, HR Alarm - High, HR Alarm - Low, Heart Rhythm, Height, Height (cm), Hematocrit, Hematocrit (serum), Hemoglobin, History of falling (within 3 mnths)*, History of slips / falls, Home TF, INR, INR(PT), IV/Saline lock, Insulin pump, Intravenous / IV access prior to admission, Judgement, LLE Color, LLE Temp, LLL Lung Sounds, LUE Color, LUE Temp, LUL Lung Sounds, Lactate, Lactic Acid, Living situation, Lymphocytes, MCH, MCHC, MCV, Magnesium, Mental status, Monocytes, Motor Response, NBP Alarm - High, NBP Alarm - Low, NBP Alarm Source, NBPd, NBPm, NBPs, Nares L, Nares R, Neutrophils, O2 Delivery Device(s), Oral Care, Oral Cavity, Orientation, PT, PTT, Pain Assessment Method, Pain Cause, Pain Level, Pain Level Acceptable, Pain Level Response, Pain Location, Pain Management, Pain Present, Pain Type, Parameters Checked, Phosphate, Phosphorous, Platelet Count, Position, PostTib Pulses L, PostTib Pulses R, Potassium, Potassium (serum), Potassium, Whole Blood, Pressure Reducing Device, Pressure Ulcer Present, Pupil Response L, Pupil Response R, Pupil Size Left, Pupil Size Right, RBC, RDW, RLE Color, RLE Temp, RLL Lung Sounds, RR, RUE Color, RUE Temp, RUL Lung Sounds, Radial Pulse L, Radial Pulse R, Red Blood Cells, Resp Alarm - High, Resp Alarm - Low, Respiratory Effort, Respiratory Pattern, Richmond-RAS Scale, ST Segment Monitoring On, Safety Measures, Secondary diagnosis, Self ADL, Side Rails, Skin Color, Skin Condition, Skin Integrity, Skin Temp, Sodium, Sodium (serum), SpO2, SpO2 Alarm - High, SpO2 Alarm - Low, SpO2 Desat Limit, Specific Gravity, Specimen Type, Speech, Strength L Arm, Strength L Leg, Strength R Arm, Strength R Leg, Support Systems, Temp Site, Temperature F, Therapeutic Bed, Tobacco Use History, Turn, Untoward Effect, Urea Nitrogen, Urine Source, Verbal Response, Visual / hearing deficit, WBC, White Blood Cells, Yeast, admit age, gender, pCO2, pH, pO2

C.6.3 Missingness heatmaps

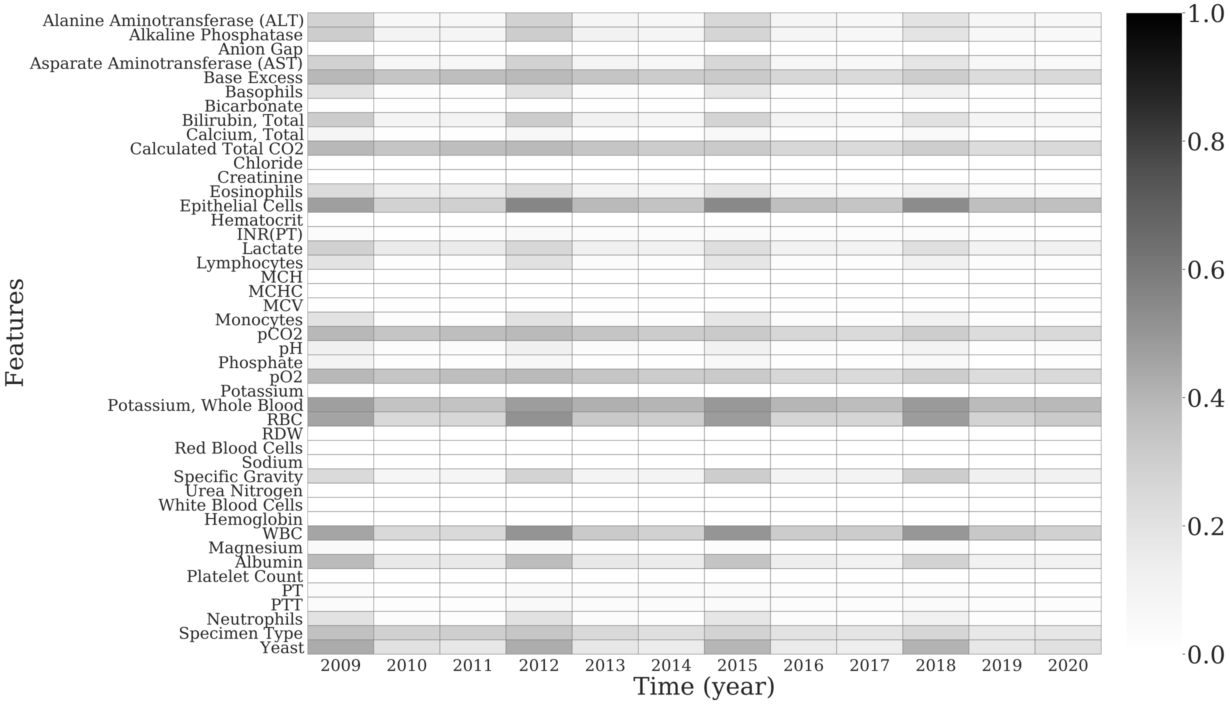


Figure C.20: Missingness over time for labevents features in MIMIC-IV dataset after cohort selection. The darker the color, the larger the proportion of missing data.

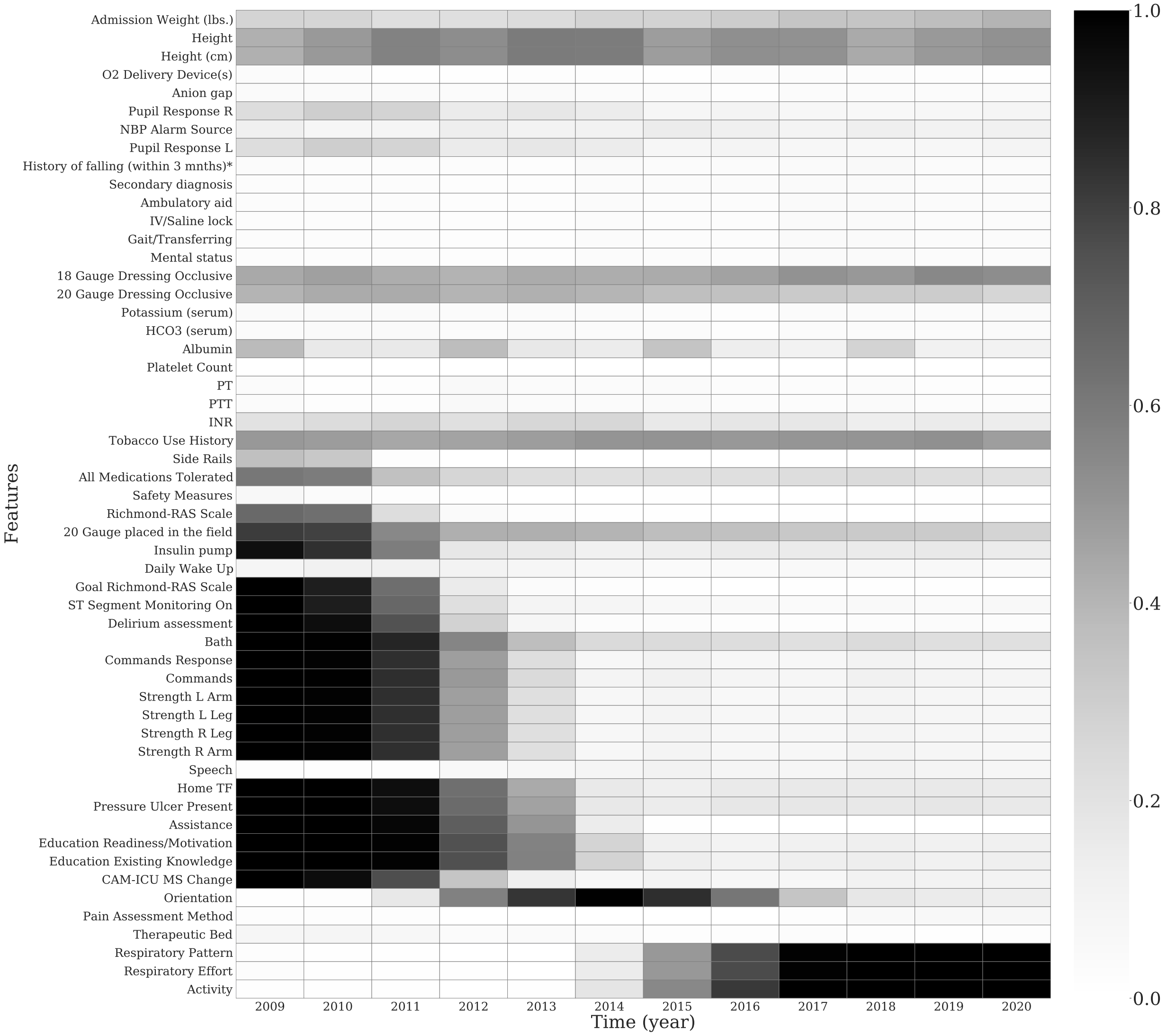


Figure C.21: Missingness over time for chartevents features in MIMIC-IV dataset after cohort selection. The darker the color, the larger the proportion of missing data. (part 1)

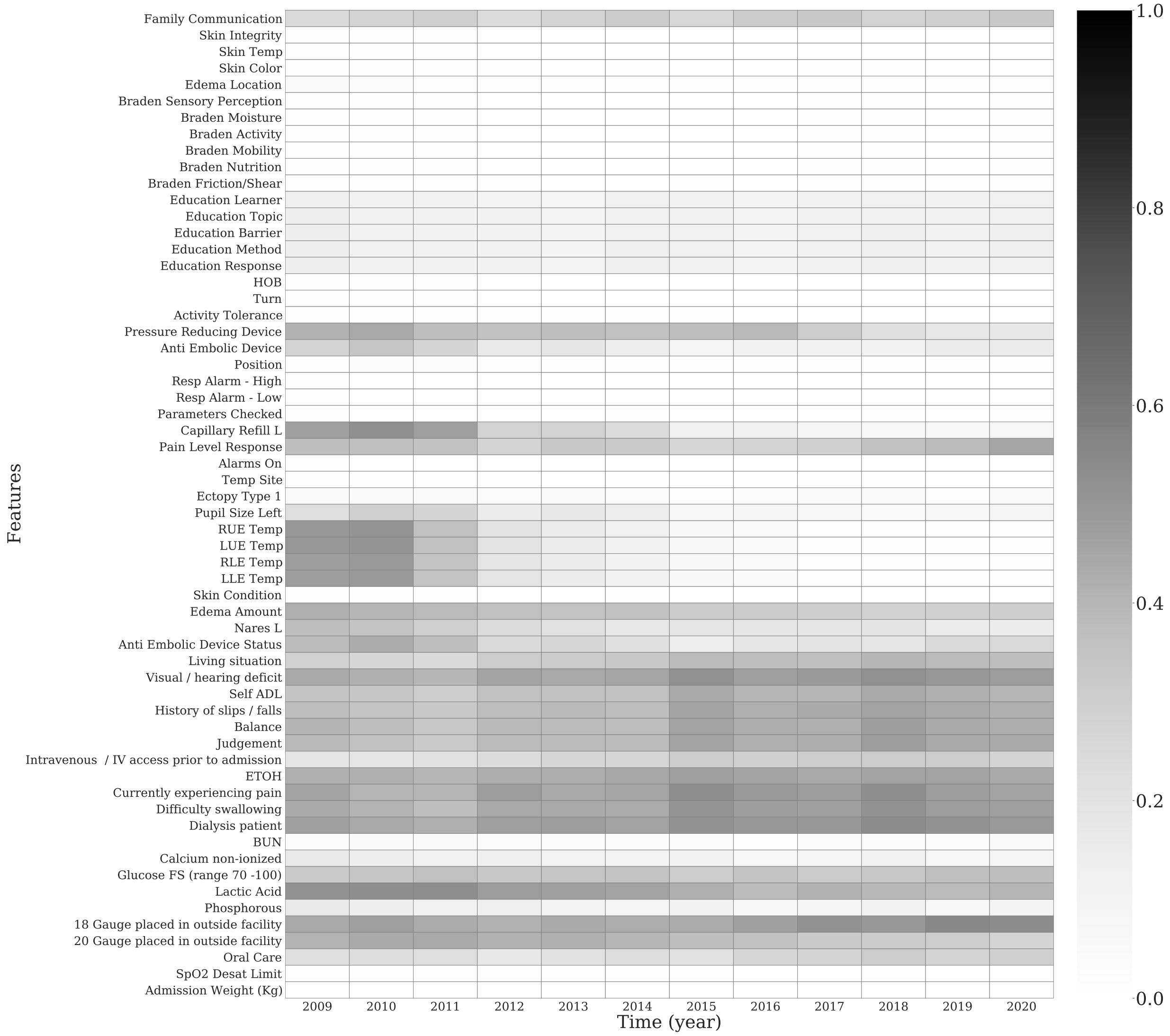


Figure C.22: Missingness over time for chartevents features in MIMIC-IV dataset after cohort selection. The darker the color, the larger the proportion of missing data. (part 2)

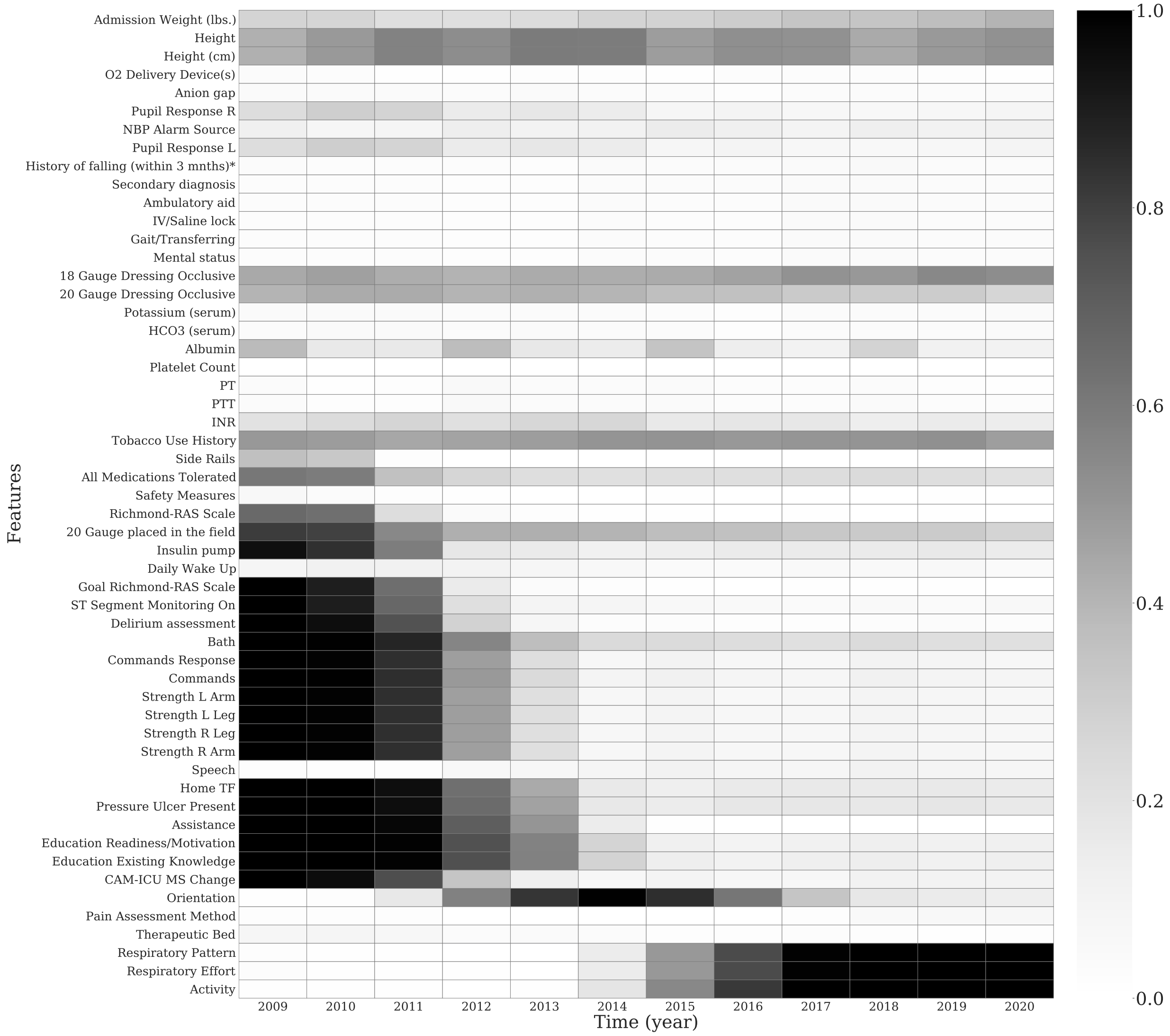


Figure C.23: Missingness over time for chartevents features in MIMIC-IV dataset after cohort selection. The darker the color, the larger the proportion of missing data. (part 3)

C.7 Additional OPTN (Liver) Data Details

The OPTN database (Organ Procurement and Transplantation Network, 2020) tracks organ donation and transplant events in the U.S. Our study uses data from candidates on the liver transplant wait list. The performance over time is evaluated on a *yearly* basis. First, we provide the disclaimer: "The data reported here have been supplied by the United Network for Organ Sharing as the contractor for the Organ Procurement and transplantation Network. The interpretation and reporting of these data are the responsibility of the author(s) and in no way should be seen as an official policy of or interpretation by the OPTN or the U.S. Government".

- Data access: After signing the Data Use Agreement - I from Organ Procedurement And Transplantation network, users can access the OPTN (Liver) dataset.
- Cohort selection: The cohort consists of liver transplant candidates on the waiting list (2005-2017). We follow the same pipeline as Byrd et al. (2021) to extract the data, except that we select the first record for each patient. Cohort selection diagrams are given in Figures C.24. When a patient is first added to the transplant list, given what we know about that patient, we woudl like to estimate their 180-day risk of mortality.
- Outcome definition: 180-day mortality from when the patient was first added to the list
- Cohort characteristics: Cohort characteristics are given in Table C.7.
- Features: We list the features used in the OPTN liver dataset in Section C.7.2. Categorical variables are converted into dummies, and numerical variables are standard scaled.
- Missingness heat maps: are given in Figures C.25 and C.26.

C.7.1 Cohort Selection and Cohort Characteristics

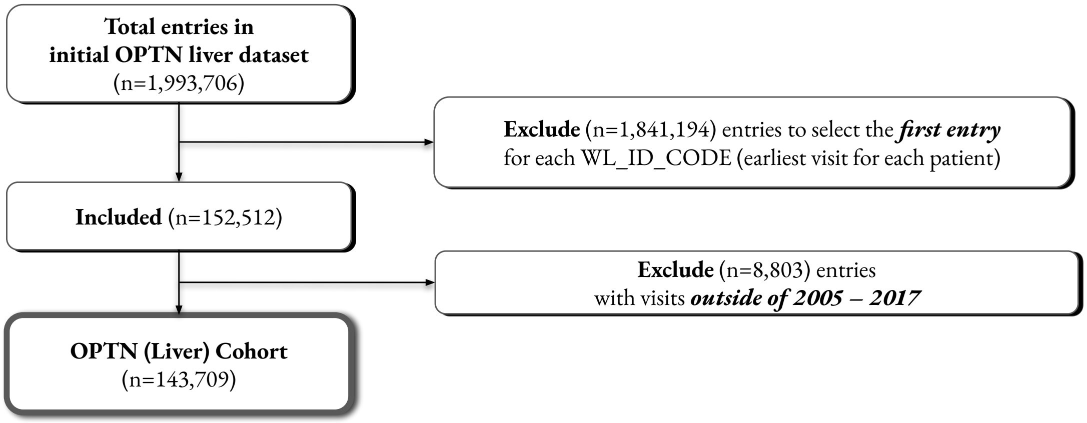

Total entries in initial OPTN liver dataset (n=1,993,706)
Exclude (n=1,841,194) entries to select the ***first entry*** for each WL_ID_CODE (earliest visit for each patient)
Included (n=152,512)
Exclude (n=8,803) entries with visits ***outside of 2005 – 2017***
OPTN (Liver) Cohort (n=143,709)

Figure C.24: Cohort selection diagram - OPTN (Liver)

Table C.7: OPTN (Liver) cohort characteristics, with count (%) or median (Q1 – Q3).

Feature name (value)		Empty (ratio)	Type
Gender			
Male	92,560 (64.4%)	–	categorical
Female	51,149 (35.6%)	–	categorical
INIT_AGE	56 (49-62)	0.0%	continuous
FUNC_STAT_TCR	2,070 (2,050-2,080)	0.0%	categorical
INIT_OPO_CTR_CODE	11,036 (3,782-19,282)	0.0%	categorical
ALBUMIN	3 (3-4)	0.0%	continuous
HCC_DIAGNOSIS_TCR			
No	31,390 (21.8%)	–	categorical
Yes	11,312 (7.9%)	–	categorical
Missing	101,007 (70.3%)	–	categorical
PERM_STATE			
CA	19,645 (13.7%)	–	categorical
TX	14,692 (10.2%)	–	categorical
NY	9,976 (6.9%)	–	categorical
GA	4,052 (2.8%)	–	categorical
MD	4,050 (2.8%)	–	categorical
FL	7,602 (5.3%)	–	categorical
PA	8,013 (5.6%)	–	categorical
MI	3,989 (2.8%)	–	categorical
Other	71,007 (49.4%)	–	categorical
EDUCATION	4 (3-5)	0.0%	categorical
ASCITES	2 (1-2)	0.0%	categorical
MORTALITY_180D			
1	4,635 (3.2%)	–	categorical
0	139,074 (96.8%)	–	categorical

C.7.2 Features

ABO, BACT PERIT TCR, CITIZENSHIP, DGN TCR, DGN2 TCR, DIAB, EDUCATION, FUNC STAT TCR, GENDER, LIFE SUP TCR, MALIG TCR, OTH LIFE SUP TCR, PERM STATE, PORTAL VEIN TCR, PREV AB SURG TCR, PRI PAYMENT TCR, REGION, TIPSS TCR, VENTILATOR TCR, WORK INCOME TCR, ETHCAT, HCC DIAGNOSIS TCR, MUSCLE WAST TCR, INIT OPO CTR CODE, WLHR, WLIN, WLKI, WLLU, WLPA, INACTIVE, ASCITES, ENCEPH, DIALYSIS PRIOR WEEK, INIT HGT CM, INIT WGT KG, INIT BMI CALC, INIT AGE, UNOS CAND STAT CD, BILIRUBIN, SERUM CREAT, INR, SERUM SODIUM, ALBUMIN, BILIRUBIN DELTA, SERUM CREAT DELTA, INR DELTA, SERUM SODIUM DELTA, ALBUMIN DELTA

C.7.3 Missingness heatmaps

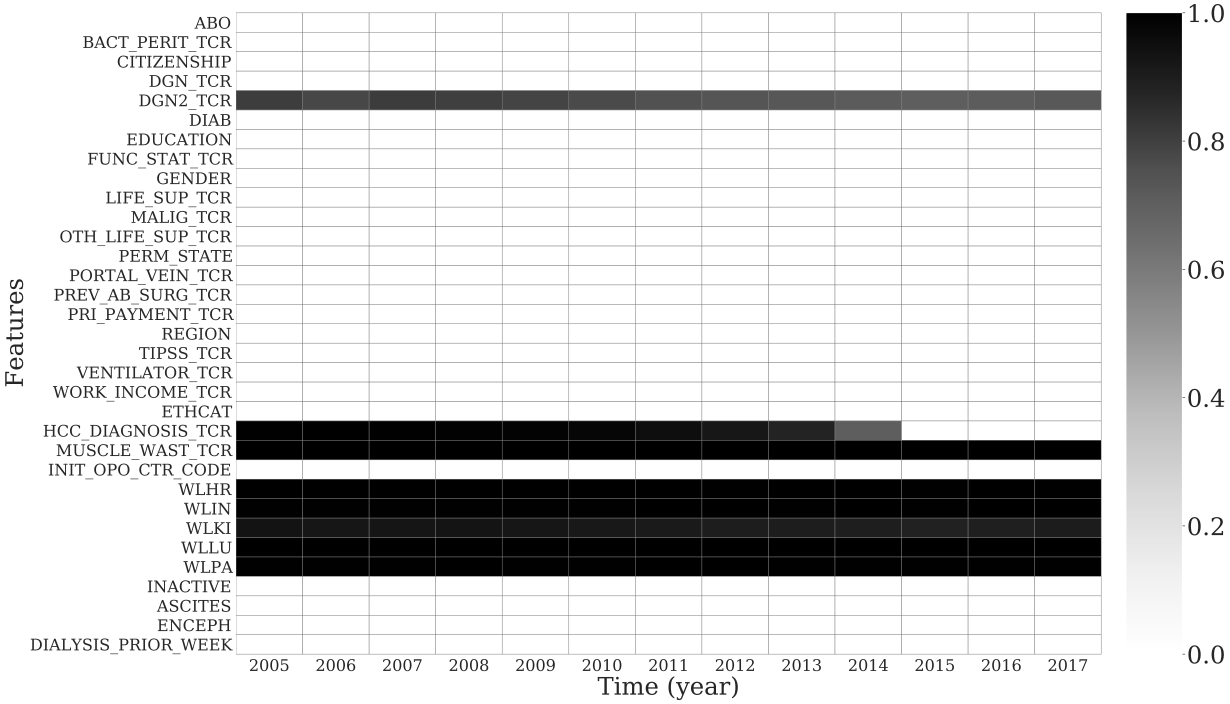


Figure C.25: Missingness over time for categorical features in OPTN (Liver) dataset after cohort selection. The darker the color, the larger the proportion of missing data.

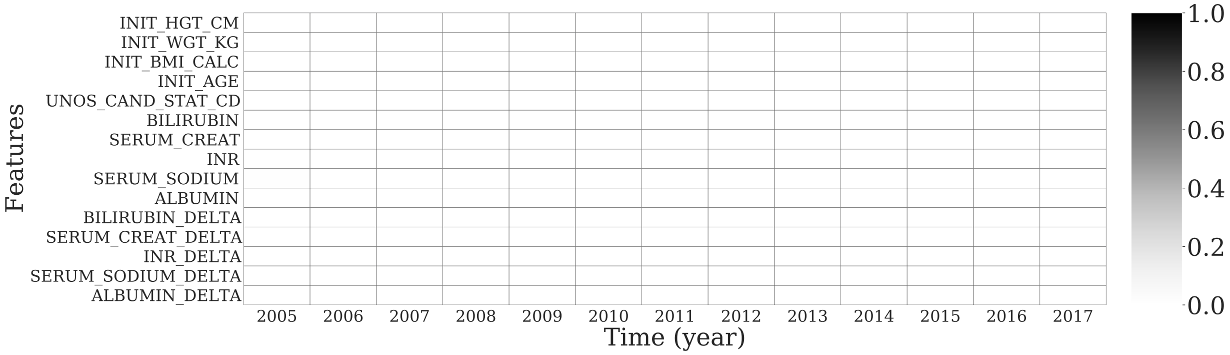


Figure C.26: Missingness over time for numerical features in OPTN (Liver) dataset after cohort selection. The darker the color, the larger the proportion of missing data. (Near-zero missingness here.)

C.8 Additional MIMIC-CXR Data Details

The MIMIC Chest X-ray (MIMIC-CXR-JPG) (Johnson et al., 2019b) is a publicly available dataset containing chest radiographs in JPG format from 2009–2018. Similar to MIMIC-IV, MIMIC-CXR add time annotations placing each sample into a three-year time range. We approximate the year of each sample by taking the midpoint of its time range. Each patient has an `anchor_year_group`, `anchor_year` and `StudyDate`. For each patient, we first calculated an offset as the difference between `StudyDate` and `anchor_year`. Then, we approximated the admit time as the midpoint of `anchor_year_group` after applying the computed offset. The performance over time is evaluated on a *yearly* basis. Our study uses MIMIC-IV-JPG-2.0. A similar training setup to that in Seyyed-Kalantari et al. (2020) was used (learning rate, architecture, data augmentation, stopping criteria, etc.).

- Data access: Users must create a Physionet account, become credentialed, and sign a data use agreement (DUA).
- Cohort selection: We removed the records from 2009 due to the tiny sample size. (Selection diagram in Figure C.27). We keep all records for each patients and split the data based on patient `subject id`.
- Outcome definition: There are 13 abnormal outcomes and 1 normal outcome: Atelectasis, Cardiomegaly, Consolidation, Edema, Enlarged Cardiomediastinum, Fracture, Lung Lesion, Lung Opacity, Pleural Effusion, Pneumonia, Pneumothorax, Pleural Other, Support Devices, No Finding
- Cohort characteristics: Cohort characteristics are given in Table C.8.

C.8.1 Cohort Selection and Cohort Characteristics

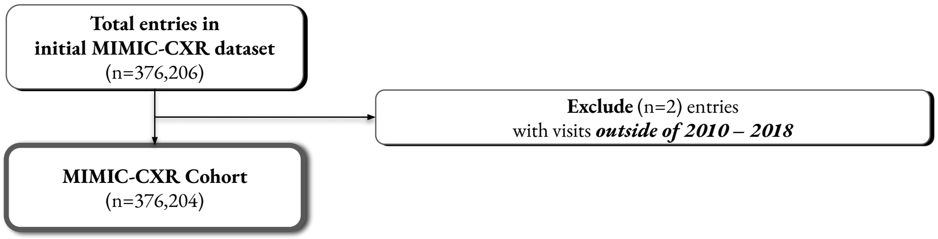

Total entries in
initial MIMIC-CXR dataset
(n=376,206)
Exclude (n=2) entries
with visits ***outside of 2010 – 2018***
MIMIC-CXR Cohort
(n=376,204)

Figure C.27: Cohort selection diagram - MIMIC-CXR

Table C.8: MIMIC-CXR cohort characteristics, with count (%) or median (Q1–Q3).

Feature name (value)	Summary statistic	Empty (ratio)	Status
Gender			
F	179,765 (47.8%)	–	categorical
M	196,439 (52.2%)	–	categorical
Age	64 (51-76)	0.0%	continuous
Diseases			
Atelectasis	65,390 (17.4%)	–	categorical
Cardiomegaly	56,404 (15.0%)	–	categorical
Consolidation	14,394 (3.8%)	–	categorical
Edema	36,026 (9.6%)	–	categorical
Enlarged Cardiomediastinum	9,821 (2.6%)	–	categorical
Fracture	6,314 (1.7%)	–	categorical
Lung Lesion	10,574 (2.8%)	–	categorical
Lung Opacity	76,074 (20.2%)	–	categorical
Pleural Effusion	75,526 (20.1%)	–	categorical
Pleural Other	3,432 (0.9%)	–	categorical
Pneumonia	25,065 (6.7%)	–	categorical
Pneumothorax	12,828 (3.4%)	–	categorical
Support Devices	69,148 (18.4%)	–	categorical
No Finding	167,116 (44.4%)	–	categorical

C.8.2 Label level AUROC over time for MIMIC-CXR

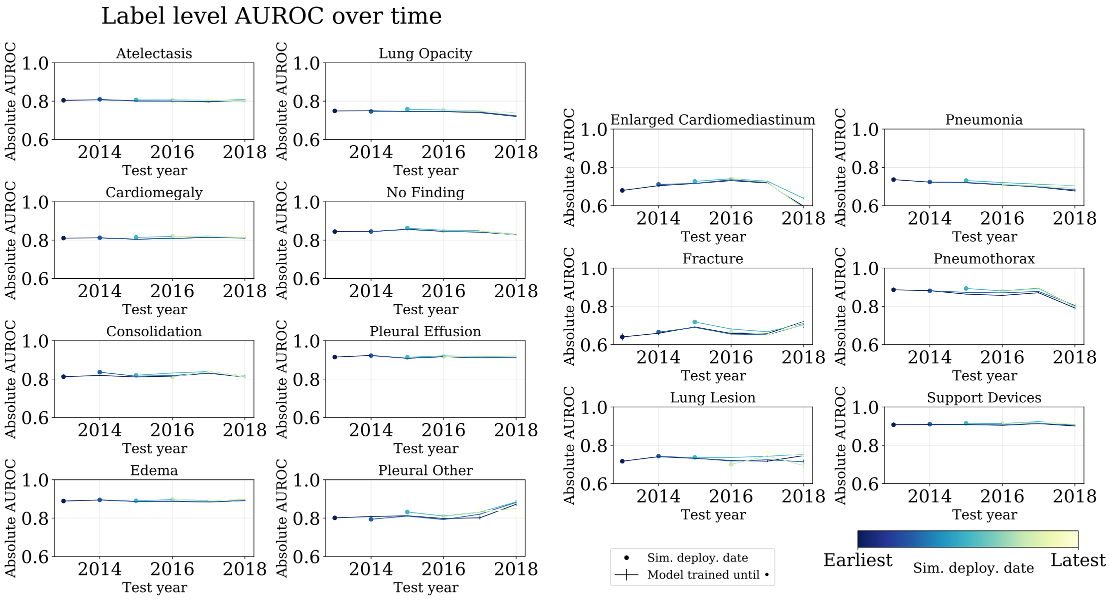


Figure C.28: Absolute AUROC over time of each label in MIMIC-CXR

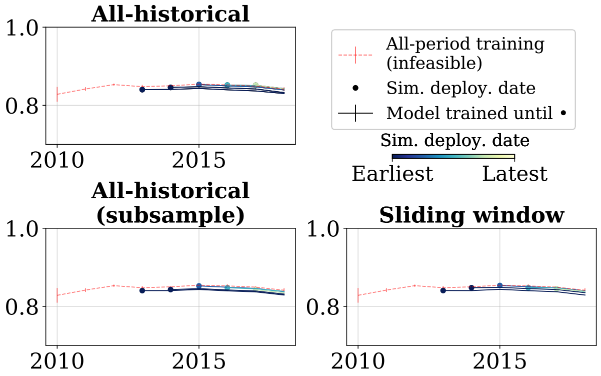


Figure C.29: Weighted test AUROC vs. year for the DenseNet architecture on MIMIC-CXR.

Table C.9: MIMIC-CXR label-level AUROC from time-agnostic evaluation of all-period training. The format is mean (±std. dev. across splits)

Label	AUROC	Label	AUROC
Atelectasis	0.826 (±0.003)	Cardiomegaly	0.837 (±0.002)
Consolidation	0.841 (±0.003)	Edema	0.904 (±0.002)
Enlarged Cardiomediastinum	0.759 (±0.005)	Fracture	0.745 (±0.006)
Lung Lesion	0.784 (±0.003)	Lung Opacity	0.770 (±0.002)
Pleural Effusion	0.929 (±0.001)	Pleural Other	0.844 (±0.009)
Pneumonia	0.755 (±0.004)	Pneumothorax	0.918 (±0.006)
Support Devices	0.928 (±0.001)	No Finding	0.876 (±0.002)

C.9 Coefficients from Splitting by Patient

To help with intuition in important features for the predictive task on each dataset, here we have the coefficients of logistic regression models trained from splitting by patient.

C.9.1 SEER (Breast)

Table C.10: SEER (Breast) top 10 important features for LR models, all-period training.

Feature	Coefficient
SEER historic stage A (1973-2015)_Distant	-2.113944
SEER historic stage A (1973-2015)_Localized	1.676493
Regional nodes examined (1988+)_95.0	-1.167844
CS lymph nodes (2004-2015)_750	1.100824
CS lymph nodes (2004-2015)_755	1.023753
Histologic Type ICD-O-3_8530	-0.913494
Histologic Type ICD-O-3_8543	0.902798
Breast - Adjusted AJCC 6th T (1988-2015)_T4d	0.899491
Histologic Type ICD-O-3_8211	0.877848
EOD 10 - extent (1988-2003)_85	-0.791136

Table C.11: SEER (Colon) top 10 important features for LR models, all-period training.

Feature	Coefficient
Reason no cancer-directed surgery_Surgery performed	2.360161
Regional nodes positive (1988+)_00	1.897706
Regional nodes positive (1988+)_01	1.872008
modified AJCC stage 3rd (1988-2003)_40	-1.787481
EOD 10 - extent (1988-2003)_13	1.766066
Reason no cancer-directed surgery_Not recommended, contraindicated due to other cond; autopsy only (1973-2002)	-1.752474
EOD 10 - extent (1988-2003)_85	-1.732619
EOD 10 - extent (1988-2003)_70	-1.704333
CS mets at dx (2004-2015)_99	1.619905
CS mets at dx (2004-2015)_00	1.609454

Table C.12: SEER (Lung) top 10 important features for LR models, all-period training.

Feature	Coefficient
Histologic Type ICD-O-3_8240	2.514539
EOD 4 - nodes (1983-1987)_0	2.074730
EOD 4 - nodes (1983-1987)_7	-1.777530
EOD 10 - size (1988-2003)_140	-1.587893
Histologic Type ICD-O-3_8141	-1.546566
CS tumor size (2004-2015)_998.0	-1.515856
EOD 4 - nodes (1983-1987)_6	-1.497022
Type of Reporting Source_Nursing/convalescent home/hospice	-1.338998
CS mets at dx (2004-2015)_51	-1.326595
EOD 10 - size (1988-2003)_150	-1.326196

Table C.13: CDC COVID-19 top 10 important features for LR models, all-period training.

Feature	Coefficient
res_state_DE	2.202055
age_group_0 - 9 Years	-2.114818
age_group_80+ Years	1.965279
age_group_10 - 19 Years	-1.681099
res_state_GA	1.391469
age_group_70 - 79 Years	1.379589
res_county_WICHITA	1.290644
age_group_20 - 29 Years	-1.189734
res_county_SUMNER	-1.135073
mechvent_yn_Yes	1.117372

Table C.14: SWPA COVID-19 top 10 important features for LR models according to experiments splitting by patient.

Feature	Coefficient
age_bin_(70, 200]_0	-0.781337
age_bin_(70, 200]_1	0.780673
medication_FENTANYL (PF) 50 MCG/ML INJECTION SOLUTION_0.0	0.651419
medication_EPINEPHRINE 0.3 MG/0.3 ML INJECTION, AUTO-INJECTOR_nan	-0.627565
medication_HYDROCORTISONE SOD SUCCINATE (PF) 100 MG/2 ML SOLUTION FOR INJECTION_0.0	0.544222
medication_HYDROCODONE 5 MG-ACETAMINOPHEN 325 MG TABLET_nan	-0.520368
medication_DEXAMETHASONE SODIUM PHOSPHATE 4 MG/ML INJECTION SOLUTION_0.0	0.502954
medication_ASPIRIN 81 MG TABLET,DELAYED RELEASE_nan	-0.479100
bmi_nan	-0.427569
age_bin_(60, 70]_0	-0.380688

Table C.15: MIMIC-IV top 10 important features for LR models, all-period training.

Feature	Coefficient
O2 Delivery Device(s)_None	-0.307334
Eye Opening_None	0.301737
admit_age	0.299712
O2 Delivery Device(s)_Nasal cannula	-0.248463
Motor Response_Obeys Commands	-0.230931
Pupil Response L_Non-reactive	0.223776
Richmond-RAS Scale_ 0 Alert and calm	-0.205476
Temp Site_Blood	-0.204514
HR_0.0	0.197299
Diet Type_NPO	0.195156

Table C.16: OPTN (Liver) top 10 important features for LR models, all-period training.

Feature	Coefficient
SERUM_CREAT_DELTA	0.660589
FUNC_STAT_TCR_2020.0	0.241507
FUNC_STAT_TCR_2080.0	-0.236288
DGNC_4110.0	-0.234680
REGION_5.0	0.223940
EDUCATION_998.0	0.218549
ASCITES_3.0	0.218329
ASCITES_1.0	-0.214076
INIT_OPO_CTR_CODE_1054	-0.209265
INIT_OPO_CTR_CODE_4743	-0.207778

C.10 Diagnostic plots

We took the union of the top k most important features from each time point to be included in the diagnostic plots, where k was tuned depending on the dataset so that the resulting plots would not be overcrowded. For categorical features, we additionally highlighted (using a thicker line) features that had consistently high prevalence ($\geq p$) or experienced a large change in prevalence across one time point ($\geq \Delta$). The specific parameters of each dataset are defined in each subsection. For numerical features, we highlighted features whose average ranking across all time points was ≤ 3 (also chosen to avoid overcrowding).

C.10.1 SEER (Breast)

For SEER (Breast) diagnostic plots, important features were selected using $k = 5, p = 0.4, \Delta = 0.2$.

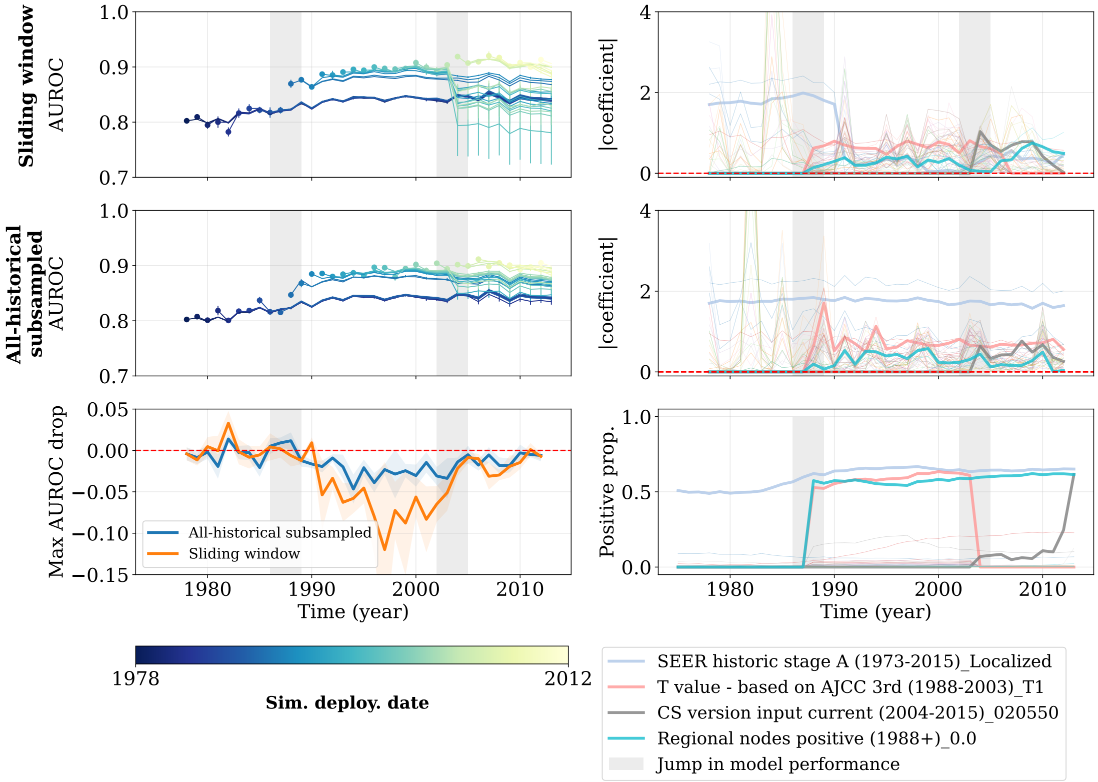


Figure C.30: Diagnostic plot of SEER (Breast) dataset. The important features are selected as the union of the top 5 features that have the highest absolute value model coefficients. The left column includes AUROC versus time for both sliding window and all-historical subsampled, and the maximum AUROC drop for each trained model. The right column provides the absolute coefficients of each trained model from both regimes, and positive proportion of the significant features over time. As shown in the gray highlighted region, there are jumps in performance around 1988 and 2003, which coincides with the introducing and removal of several features (e.g. T value - based on AJCC 3rd (1988-2003)_T1). The latency of jumps in coefficients are caused by length of sliding window.

C.10.2 SEER (Colon)

For SEER (Colon) diagnostic plots, important features were selected using $k = 3, p = 0.4, \Delta = 0.2$.

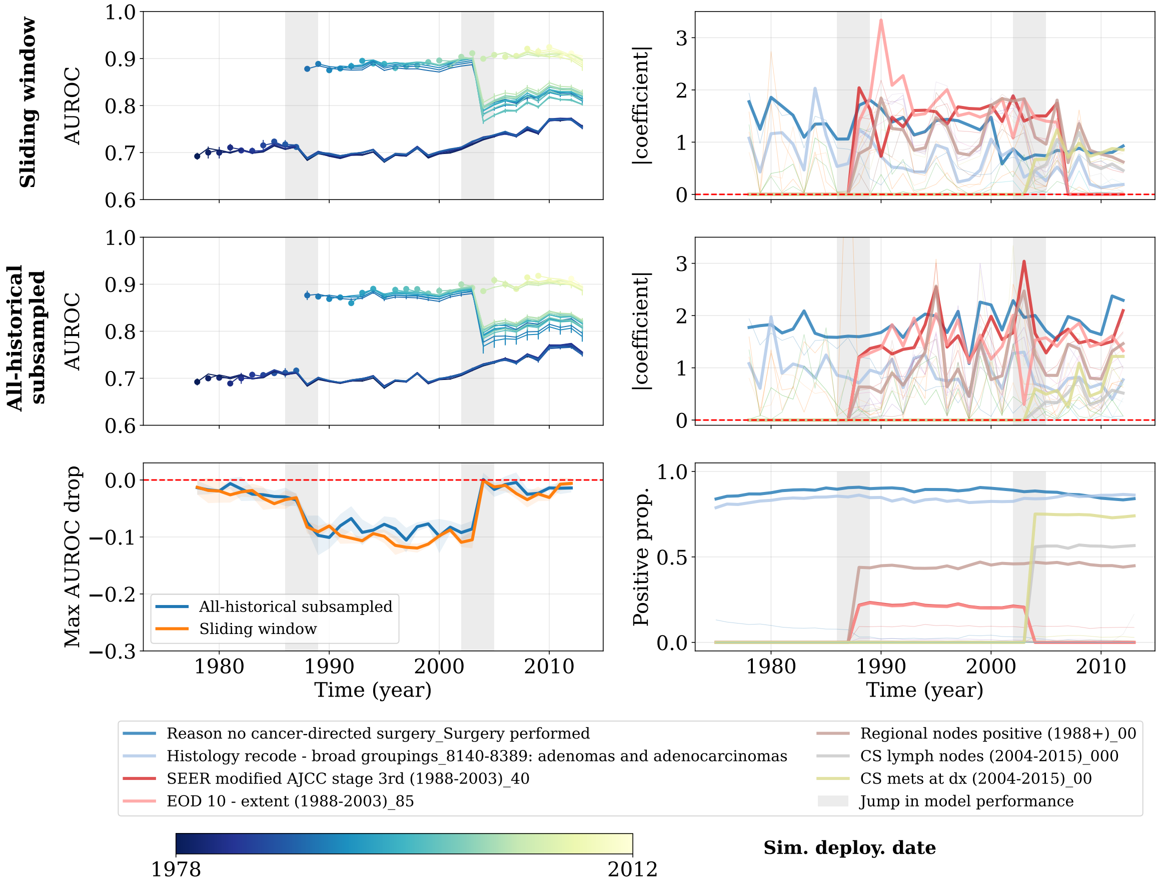


Figure C.31: Diagnostic plot of SEER (Colon) dataset. The important features are selected as the union of the top 3 features that have the highest absolute model coefficients. The left column includes AUROC versus time for both sliding window and all-historical subsampled, and the maximum AUROC drop for each trained model. The right column provides the absolute coefficients of each trained model from both regimes, and positive proportion of the significant features over time. As shown in the gray highlighted region, there are jumps in performance around 1988 and 2003, which coincides with the introducing and removal of several features (e.g. SEER modified AJCC stage 3rd (1988-2003)_40). The latency of jumps in coefficients are caused by length of sliding window.

C.10.3 SEER (Lung)

For SEER (Lung) diagnostic plots, important features were selected using $k = 5, p = 0.2, \Delta = 0.2$.

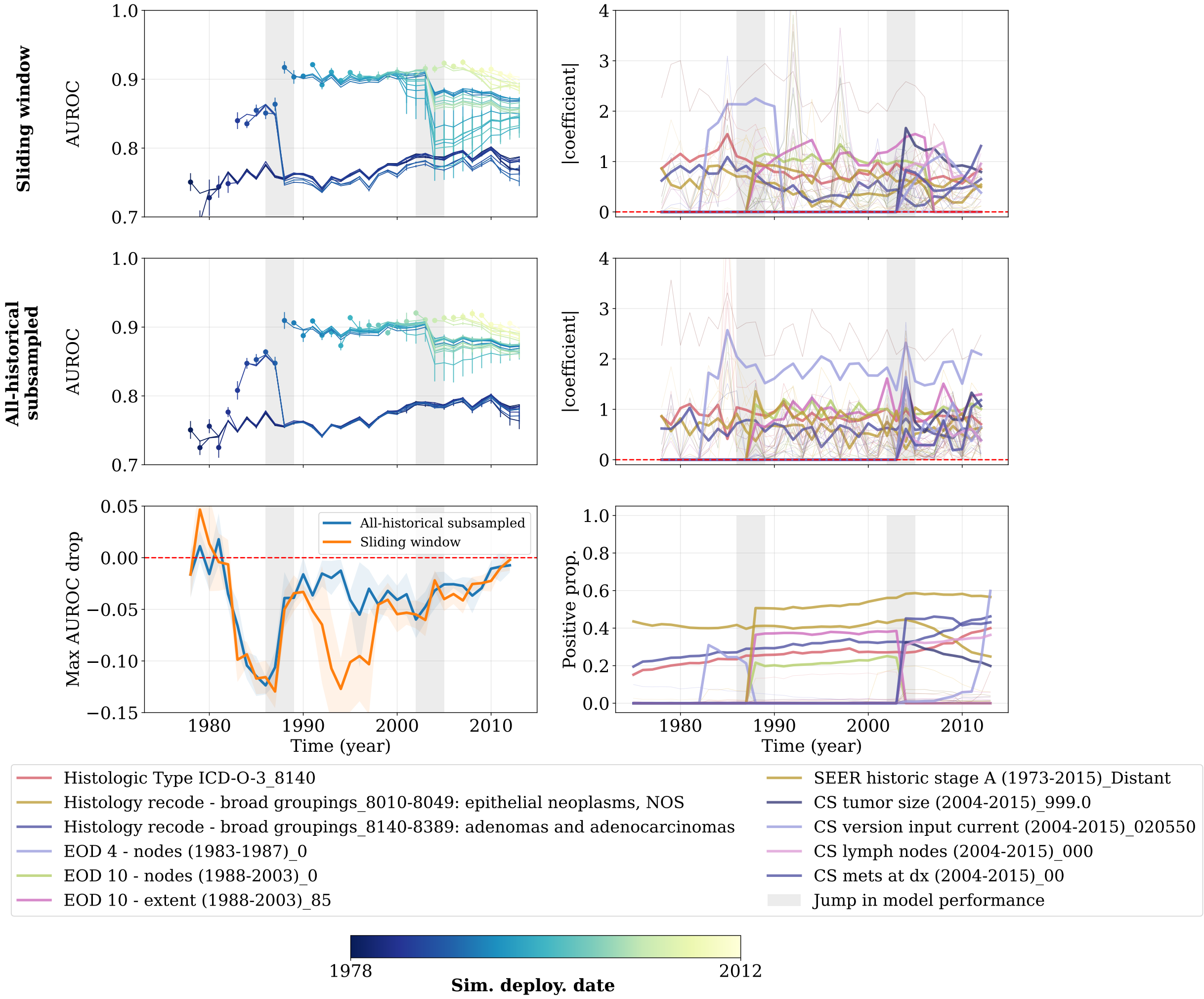


Figure C.32: Diagnostic plot of SEER (Lung) dataset. The important features are selected as the union of the top 5 features that have the highest absolute model coefficients. The left column includes AUROC versus time for both sliding window and all-historical subsampled, and the maximum AUROC drop for each trained model. The right column provides the absolute coefficients of each trained model from both regimes, and positive proportion of the significant features over time. As shown in the gray highlighted region, there are jumps in performance around 1988 and 2003, which coincides with the introducing and removal of several features (e.g. EOD 10 - nodes (1988-2013)_0 & EOD 10 - extent (1988-2003)_85). The latency of jumps in coefficients are caused by length of sliding window.

C.10.4 CDC COVID-19

For CDC COVID-19 diagnostic plots, important features were selected using $k = 5, p = 0.15, \Delta = 0.15$.

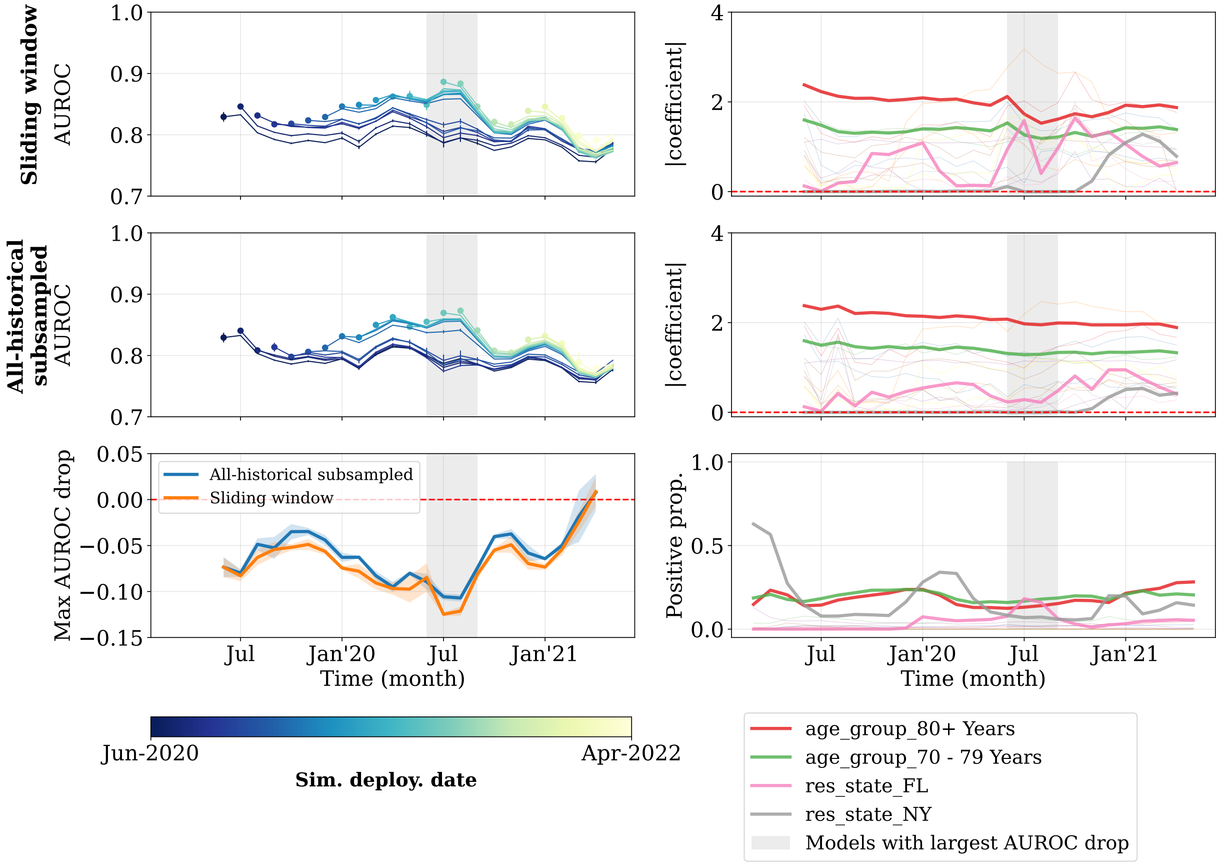


Figure C.33: Diagnostic plot of CDC COVID-19. The important features are selected as the union of the top 5 features that have the highest absolute model coefficients. The left column includes AUROC versus time for both sliding window and all-historical subsampled, and the maximum AUROC drop for each trained model. The right column provides the absolute coefficients of each trained model from both regimes, and positive proportion of the significant features over time. As shown in the gray highlighted region, the models trained around June 2021 suffer the largest maximum AUROC drop, coinciding with a shift in distribution of ages (Figure C.13a) and states (Figure C.13b). The latency of jumps in coefficients are caused by length of sliding window.

C.10.5 SWPA COVID-19

For SWPA COVID-19 diagnostic plots, important features were selected using $k = 3, p = 0.4, \Delta = 0.2$.

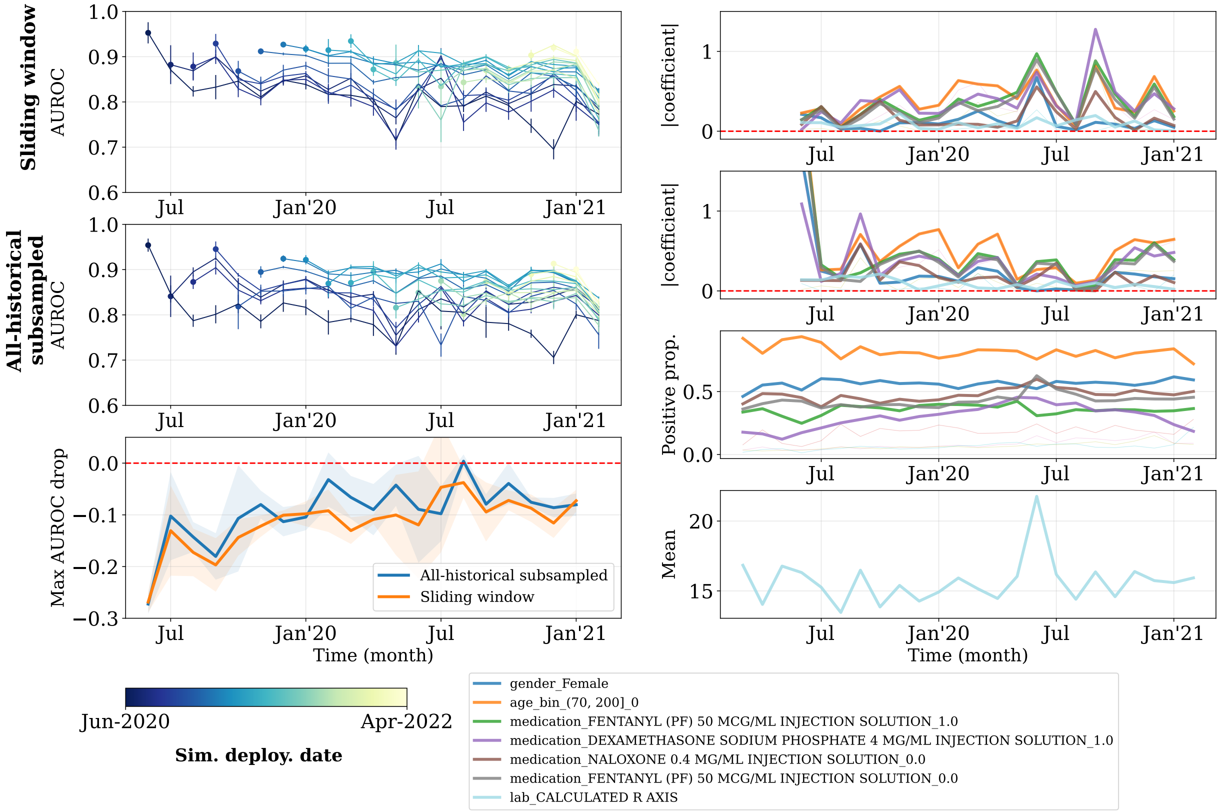


Figure C.34: Diagnostic plot of SWPA COVID-19. The important features are selected as the union of the top 3 features that have the highest absolute model coefficients. The left column includes AUROC versus time for both sliding window and all-historical subsampled, and the maximum AUROC drop for each trained model. The right column provides the absolute coefficients of each trained model from both regimes, and positive proportion of the significant features over time. One of the hypotheses for relatively large uncertainty is smaller sample size.

C.10.6 MIMIC-IV

For MIMIC-IV diagnostic plots, important features were selected using $k = 3, p = 0.4, \Delta = 0.2$.

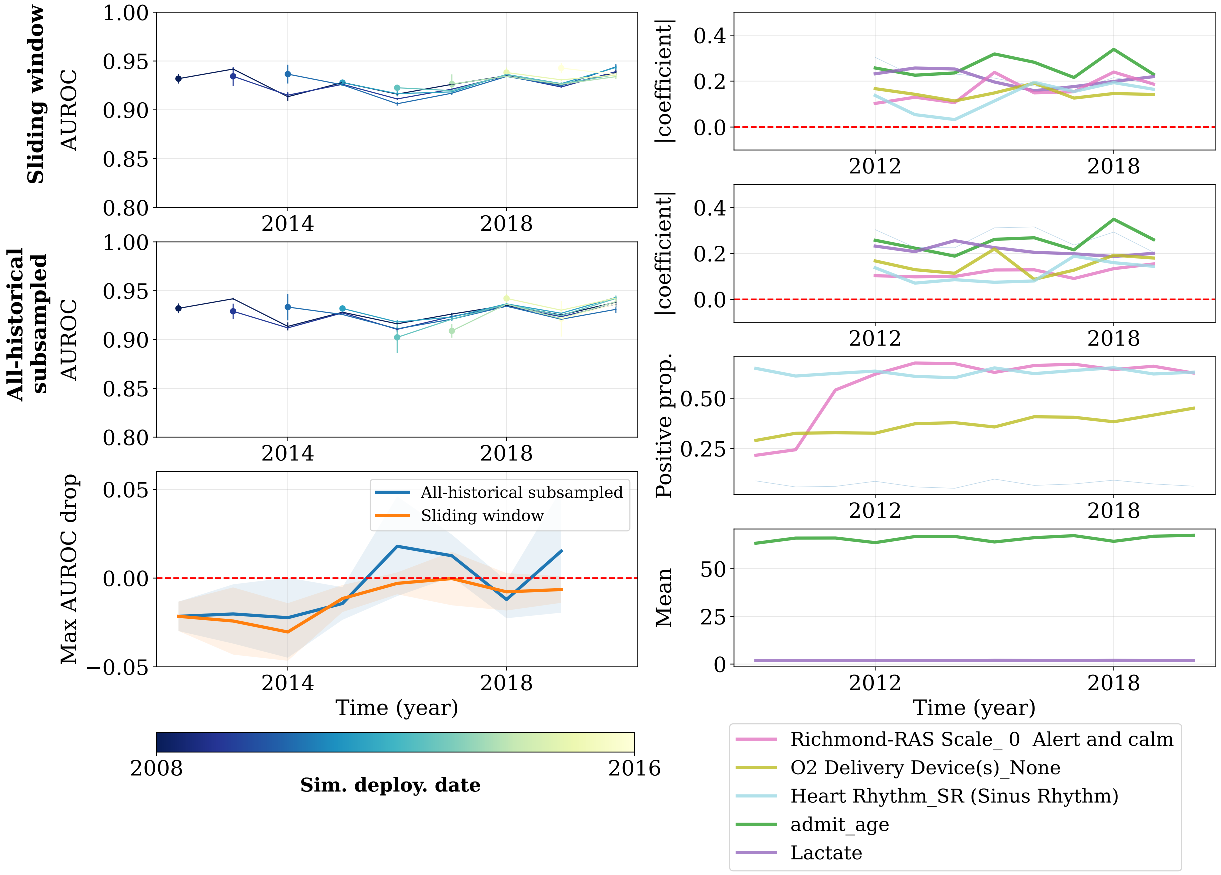


Figure C.35: Diagnostic plot of MIMIC-IV. The important features are selected as the union of the top 3 features that have the highest absolute model coefficients. The left column includes AUROC versus time for both sliding window and all-historical subsampled, and the maximum AUROC drop for each trained model. The right column provides the absolute coefficients of each trained model from both regimes, and positive proportion of the significant features over time. The model performance is relatively stable, coinciding with relatively stable distributions of a majority of important features.

C.10.7 OPTN (Liver)

OPTN (Liver) diagnostic plots, with important features selected using $k = 3, p = 0.4, \Delta = 0.2$.

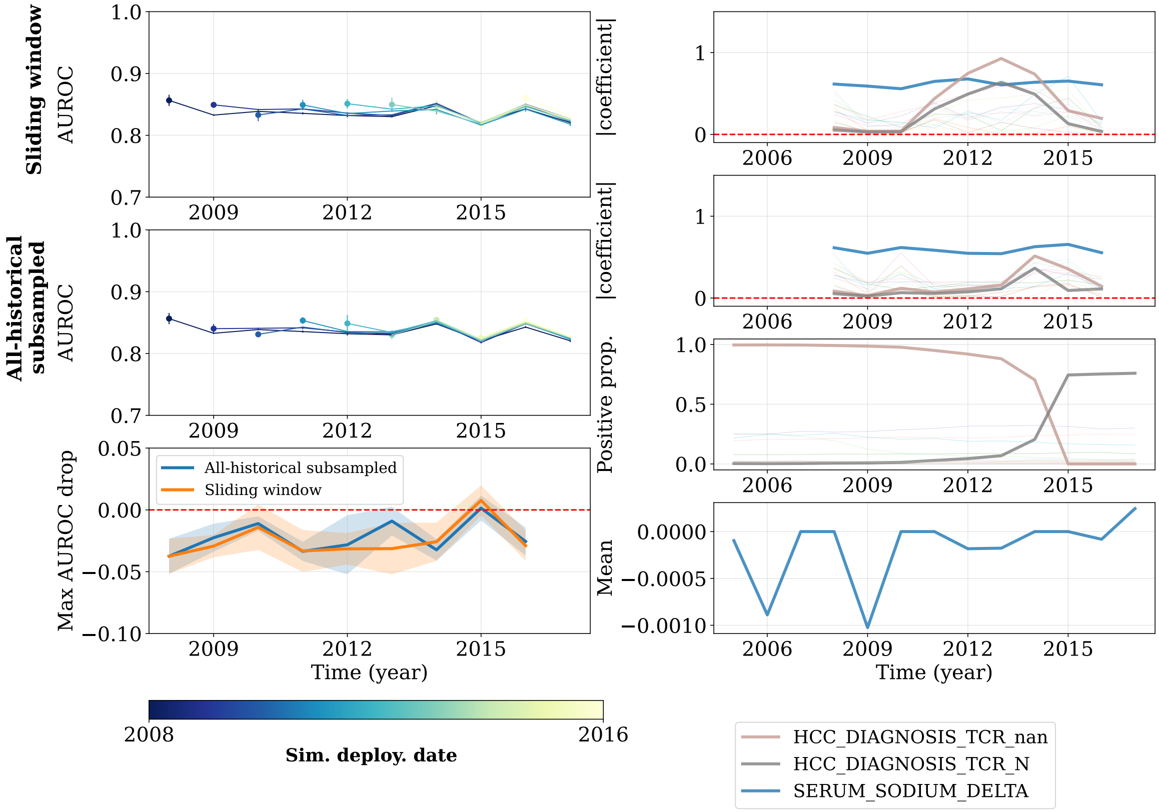


Figure C.36: Diagnostic plot of OPTN (Liver). The important features are selected as the union of the top 3 features that have the highest absolute model coefficients. The left column includes AUROC versus time for both sliding window and all-historical subsampled, and the maximum AUROC drop for each trained model. The right column provides the absolute coefficients of each trained model from both regimes, and positive proportion of the significant features over time. Although the HCC DIAGNOSIS TCR binary features change in positive proportion over time, these features were not always important, and the other important features (faded) maintain relatively stable proportions across time. Overall, model performance is quite stable over time.

C.10.8 MIMIC-CXR

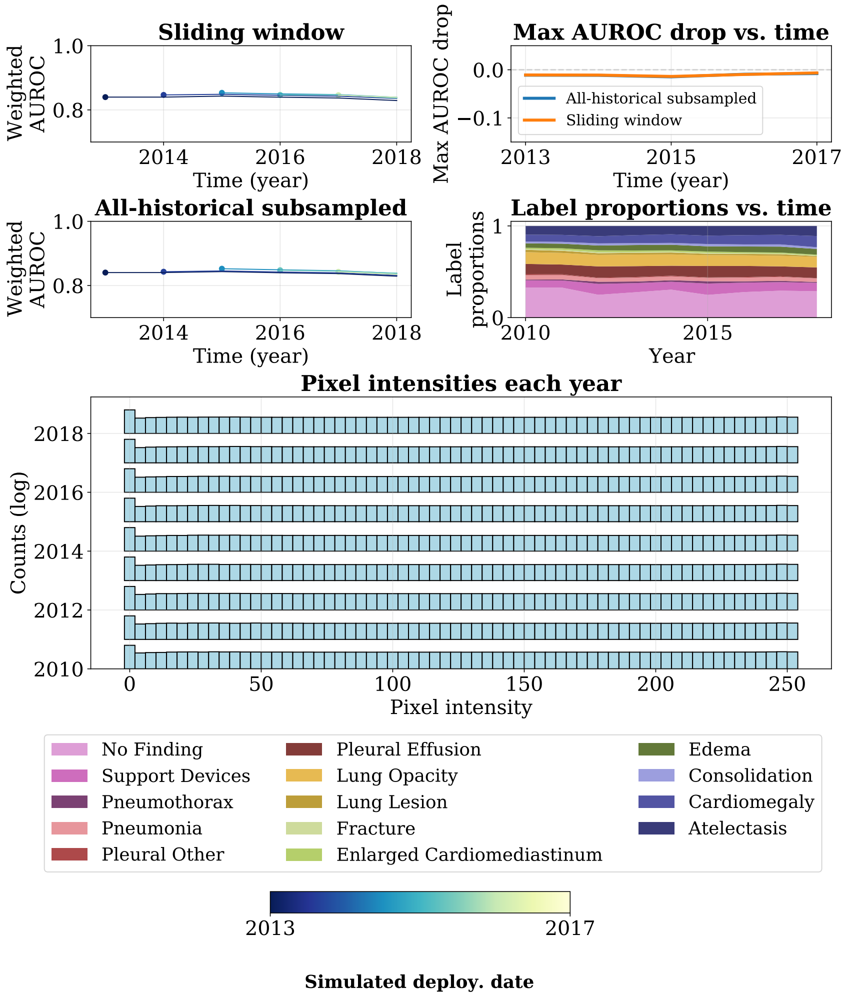


Figure C.37: Diagnostic plot of MIMIC-CXR. The top and mid left includes AUROC versus time for both sliding window and all-historical subsampled. The top right is the maximum AUROC drop for each trained model. The mid-right provides the label proportions over time. The bottom shows pixel intensities for images in each year. The histogram of pixel intensity is stable over time, which is consistent with the small variation in model performance over time

C.11 Model performance over time from three models

All plots in this section are for the all-historical training regime.

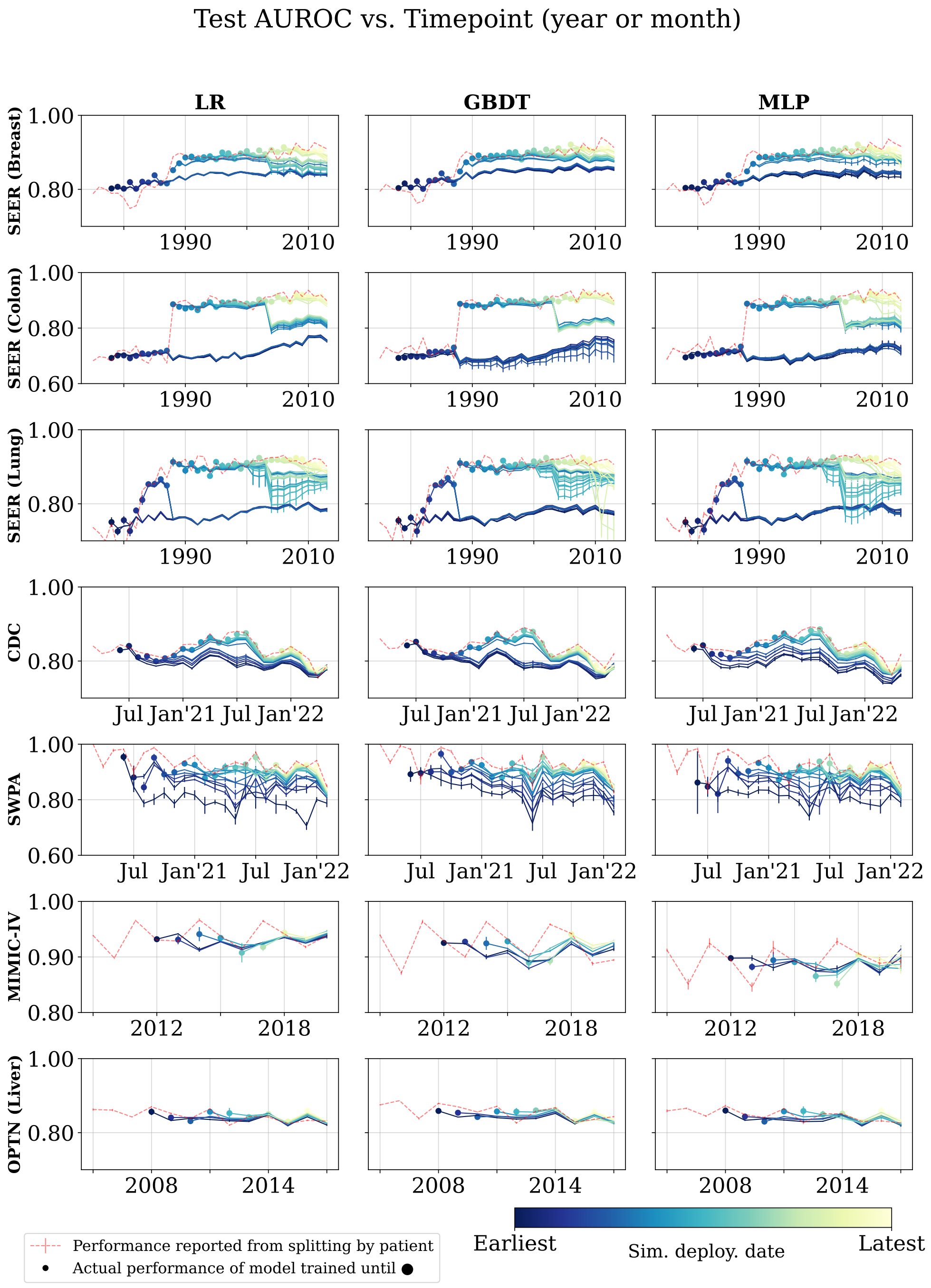


Figure C.38: AUROC versus test timepoints from three model classes on all datasets.

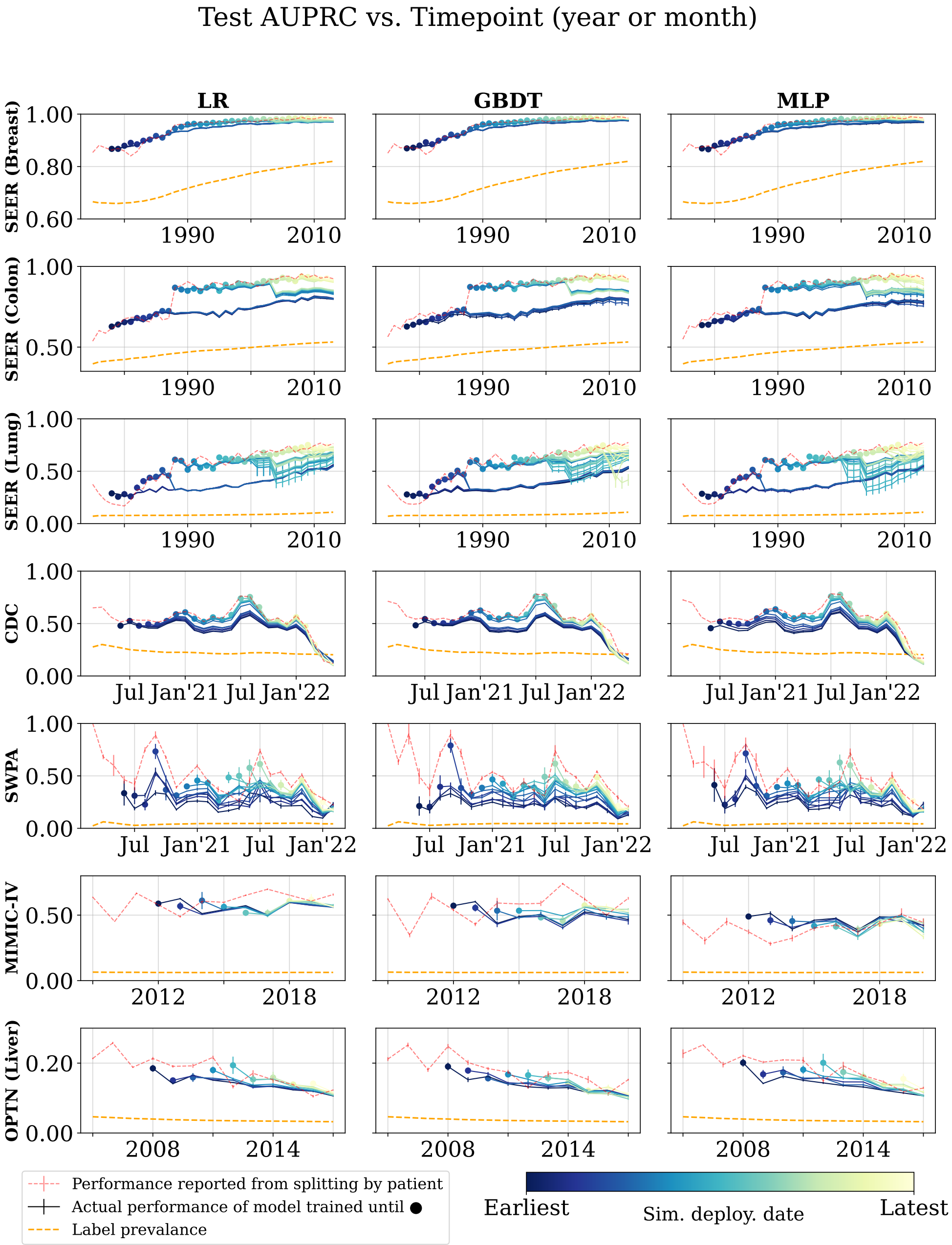


Figure C.39: AUPRC versus test timepoints from three model classes on all datasets. Label prevalance refers to the ratio of accumulated positive labels over time.

C.12 Data Split Details

Table C.17: Split ratio for each dataset for training, validation and testing (both for time-agnostic splits and in-period splits).

Dataset	Split ratio
SEER (Breast)	0.8-0.1-0.1
SEER (Colon)	0.8-0.1-0.1
SEER (Lung)	0.8-0.1-0.1
CDC COVID-19	0.8-0.1-0.1
SWPA COVID-19	0.5-0.25-0.25
MIMIC-IV	0.5-0.25-0.25
OPTN (Liver)	0.5-0.25-0.25
MIMIC-CXR	0.5-0.25-0.25

C.13 Hyperparameter Grids

Table C.18: Hyperparameter grids for model training.

Parameter	Values Considered
LR	
C	$0.01, 0.1, 1, 10, 10^2, 10^3, 10^4, 10^5$
GBDT	
n_estimators	50, 100
max_depth	3, 5
learning_rate	0.01, 0.1
MLP	
hidden_layer_sizes	3, 5
learning_rate_init	$10^{-4}, 10^{-3}, 0.01$

C.14 AUROC from full-period training

Table C.19: AUROC report from full-period training, the results are in format mean (±std. dev. across splits)

Dataset	Model	Full-period AUROC
SEER (Breast)	LR	0.888 (±0.002)
	GBDT	0.891 (±0.002)
	MLP	0.891 (±0.002)
SEER (Colon)	LR	0.863 (±0.003)
	GBDT	0.868 (±0.002)
	MLP	0.869 (±0.003)
SEER (Lung)	LR	0.894 (±0.002)
	GBDT	0.894 (±0.002)
	MLP	0.898 (±0.002)
CDC COVID-19	LR	0.837 (±0.001)
	GBDT	0.851 (±0.001)
	MLP	0.852 (±0.002)
SWPA COVID-19	LR	0.928 (±0.005)
	GBDT	0.930 (±0.004)
	MLP	0.928 (±0.006)
MIMIC-IV	LR	0.935 (±0.003)
	GBDT	0.931 (±0.002)
	MLP	0.898 (±0.008)
OPTN (Liver)	LR	0.846 (±0.005)
	GBDT	0.854 (±0.005)
	MLP	0.847 (±0.006)
MIMIC-CXR	DenseNet	0.860 (±0.001)

Appendix D

Domain Adaptation under Missingness Shift

D.1 Motivating Examples

Example 1 (Redundant Features)

Let $m_s = [1-\epsilon, \epsilon]$ and $m_t = [\epsilon, 1-\epsilon]$. Consider the following data generating process:

$$\begin{aligned} Z &= u_Z \\ X_1 &= Z & \qquad u_Z &\sim \mathcal{N}(0, \sigma_Z^2) \\ X_2 &= Z & \qquad u_Y &\sim \mathcal{N}(0, \sigma_Y^2) \\ Y &= Z + u_Y \end{aligned}$$

where Z is a latent variable, X_1 and X_2 are observed covariates, and Y is the label we wish to predict.

We start by summarizing the findings, and then provide the full algebraic justification. The optimal (risk-minimizing) linear predictor on the source data is given by:

$$\beta_*^s = \left[\frac{\epsilon}{1-\epsilon+\epsilon^2}, \frac{1-\epsilon}{1-\epsilon+\epsilon^2}\right]$$

And for the target data:

$$\beta_*^t = \left[\frac{1-\epsilon}{1-\epsilon+\epsilon^2}, \frac{\epsilon}{1-\epsilon+\epsilon^2}\right]$$

The excess risk of the source predictor on the target data is given by:

$$\begin{aligned} r^t(\beta_*^s) - r^t(\beta_*^t) &= (\beta_*^s - \beta_*^t)^\top \mathbb{E}[\widetilde{X}^{t\top} \widetilde{X}^t](\beta_*^s - \beta_*^t) \\ &= \sigma_Z^2 \cdot \frac{(1-2\epsilon)^2(1-2\epsilon+2\epsilon^2)}{(1-\epsilon+\epsilon^2)^2} \end{aligned}$$

As $\epsilon \to 0$, we have:

$$\begin{aligned}
\beta^s_* &\to [0, 1] \\
\beta^t_* &\to [1, 0] \\
r^t(\beta^s_*) - r^t(\beta^t_*) &\to \sigma_Z^2 \\
r^t([0, 0]) - r^t(\beta^t_*) &\to \sigma_Z^2
\end{aligned}$$

that is, the source classifier performs no better than simply predicting 0 (the mean of Y). Thus, $r^t(\beta^s_*) \to \sigma_Z^2 + \sigma_Y^2 = \text{Var}(Y)$

Proof. In the example, we have:

$$\begin{aligned}
\mathbb{E}[X^T X] &= \begin{bmatrix} \sigma_Z^2 & \sigma_Z^2 \\ \sigma_Z^2 & \sigma_Z^2 \end{bmatrix} \\
\mathbb{E}[X^T Y] &= \begin{bmatrix} \sigma_Z^2 \\ \sigma_Z^2 \end{bmatrix}
\end{aligned}$$

We apply the expressions for $\mathbb{E}[\widetilde{X}^T \widetilde{X}]$ and $\mathbb{E}[\widetilde{X}^\top Y]$ derived in Appendix D.8:

$$\begin{aligned}
\mathbb{E}[\widetilde{X}^\top \widetilde{X}] &= (1 - m)(1 - m)^\top \odot \mathbb{E}\left[X^\top X\right] + \text{diag}\left(m(1 - m^\top)\right) \text{diag}\left(\mathbb{E}\left[X^\top X\right]\right) \\
&= \begin{bmatrix} 1 - m_1 & (1 - m_1)(1 - m_2) \\ (1 - m_1)(1 - m_2) & 1 - m_2 \end{bmatrix} \odot \mathbb{E}\left[X^\top X\right] \\
\mathbb{E}[\widetilde{X}^\top Y] &= (1 - m) \odot \mathbb{E}[X^\top Y]
\end{aligned}$$

to get:

$$\begin{aligned}
\mathbb{E}[\widetilde{X}^{t\top} \widetilde{X}^t] &= \begin{bmatrix} 1 - \epsilon & \epsilon(1 - \epsilon) \\ \epsilon(1 - \epsilon) & \epsilon \end{bmatrix} \cdot \sigma_Z^2 \\
\mathbb{E}[\widetilde{X}^{t\top} \widetilde{X}^t]^{-1} &= \frac{1}{\sigma_Z^2 \epsilon(1 - \epsilon)(1 - \epsilon + \epsilon^2)} \begin{bmatrix} \epsilon & -\epsilon(1 - \epsilon) \\ -\epsilon(1 - \epsilon) & 1 - \epsilon \end{bmatrix} \\
\mathbb{E}[\widetilde{X}^{t\top} Y] &= \sigma_Z^2 \begin{bmatrix} 1 - \epsilon \\ \epsilon \end{bmatrix} \\
\beta^t_* &= \mathbb{E}[\widetilde{X}^{t\top} \widetilde{X}^t]^{-1} \mathbb{E}[\widetilde{X}^{t\top} Y] \\
&= \frac{1}{\epsilon(1 - \epsilon)(1 - \epsilon + \epsilon^2)} \begin{bmatrix} \epsilon(1 - \epsilon) + -\epsilon^2(1 - \epsilon) \\ -\epsilon(1 - \epsilon)^2 + \epsilon(1 - \epsilon) \end{bmatrix} \\
&= \frac{1}{\epsilon(1 - \epsilon)(1 - \epsilon + \epsilon^2)} \begin{bmatrix} \epsilon(1 - \epsilon)(1 - \epsilon) \\ \epsilon(1 - \epsilon)(-(1 - \epsilon) + 1) \end{bmatrix} \\
&= \frac{1}{1 - \epsilon + \epsilon^2} \begin{bmatrix} 1 - \epsilon \\ \epsilon \end{bmatrix}.
\end{aligned}$$

Similarly,

$$\beta^s_* = \frac{1}{1 - \epsilon + \epsilon^2} \begin{bmatrix} \epsilon \\ 1 - \epsilon \end{bmatrix},$$

so we can compute

$$\begin{aligned}\beta_*^s - \beta_*^t &= \frac{1}{1-\epsilon+\epsilon^2}\begin{bmatrix}2\epsilon-1\\-2\epsilon+1\end{bmatrix}\\&= \frac{1-2\epsilon}{1-\epsilon+\epsilon^2}\begin{bmatrix}-1\\1\end{bmatrix}\end{aligned}$$

Now, excess risk is computed as follows:

$$\begin{aligned}r^t(\beta_*^s) - r^t(\beta_*^t) &= (\beta_*^s-\beta_*^t)^\top \mathbb{E}[\widetilde{X}^{t\top}\widetilde{X}^t](\beta_*^s-\beta_*^t)\\&= \frac{(1-2\epsilon)^2}{(1-\epsilon+\epsilon^2)^2}\begin{bmatrix}-1\\1\end{bmatrix}^\top \begin{bmatrix}1-\epsilon & \epsilon(1-\epsilon)\\\epsilon(1-\epsilon) & \epsilon\end{bmatrix}\cdot\sigma_Z^2\cdot\begin{bmatrix}-1\\1\end{bmatrix}\\&= \frac{\sigma_Z^2(1-2\epsilon)^2(1-2\epsilon+2\epsilon^2)}{(1-\epsilon+\epsilon^2)^2}\end{aligned}$$

As $\epsilon \to 0$, we can see that $r^t(\beta_*^s) - r^t(\beta_*^t) \to \sigma_Z^2$.

Additionally, we can compute the excess risk of the constant zero classifier:

$$\begin{aligned}r^t([0,0]) - r^t(\beta_*^t) &= \beta_*^{t\top}\mathbb{E}[\widetilde{X}^{t\top}\widetilde{X}^t]\beta_*^t\\&= \frac{1}{(1-\epsilon+\epsilon^2)^2}\begin{bmatrix}1-\epsilon\\\epsilon\end{bmatrix}^\top \begin{bmatrix}1-\epsilon & \epsilon(1-\epsilon)\\\epsilon(1-\epsilon) & \epsilon\end{bmatrix}\cdot\sigma_Z^2\cdot\begin{bmatrix}1-\epsilon\\\epsilon\end{bmatrix}\\&= \frac{\sigma_Z^2}{(1-\epsilon+\epsilon^2)^2}\begin{bmatrix}(1-\epsilon)^2+\epsilon^2(1-\epsilon)\\\epsilon(1-\epsilon)^2+\epsilon^2\end{bmatrix}^\top\begin{bmatrix}1-\epsilon\\\epsilon\end{bmatrix}\\&= \frac{\sigma_Z^2(1-\epsilon+\epsilon^2)}{(1-\epsilon+\epsilon^2)^2}\begin{bmatrix}(1-\epsilon)\\\epsilon\end{bmatrix}^\top\begin{bmatrix}1-\epsilon\\\epsilon\end{bmatrix}\\&= \frac{\sigma_Z^2(1-\epsilon+\epsilon^2)}{(1-\epsilon+\epsilon^2)^2}\left[(1-\epsilon)^2+\epsilon^2\right]\\&= \frac{\sigma_Z^2(1-2\epsilon+2\epsilon^2)}{1-\epsilon+\epsilon^2}\end{aligned}$$

As $\epsilon \to 0$, we can see that $r^t([0,0]) - r^t(\beta_*^t) \to \sigma_Z^2$. □

Example 2 (Confounded Features)

Now, suppose that $m_s = [0, 0]$ and $m_t = [1, 0]$. For some constants a, b, c consider the following data generating process:

$$\begin{aligned} X_1 &= \nu_1 & \nu_1 &\sim \mathcal{N}(0,1) \\ X_2 &= aX_1 + \nu_2 & \nu_2 &\sim \mathcal{N}(0,1) \\ Y &= bX_1 + cX_2 + \nu_Y & \nu_Y &\sim \mathcal{N}(0,1). \end{aligned}$$

We will show that the optimal source and target predictors are $\beta^s_* = [b, c]$ and $\beta^t_* = [0, \frac{ab}{a^2+1} + c]$. By setting $a = -\frac{b}{c}$, we will show that for any $\tau > 1$, there exists values of a, b, c such that $r^t(\beta^s_*) > \tau\text{Var}(Y)$.

Proof. First, we compute β^s_* (where $m_s = [0, 0]$):

$$\begin{aligned} \mathbb{E}[\widetilde{X}^{s\top}\widetilde{X}^s] &= \mathbb{E}\left[X^\top X\right] \\ &= \begin{bmatrix} 1 & a \\ a & a^2+1 \end{bmatrix} \\ \mathbb{E}[\widetilde{X}^{s\top}\widetilde{X}^s]^{-1} &= \begin{bmatrix} a^2+1 & -a \\ -a & 1 \end{bmatrix} \\ \mathbb{E}[\widetilde{X}^{s\top}Y] &= \mathbb{E}[X^\top Y] \\ &= \begin{bmatrix} b + ac \\ ab + a^2c + c \end{bmatrix} \\ \beta^s_* &= \mathbb{E}[\widetilde{X}^{s\top}\widetilde{X}^s]^{-1}\mathbb{E}[\widetilde{X}^{s\top}Y] \\ &= \begin{bmatrix} a^2+1 & -a \\ -a & 1 \end{bmatrix} \cdot \begin{bmatrix} b + ac \\ ab + a^2c + c \end{bmatrix} \\ &= \begin{bmatrix} b \\ c \end{bmatrix}. \end{aligned}$$

Thus, $\beta^s_* = [b, c]$.

Now, let us compute β^t_* (where $m_t = [1, 0]$). Since X_1 is entirely missing and X_2 is completely observed, we only regress on X_2:

$$\begin{aligned} \mathbb{E}[\widetilde{X}^{t\top}_2\widetilde{X}^t_2] &= \mathbb{E}[X_2^\top X_2] \\ &= (a^2+1) \\ \mathbb{E}[\widetilde{X}^{t\top}_2\widetilde{X}^t_2]^{-1} &= \frac{1}{a^2+1} \\ \mathbb{E}[\widetilde{X}^{t\top}_2 Y] &= ab + a^2c + c \\ \mathbb{E}[\widetilde{X}^{t\top}_2\widetilde{X}^t_2]^{-1}\mathbb{E}[\widetilde{X}^{t\top}_2 Y] &= \frac{ab + a^2c + c}{a^2+1} \\ &= \frac{ab}{a^2+1} + c. \end{aligned}$$

Thus, $\beta_*^t = \left[0, \frac{ab}{a^2+1} + c\right]$.

Now, let us compute $\mathrm{Var}(Y)$. Note that $\mathbb{E}[Y] = 0$, so $\mathrm{Var}(Y) = \mathbb{E}[Y^2]$. Also, note that ν_1, ν_2, ν_Y are independent:

$$\begin{aligned}
\mathrm{Var}(Y) &= \mathrm{Var}(bX_1 + cX_2 + \nu_Y) \\
&= \mathbb{E}[(b\nu_1 + c(a\nu_1 + \nu_2) + \nu_Y)^2] \\
&= (b + ac)^2 + c^2 + 1 \\
&= b^2 + 2abc + a^2c^2 + c^2 + 1.
\end{aligned}$$

Thus, $\mathrm{Var}(Y) = b^2 + 2abc + a^2c^2 + c^2 + 1$.

Now, let us compute $r^t(\beta_*^s)$. Let $[\beta_*^s]_2$ denote the second dimension of β_*^s. We have:

$$\begin{aligned}
r^t(\beta_*^s) &= \mathbb{E}[(Y - \widetilde{X}_2^t[\beta_*^s]_2)^2] \\
&= \mathbb{E}[Y^2] - 2\mathbb{E}[\widetilde{X}_2^t[\beta_*^s]_2 Y] + \mathbb{E}[(\widetilde{X}_2^t[\beta_*^s]_2)^2] \\
&= \mathrm{Var}[Y^2] - 2\mathbb{E}[X_2[\beta_*^s]_2 Y] + \mathbb{E}[(X_2[\beta_*^s]_2)^2] \\
&= \mathrm{Var}[Y^2] - 2\mathbb{E}[(a\nu_1 + \nu_2)c(b\nu_1 + c(a\nu_1 + \nu_2) + \nu_Y)] + \mathbb{E}[((a\nu_1 + \nu_2)c)^2] \\
&= b^2 + 2abc + a^2c^2 + c^2 + 1 - 2[ac(b + ac) + c^2] + [a^2c^2 + c^2] \\
&= b^2 + 1.
\end{aligned}$$

Thus, we have $\frac{r^t(\beta_*^s)}{\mathrm{Var}(Y)} = \frac{b^2+1}{b^2+2abc+a^2c^2+c^2+1}$. If we set $a = -\frac{b}{c}$, then we have:

$$\begin{aligned}
\frac{r^t(\beta_*^s)}{\mathrm{Var}(Y)} &= \frac{b^2 + 1}{b^2 + 2abc + a^2c^2 + c^2 + 1} \\
&= \frac{b^2 + 1}{b^2 - 2b^2 + b^2 + c^2 + 1} \\
&= \frac{b^2 + 1}{c^2 + 1}.
\end{aligned}$$

Now suppose that for some $\tau > 1$, we would like $r^t(\beta_*^s) > \tau\mathrm{Var}(Y)$. Then, it is easy to see that we can simply choose b large enough, c small enough, and $a = -\frac{b}{c}$, such that $\frac{b^2+1}{c^2+1} > \tau$. □

D.2 DAMS with Indicators as an Instance of Covariate Shift

This section contains a proof of Proposition 1: Assume we observe ξ. Let us consider an augmented set of covariates $\tilde{x}' = (\tilde{x}, \xi)$. When ξ is drawn independently of other covariates or depending only on other completely observed covariates, we will show that missingness shift satisfies the covariate shift assumption, i.e, $P^s(Y|\widetilde{X}' = \tilde{x}') = P^t(Y|\widetilde{X}' = \tilde{x}')$.

First, let us formalize what it means for ξ to be drawn independently of other covariates or depending only on other completely observed covariates:

(a) **Independent of other covariates** When ξ is drawn independently of other covariates, as described in the DAMS with UCAR setup (Section 5.3), we have that $\xi \sim \text{Bernoulli}(1-m)$ for some constant vector of missingness rates $m \in [0,1]^d$.

(b) **Depending only on other completely observed covariates** Now, suppose that some subset of covariates $X_c \subseteq X$ is completely observed (i.e. no missingness), and the missingness of the other covariates $X_m = X \setminus X_c$ depends on X_c. That is, $\xi \sim \text{Bernoulli}(f(X_c))$ for some function $f : \mathbb{R}^{|X_c|} \to [0,1]^{|X_m|}$.

Since (b) is more general than (a), we adopt notation from (b) throughout our proof, and then argue why it also holds for (a).

Proof. Consider some augmented set of covariates taking values $\tilde{x}' = (\tilde{x}_m, \xi, x_c)$. To prove that the covariate shift assumption holds, let us start by considering the left-hand side of the equation. Applying Bayes' Rule, we have:

$$P^s(Y|\widetilde{X}' = \widetilde{x}') = P^s(Y|\widetilde{X}^s_m = \widetilde{x}_m, \xi^s = \xi, X_c = x_c) = \frac{P^s(Y, \widetilde{X}^s_m = \widetilde{x}_m, \xi^s = \xi, X_c = x_c)}{\sum_y P^s(Y = y, \widetilde{X}^s_m = \widetilde{x}_m, \xi^s = \xi, X_c = x_c)}$$

We can rewrite the numerator as follows:

$$\begin{aligned}
P^s(Y, \widetilde{X}^s_m = \widetilde{x}_m, \xi^s = \xi, X_c = x_c) &= \sum_{x_m : x_m \odot \xi = \widetilde{x}_m} P(Y, \widetilde{X}^s_m = \widetilde{x}_m, \xi^s = \xi, X_m = x_m, X_c = x_c) \\
&= \sum_{x_m : x_m \odot \xi = \widetilde{x}_m} P(Y, \xi^s = \xi, X_m = x_m, X_c = x_c) \\
&= \sum_{x_m : x_m \odot \xi = \widetilde{x}_m} P(\xi^s = \xi | Y, X_m = x_m, X_c = x_c) \cdot P(Y, X_m = x_m, X_c = x_c) \\
&= \sum_{x_m : x_m \odot \xi = \widetilde{x}_m} P(\xi^s = \xi | X_c = x_c) \cdot P(Y, X_m = x_m, X_c = x_c) \\
&= P(\xi^s = \xi | X_c = x_c) \sum_{x_m : x_m \odot \xi = \widetilde{x}_m} P(Y, X_m = x_m, X_c = x_c),
\end{aligned}$$

where the first line follows from marginalizing over all possible values of X_m, the second line comes from the fact that $\widetilde{x}_m$ is determined given x_m and ξ, the third line comes from Bayes'

Rule, the fourth line comes the fact that ξ only depends on X_c, and the last line comes from pulling the first term out of the summation.

Plugging back into the expression for $P^s(Y|\widetilde{X}' = \widetilde{x}')$, we have:

$$\begin{aligned}
P^s(Y|\widetilde{X}' = \widetilde{x}') &= \frac{P^s(Y, \widetilde{X}^s_m = \widetilde{x}_m, \xi^s = \xi, X_c = x_c)}{\sum_y P^s(Y = y, \widetilde{X}^s_m = \widetilde{x}_m, \xi^s = \xi, X_c = x_c)} \\
&= \frac{P(\xi^s = \xi | X_c = x_c) \sum_{x_m : x_m \odot \xi = \widetilde{x}_m} P(Y, X_m = x_m, X_c = x_c)}{\sum_y P(\xi^s = \xi | X_c = x_c) \sum_{x_m : x_m \odot \xi = \widetilde{x}_m} P(Y = y, X_m = x_m, X_c = x_c)} \\
&= \frac{\sum_{x_m : x_m \odot \xi = \widetilde{x}_m} P(Y, X_m = x_m, X_c = x_c)}{\sum_{x_m : x_m \odot \xi = \widetilde{x}_m} P(Y = y, X_m = x_m, X_c = x_c)},
\end{aligned}$$

which does not contain source-specific quantities (everything is in terms of the underlying distribution). By the same logic,

$$P^t(Y|\widetilde{X}' = \widetilde{x}') = \frac{\sum_{x_m : x_m \odot \xi = \widetilde{x}_m} P(Y, X_m = x_m, X_c = x_c)}{\sum_{x_m : x_m \odot \xi = \widetilde{x}_m} P(Y = y, X_m = x_m, X_c = x_c)}.$$

Thus, $P^s(Y|\widetilde{X}' = \widetilde{x}') = P^t(Y|\widetilde{X}' = \widetilde{x}')$ as desired. When ξ is instead drawn independently of other covariates, as in (a) above, we note that all of the steps of the proof follow through simply by removing X_c. Additionally, while all of the above expressions apply to discrete X, extension to continuous X is straightforward (e.g. replace summations with integrals, and constants with sets or intervals). □

D.3 Constant Missingness as L2 Regularization

This section contains a proof of Theorem 5.4.1. This proof is based off of that presented in Wager et al. (2013)'s work showing dropout to be a form of adaptive regularization. Instead of assuming a single constant dropout rate across all covariates, however, our proof extends to varying rates of missingness (i.e. different constant dropout rates) for different covariates.

Proof. Assume we know the constant missingness rates m. For mathematical convenience, we preprocess $\widetilde{x}$ by multiplying each dimension by the corresponding $\frac{1}{1-m_j}$. For the remainder of this derivation, this preprocessed data is referred to as $\widetilde{x}$.

Similar to Wager et al. (2013), we start with an analysis of generalized linear models and then consider the case of linear regression. Minimizing the expected negative log likelihood $l_{\widetilde{x}^{(i)},y^{(i)}}(\beta)$ of a generalized linear model $p_\beta(y|x) = h(y) \exp\{yx \cdot \beta - A(x \cdot \beta)\}$, we have:

$$\widehat{\beta} = \arg \min_{\beta \in \mathbb{R}^d} \sum_{i=1}^{n} \mathbb{E}_\xi [l_{\widetilde{x}^{(i)},y^{(i)}}(\beta)]$$

$$\begin{aligned}
\sum_{i=1}^{n} \mathbb{E}_\xi[l_{\widetilde{x}^{(i)},y^{(i)}}(\beta)] &= \sum_{i=1}^{n} \mathbb{E}_\xi[-\log p_\beta(y^{(i)}|\widetilde{x}^{(i)})] \\
&= \sum_{i=1}^{n} \mathbb{E}_\xi[-(\log h(y^{(i)}) + y^{(i)}\widetilde{x}^{(i)}\beta - A(\widetilde{x}^{(i)} \cdot \beta))] \\
&= \sum_{i=1}^{n} -\log h(y^{(i)}) - y^{(i)}\mathbb{E}_\xi[\widetilde{x}^{(i)}]\beta + \mathbb{E}_\xi[A(\widetilde{x}^{(i)} \cdot \beta)] \\
&= \sum_{i=1}^{n} -\log h(y^{(i)}) - y^{(i)}\left(x^{(i)} \odot \frac{1-m}{1-m}\right)\beta + \mathbb{E}_\xi[A(\widetilde{x}^{(i)} \cdot \beta)] \\
&= \sum_{i=1}^{n} -(\log h(y^{(i)}) + y^{(i)}x^{(i)}\beta - A(x^{(i)}\beta)) - A(x^{(i)}\beta) + \mathbb{E}_\xi[A(\widetilde{x}^{(i)} \cdot \beta)] \\
&= \sum_{i=1}^{n} l_{x^{(i)},y^{(i)}}(\beta) + \mathbb{E}_\xi[A(\widetilde{x}^{(i)} \cdot \beta)] - A(x^{(i)}\beta) \\
&= \sum_{i=1}^{n} l_{x^{(i)},y^{(i)}}(\beta) + R(\beta)
\end{aligned}$$

where $R(\beta) \triangleq \sum_{i=1}^{n} \mathbb{E}_\xi[A(\widetilde{x}^{(i)} \cdot \beta)] - A(x^{(i)}\beta)$. How do we interpret $R(\beta)$?

First, we do a second order Taylor expansion of A around $x\beta$. Note that linear regression has a second order log partition function. Thus, for linear regression this expansion is exact:

$$\begin{aligned}
A(y) &\approx A(x\beta) + A'(x\beta)(y - x\beta) + \frac{1}{2}A''(x\beta)(y - x\beta)^2 \\
A(\widetilde{x}\beta) &\approx A(x\beta) + A'(x\beta)(\widetilde{x}\beta - x\beta) + \frac{1}{2}A''(x\beta)(\widetilde{x}\beta - x\beta)^2 \\
&= A(x\beta) + A'(x\beta)(\widetilde{x} - x)\beta + \frac{1}{2}A''(x\beta)(\widetilde{x}\beta - x\beta)^2
\end{aligned}$$

Now, we can compute the first term of $R(\beta)$:

$$\begin{aligned}
\mathbb{E}_\xi[A(\widetilde{x} \cdot \beta)] &\approx \mathbb{E}_\xi[A(x\beta)] + \mathbb{E}_\xi[A'(x\beta)(\widetilde{x} - x)\beta] + \mathbb{E}_\xi[\frac{1}{2}A''(x\beta)(\widetilde{x}\beta - x\beta)^2] \\
&= A(x\beta) + 0 + \frac{1}{2}A''(x\beta)\mathbb{E}_\xi[(\widetilde{x}\beta - x\beta)^2] \\
&= A(x\beta) + \frac{1}{2}A''(x\beta)\text{Var}_\xi(\widetilde{x}\beta)
\end{aligned}$$

where the second step follows because $\mathbb{E}_\xi[\widetilde{x}] = x$. Thus, $R(\beta)$ is given by:

$$\begin{aligned}
R(\beta) &= \sum_{i=1}^{n} \mathbb{E}_\xi[A(\widetilde{x}^{(i)} \cdot \beta)] - A(x^{(i)}\beta) \\
&\approx \sum_{i=1}^{n} A(x^{(i)}\beta) + \frac{1}{2}A''(x^{(i)}\beta)\text{Var}_\xi(\widetilde{x}^{(i)}\beta) - A(x^{(i)}\beta)
\end{aligned}$$

$$
\begin{aligned}
&= \sum_{i=1}^{n} \frac{1}{2} A''(x^{(i)}\beta)\text{Var}_\xi(\widetilde{x}^{(i)}\beta) \\
&\triangleq R^q(\beta).
\end{aligned}
$$

Note that the first term corresponds to variance of $y^{(i)}$, and the second term corresponds to the variance of the estimated GLM parameter due to noising, or in the linear case, $\text{Var}(y^{(i)})$. Additionally, note that for linear regression $R(\beta) = R^q(\beta)$ since the approximate equality comes from the Taylor series approximation.

Analyzing $\text{Var}_\xi(\widetilde{x}^{(i)}\beta)$,

$$
\begin{aligned}
\text{Var}_\xi(\widetilde{x}^{(i)}\beta) &= \sum_{j=1}^{d} \text{Var}_\xi(\widetilde{x}_j^{(i)}\beta_j) \\
&= \sum_{j=1}^{d} \text{Var}_\xi\left(\frac{x_j^{(i)}}{1-m_j} \cdot b_j \cdot \beta_j\right) \\
&= \sum_{j=1}^{d} \left(\frac{x_j^{(i)}}{1-m_j}\right)^2 \beta_j^2 (1-m_j)(m_j) \\
&= \sum_{j=1}^{d} \frac{m_j}{1-m_j} \left(x_j^{(i)}\right)^2 \beta_j^2
\end{aligned}
$$

where $b_j \sim$ Bernoulli$(1-m_j)$. Thus, $R^q(\beta)$ is given by:

$$
R^q(\beta) = \frac{1}{2} \sum_{i=1}^{n} A''(x^{(i)}\beta) \sum_{j=1}^{d} \frac{m_j}{1-m_j} \left(x_j^{(i)}\right)^2 \beta_j^2.
$$

Let $V(\beta) \in \mathbb{R}^{n\times n}$ be diagonal with entries $A''(x^{(i)}\beta)$, and $X \in \mathbb{R}^{n\times d}$ be the design matrix with rows $x^{(i)}$. For linear regression, $V(\beta)$ is given by the identity matrix. Then, we can rewrite $R^q(\beta)$ as:

$$
\begin{aligned}
R^q(\beta) &= \frac{1}{2}\left(\beta \odot \sqrt{\frac{m}{1-m}}\right)^\top \text{diag}(X^\top V(\beta) X) \left(\beta \odot \sqrt{\frac{m}{1-m}}\right) \\
R^q(\beta) &= \frac{1}{2}\left(\beta \odot \frac{m}{1-m}\right)^\top \text{diag}(I) \left(\beta \odot \frac{m}{1-m}\right) \\
&= \frac{1}{2}\left(\text{diag}(I)^{1/2}\beta \odot \frac{m}{1-m}\right)^\top \left(\text{diag}(I)^{1/2}\beta \odot \frac{m}{1-m}\right) \\
&= \frac{1}{2}\left(\beta\widetilde{\Delta}_{\text{diag}}\right)^\top \left(\beta\widetilde{\Delta}_{\text{diag}}\right)
\end{aligned}
$$

where $\widetilde{\Delta}_{\text{diag}} = \text{diag}\left(\sqrt{\frac{m}{1-m}}\right)\text{diag}(I)^{1/2}$, where $\text{diag}\left(\sqrt{\frac{m}{1-m}}\right)$ refers to a diagonal matrix with the vector quantities on the diagonal, and $\text{diag}(I)^{1/2}$ refers to the square root of the diagonal of the Fisher information matrix. Thus, for linear regression, applying missingness rates $m \in [0,1]^d$ to data scaled by $\frac{1}{1-m}$ can be viewed as an attempt to apply L2 regularization of β scaled by $\widetilde{\Delta}_{\text{diag}}$. □

D.4 Identification of Clean Distribution from Corrupted Distribution

This section proves Lemma 5.5.1, which states that the clean distribution p is identified from the corrupted distribution $\widetilde{p}$ given missingness rates m, and $m \prec 1$.

Proof. Let $\mathcal{A}^k$ denote the set of possible values of x where at most k of the dimensions of x are 0. We would like to show that $\forall k \in \{0, 1, ..., d\}$, $\forall a \in \mathcal{A}^k$, the clean distribution $p_{a,y}$ is identifiable (and hence $p_{x,y}$ is identifiable) for all values of x and y. We proceed with a proof by induction on k.

- *Base case ($k = 0$)*:

  Consider $\mathcal{A}^0$, the set of possible values of x where none of the dimensions of x are 0. For any subset $a \subseteq \mathcal{A}^0$, we can write:

  $$\widetilde{p}_{a,y} = \prod_{j=1}^{d} (1 - m_j) p_{a,y}$$

  which can be rearranged to recover p_a from $\widetilde{p}_a$ and m, which are both known:

  $$p_{a,y} = \prod_{j=1}^{d} \frac{1}{1 - m_j} \widetilde{p}_{a,y}.$$

  Thus $p_{a,y}$ is identified for $a \subseteq \mathcal{A}^0$.

- *Inductive Step*: Assume $p_{a,y}$ is identified for $a \subseteq \mathcal{A}^k$. Consider some $a' \subseteq \mathcal{A}^{k+1}$. Using equation (5.1), we have:

  $$\begin{aligned}\widetilde{p}_{a',y} &= \sum_{b: b \rightsquigarrow a'} p_{b,y} \cdot \prod_{j=1}^{d} (1 - m_j)^{[a'_j]_{\neq 0}} m_j^{[b_j]_{\neq 0} - [a'_j]_{\neq 0}} \\ &= p_{a',y} \cdot \prod_{j=1}^{d} (1 - m_j)^{[a'_j]_{\neq 0}} + \sum_{\substack{b: b \rightsquigarrow a', \\ b \neq a'}} p_{b,y} \cdot \prod_{j=1}^{d} (1 - m_j)^{[a'_j]_{\neq 0}} m_j^{[b_j]_{\neq 0} - [a'_j]_{\neq 0}}\end{aligned}$$

  Recall from Remark 3 that if $b \rightsquigarrow a'$, then the dimensions of b that are 0 must be a subset of the ones that are 0 in a'. Additionally, any dimensions that are nonzero in both b and

a' must match in value. This implies that if there are the same number of zeros in b and a', then $b = a'$. The remaining b where $b \rightsquigarrow a'$ have at least one less zero than a'. Thus, the set of $\{b : b \rightsquigarrow a', b \neq a'\} \in \mathcal{A}^k$, and by our inductive hypothesis, $p_{b,y}$ are identified when $b \in \mathcal{A}^k$. As a result, we can identify the second term in the equation above (the summation over b's), and rearranging the equation, we can identify $p_{a',y}$ as $\widetilde{p}$ and m are known.

Thus, by the principle of mathematical induction, p_a is identified for $a \in \mathcal{A}^k$, $\forall k \in \{0, 1, ..., d\}$. Therefore, given m, we have identified the clean distribution from the corrupted distribution. Additionally, while all of the above expressions apply to discrete X, extension to continuous X is straightforward (e.g. replace summations with integrals, and constants with sets or intervals). □

D.5 Identification of Labeled Target Distribution from the Labeled Source Distribution

Here we prove Theorem 5.5.2, which states that:

$$\widetilde{p}^{t}_{x,y} = \sum_{z:z\rightsquigarrow x} \widetilde{p}^{s}_{z,y} \cdot \prod_{j=1}^{d} (1 - r_j^{s\to t})^{[x_j]\neq 0} (r_j^{s\to t})^{[z_j]\neq 0 - [x_j]\neq 0}$$

Proof. Applying equation (5.1), the corrupted source and target distributions can be written as:

$$\widetilde{p}^{s}_{a,y} = \sum_{b:b\rightsquigarrow a} p_{b,y} \cdot \prod_{j=1}^{d} (1 - m_{sj})^{[a_j]\neq 0} m_{sj}^{[b_j]\neq 0 - [a_j]\neq 0}$$

$$\widetilde{p}^{t}_{a,y} = \sum_{c:c\rightsquigarrow a} p_{c,y} \cdot \prod_{j=1}^{d} (1 - m_{tj})^{[a_j]\neq 0} m_{tj}^{[c_j]\neq 0 - [a_j]\neq 0}$$

We apply relative missingness $r = r^{s\to t} = \frac{m_t - m_s}{1 - m_s}$ to source distribution $\widetilde{p}^{s}$, denoting this new distribution as $\widetilde{p}^{s\to t}$:

$$\begin{aligned}
\widetilde{p}^{s\to t}_{a,y} &= \sum_{b:b\rightsquigarrow a} \widetilde{p}^{s}_{b,y} \cdot \prod_{j=1}^{d} (1 - r_j)^{[a_j]\neq 0} r_j^{[b_j]\neq 0 - [a_j]\neq 0} \\
&= \sum_{b:b\rightsquigarrow a} \sum_{c:c\rightsquigarrow b} p_{c,y} \cdot \prod_{j=1}^{d} (1 - m_{sj})^{[b_j]\neq 0} m_{sj}^{[c_j]\neq 0 - [b_j]\neq 0} \cdot \prod_{j=1}^{d} (1 - r_j)^{[a_j]\neq 0} r_j^{[b_j]\neq 0 - [a_j]\neq 0} \\
&= \sum_{c:c\rightsquigarrow b} p_{c,y} \sum_{b:b\rightsquigarrow a} \cdot \prod_{j=1}^{d} (1 - m_{sj})^{[b_j]\neq 0} m_{sj}^{[c_j]\neq 0 - [b_j]\neq 0} \cdot \prod_{j=1}^{d} (1 - r_j)^{[a_j]\neq 0} r_j^{[b_j]\neq 0 - [a_j]\neq 0}
\end{aligned}$$

$$
\begin{aligned}
&= \sum_{c:c\rightsquigarrow b} p_{c,y} \sum_{b:b\rightsquigarrow a} \cdot \prod_{j=1}^{d} (1-m_{sj})^{[b_j]_{\neq 0}} m_{sj}^{[c_j]_{\neq 0}-[b_j]_{\neq 0}} \cdot \prod_{j=1}^{d} \left(\frac{1-m_{tj}}{1-m_{sj}}\right)^{[a_j]_{\neq 0}} \left(\frac{m_{tj}-m_{sj}}{1-m_{sj}}\right)^{[b_j]_{\neq 0}-[a_j]_{\neq 0}} \\
&= \sum_{c:c\rightsquigarrow b} p_{c,y} \sum_{b:b\rightsquigarrow a} \prod_{j=1}^{d} (1-m_{sj})^{\mathbb{1}\left\{[c_j]_{\neq 0}=[b_j]_{\neq 0}=1,[a_j]_{\neq 0}=0\right\}+\mathbb{1}\left\{[c_j]_{\neq 0}=[b_j]_{\neq 0}=[a_j]_{\neq 0}=1\right\}} \\
&\qquad\qquad\qquad \cdot m_{sj}^{\mathbb{1}\left\{[c_j]_{\neq 0}=1,[b_j]_{\neq 0}=[a_j]_{\neq 0}=0\right\}} \\
&\qquad\qquad\qquad \cdot \left(\frac{1-m_{tj}}{1-m_{sj}}\right)^{\mathbb{1}\left\{[c_j]_{\neq 0}=[b_j]_{\neq 0}=[a_j]_{\neq 0}=1\right\}} \\
&\qquad\qquad\qquad \cdot \left(\frac{m_{tj}-m_{sj}}{1-m_{sj}}\right)^{\mathbb{1}\left\{[c_j]_{\neq 0}=[b_j]_{\neq 0}=1,[a_j]_{\neq 0}=0\right\}} \\
&= \sum_{c:c\rightsquigarrow b} p_{c,y} \sum_{b:b\rightsquigarrow a} \prod_{j=1}^{d} m_{sj}^{\mathbb{1}\left\{[c_j]_{\neq 0}=1,[b_j]_{\neq 0}=[a_j]_{\neq 0}=0\right\}} \\
&\qquad\qquad\qquad \cdot (1-m_{tj})^{\mathbb{1}\left\{[c_j]_{\neq 0}=[b_j]_{\neq 0}=[a_j]_{\neq 0}=1\right\}} \\
&\qquad\qquad\qquad \cdot (m_{tj}-m_{sj})^{\mathbb{1}\left\{[c_j]_{\neq 0}=[b_j]_{\neq 0}=1,[a_j]_{\neq 0}=0\right\}} \\
&= \sum_{c:c\rightsquigarrow a} p_{c,y} \cdot \left(\prod_{j:[c_j]_{\neq 0}=[a_j]_{\neq 0}=1} 1-m_{tj}\right) \cdot \left(\prod_{j:[c_j]_{\neq 0}=[a_j]_{\neq 0}=0} 1\right) \\
&\qquad\qquad \cdot \sum_{b:b\rightsquigarrow a} \left(\prod_{j:[c_j]_{\neq 0}=1,[a_j]_{\neq 0}=0} m_{sj}^{1-[b_j]_{\neq 0}} (m_{tj}-m_{sj})^{[b_j]_{\neq 0}}\right) \\
&= \sum_{c:c\rightsquigarrow a} p_{c,y} \cdot \left(\prod_{j:[c_j]_{\neq 0}=[a_j]_{\neq 0}=1} 1-m_{tj}\right) \cdot \left(\prod_{j:[c_j]_{\neq 0}=[a_j]_{\neq 0}=0} 1\right) \\
&\qquad\qquad \cdot \sum_{[b]_{\neq 0}\in\{0,1\}^d} \left(\prod_{j:[c_j]_{\neq 0}=1,[a_j]_{\neq 0}=0} m_{sj}^{1-[b_j]_{\neq 0}} (m_{tj}-m_{sj})^{[b_j]_{\neq 0}}\right) \\
&= \sum_{c:c\rightsquigarrow a} p_{c,y} \cdot \left(\prod_{j:[c_j]_{\neq 0}=[a_j]_{\neq 0}=1} 1-m_{tj}\right) \cdot \left(\prod_{j:[c_j]_{\neq 0}=[a_j]_{\neq 0}=0} 1\right) \cdot \left(\prod_{j:[c_j]_{\neq 0}=1,[a_j]_{\neq 0}=0} m_{t_j}\right) \\
&= \sum_{c:c\rightsquigarrow a} p_{c,y} \prod_{j=1}^{d} (1-m_{tj})^{[a_j]_{\neq 0}} m_{tj}^{[c_j]_{\neq 0}-[a_j]_{\neq 0}} \\
&= \widetilde{p}_{a,y}^{t}
\end{aligned}
$$

as desired. The steps are explained in words below:

- Plug in equation for corrupted source distribution.
- Switch summation order and factor out $p_{c,y}$.

- Plug in for r.
- Note that $[c_j]_{\neq 0} - [b_j]_{\neq 0} = 1$ only if $[c_j]_{\neq 0} = 1$ and $[b_j]_{\neq 0} = 0$. Use similar reasoning for the remaining, keeping in mind that $[c]_{\neq 0} \succeq [b]_{\neq 0} \succeq [a]_{\neq 0}$. Simplify.
- Since all elements of the sum have $\mathbb{1}\{[c]_{\neq 0} \succeq [b]_{\neq 0} \succeq [a]_{\neq 0}\}$, it is also true that $\mathbb{1}\{[c]_{\neq 0} \succeq [a]_{\neq 0}\}$.
- If $[a_i]_{\neq 0} = [c_i]_{\neq 0} = 1$, then $[b_i]_{\neq 0} = 1$ necessarily.
- Note that if $c \rightsquigarrow b \rightsquigarrow a$ and $[c_i]_{\neq 0} = 1, [a_i]_{\neq 0} = 0$, then $\forall i, b_i \in \{0, c_i\}$. We can then perform a change of variables in the summation, now summing over $[b]_{\neq 0} \in \{0,1\}^d$ instead.
- We use the following identity for arbitrary d-dimensional vectors a and b:

  $$\sum_{u \in \{0,1\}^d} \prod_j a_j^{u_j} b_j^{1-u_j} = \prod_j (a_j + b_j)$$

  To gain intuition for why this is the case, let's start with $d = 2$:

  $$\begin{aligned} LHS &= \sum_{u \in \{0,1\}^d} \prod_j a_j^{u_j} b_j^{1-u_j} \\ &= \sum_{u \in \{0,1\}^2} a_1^{u_1} b_1^{1-u_1} a_2^{u_2} b_2^{(1-u_2)} \\ &= \sum_{u \in [(1,1),(1,0),(0,1),(0,0)]} a_1^{u_1} b_1^{1-u_1} a_2^{u_2} b_2^{(1-u_2)} \\ &= a_1 a_2 + a_1 b_2 + b_1 a_2 + b_1 b_2 \\ RHS &= \prod_j (a_j + b_j) \\ &= (a_1 + b_1)(a_2 + b_2) \\ &= a_1 a_2 + a_1 b_2 + b_1 a_2 + b_1 b_2 \end{aligned}$$

  Notice that the right-hand side is a product of sums $(a_j + b_j)$, of which there are d terms. When expanding this product of sums into a sum of products, each term in the sum of products will include either a_j or b_j for all $j \in 1, 2, ..., d$. Summing over all possible choices of either a_j or b_j for all j is then equivalent to summing over all possible values of a binary d-dimensional vector u. Thus, we get the left-hand side of the identity.
- The remaining steps are straightforward simplifications to get a form matching equation (5.3).
- Note that while all of the above expressions apply to discrete X, extension to continuous X is straightforward (e.g. replace summations with integrals, and constants with sets or intervals).

□

D.6 Error Bound for Estimating Non-Missing Proportions

This is a proof of Theorem 5.6.1. To estimate the non-missingness proportion $q = P(\widetilde{X} = 1)$ within ϵ of the true non-missingness proportion with probability at least $1-\delta$, we use Hoeffding's bound to show:

$$\begin{aligned}
P(|\widehat{q} - q| \geq \epsilon) &\leq 2\exp(-2ne^2) = \delta \\
\implies -2n\epsilon^2 &= \log(\delta/2) \\
\implies n &= \frac{\log(2/\delta)}{2\epsilon^2} \\
\implies |\widehat{q} - q| &= \sqrt{\frac{\log(2/\delta)}{2n}}.
\end{aligned}$$

Now, we show that with high probability, the estimate for $1 - r^{s\to t} = \frac{q_t}{q_s}$ is close to the true value. This part of the derivation is similar to that used in Garg et al., 2021. Using triangle inequality,

$$\begin{aligned}
\left|\frac{\widehat{q}_t}{\widehat{q}_s} - \frac{q_t}{q_s}\right| &= \left|\frac{q_s\widehat{q}_t - \widehat{q}_s q_t}{\widehat{q}_s q_s}\right| \\
&= \frac{1}{\widehat{q}_s q_s}\left|q_s\widehat{q}_t - q_s q_t + q_s q_t - \widehat{q}_s q_t\right| \\
&\leq \frac{1}{\widehat{q}_s q_s}\left|q_s\widehat{q}_t - q_s q_t\right| + \frac{1}{\widehat{q}_s q_s}\left|q_s q_t - \widehat{q}_s q_t\right| \\
&\leq \frac{1}{\widehat{q}_s}\left|\widehat{q}_t - q_t\right| + \frac{q_t}{\widehat{q}_s q_s}\left|q_s - \widehat{q}_s\right|.
\end{aligned}$$

On the right hand side, we use the union bound and plug in $\delta/2$ for δ in Hoeffding's bound. Plugging in, we then have that with probability at least $1 - \delta$,

$$\begin{aligned}
\left|\frac{\widehat{q}_t}{\widehat{q}_s} - \frac{q_t}{q_s}\right| &\leq \frac{1}{\widehat{q}_s}\left(\sqrt{\frac{\log(4/\delta)}{2n_t}} + \frac{q_t}{q_s}\sqrt{\frac{\log(4/\delta)}{2n_s}}\right) \\
\implies \left|\widehat{r}^{s\to t} - r^{s\to t}\right| &\leq \frac{1}{\widehat{P}^s(\widetilde{x} = 1)}\left(\sqrt{\frac{\log(4/\delta)}{2n_t}} + (1 - r^{s\to t})\sqrt{\frac{\log(4/\delta)}{2n_s}}\right).
\end{aligned}$$

D.7 Justification for the Non-parametric Procedure with Non-Negative Relative Missingness

Simple Justification Since (5.3) matches the form of (5.1) except with $m = r^{s\to t}$, applying missingness with rate $r^{s\to t}$ to the source distribution will yield samples independent and identically distributed to the target distribution. That is, plugging in $\widetilde{p}^s$ for p and $r^{s\to t}$ for m, we have:

$$\begin{aligned}
\widetilde{p}_{x,y} &= \sum_{z:z\rightsquigarrow x} p_{z,y} \cdot \prod_{j=1}^{d} (1-m_j)^{[x_j]\neq 0} m_j^{[z_j]\neq 0 - [x_j]\neq 0} \\
&= \sum_{z:z\rightsquigarrow x} \widetilde{p}^s_{z,y} \cdot \prod_{j=1}^{d} (1-r_j^{s\to t})^{[x_j]\neq 0} (r_j^{s\to t})^{[z_j]\neq 0 - [x_j]\neq 0} \\
&= \widetilde{p}^t_{x,y}
\end{aligned}$$

where the first line is (5.1) and the third line follows from (5.3).

Alternative Justification Suppose that $m^t \succeq m^s$, where $\succeq$ denotes whether all elements of m^t are greater than or equal to all corresponding elements of m^s, that is, $m^t_j \geq m^s_j$ for $j = 1, 2, ..., d$. Below, we show that the data generating process for the target data is equivalent to applying a missingness filter with relative missingness rate $r^{s\to t}$ applied to the source data. To draw a point from the source, target, and transformed distribution, respectively, one first draws a clean data point $(x, y) \sim P(X, Y)$, where $x \in \mathbb{R}^d, y \in \mathbb{R}$, and then applies the respective missingness filter to the clean covariates:

$$\begin{aligned}
\widetilde{x}^s &= \nu_s(x) = x \odot \xi^s \\
\widetilde{x}^t &= \nu_t(x) = x \odot \xi^t \\
\widetilde{x}^{s\to t} &= \nu_{s\to t}(\nu_s(x)) = x \odot \xi^s \odot \xi^{s\to t}
\end{aligned}$$

where $\xi^t \sim \text{Bernoulli}(1-m^t)$, $\xi^s \sim \text{Bernoulli}(1-m^s)$, and $\xi^{s\to t} \sim \text{Bernoulli}(1-r^{s\to t})$. Combining Bernoullis, we have:

$$\begin{aligned}
\xi^s \odot \xi^{s\to t} &= \begin{cases} 1 & \text{w.p. } \left(1 - \frac{m^t - m^s}{1-m^s}\right) \cdot (1-m^s) \\ 0 & otherwise \end{cases} \\
&= \begin{cases} 1 & \text{w.p. } (1-m^t) \\ 0 & otherwise \end{cases} = \xi^t
\end{aligned}$$

Thus, for true relative missing rates $r^{s\to t}$, we have $\nu_t(x) = \nu_{s\to t}(\nu_s(x))$. Since the data generating process after applying $\nu_{s\to t}$ to source data is now identical to the data generating process of the target dataset, we have $\{(\nu_{s\to t}(\widetilde{X}^{s,i}), Y^{s,i})\}_{i=1}^{n_s}$ drawn independent and identically distributed to $P^t(\widetilde{X}, Y)$.

D.8 Optimal Linear Predictors

D.8.1 Optimal linear target predictor, derived from target covariances

For each dimension j, the covariance between corrupted data $\widetilde{X}_j$ with missingness rate m and its labels Y is $\text{Cov}(\widetilde{X}_j, Y) = \text{Cov}(X_j \cdot \xi_j, Y) = (1 - m_j)\text{Cov}\left(X_j, Y\right)$. Thus,

$$\begin{aligned}
\text{Cov}(X, Y) &= \frac{1}{1-m} \odot \text{Cov}(\widetilde{X}, Y) \\
\mathbb{E}[X^\top Y] &= \text{Cov}\left(X, Y\right) + \mathbb{E}[X]^\top \mathbb{E}[Y] \\
&= \frac{1}{1-m} \odot \text{Cov}\left(\widetilde{X}, Y\right) + \frac{1}{1-m} \odot \mathbb{E}[\widetilde{X}]^\top \mathbb{E}[Y] \\
&= \frac{1}{1-m} \odot \mathbb{E}[\widetilde{X}^\top Y].
\end{aligned}$$

Plugging into the ordinary least squares regression solution,

$$\begin{aligned}
\beta_*^t &= \mathbb{E}[\widetilde{X}^{t^\top} \widetilde{X}^t]^{-1} \mathbb{E}[\widetilde{X}^{t\top} Y^t] \\
&= \mathbb{E}[\widetilde{X}^{t\top} \widetilde{X}^t]^{-1} \left((1 - m_t) \odot \mathbb{E}[X^\top Y]\right) \\
&= \mathbb{E}[\widetilde{X}^{t\top} \widetilde{X}^t]^{-1} \left(\frac{1 - m_t}{1 - m_s} \odot \mathbb{E}[\widetilde{X}^{s\top} Y^s]\right) \\
&= \mathbb{E}[\widetilde{X}^{t\top} \widetilde{X}^t]^{-1} \left(r^{s \to t} \odot \mathbb{E}[\widetilde{X}^{s\top} Y^s]\right).
\end{aligned}$$

The remainder of this section derives the optimal linear target predictor, where the corrupted target covariance is derived from the corrupted source covariance.

D.8.2 Means, Variances, and Covariances

This section begins by deriving the relationships between the means, covariances, and variances of the corrupted and clean data. Then, it derives the relationships between corrupted and clean $\mathbb{E}[X^\top X]$. Finally, the derived first and second moments are summarized in Table D.1.

Recall that for any covariate x_j, we have:

$$\begin{aligned}
\widetilde{x}_j &= \begin{cases} 0 & \text{w.p. } m_j \\ x_j & \text{w.p. } 1 - m_j \end{cases} \\
&= b_j x_j
\end{aligned}$$

where $b_j \sim \text{Bernoulli}(1 - m_j)$. The **mean** of the corrupted data is given by:

$$\mathbb{E}[\widetilde{X}] = (1 - m) \odot \mathbb{E}\left[X\right]$$

To derive the covariance matrix of the corrupted data, consider the covariance between two arbitrary distinct covariate dimensions $\widetilde{x}_1$ and $\widetilde{x}_2$. Let $A = b_1$, $B = x_1$, $C = b_2$, and $D = x_2$. Note that A and C are independent of all other variables. Thus,

$$\begin{aligned}
\text{Cov}(\widetilde{x}_1, \widetilde{x}_2) &= \text{Cov}(AB, CD) \\
&= \mathbb{E}[ABCD] - \mathbb{E}[AB]\mathbb{E}[CD] \\
&= \mathbb{E}[ABCD] - \mathbb{E}[A]\mathbb{E}[B]\mathbb{E}[C]\mathbb{E}[D] \\
&= \mathbb{E}[A]\mathbb{E}[C](\mathbb{E}[BD] - \mathbb{E}[B]\mathbb{E}[D]) \\
&= \mathbb{E}[A]\mathbb{E}[C]\text{Cov}(B, D) \\
&= (1 - m_1)(1 - m_2)\text{Cov}\left(x_1, x_2\right) \\
\implies \text{Cov}(x_1, x_2) &= \frac{1}{(1 - m_1)(1 - m_2)}\text{Cov}(\widetilde{x}_1, \widetilde{x}_2)
\end{aligned}$$

And similarly,

$$\begin{aligned}
\text{Cov}(\widetilde{x}_1, y) &= (1 - m_1)\text{Cov}\left(x_1, y\right) \\
\implies \text{Cov}\left(x_1, y\right) &= \frac{1}{1 - m_1}\text{Cov}(\widetilde{x}_1, y)
\end{aligned}$$

The variance (entries along the diagonal of the covariance matrix) is given by:

$$\begin{aligned}
\text{Var}(\widetilde{x}_1) &= \text{Var}\left(b_1 x_1\right) \\
&= \text{Var}(AB) \\
&= (\sigma_A^2 + \mu_A^2)(\sigma_B^2 + \mu_B^2) - \mu_A^2\mu_B^2 \\
&= \left(m_1(1 - m_1) + (1 - m_1)^2\right)\left(\text{Var}(x_1) + \mathbb{E}[x_1]^2\right) - (1 - m_1)^2\mathbb{E}[x_1]^2 \\
&= (1 - m_1)\left(\text{Var}(x_1) + \mathbb{E}[x_1]^2\right) - (1 - m_1)^2\mathbb{E}[x_1]^2 \\
&= (1 - m_1)\left(\text{Var}(x_1) + \mathbb{E}[x_1]^2 - (1 - m_1)\mathbb{E}[x_1]^2\right) \\
&= (1 - m_1)\left(\text{Var}(x_1) + \mathbb{E}[x_1]^2 - \mathbb{E}[x_1]^2 + m_1\mathbb{E}[x_1]^2\right) \\
&= (1 - m_1)\left(\text{Var}(x_1) + m_1\mathbb{E}[x_1]^2\right) \\
&= (1 - m_1)\text{Var}(x_1) + m_1(1 - m_1)\mathbb{E}[x_1]^2 \\
\text{Var}(x_1) &= \frac{\text{Var}(\widetilde{x}_1)}{1 - m_1} - m_1\mathbb{E}[x_1]^2 \\
&= \frac{\text{Var}(\widetilde{x}_1)}{1 - m_1} - \frac{m_1}{(1 - m_1)^2}\mathbb{E}[\widetilde{x}_1]^2
\end{aligned}$$

Putting this together, the variance-covariance matrix is given by (elementwise division below):

$$\text{Cov}(\widetilde{X}, \widetilde{X}) = (1 - m)(1 - m)^\top \odot \text{Cov}(X, X)$$

$$\begin{aligned}
&\quad+\text{diag}(((1-m)-(1-m)^2)\text{Var}(X)+m(1-m)\mathbb{E}[x_1]^2)\\
&=(1-m)(1-m)^\top\odot\text{Cov}(X,X)+\text{diag}(m(1-m)(\text{Var}(X)+\mathbb{E}[X]^2))\\
&=(1-m)(1-m)^\top\odot\text{Cov}(X,X)\\
&\quad+\text{diag}(m(1-m)^\top)\text{diag}(\text{Cov}(X,X)+\mathbb{E}[X]^\top\mathbb{E}[X])\\
&=(1-m)(1-m)^\top\odot\text{Cov}(X,X)+\text{diag}(m(1-m)^\top)\text{diag}(\mathbb{E}[X^\top X])\\
\implies \text{Cov}(X,X)&=\left(\frac{1}{1-m}\right)\left(\frac{1}{1-m}\right)^\top\odot\text{Cov}(\widetilde{X},\widetilde{X})\\
&\quad+\text{diag}\left(-\frac{\text{Var}(\widetilde{X})}{(1-m)^2}+\frac{\text{Var}(\widetilde{X})}{1-m}-\frac{m\mathbb{E}\left[\widetilde{X}\right]^2}{(1-m)^2}\right)\\
&=\left(\frac{1}{1-m}\right)\left(\frac{1}{1-m}\right)^\top\odot\text{Cov}(\widetilde{X},\widetilde{X})-\text{diag}\left(\frac{m}{(1-m)^2}(\text{Var}(\widetilde{X})+\mathbb{E}[\widetilde{X}]^2)\right)
\end{aligned}$$

Thus far, we have been working with the covariance matrix. How do the expressions for covariance relate to $\widetilde{X}^\top\widetilde{X}$ and $\widetilde{X}^\top Y$? We have:

$$\begin{aligned}
\text{Cov}(\widetilde{X},\widetilde{X})&=(1-m)(1-m)^\top\odot\text{Cov}(X,X)+\text{diag}\left(m(1-m)^\top\right)\text{diag}(\mathbb{E}[X^\top X])\\
\mathbb{E}[\widetilde{X}^\top\widetilde{X}]&=\text{Cov}\left(\widetilde{X},\widetilde{X}\right)+\mathbb{E}[\widetilde{X}]^\top\mathbb{E}[\widetilde{X}]\\
&=(1-m)(1-m)^\top\odot(\text{Cov}(X,X)+\mathbb{E}[X]^\top\mathbb{E}[X])\text{diag}\left(m(1-m)^\top\right)\text{diag}\left(\mathbb{E}\left[X^\top X\right]\right)\\
&=(1-m)(1-m)^\top\odot\mathbb{E}\left[X^\top X\right]+\text{diag}\left(m(1-m^\top)\right)\text{diag}\left(\mathbb{E}\left[X^\top X\right]\right)
\end{aligned}$$

Additionally,

$$\begin{aligned}
\text{Cov}\left(X,X\right)&=\left(\frac{1}{1-m}\right)\left(\frac{1}{1-m}\right)^\top\odot\text{Cov}(\widetilde{X},\widetilde{X})+\text{diag}\left(-\frac{m}{(1-m)^2}\right)\text{diag}\left(\text{Var}(\widetilde{X})+\mathbb{E}[\widetilde{X}]^2\right)\\
\mathbb{E}\left[X^\top X\right]&=\text{Cov}\left(X,X\right)+\mathbb{E}\left[X\right]^\top\mathbb{E}\left[X\right]\\
&=\left(\frac{1}{1-m}\right)\left(\frac{1}{1-m}\right)^\top\odot\left(\text{Cov}(\widetilde{X},\widetilde{X})+\mathbb{E}[\widetilde{X}]^\top\mathbb{E}[\widetilde{X}]\right)\\
&\quad+\text{diag}\left(-\frac{m}{(1-m)^2}\right)\text{diag}\left(\text{Var}(\widetilde{X})+\mathbb{E}[\widetilde{X}]^2\right)\\
&=\left(\frac{1}{1-m}\right)\left(\frac{1}{1-m}\right)^\top\odot\mathbb{E}[\widetilde{X}^\top\widetilde{X}]-\text{diag}\left(\frac{m}{(1-m)^2}\right)\text{diag}\left(\mathbb{E}[\widetilde{X}^\top\widetilde{X}]\right)
\end{aligned}$$

D.8.3 Closed Form Solution

Using results from previous sections, we can now derive a closed form solution for the optimal linear classifier for a target domain with missing rates m_t, given labeled data from a source domain with missing rates m_s. We break down this problem by going from corrupted data with some missingness rate to clean data with 0 missingness, and then from clean data with 0 missingness to corrupted data with another level of missingness.

Table D.1: Summary of 1st and 2nd moments of corrupted data and clean data

Quantity of Interest	Expression
$\mathbb{E}\left[X\right]$	$\frac{1}{1-m} \odot \mathbb{E}\left[\widetilde{X}\right]$
$\mathbb{E}\left[\widetilde{X}\right]$	$(1-m) \odot \mathbb{E}\left[X\right]$
$\mathbb{E}\left[X^\top X\right]$	$\left(\frac{1}{1-m}\right)\left(\frac{1}{1-m}\right)^\top \odot \mathbb{E}\left[\widetilde{X}^\top \widetilde{X}\right] - \text{diag}\left(\frac{m}{(1-m)^2}\right) \text{diag}\left(\mathbb{E}[\widetilde{X}^\top \widetilde{X}]\right)$
$\mathbb{E}\left[\widetilde{X}^\top \widetilde{X}\right]$	$(1-m)(1-m)^\top \odot \mathbb{E}\left[X^\top X\right] + \text{diag}\left(m(1-m)^\top\right) \text{diag}\left(\mathbb{E}\left[X^\top X\right]\right)$

Suppose we are going from corrupted data $\widetilde{X}$ with missing rate m to clean data X with 0 missingness:

$$\begin{aligned}
\text{Cov}\left(X, y\right) &= \frac{1}{1-m} \odot \text{Cov}\left(\widetilde{X}, y\right) \\
\mathbb{E}\left[X^\top y\right] &= \text{Cov}\left(X, y\right) + \mathbb{E}\left[X\right]^\top \mathbb{E}\left[y\right] \\
&= \frac{1}{1-m} \odot \text{Cov}\left(\widetilde{X}, y\right) + \frac{1}{1-m} \odot \mathbb{E}\left[\widetilde{X}\right]^\top \mathbb{E}\left[y\right] \\
&= \frac{1}{1-m} \odot \mathbb{E}\left[\widetilde{X}^\top y\right] \\
\mathbb{E}\left[X^\top X\right] &= \left(\frac{1}{1-m}\right)\left(\frac{1}{1-m}\right)^\top \odot \mathbb{E}\left[\widetilde{X}^\top \widetilde{X}\right] - \text{diag}\left(\frac{m}{(1-m)^2} \odot \mathbb{E}\left[\widetilde{X}^\top \widetilde{X}\right]\right) \\
\implies \beta &= \left\{\left(\frac{1}{1-m}\right)\left(\frac{1}{1-m}\right)^\top \odot \mathbb{E}\left[\widetilde{X}^\top \widetilde{X}\right] - \text{diag}\left(\frac{m}{(1-m)^2} \odot \mathbb{E}\left[\widetilde{X}^\top \widetilde{X}\right]\right)\right\}^{-1} \frac{1}{1-m} \odot \mathbb{E}\left[\widetilde{X}^\top y\right]
\end{aligned}$$

Going from clean to corrupted data, we have:

$$\begin{aligned}
\mathbb{E}[\widetilde{X}^\top y] &= \text{Cov}(\widetilde{X}, y) + \mathbb{E}[\widetilde{X}]^\top \mathbb{E}\left[y\right] \\
&= (1-m) \odot \text{Cov}\left(X, y\right) + (1-m) \odot \mathbb{E}\left[\widetilde{X}\right]^\top \mathbb{E}\left[y\right] \\
\mathbb{E}[\widetilde{X}^\top \widetilde{X}] &= (1-m)(1-m)^\top \odot \mathbb{E}\left[X^\top X\right] + \text{diag}\left(m(1-m^\top)\right) \text{diag}\left(\mathbb{E}\left[X^\top X\right]\right) \\
\implies \widetilde{\beta} &= \left[(1-m)(1-m)^\top \odot \mathbb{E}\left[X^\top X\right] + \text{diag}\left(m(1-m^\top)\right) \text{diag}\left(\mathbb{E}\left[X^\top X\right]\right)\right]^{-1} (1-m) \odot \mathbb{E}\left[X^\top y\right]
\end{aligned}$$

Now, we put all of these equations together, going from source corrupted data (S), to clean data (C), to target corrupted data (T).

(S) $\rightarrow$ (C):

$$\begin{aligned}
\mathbb{E}[X^\top X] &= \left(\frac{1}{1-m_s}\right)\left(\frac{1}{1-m_s}\right)^\top \odot \mathbb{E}[\widetilde{X}^{s\top} \widetilde{X}^s] - \text{diag}\left(\frac{m_s}{(1-m_s)^2} \odot (\mathbb{E}[\widetilde{X}^{s\top} \widetilde{X}^s])\right) \\
\mathbb{E}[X^\top y] &= \frac{1}{1-m_s} \odot \text{Cov}\left(\widetilde{X}^s, y\right) + \frac{1}{1-m_s} \odot \mathbb{E}[\widetilde{X}^s]^\top \mathbb{E}\left[y\right]
\end{aligned}$$

(C) → (T):

$$\begin{aligned}
\mathbb{E}\left[\widetilde{X}^{t\top}\widetilde{X}^{t}\right] &= (1-m_t)(1-m_t)^\top \odot \mathbb{E}\left[X^\top X\right] + \text{diag}\left(m_t(1-m_t^\top)\right)\text{diag}\left(\mathbb{E}\left[X^\top X\right]\right) \\
&= (1-m_t)(1-m_t)^\top \odot \left[\left(\frac{1}{1-m_s}\right)\left(\frac{1}{1-m_s}\right)^\top \odot \mathbb{E}\left[\widetilde{X}^{s\top}\widetilde{X}^{s}\right] - \text{diag}\left(\frac{m_s}{(1-m_s)^2}\mathbb{E}\left[\widetilde{X}^{s\top}\widetilde{X}^{s}\right]\right)\right] \\
&\qquad + \text{diag}\left(\frac{m_t}{1-m_t}\right)\odot\text{diag}\left(\mathbb{E}\left[\widetilde{X}^{s\top}\widetilde{X}^{s}\right] - \text{diag}\left(m_s\right)\mathbb{E}\left[\widetilde{X}^{s\top}\widetilde{X}^{s}\right]\right) \\
&= (1-m_t)(1-m_t)^\top \odot \left(\frac{1}{1-m_s}\right)\left(\frac{1}{1-m_s}\right)^\top \odot \mathbb{E}\left[\widetilde{X}^{s\top}\widetilde{X}^{s}\right] \\
&\qquad - (1-m_t)(1-m_t)^\top \odot \text{diag}\left(\frac{m_s}{(1-m_s)^2}\odot\mathbb{E}\left[\widetilde{X}^{s\top}\widetilde{X}^{s}\right]\right) \\
&\qquad + \text{diag}\left(\frac{m_t(1-m_t)}{1-m_s}\odot\mathbb{E}\left[\widetilde{X}^{s\top}\widetilde{X}^{s}\right]\right)
\end{aligned}$$

For $i \neq j$, the off-diagonal entries of the above expression are given by:

$$\mathbb{E}\left[\widetilde{X}^{t\top}\widetilde{X}^{t}\right]_{ij} = \left(\frac{1-m_{ti}}{1-m_{si}}\right)\left(\frac{1-m_{tj}}{1-m_{sj}}\right)\mathbb{E}\left[\widetilde{X}^{s\top}\widetilde{X}^{s}\right]_{ij} = (1-r_i^{s\to t})(1-r_j^{s\to t})\mathbb{E}\left[\widetilde{X}^{s\top}\widetilde{X}^{s}\right]_{ij}$$

The diagonal entries of the above expression are given by:

$$\begin{aligned}
\mathbb{E}\left[\widetilde{X}^{t\top}\widetilde{X}^{t}\right]_{ii} &= \mathbb{E}\left[\widetilde{X}^{s\top}\widetilde{X}^{s}\right]_{ii}\left(\left(\frac{1-m_{ti}}{1-m_{si}}\right)^2 - \frac{m_{si}(1-m_{ti})^2}{(1-m_{si})^2} + \frac{m_{ti}(1-m_{ti})}{1-m_{si}}\right) \\
&= \mathbb{E}\left[\widetilde{X}^{s\top}\widetilde{X}^{s}\right]_{ii}\left((1-r_i^{s\to t})^2 - m_{si}(1-r_i^{s\to t})^2 + m_{ti}(1-r_i^{s\to t})\right) \\
&= \mathbb{E}\left[\widetilde{X}^{s\top}\widetilde{X}^{s}\right]_{ii}(1-r_i^{s\to t})\left((1-r_i^{s\to t}) - m_{si}(1-r_i^{s\to t}) + m_{ti}\right) \\
&= \mathbb{E}\left[\widetilde{X}^{s\top}\widetilde{X}^{s}\right]_{ii}(1-r_i^{s\to t})\left(\frac{1-m_{ti}}{1-m_{si}} - \frac{m_{si}-m_{si}m_{ti}}{1-m_{si}} + \frac{m_{ti}-m_{si}m_{ti}}{1-m_{si}}\right) \\
&= \mathbb{E}\left[\widetilde{X}^{s\top}\widetilde{X}^{s}\right]_{ii}(1-r_i^{s\to t})\left(\frac{1-m_{si}}{1-m_{si}}\right) \\
&= \mathbb{E}\left[\widetilde{X}^{s\top}\widetilde{X}^{s}\right]_{ii}(1-r_i^{s\to t})
\end{aligned}$$

Additionally,

$$\begin{aligned}
\mathbb{E}\left[\widetilde{X}^{t\top}y\right] &= (1-m_t)\odot\mathbb{E}\left[X^\top y\right] \\
&= (1-m_t)\odot\left(\frac{1}{1-m_s}\odot\text{Cov}\left(\widetilde{X}^{s}, y\right) + \frac{1}{1-m_s}\odot\mathbb{E}\left[\widetilde{X}^{s}\right]^\top\mathbb{E}\left[y\right]\right) \\
&= \frac{1-m_t}{1-m_s}\odot\mathbb{E}\left[\widetilde{X}^{s\top}y\right]
\end{aligned}$$

D.9 Experiment Details

Experiments were run on a machine with 28 CPU cores. The linear regression models were implemented from scratch and validated against that of sklearn. The MLPRegressor class from the scikit-learn Python package was used with default hyperparameters, and the XGBoost class from the xgboost Python package was used with default hyperparameters. All experiments (except imputation) are feasible to run within a few hours.

Semi-synthetic experiments on linear models included 10 samples of β, and 50 samples of missingness rates under each regime ($m^s \preceq m^t$ and m^s ? m^t). Semi-synthetic experiments on nonlinear models (XGB, NN) included 5 samples of β and 20 samples of missingness rates under each regime. Across these runs, 95% confidence intervals were computed.

In the imputation experiments, a MissForest imputer from the missingpy Python package was trained on the combination of the source training set and target training set (just on the covariates, without labels). This imputer was then applied to both the source and target test sets. Finally, we train a source classifier on the imputed source labeled data and evaluate its performance on the target unlabeled data. We note that in our experience with the imputation experiments, imputation was somewhat slow (2-3 minutes for each imputation), and so all of our imputed results are reported on 5 samples of β and 20 samples of missingness rates under each regime, across all semi-synthetic datasets.

D.9.1 Synthetic Data Experiments

Table D.2: MSE/Var(Y) on Redundant Features and Confounded Features settings, with 95% confidence intervals computed over varying ϵ between 0.05 to 0.95.

	$m^s \preceq m^t$	m^s ? m^t
Lin. Reg. (oracle)	0.178 (0.172 − 0.185)	0.206 (0.199 − 0.213)
Lin. Reg. (source)	1.259 (1.231 − 1.286)	1.103 (1.076 − 1.129)
Lin. Reg. (imputed)	1.002 (1.002 − 1.002)	0.918 (0.915 − 0.921)
Lin. Reg. (closed-form adj.)	**0.186 (0.180 − 0.193)**	**0.209 (0.205 − 0.213)**
Lin. Reg. (non-param. adj.)	0.473 (0.471 − 0.476)	0.492 (0.489 − 0.495)
XGBoost (oracle)	0.166 (0.160 − 0.172)	0.200 (0.193 − 0.208)
XGBoost (source)	**0.166 (0.160 − 0.172)**	0.475 (0.458 − 0.492)
XGBoost (imputed)	1.002 (1.002 − 1.002)	1.157 (1.102 − 1.211)
XGBoost (non-param. adj.)	0.425 (0.422 − 0.428)	**0.473 (0.468 − 0.478)**
MLP (oracle)	0.166 (0.160 − 0.172)	0.201 (0.195 − 0.208)
MLP (source)	**0.184 (0.165 − 0.202)**	**0.321 (0.300 − 0.342)**
MLP (imputed)	1.003 (1.002 − 1.003)	0.924 (0.918 − 0.930)
MLP (non-param. adj.)	0.436 (0.428 − 0.444)	0.470 (0.465 − 0.474)

D.9.2 Semi-Synthetic Data Experiments

The UCI datasets Dua and Graff, 2017 used in this work are:

- Adult Data Set: The classification task is whether an individual's income exceeds $50K a year based on census data. The dataset contains categorical variables (occupation, education, marital status, etc.), as well as continuous variables (age, hours per week, etc.)
- Bank Marketing Data Set: The classification task is whether a client will subscribe a term deposit. This dataset contains categorical features such as type of job, marital status, education, whether they have a housing loan, etc., as well as continuous variables such as age, number of contacts performed, etc.
- Thyroid Disease Data Set: The classification task is of increased vs. decreased binding protein. This dataset contains binary variables such as whether the patient is pregnant, is male, on thyroxine, has a tumor, etc., as well as continuous variables such as age, TSH, T3, TT4, etc.

For semi-synthetic experiments, we pre-process the UCI data by creating dummy variables from categorical variables, dropping redundant columns, normalizing numerical variables, dropping binary variables with low frequency ($< 5\%$, since we apply additional synthetic missingness in our experiments), and dropping columns with low variance ($< 5\%$). We additionally generate synthetic labels by sampling coefficients $\beta_j \sim \text{Uniform}(0, 10), \forall j \in \{0, 1, 2, ..., d\}$ and computing new synthetic labels $y_{new} = X\beta$. Table D.3 contains the MSE/Var(Y) and 95% confidence intervals (from sampling several β and m^s, m^t) of the adult dataset, Table D.4 contains the MSE/Var(Y) and 95% confidence intervals of the bank dataset, and Table D.5 contains the MSE/Var(Y) and 95% confidence intervals of the thyroid dataset.

Table D.3: MSE/Var(Y) on UCI Adult Semi-synthetic Setting, with 95% confidence intervals computed over multiple samples of β and m^s, m^t (described in Section 7.5).

	$m^s \preceq m^t$	m^s ? m^t
Lin. Reg. (oracle)	0.420 (0.415 – 0.424)	0.362 (0.356 – 0.367)
Lin. Reg. (source)	0.437 (0.433 – 0.442)	0.380 (0.373 – 0.386)
Lin. Reg. (imputed)	0.490 (0.471 – 0.509)	0.483 (0.475 – 0.491)
Lin. Reg. (closed-form adj.)	0.422 (0.417 – 0.426)	**0.363 (0.358 – 0.368)**
Lin. Reg. (non-param. adj.)	**0.420 (0.415 – 0.424)**	0.373 (0.367 – 0.379)
XGBoost (oracle)	0.398 (0.386 – 0.409)	0.354 (0.344 – 0.363)
XGBoost (source)	0.399 (0.387 – 0.410)	0.379 (0.369 – 0.388)
XGBoost (imputed)	0.512 (0.491 – 0.534)	0.521 (0.508 – 0.535)
XGBoost (non-param. adj.)	0.399 (0.387 – 0.410)	0.392 (0.382 – 0.402)
MLP (oracle)	0.389 (0.378 – 0.401)	0.343 (0.334 – 0.352)
MLP (source)	0.399 (0.387 – 0.410)	0.357 (0.348 – 0.367)
MLP (imputed)	0.480 (0.461 – 0.499)	0.468 (0.456 – 0.481)
MLP (non-param. adj.)	**0.389 (0.378 – 0.400)**	**0.355 (0.346 – 0.364)**

Table D.4: MSE/Var(Y) on UCI Bank Semi-synthetic Setting, with 95% confidence intervals computed over multiple samples of β and m^s, m^t (described in Section 7.5).

	$m^s \preceq m^t$	m^s ? m^t
Lin. Reg. (oracle)	0.338 (0.336 – 0.340)	0.433 (0.426 – 0.440)
Lin. Reg. (source)	0.371 (0.369 – 0.373)	0.480 (0.472 – 0.487)
Lin. Reg. (imputed)	0.501 (0.491 – 0.511)	0.592 (0.583 – 0.602)
Lin. Reg. (closed-form adj.)	0.339 (0.337 – 0.340)	**0.442 (0.436 – 0.449)**
Lin. Reg. (non-param. adj.)	**0.338 (0.336 – 0.340)**	0.459 (0.453 – 0.466)
XGBoost (oracle)	0.287 (0.279 – 0.295)	0.453 (0.438 – 0.468)
XGBoost (source)	0.305 (0.297 – 0.313)	**0.500 (0.484 – 0.516)**
XGBoost (imputed)	0.492 (0.482 – 0.503)	0.708 (0.684 – 0.732)
XGBoost (non-param. adj.)	**0.287 (0.279 – 0.295)**	0.503 (0.486 – 0.519)
MLP (oracle)	0.295 (0.287 – 0.303)	0.458 (0.442 – 0.473)
MLP (source)	0.322 (0.314 – 0.330)	0.499 (0.483 – 0.516)
MLP (imputed)	0.484 (0.474 – 0.494)	0.668 (0.645 – 0.690)
MLP (non-param. adj.)	**0.294 (0.286 – 0.302)**	**0.487 (0.471 – 0.503)**

Table D.5: MSE/Var(Y) on UCI Thyroid Semi-synthetic Setting, with 95% confidence intervals computed over multiple samples of β and m^s, m^t (described in Section 7.5).

	$m^s \preceq m^t$	m^s ? m^t
Lin. Reg. (oracle)	0.298 (0.292 – 0.303)	0.251 (0.246 – 0.256)
Lin. Reg. (source)	0.350 (0.342 – 0.357)	0.320 (0.314 – 0.326)
Lin. Reg. (imputed)	0.306 (0.298 – 0.313)	0.358 (0.351 – 0.365)
Lin. Reg. (closed-form adj.)	0.316 (0.310 – 0.322)	**0.291 (0.286 – 0.295)**
Lin. Reg. (non-param. adj.)	**0.293 (0.288 – 0.298)**	**0.291 (0.286 – 0.296)**
XGBoost (oracle)	0.316 (0.304 – 0.328)	0.274 (0.265 – 0.282)
XGBoost (source)	**0.310 (0.298 – 0.322)**	**0.352 (0.341 – 0.362)**
XGBoost (imputed)	0.355 (0.346 – 0.364)	0.441 (0.430 – 0.452)
XGBoost (non-param. adj.)	**0.310 (0.298 – 0.321)**	0.381 (0.370 – 0.392)
MLP (oracle)	0.279 (0.269 – 0.288)	0.230 (0.223 – 0.236)
MLP (source)	0.320 (0.308 – 0.331)	0.303 (0.294 – 0.311)
MLP (imputed)	0.304 (0.296 – 0.311)	0.345 (0.336 – 0.355)
MLP (non-param. adj.)	**0.278 (0.268 – 0.288)**	**0.272 (0.265 – 0.279)**

D.9.3 Real Data Experiments

The data for these experiments were derived from eICU-CRD (Pollard et al., 2018a), a multi-hospital critical care database which uses the PhysioNet Credentialed Health Data License Version 1.5.0. We extract data for predicting 48-hour mortality through the FIDDLE (Tang et al., 2020) preprocessing pipeline with default parameters. FIDDLE extracts both time-varying and fixed features. We collapse the time-varying features by taking the maximum value (note that most features are binary, and none take values less than 0). We extract data from two of the hospitals with the most data, the first of which contains 3,006 data points, and the second of which contains 2,663 data points. The rate of 48-hour mortality in the first hospital is 0.097, and the rate of 48-hour mortality in the second hospital is 0.100. Additionally, we threshold for features that are present that have a prevalence of at least 5% in either of the hospitals and at least 1% in both of the hospitals. Code is provided at https://github.com/acmi-lab/Missingness-Shift. We used target unlabeled data ($\alpha_t = 1, \alpha_s = 0$) to estimate $\mathbb{E}[\widetilde{X}^{t\top}\widetilde{X}^t]$ for the adjusted linear closed form model because we noticed that the estimation error with limited data made the source estimates less reliable. Due to limited positive samples, in order to evaluate cross-domain performance, a model was trained on all data from one domain and tested on all data from the other. Oracle performance (training and testing on the same domain) was computed from training on a randomly sampled 80% of the data and testing on the remaining 20%. Table D.6 contains the estimated relative non-missingness of the top five coefficients for the oracle models from each hospital.

Table D.6: The estimated proportion of nonzeros in Hospital 1 (q_1) and Hospital 2 (q_2), estimated relative non-missingness rates $q_2/q_1 = 1 - r^{1\to 2}$, Hospital 1 Oracle coefficient (β_1), and Hospital 2 Oracle coefficient (β_2) for each of the top five features (measure by magnitude of coefficient) from the Oracle linear predictors of Hospital 1 and 2.

	β_1	β_2	q_1	q_2	q_2/q_1
noninvasivemean_max_(78.0, 86.0]	-0.279	-0.364	0.754	0.938	1.244
systemicsystolic_mean_(-94.001, 99.667]	0.271	-0.362	0.333	0.134	0.404
unittype...Neuro ICU	0.055	-0.577	0.194	0.315	1.629
ethnicity...African American	-0.275	0.361	0.141	0.071	0.506
...Intake (ml)...(100.0, 150.0]	0.070	-0.732	0.318	0.045	0.142
...Invasive BP Systolic...(-59.001, 101.0]	-0.571	0.474	0.350	0.130	0.372
cvp_max_(8.0, 12.0]	0.536	-0.476	0.262	0.125	0.477

Appendix E

Business Metric-Aware Forecasting

E.1 Differentiable Computation of Inventory Performance Metrics

In this appendix section we derive equations (6.5) and (6.4), and walk through differentiable computation of all inventory system variables.

Consider a model trained on N univariate time-series, each with at most T time points. Each time point, the model makes a lead-time forecast for the demand across a forecast horizon L. Thus, the model outputs a tensor $\widehat{d} = \mathbb{R}^{N \times T \times L}$, where an entry $\widehat{d}[i, t, l]$ corresponds to the forecasted demand in the ith series for time $t + l$ at time t.

E.1.1 Computing forecasted lead-time demand

The forecasted lead-time demand is $\widehat{D}_t^L = \sum_{l=1}^{L} \hat{d}[:, t, l]$, i.e. summation along the last axis of the $\widehat{d}$ tensor.

E.1.2 Computing orders

Here we derive equation (6.5). Alternating plugging in the inventory position equation (6.4) into the order-up-to policy equation (6.1) and recursively plugging in (6.1) to itself, we can expand the expression for orders into a closed form:

$$\begin{aligned}
o_t &= \widehat{D}_t^L + ss_t - ip_t \\
&= \widehat{D}_t^L + ss_t - (ip_{t-1} + o_{t-1} - d_t) \\
&= \widehat{D}_t^L + ss_t - ip_{t-1} - (\widehat{D}_{t-1}^L + ss_{t-1} - ip_{t-1}) + d_t
\end{aligned}$$

$$
\begin{aligned}
&= (\widehat{D}_t^L - \widehat{D}_{t-1}^L) + (ss_t - ss_{t-1}) + d_t \\
&= (\widehat{D}_t^L - \widehat{D}_{t-1}^L) + \Phi^{-1}(\alpha_s) \cdot (\sigma_{e,t} - \sigma_{e,t-1}) + d_t,
\end{aligned}
$$

where the last step plugs in the safety stock definition (6.3). Using this equation we have derived, it is now possible to compute o_t given just a tensor of demand forecasts $\widehat{d}$ and true demands d.

E.1.3 Computing net inventory

Here we derive the closed-form net inventory equation (6.6). Recursively plugging in the net inventory equation as well as the closed form expression for orders (6.5), we have:

$$
\begin{aligned}
i_t &= i_{t-1} + o_{t-L} - d_t \\
&= (i_{t-2} + o_{t-1-L} - d_{t-1}) + o_{t-L} - d_t \\
&= i_0 + \sum_{j=0}^{t-L} o_j - \sum_{k=1}^{t} d_k \\
&= i_0 + \sum_{j=0}^{t-L} \Big((\widehat{D}_j^L - \widehat{D}_{j-1}^L) + (ss_j - ss_{j-1}) + d_j \Big) - \sum_{k=1}^{t} d_k \\
&= i_0 + \sum_{j=0}^{t-L} (\widehat{D}_j^L - \widehat{D}_{j-1}^L + ss_j - ss_{j-1}) + d_0 - \sum_{k=t-L+1}^{t} d_k \\
&= i_0 + \widehat{D}_{t-L}^L + \Phi^{-1}(\alpha_s) \cdot \sigma_{e,t-L} + d_0 - \sum_{k=t-L+1}^{t} d_k,
\end{aligned}
$$

assuming that all quantities at time $t = -1$ are equal to 0. Then, $i_0 = i_{-1} + o_{-L} - d_0 = -d_0$. Simplifying,

$$
i_t = \widehat{D}_{t-L}^L + \Phi^{-1}(\alpha_s) \cdot \sigma_{e,t-L} - \sum_{k=t-L+1}^{t} d_k.
$$

Intuitively this makes sense, as the net inventory is determined by the lead time forecast from L time steps prior, with additional safety stock estimated at the time, subtracting the interim demand leading up to the current time step. This closed form equation is much more efficient to implement in terms of tensor operations than the original recursive equation for inventory position. We have already described how to compute the first term, and the last term is simply the sum of true demands in a window of size L leading up to time t. The 2nd term (safety stock) is computed by some constant $\Phi^{-1}(\alpha_s)$ dependent on desired service level α_s, multiplied by the standard deviation of forecast errors up until the previous time $\sigma_{e,t-1}$. Similar to the net inventory, we can also derive a closed-form expression for the inventory position: $ip_t = \widehat{D}_{t-1}^L + \Phi^{-1}(\alpha_s) \cdot \sigma_{e,t-1} - d_t$, and the work in progress can simply be derived as $w_t = ip_t - i_t$.

E.1.4 Computing safety stock

The inverse CDF of the target service level $\Phi^{-1}(\alpha_s)$ is a constant and straightforward to compute. The standard deviations of forecast errors $\sigma_{e,t}$ are more involved since forecasts are made at each time point for some horizon, but can be computed as follows:

- Create an $N \times T$ tensor M where $M[:, t] = t$.
- Construct sliding windows of size L along the time dimension. This will create a tensor M' with the same shape as $\widehat{d}$ ($N \times T \times L$), of the time of each entry.
- Repeat the tensor T times in a new (fourth) dimension, thresholding to create a binary mask M'' for each $t \in \{1, 2, ..., T\}$ corresponding to whether that time has occurred. That is, $M''[i, t, l, t'] = \mathbf{1}\{M'[i, t, l, t'] \leq t'\}$.
- Compute per-element squared error E of the $N \times T \times L$ predictions. Copy T times to get E' ($N \times T \times L \times T$).
- Multiply the repeated error E' element-wise with the binary mask M'' (both should be $N \times T \times L \times T$). Sum along the second and third dimensions (of size T and L), and divide by the sum of the binary mask along the second and third dimensions. This gives the average squared forecast errors for each time point, for each time-series. Take the square root to get the standard deviation.

E.1.5 Computing inventory performance metrics

There are three main aspects of inventory performance we examine:

1. Holding cost: $C_h = c_h \cdot \mathbb{E}[\max(0, i_t)]$
2. Stockout cost: $C_s = c_s \cdot \mathbb{E}[\max(0, -i_t)]$
3. Order variance cost: $C_v = c_v \cdot \mathrm{Var}(o_t)$

The holding cost can be computed by passing the computed inventory positions through a ReLU activation, and then taking the average across times and series. The variance of orders can be computed by taking the average orders for each series, subtracting them from the orders, squaring, and then taking the expectation.

The total cost (TC) and relative RMS (RRMS) objectives combine these three components in differentiable ways (either summation or subtracting and dividing by a constant, squaring, and summing).

E.2 Double-Rollout Supervision

Here we describe the double-rollout supervision technique. Since some objectives must be computed holistically across multiple time points (e.g. order variance), a single series of points may only yield one inventory performance value. If one model is being trained per time series (as is the case in M3 univariate data), this provides very little supervision for the model. The double-rollout supervision technique addresses this problem by having the model predict a long forecasting horizon H, which is then treated as the series to compute inventory performance over (Figure E.1, top). A sliding window of size L is taken over the H time points (Figure E.1, bottom) in order to compute lead-time demands across the H time points (Figure E.2). Thus, we choose an $H > L$. If there are $t - W$ decoding points, then this gives $t - W$ series of length $\leq H$, which can provide $t - W$ measures of inventory performance to help supervise learning.

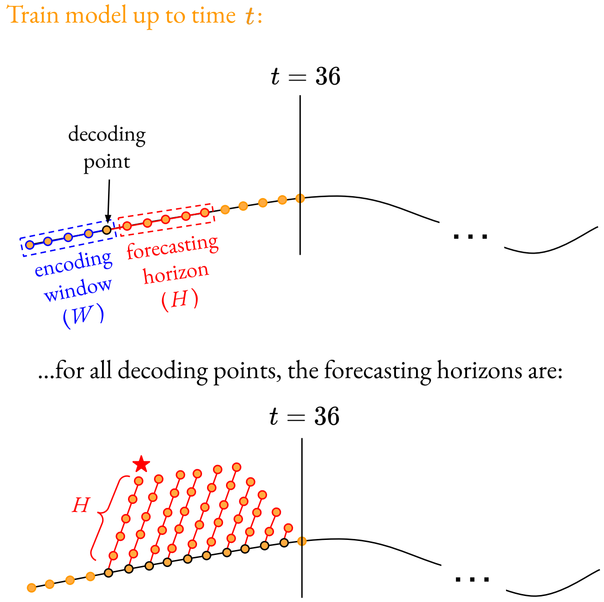

Train model up to time t:
$t = 36$
decoding point
encoding window (W)
forecasting horizon (H)
...
...for all decoding points, the forecasting horizons are:
$t = 36$
H
★
...

Figure E.1: First step of the double-rollout training procedure. At each decoding point, the model forecasts a long forecasting horizon H.

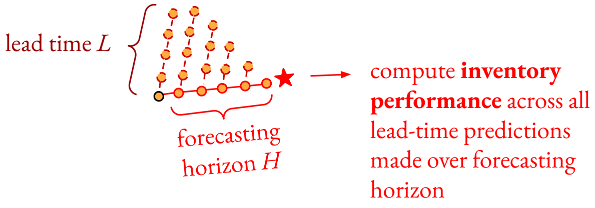

lead time L
★
forecasting horizon H
compute **inventory performance** across all lead-time predictions made over forecasting horizon

Figure E.2: Second step of the double-rollout training procedure. For each forecasted horizon of H time points, take a sliding window of size L as the lead-time demand forecasts and compute inventory performance across the H time points.

E.3 Roll-Forward Evaluation

In real-world settings, models are updated over time as new data are collected. To better simulate the process of how forecasting models could feasibly be updated over time, we employ a training procedure which rolls forward in time.

At each time point $t = 1, 2, ..., T$, the model is updated with the most recent data by taking additional gradient steps based on all data up to time t (i.e. fine-tuning to the latest dataset each time). Given a model trained on the most recent demand data up time t, predictions are made for the next lead-time L time steps (Figure E.3), giving $\widehat{D}_t^L$ (Figure E.4).

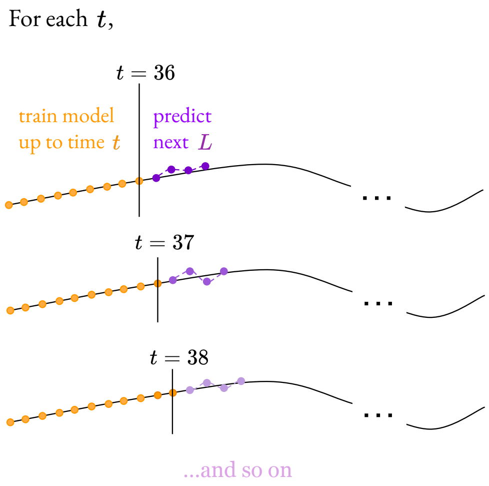


Figure E.3: Each time t, the model is updated using all data available up until t, and forecasts the next L time steps.

These predictions are used by the order-up-to policy (6.1) to determine how many orders to place. This way, when inventory performance is computed across all t, the predictions are made using the most up-to-date models that could have been trained at each time point.

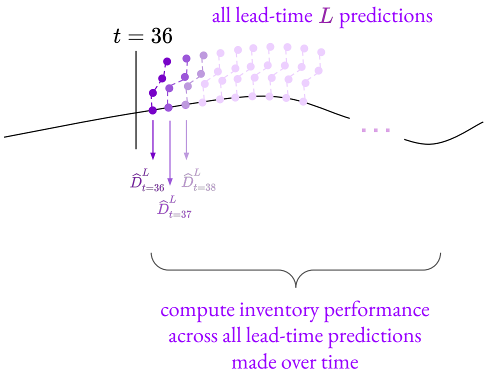


Figure E.4: Predictions resulting from the roll-forward evaluation procedure after rolling across the whole series.

E.4 Hyperparameter Selection

Ten iterations of random search with the MSE, TC, and RRMS objectives are used to choose hyperparameters for the M3 and Favorita datasets. For the Favorita dataset, a subset of 10,000 time series is used in order to tune hyperparameters more quickly. For the Naive Seasonal Scaler, the grid of hyperparameters considered is batchsize (100, 200, 300) and learning rate (0.01, 0.001, 0.0001). For the LSTM, the grid of hyperparameters considered is batchsize (100, 200, 300, 500), hidden size (32, 64, 128), and learning rate (0.01, 0.001, 0.0001, 0.00001). A small amount of subsequent manual tuning was done if the selected parameter was at the edge of the grid.

For the M3 dataset, a hidden size of 20 is used for both the encoder and decoder. For the Favorita dataset, a hidden size of 64 is used for both the encoder and decoder, and the embedding size is 10. Categorical variables are embedded such that each possible value is stored as a different learnable 10-dimensional vector, and numerical variables are passed through a linear layer with 10-dimensional output. The `run_all.sh` script in the included code supplement contains commands to run all of the experiments, with exact hyperparameter settings included.

E.5 Training Details

On Favorita, LSTM encoder-decoder models are trained on machines with A100 GPUs, and require about 10GB of memory. On M3, LSTM encoder-decoder models are trained on machines with 32 CPUs, and since one model is trained for each series, model training is done in parallel, taking about 2GB of memory at a time. All model training is done on the Ubuntu operating system. Package versions are included in the code package.

For M3, the training and validation cutoff points are chosen so that there are substantial time points in the training set (72) while still leaving enough to evaluate validation (36) and test (36) performance. For Favorita, training and validation cutoffs are the same as in Lim et al. (2021).

E.6 Architecture

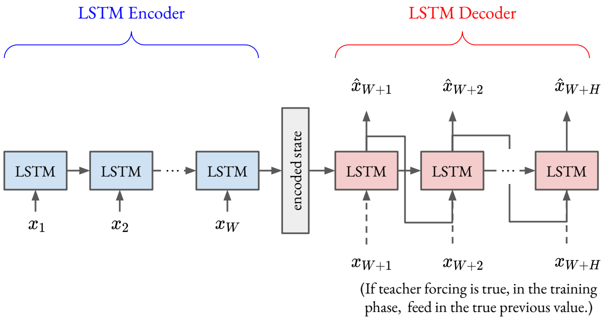


Figure E.5: LSTM encoder-decoder architecture.

E.7 Additional Results: Average Forecasts

Business-aware forecasts are visualized in Figure E.6.

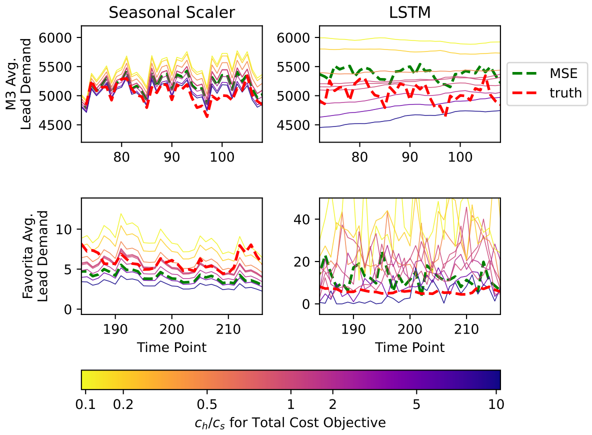


Figure E.6: Forecasted and true lead demand, averaged across all series in M3 (first row) and Favorita (second row). True demand is the red dotted line, and forecasts from the MSE objective are the green dotted line. Forecasts from the TC objective are solid lines, for several unit cost tradeoffs—fixing $c_v = 10^{-6}$ in the M3 dataset and $c_v = 10^{-2}$ in the Favorita dataset, ratios of c_h/c_s are indicated by line color.

E.8 Relative Improvements with Different Tradeoffs

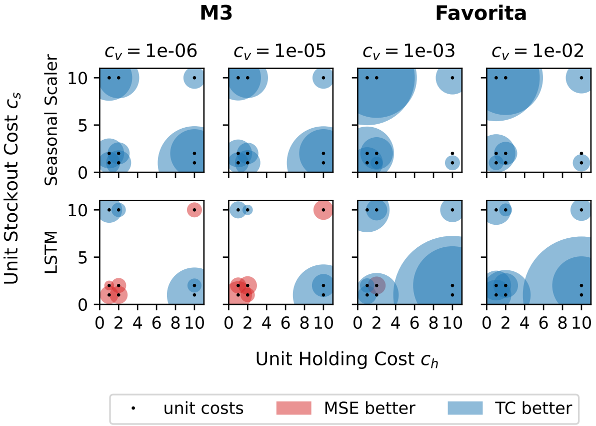


Figure E.7: Improvements in total cost when using the TC objective vs. the MSE objective, under various cost tradeoffs. Left two columns are for different c_v's on the M3 dataset, and right two columns are for the Favorita dataset. Radius is ten times the relative proportional improvement.

Appendix F

Timing as an Action: Learning When to Observe and Act

F.1 Proofs for Section 7.4

F.1.1 Proof of Lemma 7.4.1

Proof. Let us show that the Bellman backup operator H described in (7.2) is a contraction mapping in the finite space $(\mathbb{R}, L_\infty)$.

$$(Hq)(s,a,k) = \sum_{s' \in \mathcal{S}} P_{a,k}(s'|s) \left[\mathbb{E}[G(s,a,k)] + \gamma^k \max_{a',k'} q(s',a',k') \right]$$

$$\begin{aligned}
&||Hq_1 - Hq_2||_\infty \\
&\quad = \max_{s,a,k} \left| \sum_{s' \in \mathcal{S}} P_{a,k}(s'|s) \left[\mathbb{E}[G(s,a,k)] + \gamma^k \max_{a',k'} q_1(s',a',k') - \mathbb{E}[G(s,a,k)] - \gamma^k \max_{a',k'} q_2(s',a',k') \right] \right| \\
&\quad = \max_{s,a,k} \left| \sum_{s' \in \mathcal{S}} P_{a,k}(s'|s) \left[\gamma^k (\max_{a',k'} q_1(s',a',k') - \max_{a',k'} q_2(s',a',k')) \right] \right| \\
&\quad \leq \max_{s,a,k} \gamma^k \sum_{s' \in \mathcal{S}} P_{a,k}(s'|s) \left| \max_{a',k'} q_1(s',a',k') - \max_{a',k'} q_2(s',a',k') \right| \\
&\quad \leq \max_{s,a,k} \gamma^k \sum_{s' \in \mathcal{S}} P_{a,k}(s'|s) \max_{a',k'} |q_1(s',a',k') - q_2(s',a',k')| \\
&\quad = \max_{s,a,k} \gamma^k \sum_{s' \in \mathcal{S}} P_{a,k}(s'|s) ||q_1 - q_2||_\infty \\
&\quad = \gamma^k ||q_1 - q_2||_\infty \\
&\quad \leq \gamma ||q_1 - q_2||_\infty
\end{aligned}$$

where the last line follows from the fact that $k \geq 1$. Thus, the operator H is a contraction. Hence, by the Banach fixed point theorem, there exists a unique optimal Q^*. □

F.1.2 Proofs for Lemma 7.4.2 and Lemma 7.4.3

First we prove Lemma 7.4.3. Applying Lemma F.1.1 to $\{\mathcal{D}_i\}_{i=1}^N$ drawn as in the generative setting (Definition 3), the LHS becomes

$$\sum_{i=1}^{N} \mathbb{E}_{(s,a,k)\sim\mathcal{D}_i} \left\|\widehat{P}(s'|s,a,k) - P(s'|s,a,k)\right\|_1^2 = n \sum_{s,a,k} \left\|\widehat{P}(s'|s,a,k) - P(s'|s,a,k)\right\|_1^2$$

and we have the bound that with probability at least $1-\delta$,

$$\sum_{s,a,k} \left\|\widehat{P}(s'|s,a,k) - P(s'|s,a,k)\right\|_1^2 \leq \frac{12\log(|\overline{\mathcal{P}}_1|/\delta)}{n} + 6\epsilon SAK + \epsilon^2 SAK$$

Then choosing $\epsilon = 1/nSAK$, Lemma F.1.4 states that $|\overline{\mathcal{P}}_1|$ has cardinality $\leq (nS^2AK^2)^{S^2A}$, thus

$$\sum_{s,a,k} \left\|\widehat{P}(s'|s,a,k) - P(s'|s,a,k)\right\|_1^2 \leq \frac{12S^2A\log(S^2AK^2n/\delta)}{n} + \frac{6}{n} + \frac{1}{SAKn^2},$$

which implies that (suppressing log factors)

$$\max_{s,a,k} \left\|\widehat{P}(s'|s,a,k) - P(s'|s,a,k)\right\|_1 \lesssim S\sqrt{\frac{A\log(K/\delta)}{n}}$$

The proof of Lemma 7.4.2 proceeds similarly but with $|\overline{\mathcal{P}}| \leq (nS^2AK)^{S^2AK}$ from Lemma F.1.4.

Maximum likelihood estimation We state and prove a general MLE guarantee for conditional probability estimation. This section is takes inspiration from the results in Agarwal et al., 2020a; Liu et al., 2022; Liu et al., 2023; Huang et al., 2023. We consider the problem of estimating the conditional density $f^*(y|x)$, for all $x \in \mathcal{X}$ (the input space) and $y \in \mathcal{Y}$ (the target space). We are given a function class $\mathcal{F} : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$, and suppose $f^* \in \mathcal{F}$. In addition, we have an adaptive dataset $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^N$ where $x_i \sim \mathcal{D}_i(x_{<i}, y_{<i})$, and $y_i \sim f^*(\cdot|x_i)$. We output $\widehat{f} = \text{argmax}_{f\in\mathcal{F}} \sum_{i=1}^n \log f(y_i|x_i)$. Note that this is analogous to the model-based estimation in (??) and (??) with function classes $\mathcal{P}_1$ and $\mathcal{P}$, respectively, and $\mathcal{X} = \mathcal{S} \times \mathcal{A} \times \mathcal{K}$ and $\mathcal{Y} = \mathcal{S}$.

We allow $\mathcal{F}$ to be an infinite function class, and the MLE bound will depend on the statistical complexity of the function class $\mathcal{F}$, which is quantified using the ℓ_1 optimistic cover, defined below:

Definition 4 (ℓ_1 optimistic cover). *For a function class $\mathcal{F} \subseteq (\mathcal{X} \to \mathbb{R})$, we call function class $\overline{\mathcal{F}}$ an ℓ_∞ optimistic cover of $\mathcal{F}$ with scale ϵ, if for any $f \in \mathcal{F}$ there exists $\overline{f} \in \overline{\mathcal{F}}$, such that* $\max_{x \in \mathcal{X}} \|f(\cdot|x) - \overline{f}(\cdot|x)\|_1 \leq \epsilon$ *and* $f(y|x) \leq \overline{f}(y|x), \forall x \in \mathcal{X}, y \in \mathcal{Y}$.

The formal bound for MLE estimation is stated below:

Lemma F.1.1 (MLE guarantee). *Suppose $\mathcal{F}$ satisfies: (i) $f^* \in \mathcal{F}$, (ii) each function $f \in \mathcal{F}$ is a valid probability distribution over $\mathcal{Y}$ given x (i.e., $f(\cdot|x) \in \Delta(\mathcal{Y})$ for all $x \in \mathcal{X}$), and (iii) $\mathcal{F}$ has a finite ℓ_1 optimistic cover (Definition 4) $\overline{\mathcal{F}}$ with scale ϵ and $\overline{\mathcal{F}} \subseteq (\mathcal{X} \to \mathbb{R}_{\geq 0})$. Then with probability at least $1 - \delta$, the MLE solution $\widehat{f}$ has an ℓ_1 error guarantee*

$$\sum_{i=1}^{N} \mathbb{E}_{x_i \sim \mathcal{D}_i} \left\| \widehat{f}(\cdot|x_i) - f^*(\cdot|x_i) \right\|_1^2 \leq 12 \log(|\overline{\mathcal{F}}|/\delta) + 6\epsilon N + \epsilon^2 N$$

Proof of Lemma F.1.1. First, define

$$\mathcal{L}(f, \mathcal{D}) = \frac{1}{2} \sum_{i=1}^{N} \log \left(\frac{f(y_i|x_i)}{f^*(y_i|x_i)} \right).$$

Next, we decouple the dependencies between samples, and state the following result from Agarwal et al., 2020a without proof:

Lemma F.1.2 (Lemma 24 of Agarwal et al., 2020a). *Let $\mathcal{D}$ be a dataset of N examples, and let $\mathcal{D}'$ be a tangent sequence. A tangent sequence $\{(x'_i, y'_i)\}_{i=1}^N$ is sampled as $x'_i \sim \mathcal{D}_i(x_{1:i-1}, y_{1:i-1})$ and $y'_i \sim f^*(\cdot|x'_i)$, which is independent conditioned on $\mathcal{D}$. Let $\mathcal{L}(f, \mathcal{D}) = \frac{1}{N} \sum_{i=1}^{N} l(f, (x_i, y_i))$ be any function that decomposes additively across examples, where l is any function, and let $f(\mathcal{D})$ be any estimator taking as input the random variable $\mathcal{D}$ and with range $\mathcal{F}$. Then*

$$\mathbb{E}_{\mathcal{D}} \left[\exp\left(\mathcal{L}(f(\mathcal{D}), \mathcal{D}) - \log \mathbb{E}_{\mathcal{D}'} \exp(\mathcal{L}(f(\mathcal{D}), \mathcal{D}')) - \log |\mathcal{F}| \right) \right] \leq 1.$$

Using Chernoff's method with union bound over $\overline{\mathcal{F}}$, with probability at least $1 - \delta$ we have for any $f \in \overline{\mathcal{F}}$ that

$$-\log \mathbb{E}_{\mathcal{D}'} \exp\left(\mathcal{L}(f(\mathcal{D}), \mathcal{D}') \right) \leq -\mathcal{L}(f(\mathcal{D}), \mathcal{D}) + \log(|\overline{\mathcal{F}}|/\delta)$$

Now let $\overline{f} \in \overline{\mathcal{F}}$ be the ϵ-close ℓ_1 optimistic approximator of the MLE solution $\widehat{f} \in \mathcal{F}$. Applying the above to $\overline{f}$, in the RHS we have

$$\begin{aligned}
-\mathcal{L}(\overline{f}(\mathcal{D}), \mathcal{D}) &= \frac{1}{2} \sum_{i=1}^{N} \log \frac{f^*(y_i|x_i)}{\overline{f}(y_i|x_i)} \\
&\leq \frac{1}{2} \sum_{i=1}^{N} \log \frac{f^*(y_i|x_i)}{\widehat{f}(y_i|x_i)} && (\overline{f} \text{ is optimistic cover}) \\
&= \frac{1}{2} \left(\sum_{i=1}^{N} \log f^*(y_i|x_i) - \sum_{i=1}^{N} \log \widehat{f}(y_i|x_i) \right)
\end{aligned}$$

$$\le 0 \qquad (\widehat{f} \text{ optimal})$$

Combining inequalities, we have

$$\begin{aligned}\log(|\overline{\mathcal{F}}|/\delta) \ge & -\log \mathbb{E}_{\mathcal{D}'} \exp\left(\mathcal{L}(\overline{f}(\mathcal{D}), \mathcal{D}')\right) \\ = & -\log \mathbb{E}_{\mathcal{D}'}\left[\exp\left(\frac{1}{2}\sum_{i=1}^{N}\log\left(\frac{\overline{f}(y_i'|x_i')}{f^*(y_i'|x_i')}\right)\right)|\mathcal{D}\right] \\ = & -\sum_{i=1}^{N}\log \mathbb{E}_{x\sim\mathcal{D}_i, y\sim f^*(\cdot|x_i)}\left[\sqrt{\frac{\overline{f}(y|x)}{f^*(y|x)}}\right] \end{aligned} \tag{F.1}$$

Next, we show that for any $\mathcal{D}$, we have

$$\mathbb{E}_{x\sim\mathcal{D}}\left\|\overline{f}(\cdot|x) - f^*(\cdot|x)\right\|_1^2 \le -12\log \mathbb{E}_{x\sim\mathcal{D}, y\sim f^*(\cdot|x)}\left[\sqrt{\frac{\overline{f}(y|x)}{f^*(y|x)}}\right] + 6\epsilon \tag{F.2}$$

Combining (F.1) and (F.2), we have

$$\sum_{i=1}^{N}\mathbb{E}_{x\sim\mathcal{D}_i}\left\|\overline{f}(\cdot|x) - f^*(\cdot|x)\right\|_1^2 \le 12\log(|\overline{\mathcal{F}}|/\delta) + 6\epsilon N \tag{F.3}$$

Finally, using the triangle inequality, we have

$$\begin{aligned}\sum_{i=1}^{N}\mathbb{E}_{x\sim\mathcal{D}_i}\left\|\widehat{f}(\cdot|x) - f^*(\cdot|x)\right\|_1^2 &\le \sum_{i=1}^{N}\mathbb{E}_{x\sim\mathcal{D}_i}\left\|\widehat{f}(\cdot|x) - \overline{f}(\cdot|x)\right\|_1^2 + \sum_{i=1}^{N}\mathbb{E}_{x\sim\mathcal{D}_i}\left\|\overline{f}(\cdot|x) - f^*(\cdot|x)\right\|_1^2 \\ &\le \epsilon^2 N + 12\log(|\overline{\mathcal{F}}|/\delta) + 6\epsilon N\end{aligned}$$

□

Lemma F.1.3 (Optimistic cover for $\mathcal{P}$)**.** *For the function class $\mathcal{P}$ (from (7.4)) there exists an ℓ_1 optimistic cover (Definition 4) with scale ϵ of size $\left(\lceil\frac{S}{\epsilon}\rceil\right)^{S^2AK}$.*

Lemma F.1.4 (Optimistic cover for $\mathcal{P}_1$)**.** *For the function class $\mathcal{P}_1$ (from (7.5)) there exists an ℓ_1 optimistic cover (Definition 4) with scale ϵ of size $\left(\lceil\frac{KS}{\epsilon}\rceil\right)^{S^2A}$.*

Proof of Lemma F.1.3. Denote $\mathcal{P} = \{\mathcal{P}_k\}_{k\in[K]}$, where $\mathcal{P}_k$ denotes the model class for the k-step transitions, and we will construct $\overline{\mathcal{P}} = \{\overline{\mathcal{P}}_k\}_{k\in[K]}$ its optimistic covering set. For any $P \in \mathcal{P}_k$, set its optimistic covering function to be $\overline{P}(s'|s,a,k) = \epsilon'\lceil\frac{P(s'|s,a,k)}{\epsilon'}\rceil$ and include this $\overline{P}$ in $\overline{\mathcal{P}}_k$. Clearly for any (s,a,k,s') we have $\overline{P}(s'|s,a,k) \ge P(s'|s,a,k)$, and $\|P(\cdot|s,a,k) - \overline{P}(\cdot|s,a,k)\|_1 \le \epsilon|S|$ so we need to set $\epsilon' = \epsilon'/|S|$. Then $|\overline{\mathcal{P}}| \le \left(\lceil\frac{|S|}{\epsilon}\rceil\right)^{S^2AK}$. □

Proof of Lemma F.1.4. For any $P, P' \in \mathcal{P}_1$ and $k \in [K]$,

$$\left\|P_{a,k}(\cdot|s) - P'_{a,k}(\cdot|s)\right\| = \left\|P_{a,1}^k(\cdot|s) - (P'_{a,1})^k(\cdot|s)\right\|_1 \le k\left\|P_{a,1}(\cdot|s) - P'_{a,1}(\cdot|s)\right\|_1$$

Let $\mathcal{P}_1' = \{P : \mathcal{S} \times \mathcal{A} \to \Delta(\mathcal{S})\}$ to be the set of all valid one-step transitions. Then it suffices to first find a cover $\overline{\mathcal{P}}_1'$ of $\mathcal{P}_1'$ using a grid of size $\frac{\epsilon}{|S|K}$, then set the cover of $\mathcal{P}_1$ to be $\overline{\mathcal{P}} = \{[(\overline{P}_{a,1})^k]_{a\in\mathcal{A},k\in\mathcal{K}} : \overline{P} \in \overline{\mathcal{P}}_1'\}$. Then $|\overline{\mathcal{P}}| \leq \left(\lceil\frac{K|S|}{\epsilon}\rceil\right)^{S^2A}$. □

F.1.3 Proof of Lemma 7.4.4

We treat $\widehat{P}$ as independent of $\widehat{G}$ in this section, which can be accomplished by splitting the samples N into two folds, one for $\widehat{P}$ estimation, and one for $\widehat{G}$ estimation, which dilutes the final bound by only a small constant factor without changing the dependencies, and we discuss this at the end of this section.

Let $(s_i, a_i, k_i) \sim \mu$ independently. For a fixed $\widehat{P}$, rewrite the regression as

$$\widehat{G} = \operatorname*{argmin}_{f\in\mathcal{F}_{\widehat{P}}} \frac{1}{N}\sum_{i=1}^{N}(f(s_i, a_i, k_i) - g_i)^2$$

where $\mathcal{F}_{\widehat{P}} = \{\mathcal{G}_{R',\widehat{P}} : R' \in [0,1]^{SA}\}$ is the set of aggregated rewards induced by $\widehat{P}$ and any valid one-step reward function. We will bound the error $\left\|\widehat{G} - \mathcal{G}_{R,P}\right\|_{2,\mu}^2$, starting with the decomposition

$$\begin{aligned}\left\|\widehat{G} - \mathcal{G}_{R,P}\right\|_{2,\mu}^2 &= \left\|\widehat{G} - g\right\|_{2,\mu\times G}^2 - \|\mathcal{G}_{R,P} - g\|_{2,\mu\times G}^2 \\ &= \left\|\widehat{G} - g\right\|_{2,\mu\times G}^2 - \left\|\mathcal{G}_{R,\widehat{P}} - g\right\|_{2,\mu\times G}^2 + \left\|\mathcal{G}_{R,\widehat{P}} - g\right\|_{2,\mu\times G}^2 - \|\mathcal{G}_{R,P} - g\|_{2,\mu\times G}^2 \\ &= \underbrace{\left\|\widehat{G} - g\right\|_{2,\mu\times G}^2 - \left\|\mathcal{G}_{R,\widehat{P}} - g\right\|_{2,\mu\times G}^2}_{\text{T1}} + \underbrace{\left\|\mathcal{G}_{R,\widehat{P}} - \mathcal{G}_{R,P}\right\|_{2,\mu}^2}_{\text{T2}}\end{aligned}$$

We can bound T2 as follows:

$$\left\|\mathcal{G}_{R,\widehat{P}} - \mathcal{G}_{R,P}\right\|_{2,\mu}^2 \leq \frac{1}{(1-\gamma)^2}\max_{s,a,k}\left\|\widehat{P}(\cdot|s,a,k) - P(\cdot|s,a,k)\right\|_1^2 := \frac{1}{(1-\gamma)^2}\varepsilon_P \tag{F.4}$$

We have the following bound for T1 from Lemma F.1.5:

$$\left\|\widehat{G} - g\right\|_{2,\mu\times G}^2 - \left\|\mathcal{G}_{R,\widehat{P}} - g\right\|_{2,\mu\times G}^2 \lesssim \sqrt{\frac{SAG_{\max}^2\log(1/\delta')}{N}\left\|\mathcal{G}_{R,\widehat{P}} - \mathcal{G}_{R,P}\right\|_{2,\mu}^2} + \frac{SAG_{\max}^2\log(1/\delta')}{N}$$

Then combining (F.4) and Lemma F.1.5, we have

$$\left\|\widehat{G} - \mathcal{G}_{R,P}\right\|_{2,\mu}^2 \lesssim \sqrt{\frac{SAG_{\max}^2\log(1/\delta')}{N}\left\|\mathcal{G}_{R,\widehat{P}} - \mathcal{G}_{R,P}\right\|_{2,\mu}^2} + \frac{SAG_{\max}^2\log(1/\delta')}{N} + \left\|\mathcal{G}_{R,\widehat{P}} - \mathcal{G}_{R,P}\right\|_{2,\mu}^2$$

$$\leq \frac{1}{1-\gamma}\sqrt{\frac{SAG_{\max}^2 \log(1/\delta')}{N}\varepsilon_P} + \frac{SAG_{\max}^2 \log(1/\delta')}{N} + \frac{1}{(1-\gamma)^2}\varepsilon_P$$

Finally, to translate the above inequality to the ℓ_∞ guarantee of Lemma 7.4.4 in the generative setting (Definition 3),

$$\left\|\widehat{G} - \mathcal{G}_{R,P}\right\|_{2,\mu}^2 = \frac{1}{SAK}\sum_{s,a,k}\left(\widehat{G}(s,a,k) - \mathcal{G}_{R,P}(s,a,k)\right)^2 \geq \frac{1}{SAK}\max_{s,a,k}\left(\widehat{G}(s,a,k) - \mathcal{G}_{R,P}(s,a,k)\right)^2$$

Combining the above two inequalities and rearranging gives the result.

Bounds for timing-aware model-based and timing-naive model-based methods We briefly discuss the bound for timing-aware model-based, and the bound for timing-naive model-based follows the same argument. Due to sample splitting, we call the results in Lemma 7.4.3 and Lemma 7.4.4 with $\frac{1}{2}N$ samples and $\delta' = \frac{1}{2}\delta$, then union bound over the two results. For timing-smart, we have $\varepsilon_P \lesssim \frac{S^2 A \log(K/\delta)}{n}$.

Lemma F.1.5. *Let $\widehat{R}$ be the output of (7.6) with transition $\widehat{P}$, and define $\widehat{G} = \mathcal{G}_{\widehat{R},\widehat{P}}$. With probability at least $1-\delta'$ we have*

$$\left\|\mathcal{G}_{\widehat{R},\widehat{P}} - g\right\|_{2,\mu\times G}^2 - \left\|\mathcal{G}_{R,\widehat{P}} - g\right\|_{2,\mu\times G}^2 \lesssim \sqrt{\frac{SAG_{\max}^2 \log(1/\delta')}{N}\left\|\mathcal{G}_{R,\widehat{P}} - \mathcal{G}_{R,P}\right\|_{2,\mu}^2} + \frac{SAG_{\max}^2 \log(1/\delta')}{N}$$

Proof of Lemma F.1.5. For any f, define the empirical loss $\mathcal{L}_{\mathcal{D}}(f)$ and its expectation $\mathcal{L}_\mu(f)$, respectively, as

$$\mathcal{L}_{\mathcal{D}}(f) := \frac{1}{N}\sum_{i=1}^{N}(f(s_i,a_i,k_i) - g_i)^2$$
$$\mathcal{L}_\mu(f) := \mathbb{E}_{(s,a,k)\sim\mu, g\sim G(s,a,k)}\left[(f(s,a,k)-g)^2\right].$$

Let $\overline{\mathcal{F}}_{\widehat{P}}$ be an ℓ_∞ covering of $\mathcal{F}_{\widehat{P}}$ with scale ϵ, in other words, for any $f \in \mathcal{F}_{\widehat{P}}$ there exists $\overline{f} \in \overline{\mathcal{F}}_{\widehat{P}}$ such that $|\overline{f}(s,a,k) - f(s,a,k)| \leq \epsilon$ for any (s,a,k). Lemma F.1.6 shows that such a covering exists and has cardinality $|\overline{\mathcal{F}}_{\widehat{P}}| = (\lceil 1/(1-\gamma)\epsilon \rceil)^{SA}$. For any $f \in \overline{\mathcal{F}}_{\widehat{P}}$, for a random $(s,a,k,g) \sim \mu \times G$, define

$$Z(f) := (f(s,a,k) - g)^2 - \left(\mathcal{G}_{R,\widehat{P}}(s,a,k) - g\right)^2$$

and let $Z_i(f)$ be the corresponding variable for each $(s_i,a_i,k_i,g_i) \in \mathcal{D}_i$. Observe that $\mathcal{L}_{\mathcal{D}}(f) - \mathcal{L}_{\mathcal{D}}(\mathcal{G}_{R,\widehat{P}}) = \frac{1}{N}\sum_{i=1}^{N} Z_i(f)$. Applying Bernstein's inequality with union bound over $\overline{\mathcal{F}}_{\widehat{P}}$, with probability $\geq 1-\delta$ we have that for any $f \in \overline{\mathcal{F}}_{\widehat{P}}$,

$$\mathbb{E}[Z(f)] - \frac{1}{N}\sum_{i=1}^{N} Z_i(f) \leq \sqrt{\frac{2\mathbb{V}[Z(f)]\log\frac{|\mathcal{F}_{\widehat{P}}|}{\delta}}{N}} + \frac{8G_{\max}^2 \log\frac{|\mathcal{F}_{\widehat{P}}|}{\delta}}{3N} \tag{F.5}$$

For any $f \in \overline{\mathcal{F}}_{\widehat{P}}$, we can upper bound $\mathbb{V}[Z(f)]$ as follows (with the constant $G_{\max}$ such that $g \in [-G_{\max}, G_{\max}]$):

$$\begin{aligned}
\mathbb{V}_{\mu\times P}[Z(f)] &\leq \mathbb{E}_{\mu\times G}[Z(f)^2] \\
&= \mathbb{E}_{\mu\times G}\left[\left((f(s,a,k)-g)^2 - \left(\mathcal{G}_{R,\widehat{P}}(s,a,k)-g\right)^2\right)^2\right] \\
&= \mathbb{E}_{\mu\times G}\left[\left(f(s,a,k)-\mathcal{G}_{R,\widehat{P}}(s,a,k)\right)^2\left(f(s,a,k)+\mathcal{G}_{R,\widehat{P}}(s,a,k)-2g\right)^2\right] \\
&\leq 16G_{\max}^2\mathbb{E}_{\mu}\left[\left(f(s,a,k)-\mathcal{G}_{R,\widehat{P}}(s,a,k)\right)^2\right] \\
&= 16G_{\max}^2\left\|f-\mathcal{G}_{R,\widehat{P}}\right\|_{2,\mu}^2
\end{aligned}$$

Further,

$$\begin{aligned}
\left\|f-\mathcal{G}_{R,\widehat{P}}\right\|_{2,\mu}^2 &\leq 2\left(\|f-\mathcal{G}_{R,P}\|_{2,\mu}^2 + \left\|\mathcal{G}_{R,P}-\mathcal{G}_{R,\widehat{P}}\right\|_{2,\mu}^2\right) \\
&= 2\left(\|f-\mathcal{G}_{R,P}\|_{2,\mu}^2 - \left\|\mathcal{G}_{R,\widehat{P}}-\mathcal{G}_{R,P}\right\|_{2,\mu}^2 + 2\left\|\mathcal{G}_{R,P}-\mathcal{G}_{R,\widehat{P}}\right\|_{2,\mu}^2\right) \\
&= 2\left(\mathcal{L}_\mu(f)-\mathcal{L}_\mu(\mathcal{G}_{R,P}) - \left(\mathcal{L}_\mu(\mathcal{G}_{R,\widehat{P}})-\mathcal{L}_\mu(\mathcal{G}_{R,P})\right) + 2\left\|\mathcal{G}_{R,P}-\mathcal{G}_{R,\widehat{P}}\right\|_{2,\mu}^2\right) \\
&= 2\left(\mathcal{L}_\mu(f)-\mathcal{L}_\mu(\mathcal{G}_{R,\widehat{P}}) + 2\left\|\mathcal{G}_{R,P}-\mathcal{G}_{R,\widehat{P}}\right\|_{2,\mu}^2\right) \\
&= 2\left(\mathbb{E}[Z(f)] + 2\left\|\mathcal{G}_{R,P}-\mathcal{G}_{R,\widehat{P}}\right\|_{2,\mu}^2\right)
\end{aligned}$$

To summarize the above series of inequalities, we have upper bounded the variance as:

$$\mathbb{V}_{\mu\times P}[Z(f)] \leq 32G_{\max}^2\left(\mathbb{E}[Z(f)] + 2\left\|\mathcal{G}_{R,P}-\mathcal{G}_{R,\widehat{P}}\right\|_{2,\mu}^2\right) \tag{F.6}$$

Plugging this back into (F.5), with probability $\geq 1-\delta$ for any $f \in \overline{\mathcal{F}}_{\widehat{P}}$ we have

$$\mathbb{E}[Z(f)] - \frac{1}{N}\sum_{i=1}^{N} Z_i(f) \leq \sqrt{\frac{64G_{\max}^2\left(\mathbb{E}[Z(f)] + 2\left\|\mathcal{G}_{R,P}-\mathcal{G}_{R,\widehat{P}}\right\|_{2,\mu}^2\right)\log\frac{|\overline{\mathcal{F}}_{\widehat{P}}|}{\delta}}{N}} + \frac{8G_{\max}^2\log\frac{|\overline{\mathcal{F}}_{\widehat{P}}|}{\delta}}{3N} \tag{F.7}$$

Now let $\widehat{f} = \mathcal{G}_{\widehat{R},\widehat{P}} \in \mathcal{F}_{\widehat{P}}$, and let $\overline{f}$ be its covering function. Then for any (s,a,k,g),

$$\left|Z(\overline{f}) - Z(\widehat{f})\right| = \left|\left(\overline{f}(s,a,k)-g\right)^2 - \left(\widehat{f}(s,a,k)-g\right)^2\right|$$

$$
\begin{aligned}
&= \left|\left(\overline{f}(s,a,k) - \widehat{f}(s,a,k)\right)\left(\overline{f}(s,a,k) + \widehat{f}(s,a,k) - 2g\right)\right| \\
&\leq 4G_{\max}\left\|\overline{f} - \widehat{f}\right\|_{\infty} \\
&\leq 4G_{\max}\epsilon
\end{aligned}
$$

since $\overline{\mathcal{F}}_{\widehat{P}}$ is an ℓ_∞ cover of scale ϵ. By extension, $\left|\mathbb{E}[Z(\overline{f})] - \mathbb{E}[Z(\widehat{f})]\right| \leq \epsilon$, and same for its empirical approximation. Then

$$
\begin{aligned}
&\mathbb{E}[Z(\widehat{f})] - \frac{1}{N}\sum_{i=1}^{N} Z_i(\widehat{f}) \\
&= \mathbb{E}[Z(\overline{f})] - \frac{1}{N}\sum_{i=1}^{N} Z_i(\overline{f}) + \left(\mathbb{E}[Z(\widehat{f})] - \mathbb{E}[Z(\overline{f})]\right) + \left(\frac{1}{N}\sum_{i=1}^{N} Z_i(\overline{f}) - \frac{1}{N}\sum_{i=1}^{N} Z_i(\widehat{f})\right) \\
&\leq \mathbb{E}[Z(\overline{f})] - \frac{1}{N}\sum_{i=1}^{N} Z_i(\overline{f}) + 2\epsilon \\
&\leq \sqrt{\frac{64G_{\max}^2\left(\mathbb{E}[Z(\overline{f})] + 2\left\|\mathcal{G}_{R,P} - \mathcal{G}_{R,\widehat{P}}\right\|_{2,\mu}^2\right)\log\frac{|\overline{\mathcal{F}}_{\widehat{P}}|}{\delta}}{N}} + \frac{8G_{\max}^2\log\frac{|\overline{\mathcal{F}}_{\widehat{P}}|}{\delta}}{3N} + 2\epsilon \\
&\leq \sqrt{\frac{64G_{\max}^2\left(\mathbb{E}[Z(\widehat{f})] + \epsilon + 2\left\|\mathcal{G}_{R,P} - \mathcal{G}_{R,\widehat{P}}\right\|_{2,\mu}^2\right)\log\frac{|\overline{\mathcal{F}}_{\widehat{P}}|}{\delta}}{N}} + \frac{8G_{\max}^2\log\frac{|\overline{\mathcal{F}}_{\widehat{P}}|}{\delta}}{3N} + 2\epsilon
\end{aligned}
$$

Since $\widehat{f}$ is the regression loss minimizer, $\frac{1}{N}\sum_{i=1}^{N} Z_i(\widehat{f}) \leq \frac{1}{N}\sum_{i=1}^{N} Z_i(\mathcal{G}_{R,\widehat{P}}) = 0$, and we have

$$
\mathbb{E}[Z(\widehat{f})] \leq \sqrt{\frac{64G_{\max}^2\left(\mathbb{E}[Z(\widehat{f})] + \epsilon + 2\left\|\mathcal{G}_{R,P} - \mathcal{G}_{R,\widehat{P}}\right\|_{2,\mu}^2\right)\log\frac{|\overline{\mathcal{F}}_{\widehat{P}}|}{\delta}}{N}} + \frac{8G_{\max}^2\log\frac{|\overline{\mathcal{F}}_{\widehat{P}}|}{\delta}}{3N} + 2\epsilon.
$$

Completing the square gives

$$
\mathbb{E}[Z(\widehat{f})] \leq \sqrt{\frac{128G_{\max}^2\log\frac{|\overline{\mathcal{F}}_{\widehat{P}}|}{\delta}}{N}\left(\epsilon + \left\|\mathcal{G}_{R,P} - \mathcal{G}_{R,\widehat{P}}\right\|_{2,\mu}^2\right)} + \frac{112G_{\max}^2\log\frac{|\overline{\mathcal{F}}_{\widehat{P}}|}{\delta}}{3N} + 28\epsilon
$$

and $\epsilon = \frac{1}{N}$, along with the identity of $\mathbb{E}[Z(\widehat{f})]$ and $|\overline{\mathcal{F}}_{\widehat{P}}|$ from Lemma F.1.6, gives the final bound. □

Lemma F.1.6 (ℓ_∞ cover of $\mathcal{F}_{P'}$)**.** *Let P' be a valid transition matrix in the $timing - as - an -$ action MDP, and let $\mathcal{F}_{P'} = \{\mathcal{G}_{R',\widehat{P}} : R' \in [0,1]^{SA}\}$ be the induced function class of aggregate rewards. There exists an ℓ_∞ cover $\overline{\mathcal{F}}_{P'}$, meaning that for any $f \in \mathcal{F}_{P'}$ there exists $\overline{f} \in \overline{\mathcal{F}}_{P'}$ with $\|f - \overline{f}\|_\infty \leq \epsilon$, of cardinality $\left(\lceil\frac{1}{(1-\gamma)\epsilon}\rceil\right)^{SA}$.*

Proof of Lemma F.1.6. $\overline{\mathcal{F}}_{P'}$ is induced by an ℓ_∞ covering of the one-step reward functions. Let $\mathcal{R} = \{R' : R' \in [0,1]^{SA}\}$, and let $\overline{\mathcal{R}}$ be its ℓ_∞ covering of scale ϵ', such that any $R' \in \mathcal{R}$ has $\overline{R} \in \overline{\mathcal{R}}$ with $\|R' - \overline{R}\|_\infty \le \epsilon'$. It is easy to verify that the cardinality of $\overline{\mathcal{R}}$ is $\lceil \frac{1}{\epsilon'} \rceil^{SA}$, by discretizing the interval $[0,1]$ at a scale of ϵ' for each (s,a). Then we define $\overline{\mathcal{F}}_{P'} = \{\mathcal{G}_{\overline{R},\widehat{P}} : \overline{R} \in \overline{\mathcal{R}}\}$.

For any $f = \mathcal{G}_{R',P'} \in \mathcal{F}_{P'}$, consider $\overline{f} = \mathcal{G}_{\overline{R},P'} \in \overline{\mathcal{F}}_{P'}$, where $\|\overline{R} - R'\|_\infty \le \epsilon'$. Then using the definition of $\mathcal{G}$ in (7.7), for any (s,a,k) we have

$$\begin{aligned}
\left|f(s,a,k) - \overline{f}(s,a,k)\right| &= \left|\sum_{\tau=0}^{k-1} \gamma^\tau \sum_{s'} P'(s'|s,a,k)\left(R'(s',a) - \overline{R}(s',a)\right)\right| \\
&\le \sum_{\tau=0}^{k-1} \gamma^\tau \sum_{s'} P'(s'|s,a,k)\left|R'(s',a) - \overline{R}(s',a)\right| \\
&\le \frac{1}{1-\gamma}\|R' - R\|_\infty \\
&\le \frac{\epsilon'}{1-\gamma}
\end{aligned}$$

Choosing $\epsilon' = (1-\gamma)\epsilon$ gives the result. □

F.1.4 Proof of Proposition 4

Using the fact that Q^* is the unique solution to the timing-as-an-action Bellman equation in (7.2), and the definition of $\widehat{Q}$ in (7.8), for a fixed (s,a,k) we have

$$\begin{aligned}
&\left|Q^*(s,a,k) - \widehat{Q}(s,a,k)\right| \\
&= \left|\mathcal{G}_{R,P}(s,a,k) - \mathcal{G}_{\widehat{R},\widehat{P}}(s,a,k) + \gamma^k \sum_{s'}\left(P(s'|s,a,k)\max_{a',k'} Q^*(s',a',k') - \widehat{P}(s'|s,a,k)\widehat{Q}(s',a',k')\right)\right| \\
&\le \left|\mathcal{G}_{R,P}(s,a,k) - \mathcal{G}_{\widehat{R},\widehat{P}}(s,a,k)\right| + \gamma^k \sum_{s'}\left|P(s'|s,a,k) - \widehat{P}(s'|s,a,k)\right| \max_{a',k'}|Q^*(s',a',k')| \\
&+ \gamma^k \sum_{s'} \widehat{P}(s'|s,a)\left|\max_{a',k'} Q^*(s',a',k') - \max_{a',k'} \widehat{Q}(s',a',k')\right| \\
&\le \left\|\mathcal{G}_{R,P} - \mathcal{G}_{\widehat{R},\widehat{P}}\right\|_\infty + \frac{\gamma^k}{1-\gamma}\max_{s,a,k}\left\|P(\cdot|s,a,k) - \widehat{P}(\cdot|s,a,k)\right\|_1 + \gamma^k\left\|Q^* - \widehat{Q}\right\|_\infty
\end{aligned}$$

where we use the fact $\widehat{P}$ is a valid transition and that $\|Q^*\|_\infty \le 1/(1-\gamma)$ in the last inequality above. Since this holds for any (s,a,k), we have

$$\left\|Q^* - \widehat{Q}\right\|_\infty \le \left\|\mathcal{G}_{R,P} - \mathcal{G}_{\widehat{R},\widehat{P}}\right\|_\infty + \frac{\gamma^k}{1-\gamma}\max_{s,a,k}\left\|P(\cdot|s,a,k) - \widehat{P}(\cdot|s,a,k)\right\|_1 + \gamma^k\left\|Q^* - \widehat{Q}\right\|_\infty,$$

and rearranging the above inequality gives the result.

F.2 Implementation Details

For the transition estimation experiments, optimization is done using SGD with a learning rate of 0.01. For the RL environments, optimization is done using the Adam optimizer with a batch size of 500 and initial learning rate of 10^{-3} for $\widehat{P}$ and 0.1 for $\widehat{R}$, and Q-value iteration is done to convergence, where convergence is defined as a change of less than 10^{-5} for at least two iterations. For additional details, see the code provided in the supplement, which reproduces all results in the paper. All experiments were conducted on a single machine with 24 CPUs and 1 Tital RTX GPU. Windy grid experiments were conducted on the GPU whereas all other experiments were conducted on CPU.

F.3 RL Environment Details

The true one-step transition probabilities for the disease progression simulator are as follows:

```
true_P = np.array([
    [
        [0.89, 0.1, 0.01],
        [0.15, 0.8, 0.05],
        [0.0, 0.0, 1.0]
    ],
    [
        [0.1, 0.89, 0.01],
        [0.8, 0.15, 0.05],
        [0.0, 0.0, 1.0]
    ],
])
```

which is a $A \times S \times S$ matrix, and the (i, j, k)-th element is the probability of transitioning to state k conditioned on taking action i from state j.

F.4 RL Experiment Results

Reward Estimation In the generative setting, averaged over 30 trials, we characterize the aggregate reward estimation error when the reward model is learned in conjunction with the timing-aware and timing-naive models. This estimation error is compared to when the reward model is learned but paired with an oracle transition model, as well as against simply averaging the rewards gathered from tuples of experience taking each action from each state.

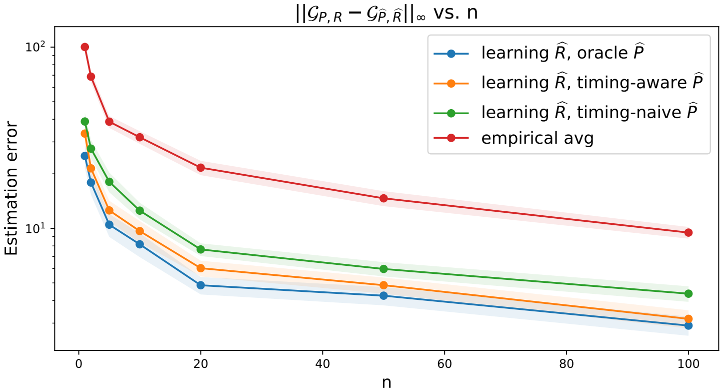


Figure F.1: Average reward estimation error $||\mathcal{G}_{P,R} - \mathcal{G}_{\widehat{P},\widehat{R}}||_\infty$ in the generative setting over 30 trials, with 95% confidence intervals. The number of repetitions $n = [1, 2, 5, 10, 20, 50, 100]$ is the number of per-(s, a, k) samples drawn for all $(s, a, k) \in \mathcal{S} \times \mathcal{A} \times \mathcal{K}$.

Overall, simply tracking the average empirical reward is the least sample efficient, followed by learning $\widehat{R}$ in conjunction with timing-naive $\widehat{P}$, followed by timing-aware $\widehat{P}$, and finally using the oracle P when learning $\widehat{R}$.

Distribution of Actions Taken Next, we include visualizations of distribution of actions taken by the policy in each environment (Figure F.2) and the policies learned in the windy grid environment (Figure F.3).

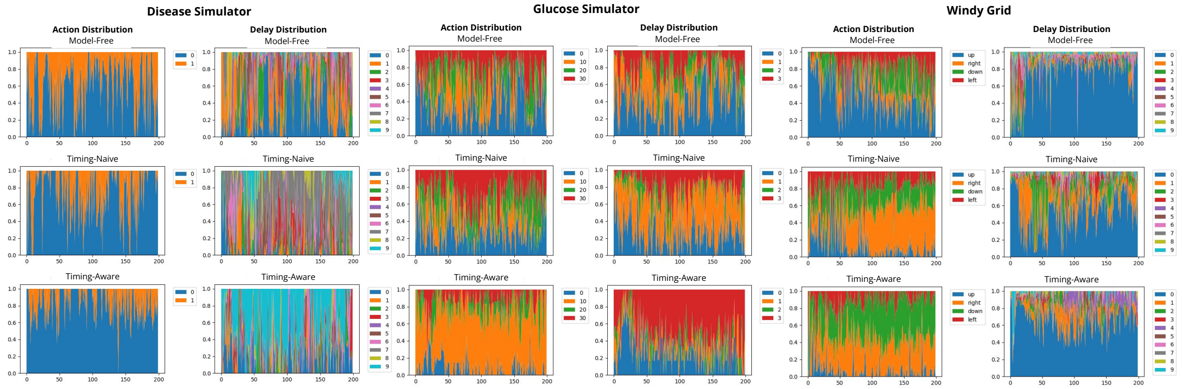


Figure F.2: Distribution of actions (sub-left) and delays (sub-right) taken by the policy in the disease progression (left), glucose monitoring (middle), and windy grid (right) environments. In the disease simulator, action 0 corresponds to "don't treat" and action 1 corresponds to "treat." Delays are 0-indexed, but delay 0 corresponds to a one-timestep delay, delay 1 corresponds to a two-timestep delay, etc.

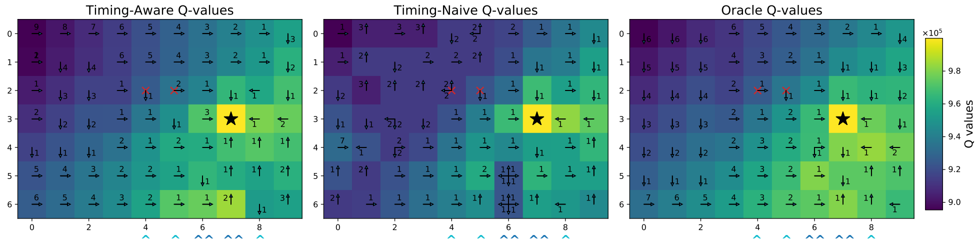


Figure F.3: Maximum Q-values learned in each state from the timing-aware and timing-naive model-based methods after 200 episodes, as well as the oracle Q values. In each grid state, the arrow gives the direction of the action, the number gives the delay, and the color gives the value. ˆ or ˆˆ indicate a stochastic wind which pushes the agent up with probability 0.5 for one or two squares, respectively. The star is the goal state, and the x's are hazard states.

Bibliography

[1] Peter R Winters. “Forecasting sales by exponentially weighted moving averages”. In: *Management science* 6.3 (1960), pp. 324–342 (cit. on p. 80).

[2] S. Eilon and J. Elmaleh. “An Evaluation of Alternative Inventory Control Policies”. In: *International Journal of Production Research* 7.1 (1968), pp. 1–14. DOI: 10.1080/00207546808929792. eprint: https://doi.org/10.1080/00207546808929792. URL: https://doi.org/10.1080/00207546808929792 (cit. on p. 80).

[3] Donald B Rubin. “Inference and missing data”. In: *Biometrika* 63.3 (1976), pp. 581–592 (cit. on pp. 60, 64).

[4] Jr Harrell Frank E. and et al. “Evaluating the Yield of Medical Tests”. In: *JAMA* 247.18 (May 1982), pp. 2543–2546. ISSN: 0098-7484 (cit. on p. 11).

[5] Everette S Gardner Jr. “Exponential smoothing: The state of the art”. In: *Journal of forecasting* 4.1 (1985), pp. 1–28 (cit. on p. 80).

[6] Mustafa Dosemeci, Sholom Wacholder, and Jay H Lubin. “Does nondifferential misclassification of exposure always bias a true effect toward the null value?” In: *American Journal of Epidemiology* 132.4 (1990), pp. 746–748 (cit. on p. 61).

[7] Christopher JCH Watkins and Peter Dayan. “Q-learning”. In: *Machine learning* 8 (1992), pp. 279–292 (cit. on p. 97).

[8] Hermann Brenner and Dana Loomis. “Varied forms of bias due to nondifferential error in measuring exposure”. In: *Epidemiology* (1994), pp. 510–517 (cit. on p. 61).

[9] James M Robins, Andrea Rotnitzky, and Lue Ping Zhao. “Estimation of regression coefficients when some regressors are not always observed”. In: *Journal of the American Statistical Association* 89.427 (1994), pp. 846–866 (cit. on p. 60).

[10] Stephen W Duffy, Hsiu-Hsi Chen, Laszlo Tabar, and Nicholas E Day. “Estimation of mean sojourn time in breast cancer screening using a Markov chain model of both entry to and exit from the preclinical detectable phase”. In: *Statistics in medicine* 14.14 (1995), pp. 1531–1543 (cit. on p. 103).

[11] PhilipS Wells, Jack Hirsh, David R Anderson, Anthony W A Lensing, Gary Foster, Clive Kearon, Jeffrey Weitz, Robert D’Ovidio, Alberto Cogo, Paolo Prandoni, et al. “Accuracy of clinical assessment of deep-vein thrombosis”. In: *The Lancet* 345.8961 (1995), pp. 1326–1330 (cit. on pp. 32, 33).

[12] Hsiu-Hsi Chen, Stephen W Duffy, and Laszlo Tabar. “A Markov chain method to estimate the tumour progression rate from preclinical to clinical phase, sensitivity and positive predictive value for mammography in breast cancer screening”. In: *Journal of the Royal Statistical Society Series D: The Statistician* 45.3 (1996), pp. 307–317 (cit. on p. 103).

[13] Leslie Pack Kaelbling, Michael L Littman, and Andrew W Moore. “Reinforcement learning: A survey”. In: *Journal of artificial intelligence research* 4 (1996), pp. 237–285 (cit. on p. 96).

[14] Donald B. Rubin. “Multiple Imputation after 18+ Years”. In: *Journal of the American Statistical Association* 91.434 (1996), pp. 473–489. DOI: 10.1080/01621459.1996.10476908 (cit. on p. 60).

[15] A. C. Davison and D. V. Hinkley. *Bootstrap Methods and their Application*. Cambridge Series in Statistical and Probabilistic Mathematics. Cambridge University Press, 1997. DOI: 10.1017/CBO9780511802843 (cit. on pp. 21, 23).

[16] Michael J Fine, Thomas E Auble, and et al. “A prediction rule to identify low-risk patients with community-acquired pneumonia”. In: *NEJM* 336.4 (1997), pp. 243–250 (cit. on pp. 8, 9, 12, 14).

[17] Sepp Hochreiter and Jürgen Schmidhuber. “Long short-term memory”. In: *Neural computation* 9.8 (1997), pp. 1735–1780 (cit. on p. 80).

[18] Hau L Lee, V Padmanabhan, and Seungjin Whang. “The Bullwhip Effect In Supply Chains1”. In: *Sloan Management Review* 38.3 (1997), pp. 93–102 (cit. on pp. 78, 80).

[19] Spyros Makridakis and Michele Hibon. “ARMA models and the Box–Jenkins methodology”. In: *Journal of forecasting* 16.3 (1997), pp. 147–163 (cit. on p. 80).

[20] Richard S Sutton, Doina Precup, and Satinder Singh. “Intra-Option Learning about Temporally Abstract Actions.” In: *ICML*. Vol. 98. 1998, pp. 556–564 (cit. on p. 95).

[21] Peter WF Wilson, Ralph B D’Agostino, Daniel Levy, Albert M Belanger, Halit Silbershatz, and William B Kannel. “Prediction of coronary heart disease using risk factor categories”. In: *Circulation* 97.18 (1998), pp. 1837–1847 (cit. on pp. 32, 33).

[22] Nicholas C Petruzzi and Maqbool Dada. “Pricing and the newsvendor problem: A review with extensions”. In: *Operations research* 47.2 (1999), pp. 183–194 (cit. on p. 79).

[23] Richard S Sutton, Doina Precup, and Satinder Singh. “Between MDPs and semi-MDPs: A framework for temporal abstraction in reinforcement learning”. In: *Artificial intelligence* 112.1-2 (1999), pp. 181–211 (cit. on p. 95).

[24] V. Assimakopoulos and K. Nikolopoulos. “The theta model: a decomposition approach to forecasting”. In: *International Journal of Forecasting* 16.4 (2000). The M3- Competition, pp. 521–530. ISSN: 0169-2070. DOI: https://doi.org/10.1016/S0169-2070(00)00066-2. URL: https://www.sciencedirect.com/science/article/pii/S0169207000000662 (cit. on p. 80).

[25] Anjali Dhond, Amar Gupta, and Sanjeev Vadhavkar. “Data mining techniques for optimizing inventories for electronic commerce”. In: *Proceedings of the sixth ACM SIGKDD international conference on Knowledge discovery and data mining*. 2000, pp. 480–486 (cit. on p. 80).

[26] Thomas G Dietterich. “Hierarchical reinforcement learning with the MAXQ value function decomposition”. In: *Journal of artificial intelligence research* 13 (2000), pp. 227–303 (cit. on p. 95).

[27] Ary L Goldberger, Luis AN Amaral, Leon Glass, Jeffrey M Hausdorff, Plamen Ch Ivanov, Roger G Mark, Joseph E Mietus, George B Moody, Chung-Kang Peng, and H Eugene Stanley. "PhysioBank, PhysioToolkit, and PhysioNet: components of a new research resource for complex physiologic signals". In: *circulation* 101.23 (2000), e215–e220 (cit. on p. 30).

[28] Spyros Makridakis and Michèle Hibon. "The M3-Competition: results, conclusions and implications". In: *International Journal of Forecasting* 16.4 (2000). The M3- Competition, pp. 451–476. ISSN: 0169-2070. DOI: https://doi.org/10.1016/S0169-2070(00)00057-1. URL: https://www.sciencedirect.com/science/article/pii/S0169207000000571 (cit. on p. 80).

[29] Spyros Makridakis and Michele Hibon. "The M3-Competition: results, conclusions and implications". In: *International Journal of Forecasting* 16.4 (2000), pp. 451–476. URL: https://ideas.repec.org/a/eee/intfor/v16y2000i4p451-476.html (cit. on p. 86).

[30] Hidetoshi Shimodaira. "Improving predictive inference under covariate shift by weighting the log-likelihood function". In: *Journal of Statistical Planning and Inference* 90.2 (2000), pp. 227–244. ISSN: 0378-3758. DOI: https://doi.org/10.1016/S0378-3758(00)00115-4. URL: https://www.sciencedirect.com/science/article/pii/S0378375800001154 (cit. on p. 2).

[31] Hidetoshi Shimodaira. "Improving predictive inference under covariate shift by weighting the log-likelihood function". In: *Journal of Statistical Planning and Inference* 90.2 (2000), pp. 227–244 (cit. on pp. 31, 61, 64).

[32] James W Taylor. "A quantile regression neural network approach to estimating the conditional density of multiperiod returns". In: *Journal of forecasting* 19.4 (2000), pp. 299–311 (cit. on p. 79).

[33] Lyn C Thomas. "A survey of credit and behavioural scoring: forecasting financial risk of lending to consumers". In: *International journal of forecasting* 16.2 (2000), pp. 149–172 (cit. on p. 77).

[34] Patrick S Kamath, Russell H Wiesner, Michael Malinchoc, Walter Kremers, Terry M Therneau, Catherine L Kosberg, Gennaro D'Amico, E Rolland Dickson, and W Ray Kim. "A model to predict survival in patients with end-stage liver disease". In: *Hepatology* 33.2 (2001), pp. 464–470 (cit. on pp. 32, 33).

[35] N Gregory Mankiw and Ricardo Reis. "Sticky information versus sticky prices: a proposal to replace the New Keynesian Phillips curve". In: *The Quarterly Journal of Economics* 117.4 (2002), pp. 1295–1328 (cit. on p. 92).

[36] Marco Saerens, Patrice Latinne, and Christine Decaestecker. "Adjusting the Outputs of a Classifier to New a Priori Probabilities: A Simple Procedure". In: *Neural Computation* (2002) (cit. on pp. 31, 61).

[37] Andrew G Barto and Sridhar Mahadevan. "Recent advances in hierarchical reinforcement learning". In: *Discrete event dynamic systems* 13.1-2 (2003), pp. 41–77 (cit. on p. 95).

[38] Rob J Hyndman and Baki Billah. "Unmasking the Theta method". In: *International Journal of Forecasting* 19.2 (2003), pp. 287–290 (cit. on p. 80).

[39] Wei Shen Lim, MM Van der Eerden, and et al. "Defining community acquired pneumonia severity on presentation to hospital: an international derivation and validation study". In: *Thorax* 58.5 (2003), pp. 377–382 (cit. on pp. 8, 9).

[40] Lim W, Van Der Eerden M, and et al. “Defining community acquired pneumonia severity on presentation to hospital: An international derivation and validation study”. In: *Thorax* 58.5 (2003), pp. 377–382 (cit. on pp. 9, 12, 14).

[41] M-F Yen, L Tabar, B Vitak, RA Smith, H-H Chen, and SW Duffy. “Quantifying the potential problem of overdiagnosis of ductal carcinoma in situ in breast cancer screening”. In: *European journal of cancer* 39.12 (2003), pp. 1746–1754 (cit. on p. 103).

[42] Alexey Tsymbal. “The problem of concept drift: definitions and related work”. In: *Computer Science Department, Trinity College Dublin* 106.2 (2004), p. 58 (cit. on pp. 31, 61).

[43] Bianca Zadrozny. “Learning and evaluating classifiers under sample selection bias”. In: *Proceedings of the twenty-first international conference on Machine learning*. 2004, p. 114 (cit. on pp. 2, 31, 61, 64).

[44] Dimitris Bertsimas and Aurélie Thiele. “A data-driven approach to newsvendor problems”. In: *Working Papere, Massachusetts Institute of Technology* 51 (2005) (cit. on p. 79).

[45] Kenneth Gilbert. “An ARIMA supply chain model”. In: *Management Science* 51.2 (2005), pp. 305–310 (cit. on p. 81).

[46] Haitao Chu, Zhaojie Wang, Stephen R Cole, and Sander Greenland. “Sensitivity analysis of misclassification: a graphical and a Bayesian approach”. In: *Annals of Epidemiology* 16.11 (2006), pp. 834–841 (cit. on p. 61).

[47] Jiayuan Huang, Arthur Gretton, Karsten Borgwardt, Bernhard Schölkopf, and Alex Smola. “Correcting sample selection bias by unlabeled data”. In: *Advances in Neural Information Processing Systems (NeurIPS)* 19 (2006) (cit. on pp. 31, 61, 64).

[48] Anne Helene Olsen, Olorunsola F Agbaje, Jonathan P Myles, Elsebeth Lynge, and Stephen W Duffy. “Overdiagnosis, sojourn time, and sensitivity in the Copenhagen mammography screening program”. In: *The Breast Journal* 12.4 (2006), pp. 338–342 (cit. on p. 103).

[49] Jack E Zimmerman, Andrew A Kramer, and et al. “Acute Physiology and Chronic Health Evaluation (APACHE) IV: hospital mortality assessment for today’s critically ill patients”. In: *Critical Care Medicine* 34.5 (2006), pp. 1297–1310 (cit. on p. 9).

[50] Sunil Chopra, Peter Meindl, and Dharam Vir Kalra. *Supply Chain Management by Pearson*. Pearson Education India, 2007 (cit. on p. 78).

[51] Clifton D Fuller, Samuel J Wang, Charles R Thomas Jr, Henry T Hoffman, Randal S Weber, and David I Rosenthal. “Conditional survival in head and neck squamous cell carcinoma: results from the SEER dataset 1973–1998”. In: *Cancer: Interdisciplinary International Journal of the American Cancer Society* 109.7 (2007), pp. 1331–1343 (cit. on p. 118).

[52] Masashi Sugiyama, Shinichi Nakajima, Hisashi Kashima, Paul Buenau, and Motoaki Kawanabe. “Direct Importance Estimation with Model Selection and Its Application to Covariate Shift Adaptation”. In: *Advances in Neural Information Processing Systems*. Ed. by J. Platt, D. Koller, Y. Singer, and S. Roweis. Vol. 20. Curran Associates, Inc., 2007. URL: https://proceedings.neurips.cc/paper_files/paper/2007/file/be83ab3ecd0db773eb2dc1b0a17836a1-Paper.pdf (cit. on p. 2).

[53] Masashi Sugiyama, Shinichi Nakajima, Hisashi Kashima, Paul Buenau, and Motoaki Kawanabe. “Direct importance estimation with model selection and its application to covariate shift adaptation”. In: *Advances in Neural Information Processing Systems (NeurIPS)* 20 (2007) (cit. on pp. 31, 61, 64).

[54] Patrick G. P. Charles, Rory Wolfe, and et al. “SMART-COP: A Tool for Predicting the Need for Intensive Respiratory or Vasopressor Support in Community-Acquired Pneumonia”. In: *Clinical Infectious Diseases* 47.3 (2008), pp. 375–384 (cit. on pp. 8, 9, 12, 14).

[55] Mehee Choi, Clifton D Fuller, Charles R Thomas Jr, and Samuel J Wang. “Conditional survival in ovarian cancer: results from the SEER dataset 1988–2001”. In: *Gynecologic oncology* 109.2 (2008), pp. 203–209 (cit. on p. 118).

[56] Charles Elkan and Keith Noto. “Learning classifiers from only positive and unlabeled data”. In: *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*. 2008, pp. 213–220 (cit. on pp. 48, 50).

[57] Charles Elkan and Keith Noto. “Learning classifiers from only positive and unlabeled data”. In: *SIGKDD International Conference on Knowledge Discovery and Data Mining*. 2008, pp. 213–220 (cit. on p. 61).

[58] AJ Six, BE Backus, and JC Kelder. “Chest pain in the emergency room: value of the HEART score”. In: *Netherlands Heart Journal* 16.6 (2008), pp. 191–196 (cit. on p. 33).

[59] Nancy L Stokey. *The Economics of Inaction: Stochastic Control models with fixed costs*. Princeton University Press, 2008 (cit. on p. 92).

[60] William Clarke and Boris Kovatchev. “Statistical tools to analyze continuous glucose monitor data”. In: *Diabetes technology & therapeutics* 11.S1 (2009), S–45 (cit. on p. 103).

[61] Arthur Gretton, Alex Smola, Jiayuan Huang, Marcel Schmittfull, Karsten Borgwardt, Bernhard Schölkopf, et al. “Covariate shift by kernel mean matching”. In: *Dataset shift in machine learning* 3.4 (2009), p. 5 (cit. on pp. 2, 31, 61, 64).

[62] Spencer S Jones, R Scott Evans, Todd L Allen, Alun Thomas, Peter J Haug, Shari J Welch, and Gregory L Snow. “A multivariate time series approach to modeling and forecasting demand in the emergency department”. In: *Journal of biomedical informatics* 42.1 (2009), pp. 123–139 (cit. on pp. 77, 80).

[63] Amos Storkey et al. “When training and test sets are different: characterizing learning transfer”. In: *Dataset shift in machine learning* 30.3-28 (2009), p. 6 (cit. on pp. 2, 31, 61).

[64] Aris A Syntetos, John E Boylan, and Stephen M Disney. “Forecasting for inventory planning: a 50-year review”. In: *Journal of the Operational Research Society* 60 (2009), S149–S160 (cit. on p. 78).

[65] Matthew J Bizzarro, Steven A Conrad, and et al. “Infections acquired during extracorporeal membrane oxygenation in neonates, children, and adults”. In: *Pediatric Critical Care Medicine* 12.3 (2011), pp. 277–281 (cit. on p. 9).

[66] Marc Peter Deisenroth, Carl Edward Rasmussen, and Dieter Fox. “Learning to control a low-cost manipulator using data-efficient reinforcement learning”. In: *Robotics: Science and Systems VII* 7 (2011), pp. 57–64 (cit. on p. 96).

[67] Christopher Jackson. “Multi-state models for panel data: the msm package for R”. In: *Journal of statistical software* 38 (2011), pp. 1–28 (cit. on p. 93).

[68] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. “Scikit-learn: Machine learning in Python”. In: *the Journal of machine Learning research* 12 (2011), pp. 2825–2830 (cit. on p. 50).

[69] Noah Simon, Jerome Friedman, and et al. “Regularization Paths for Cox’s Proportional Hazards Model via Coordinate Descent”. In: *Journal of Statistical Software* 39.5 (2011), pp. 1–13 (cit. on p. 11).

[70] Daniel J. Stekhoven and Peter Bühlmann. “MissForest—non-parametric missing value imputation for mixed-type data”. In: *Bioinformatics* 28.1 (Oct. 2011), pp. 112–118. ISSN: 1367-4803 (cit. on p. 10).

[71] Stef Van Buuren and Karin Groothuis-Oudshoorn. “MICE: Multivariate imputation by chained equations in R”. In: *Journal of Statistical Software* 45 (2011), pp. 1–67 (cit. on p. 60).

[72] Mohammad Gheshlaghi Azar, Rémi Munos, and Bert Kappen. “On the sample complexity of reinforcement learning with a generative model”. In: *arXiv preprint arXiv:1206.6461* (2012) (cit. on pp. 97, 100).

[73] Christoph Bergmeir and José M. Benítez. “On the use of cross-validation for time series predictor evaluation”. In: *Information Sciences* 191 (2012). Data Mining for Software Trustworthiness, pp. 192–213 (cit. on p. 32).

[74] Gidon Eshel. *Spatio-temporal data analysis*. Princeton University Press, 2012 (cit. on p. 21).

[75] Rolf HH Groenwold, Ian R White, A Rogier T Donders, James R Carpenter, Douglas G Altman, and Karel GM Moons. “Missing covariate data in clinical research: when and when not to use the missing-indicator method for analysis”. In: *Canadian Medical Association Journal* 184.11 (2012), pp. 1265–1269 (cit. on p. 61).

[76] Christophe Marti, Nicolas Garin, and et al. “Prediction of severe community-acquired pneumonia: A systematic review and meta-analysis”. In: *Critical Care* 16.4 (2012). ISSN: 13648535 (cit. on p. 8).

[77] Daniel J Stekhoven and Peter Bühlmann. “MissForest—non-parametric missing value imputation for mixed-type data”. In: *Bioinformatics* 28.1 (2012), pp. 112–118 (cit. on p. 61).

[78] L Suganthi and Anand A Samuel. “Energy models for demand forecasting—A review”. In: *Renewable and sustainable energy reviews* 16.2 (2012), pp. 1223–1240 (cit. on p. 80).

[79] Nazli Turken, Yinliang Tan, Asoo J Vakharia, Lan Wang, Ruoxuan Wang, and Arda Yenipazarli. “The multi-product newsvendor problem: Review, extensions, and directions for future research”. In: *Handbook of Newsvendor Problems: Models, Extensions and Applications* (2012), pp. 3–39 (cit. on p. 79).

[80] Özden Gür Ali and Kübra Yaman. “Selecting rows and columns for training support vector regression models with large retail datasets”. In: *European Journal of Operational Research* 226.3 (2013), pp. 471–480 (cit. on p. 79).

[81] Michael G Marmot, DG Altman, DA Cameron, JA Dewar, SG Thompson, and Maggie Wilcox. “The benefits and harms of breast cancer screening: an independent review”. In: *British journal of cancer* 108.11 (2013), pp. 2205–2240 (cit. on p. 93).

[82] Razvan Pascanu, Tomas Mikolov, and Yoshua Bengio. “On the difficulty of training recurrent neural networks”. In: *International conference on machine learning*. Pmlr. 2013, pp. 1310–1318 (cit. on p. 84).

[83] Cosma Shalizi. *Advanced data analysis from an elementary point of view*. 2013 (cit. on p. 23).

[84] Stefan Wager, Sida Wang, and Percy S Liang. "Dropout Training as Adaptive Regularization". In: *Advances in Neural Information Processing Systems (NeurIPS).* 2013 (cit. on pp. 65, 172).

[85] Xiaodan Yu, Zhiquan Qi, and Yuanmeng Zhao. "Support vector regression for newspaper/magazine sales forecasting". In: *Procedia Computer Science* 17 (2013), pp. 1055–1062 (cit. on p. 79).

[86] Kun Zhang, Bernhard Schölkopf, Krikamol Muandet, and Zhikun Wang. "Domain adaptation under target and conditional shift". In: *International conference on machine learning.* PMLR. 2013, pp. 819–827 (cit. on pp. 2, 31, 61).

[87] Emmanuel A Donkor, Thomas A Mazzuchi, Refik Soyer, and J Alan Roberson. "Urban water demand forecasting: review of methods and models". In: *Journal of Water Resources Planning and Management* 140.2 (2014), pp. 146–159 (cit. on p. 80).

[88] João Gama, Indrė Žliobaitė, Albert Bifet, Mykola Pechenizkiy, and Abdelhamid Bouchachia. "A survey on concept drift adaptation". In: *ACM computing surveys (CSUR)* 46.4 (2014), pp. 1–37 (cit. on pp. 31, 61).

[89] Chiara Dalla Man, Francesco Micheletto, Dayu Lv, Marc Breton, Boris Kovatchev, and Claudio Cobelli. "The UVA/PADOVA type 1 diabetes simulator: new features". In: *Journal of diabetes science and technology* 8.1 (2014), pp. 26–34 (cit. on p. 103).

[90] Xiang Wang, David Sontag, and Fei Wang. "Unsupervised learning of disease progression models". In: *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining.* 2014, pp. 85–94 (cit. on p. 1).

[91] Toshiyuki Aokage, Kenneth Palmér, and et al. "Extracorporeal membrane oxygenation for acute respiratory distress syndrome". In: *Journal of Intensive Care* 3.1 (2015). ISSN: 20520492 (cit. on p. 9).

[92] George EP Box, Gwilym M Jenkins, Gregory C Reinsel, and Greta M Ljung. *Time series analysis: forecasting and control.* John Wiley & Sons, 2015 (cit. on p. 80).

[93] Alex Braylan, Mark Hollenbeck, Elliot Meyerson, and Risto Miikkulainen. "Frame skip is a powerful parameter for learning to play atari". In: *Workshops at the twenty-ninth AAAI conference on artificial intelligence.* 2015 (cit. on p. 96).

[94] Kenneth Jung and Nigam H Shah. "Implications of non-stationarity on predictive modeling using EHRs". In: *Journal of biomedical informatics* 58 (2015), pp. 168–174 (cit. on p. 32).

[95] Surveillance Research Program. "National Cancer Institute SEER* Stat software". In: *Surveillance Research Program* (2015) (cit. on pp. 34, 119).

[96] Matthieu Schmidt, Aidan Burrell, and et al. "Predicting survival after ECMO for refractory cardiogenic shock: The survival after veno-arterial-ECMO (SAVE)-score". In: *European Heart Journal* 36.33 (2015), pp. 2246–2256. ISSN: 15229645 (cit. on p. 9).

[97] Emanuela Taioli, Andrea S Wolf, Marlene Camacho-Rivera, Andrew Kaufman, Dong-Seok Lee, Daniel Nicastri, Kenneth Rosenzweig, and Raja M Flores. "Determinants of survival in malignant pleural mesothelioma: a surveillance, epidemiology, and end results (SEER) study of 14,228 patients". In: *PloS one* 10.12 (2015), e0145039 (cit. on p. 118).

[98] Graham Elliott and Allan Timmermann. "Forecasting in economics and finance". In: *Annual Review of Economics* 8 (2016), pp. 81–110 (cit. on p. 77).

[99] Kris Johnson Ferreira, Bin Hong Alex Lee, and David Simchi-Levi. "Analytics for an online retailer: Demand forecasting and price optimization". In: *Manufacturing & service operations management* 18.1 (2016), pp. 69–88 (cit. on p. 80).

[100] Varun Gulshan, Lily Peng, Marc Coram, Martin C Stumpe, Derek Wu, Arunachalam Narayanaswamy, Subhashini Venugopalan, Kasumi Widner, Tom Madams, Jorge Cuadros, Ramasamy Kim, Rajiv Raman, Philip Q Nelson, Jessica Mega, and Dale Webster. "Development and Validation of a Deep Learning Algorithm for Detection of Diabetic Retinopathy in Retinal Fundus Photographs". In: *JAMA* (2016) (cit. on pp. 1, 32).

[101] Yoni Halpern, Steven Horng, Youngduck Choi, and David Sontag. "Electronic medical record phenotyping using the anchor and learn framework". In: *Journal of the American Medical Informatics Association* 23.4 (2016), pp. 731–740 (cit. on pp. 48–50).

[102] Alistair E. W. Johnson, Tom J. Pollard, and et al. "MIMIC-III, a freely accessible critical care database". In: *Scientific Data* 3.1 (2016), p. 160035 (cit. on p. 10).

[103] Zachary C Lipton, David C Kale, Charles Elkan, and Randall Wetzel. "Learning to diagnose with LSTM recurrent neural networks". In: *International Conference on Learning Representations (ICLR)*. 2016 (cit. on pp. 1, 32).

[104] Zachary C Lipton, David C Kale, Randall Wetzel, et al. "Modeling missing data in clinical time series with rnns". In: *Machine Learning for Healthcare* 56 (2016) (cit. on p. 61).

[105] Aaron van den Oord, Sander Dieleman, Heiga Zen, Karen Simonyan, Oriol Vinyals, Alex Graves, Nal Kalchbrenner, Andrew Senior, and Koray Kavukcuoglu. "Wavenet: A generative model for raw audio". In: *arXiv preprint arXiv:1609.03499* (2016) (cit. on p. 80).

[106] Xun Wang and Stephen M. Disney. "The bullwhip effect: Progress, trends and directions". In: *European Journal of Operational Research* 250.3 (2016), pp. 691–701. ISSN: 0377-2217. DOI: https://doi.org/10.1016/j.ejor.2015.07.022. URL: https://www.sciencedirect.com/science/article/pii/S0377221715006554 (cit. on p. 78).

[107] Nan Xiao, Qing-Song Xu, and Miao-Zhu Li. "hdnom: Building Nomograms for Penalized Cox Models with High-Dimensional Survival Data". In: *bioRxiv* (2016) (cit. on p. 11).

[108] John Alberg and Zachary C Lipton. "Improving factor-based quantitative investing by forecasting company fundamentals". In: *arXiv preprint arXiv:1711.04837* (2017) (cit. on p. 33).

[109] Yaseen M Arabi, Awad Al-Omari, and et al. "Critically ill patients with the Middle East Respiratory Syndrome: a multicenter retrospective cohort study". In: *Critical Care Medicine* 45.10 (2017), pp. 1683–1695 (cit. on p. 9).

[110] Pierre-Luc Bacon, Jean Harb, and Doina Precup. "The option-critic architecture". In: *Proceedings of the AAAI conference on artificial intelligence*. Vol. 31. 1. 2017 (cit. on p. 95).

[111] Joos-Hendrik Böse, Valentin Flunkert, Jan Gasthaus, Tim Januschowski, Dustin Lange, David Salinas, Sebastian Schelter, Matthias Seeger, and Yuyang Wang. "Probabilistic Demand Forecasting at Scale". In: *Proc. VLDB Endow.* 10.12 (Aug. 2017), pp. 1694–1705. ISSN: 2150-8097. DOI: 10.14778/3137765.3137775. URL: https://doi.org/10.14778/3137765.3137775 (cit. on p. 79).

[112] Dheeru Dua and Casey Graff. *UCI Machine Learning Repository*. 2017. URL: http://archive.ics.uci.edu/ml (cit. on pp. 70, 187).

[113] Laura J Esserman. “The WISDOM Study: breaking the deadlock in the breast cancer screening debate”. In: *NPJ breast cancer* 3.1 (2017), p. 34 (cit. on p. 93).

[114] Favorita. *Corporación Favorita Grocery Sales forecasting*. 2017. URL: https://www.kaggle.com/c/favorita-grocery-sales-forecasting (cit. on pp. 80, 86).

[115] Iman Ghalehkhondabi, Ehsan Ardjmand, Gary R Weckman, and William A Young. “An overview of energy demand forecasting methods published in 2005–2015”. In: *Energy Systems* 8 (2017), pp. 411–447 (cit. on p. 80).

[116] Aravind Lakshminarayanan, Sahil Sharma, and Balaraman Ravindran. “Dynamic action repetition for deep reinforcement learning”. In: *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 31. 1. 2017 (cit. on p. 96).

[117] Afshin Oroojlooyjadid, M Nazari, Lawrence Snyder, and Martin Takáč. “A deep q-network for the beer game: A reinforcement learning algorithm to solve inventory optimization problems”. In: *arXiv preprint arXiv:1708.05924* 5 (2017) (cit. on pp. 79, 80).

[118] Konstantinos Sechidis, Matthew Sperrin, Emily S. Petherick, Mikel Luján, and Gavin Brown. “Dealing with under-reported variables: An information theoretic solution”. In: *International Journal of Approximate Reasoning* (2017) (cit. on p. 61).

[119] Sahil Sharma, Aravind Srinivas, and Balaraman Ravindran. “Learning to repeat: Fine grained action repetition for deep reinforcement learning”. In: *arXiv preprint arXiv:1702.06054* (2017) (cit. on p. 96).

[120] Alexander Sasha Vezhnevets, Simon Osindero, Tom Schaul, Nicolas Heess, Max Jaderberg, David Silver, and Koray Kavukcuoglu. “Feudal networks for hierarchical reinforcement learning”. In: *International Conference on Machine Learning*. PMLR. 2017, pp. 3540–3549 (cit. on p. 95).

[121] Shaojie Bai, J Zico Kolter, and Vladlen Koltun. “An empirical evaluation of generic convolutional and recurrent networks for sequence modeling”. In: *arXiv preprint arXiv:1803.01271* (2018) (cit. on p. 80).

[122] Edward Choi, Cao Xiao, Walter Stewart, and Jimeng Sun. “Mime: Multilevel medical embedding of electronic health records for predictive healthcare”. In: *Advances in neural information processing systems* 31 (2018) (cit. on p. 48).

[123] Alain Combes, David Hajage, and et al. “ECMO for severe acute respiratory distress syndrome”. In: *NEJM* 378.21 (2018), pp. 1965–1975 (cit. on p. 8).

[124] Omer Gottesman, Fredrik Johansson, Joshua Meier, Jack Dent, Donghun Lee, Srivatsan Srinivasan, Linying Zhang, Yi Ding, David Wihl, Xuefeng Peng, et al. “Evaluating reinforcement learning algorithms in observational health settings”. In: *arXiv preprint arXiv:1805.12298* (2018) (cit. on p. 2).

[125] David Ha and Jürgen Schmidhuber. “World models”. In: *arXiv preprint arXiv:1803.10122* (2018) (cit. on p. 96).

[126] Stefan Hegselmann, Leonard Gruelich, Julian Varghese, and Martin Dugas. “Reproducible survival prediction with SEER cancer data”. In: *Machine Learning for Healthcare Conference*. PMLR. 2018, pp. 49–66 (cit. on pp. 32, 118, 119).

[127] Jonas Hermansson and Thomas Kahan. “Systematic review of validity assessments of Framingham risk score results in health economic modelling of lipid-modifying therapies in Europe”. In: *Pharmacoeconomics* 36 (2018), pp. 205–213 (cit. on p. 32).

[128] Mo Huang, Jingshu Wang, Eduardo Torre, Hannah Dueck, Sydney Shaffer, Roberto Bonasio, John I Murray, Arjun Raj, Mingyao Li, and Nancy R Zhang. “SAVER: gene expression recovery for single-cell RNA sequencing”. In: *Nature methods* 15.7 (2018), pp. 539–542 (cit. on p. 61).

[129] Wei Vivian Li and Jingyi Jessica Li. “An accurate and robust imputation method scImpute for single-cell RNA-seq data”. In: *Nature communications* 9.1 (2018), pp. 1–9 (cit. on p. 61).

[130] Zachary Lipton, Yu-Xiang Wang, and Alexander Smola. “Detecting and correcting for label shift with black box predictors”. In: *International conference on machine learning*. PMLR. 2018, pp. 3122–3130 (cit. on pp. 2, 31, 61).

[131] Tom J Pollard, Alistair EW Johnson, Jesse D Raffa, Leo A Celi, Roger G Mark, and Omar Badawi. “The eICU Collaborative Research Database, a freely available multi-center database for critical care research”. In: *Scientific data* 5.1 (2018), pp. 1–13 (cit. on pp. 73, 189).

[132] Tom J. Pollard, Alistair E.W. Johnson, and et al. “The eICU collaborative research database, a freely available multi-center database for critical care research”. In: *Scientific Data* 5 (2018), pp. 1–13. ISSN: 20524463 (cit. on p. 10).

[133] Syama Sundar Rangapuram, Matthias W Seeger, Jan Gasthaus, Lorenzo Stella, Yuyang Wang, and Tim Januschowski. “Deep state space models for time series forecasting”. In: *Advances in neural information processing systems* 31 (2018) (cit. on p. 80).

[134] John Co-Reyes, YuXuan Liu, Abhishek Gupta, Benjamin Eysenbach, Pieter Abbeel, and Sergey Levine. “Self-Consistent Trajectory Autoencoder: Hierarchical Reinforcement Learning with Trajectory Embeddings”. In: *Proceedings of the 35th International Conference on Machine Learning*. Ed. by Jennifer Dy and Andreas Krause. Vol. 80. Proceedings of Machine Learning Research. PMLR, July 2018, pp. 1009–1018. URL: https://proceedings.mlr.press/v80/co-reyes18a.html (cit. on p. 95).

[135] Davide Risso, Fanny Perraudeau, Svetlana Gribkova, Sandrine Dudoit, and Jean-Philippe Vert. “A general and flexible method for signal extraction from single-cell RNA-seq data”. In: *Nature Communications* 9.1 (2018), pp. 1–17 (cit. on p. 61).

[136] Richard S Sutton and Andrew G Barto. *Reinforcement learning: An introduction.* MIT press, 2018 (cit. on pp. 1, 96, 103).

[137] David Van Dijk, Roshan Sharma, Juozas Nainys, Kristina Yim, Pooja Kathail, Ambrose J Carr, Cassandra Burdziak, Kevin R Moon, Christine L Chaffer, Diwakar Pattabiraman, et al. “Recovering gene interactions from single-cell data using data diffusion”. In: *Cell* 174.3 (2018), pp. 716–729 (cit. on p. 61).

[138] Jinyu Xie. *Simglucose v0.2.1.* 2018. URL: https://github.com/jxx123/simglucose (cit. on p. 103).

[139] Roy Adams, Yuelong Ji, Xiaobin Wang, and Suchi Saria. “Learning models from data with measurement error: Tackling underreporting”. In: *International Conference on Machine Learning (ICML)*. PMLR. 2019, pp. 61–70 (cit. on p. 61).

[140] Alekh Agarwal, Nan Jiang, Sham M Kakade, and Wen Sun. “Reinforcement learning: Theory and algorithms”. In: *CS Dept., UW Seattle, Seattle, WA, USA, Tech. Rep* 32 (2019) (cit. on pp. 96, 97, 100).

[141] Gah-Yi Ban and Cynthia Rudin. “The big data newsvendor: Practical insights from machine learning”. In: *Operations Research* 67.1 (2019), pp. 90–108 (cit. on p. 79).

[142] Dimitris Bertsimas, Jerry Kung, Nikolaos Trichakis, Yuchen Wang, Ryutaro Hirose, and Parsia A Vagefi. “Development and validation of an optimized prediction of mortality for candidates awaiting liver transplantation”. In: *American Journal of Transplantation* 19.4 (2019), pp. 1109–1118 (cit. on p. 32).

[143] Cameron Davidson-Pilon. “lifelines: survival analysis in Python”. In: *Journal of Open Source Software* 4.40 (2019), p. 1317 (cit. on p. 51).

[144] Ben Eysenbach, Russ R Salakhutdinov, and Sergey Levine. “Search on the replay buffer: Bridging planning and reinforcement learning”. In: *Advances in Neural Information Processing Systems* 32 (2019) (cit. on p. 95).

[145] Chenyou Fan, Yuze Zhang, Yi Pan, Xiaoyue Li, Chi Zhang, Rong Yuan, Di Wu, Wensheng Wang, Jian Pei, and Heng Huang. “Multi-horizon time series forecasting with temporal attention learning”. In: *Proceedings of the 25th ACM SIGKDD International conference on knowledge discovery & data mining*. 2019, pp. 2527–2535 (cit. on p. 80).

[146] Omer Gottesman, Fredrik Johansson, Matthieu Komorowski, Aldo Faisal, David Sontag, Finale Doshi-Velez, and Leo Anthony Celi. “Guidelines for reinforcement learning in healthcare”. In: *Nature medicine* 25.1 (2019), pp. 16–18 (cit. on p. 2).

[147] Lingxi Guo, Dong Wei, and et al. “Clinical Features Predicting Mortality Risk in Patients With Viral Pneumonia: The MuLBSTA Score”. In: *Frontiers in Microbiology* 10.December (2019), pp. 1–10. issn: 1664302X (cit. on p. 9).

[148] A Johnson, T Pollard, R Mark, S Berkowitz, and S Horng. *MIMIC-CXR Database (version 2.0. 0). PhysioNet.* 2019 (cit. on p. 30).

[149] Alistair E. W. Johnson, Tom J. Pollard, Nathaniel R. Greenbaum, Matthew P. Lungren, Chih-ying Deng, Yifan Peng, Zhiyong Lu, Roger G. Mark, Seth J. Berkowitz, Steven Horng, and et al. *MIMIC-CXR-JPG, a large publicly available database of labeled chest radiographs.* Nov. 2019 (cit. on pp. 30, 35, 148).

[150] Adil Khan, Jiang Feng, Shaohui Liu, Muhammad Zubair Asghar, et al. “Optimal skipping rates: training agents with fine-grained control using deep reinforcement learning”. In: *Journal of Robotics* 2019 (2019) (cit. on p. 96).

[151] Shiyang Li, Xiaoyong Jin, Yao Xuan, Xiyou Zhou, Wenhu Chen, Yu-Xiang Wang, and Xifeng Yan. “Enhancing the locality and breaking the memory bottleneck of transformer on time series forecasting”. In: *Advances in neural information processing systems* 32 (2019) (cit. on p. 80).

[152] Joni V Lindbohm, Pyry N Sipilä, Nina J Mars, Jaana Pentti, Sara Ahmadi-Abhari, Eric J Brunner, Martin J Shipley, Archana Singh-Manoux, Adam G Tabak, and Mika Kivimäki. “5-year versus risk-category-specific screening intervals for cardiovascular disease prevention: a cohort study”. In: *The Lancet Public Health* 4.4 (2019), e189–e199 (cit. on p. 93).

[153] Roderick JA Little and Donald B Rubin. *Statistical analysis with missing data.* Vol. 793. John Wiley & Sons, 2019 (cit. on pp. 60, 61).

[154] Marco Lorenzi, Maurizio Filippone, Giovanni B. Frisoni, Daniel C. Alexander, and Sebastien Ourselin. “Probabilistic disease progression modeling to characterize diagnostic

uncertainty: Application to staging and prediction in Alzheimer's disease". In: *NeuroImage* 190 (2019). Mapping diseased brains, pp. 56–68. ISSN: 1053-8119. DOI: https://doi.org/10.1016/j.neuroimage.2017.08.059. URL: https://www.sciencedirect.com/science/article/pii/S1053811917307061 (cit. on p. 93).

[155] Bret Nestor, Matthew BA McDermott, Willie Boag, Gabriela Berner, Tristan Naumann, Michael C Hughes, Anna Goldenberg, and Marzyeh Ghassemi. "Feature robustness in non-stationary health records: caveats to deployable model performance in common clinical machine learning tasks". In: *Machine Learning for Healthcare Conference*. PMLR. 2019, pp. 381–405 (cit. on pp. 31, 32).

[156] Boris N Oreshkin, Dmitri Carpov, Nicolas Chapados, and Yoshua Bengio. "N-BEATS: Neural basis expansion analysis for interpretable time series forecasting". In: *arXiv preprint arXiv:1905.10437* (2019) (cit. on p. 80).

[157] Marco V Perez, Kenneth W Mahaffey, Haley Hedlin, John S Rumsfeld, Ariadna Garcia, Todd Ferris, Vidhya Balasubramanian, Andrea M Russo, Amol Rajmane, Lauren Cheung, et al. "Large-scale assessment of a smartwatch to identify atrial fibrillation". In: *New England Journal of Medicine* 381.20 (2019), pp. 1909–1917 (cit. on p. 1).

[158] Fotios Petropoulos, Xun Wang, and Stephen M. Disney. "The inventory performance of forecasting methods: Evidence from the M3 competition data". In: *International Journal of Forecasting* 35.1 (2019). Special Section: Supply Chain Forecasting, pp. 251–265. ISSN: 0169-2070. DOI: https://doi.org/10.1016/j.ijforecast.2018.01.004. URL: https://www.sciencedirect.com/science/article/pii/S0169207018300232 (cit. on pp. 80, 81, 86, 91).

[159] Nicholas G. Reich, Logan C. Brooks, Spencer J. Fox, Sasikiran Kandula, Craig J. McGowan, Evan Moore, Dave Osthus, Evan L. Ray, Abhinav Tushar, Teresa K. Yamana, Matthew Biggerstaff, Michael A. Johansson, Roni Rosenfeld, and Jeffrey Shaman. "A collaborative multiyear, multimodel assessment of seasonal influenza forecasting in the United States". In: *Proceedings of the National Academy of Sciences* 116.8 (2019), pp. 3146–3154. DOI: 10.1073/pnas.1812594116. eprint: https://www.pnas.org/doi/pdf/10.1073/pnas.1812594116. URL: https://www.pnas.org/doi/abs/10.1073/pnas.1812594116 (cit. on p. 77).

[160] Eric J Topol. "High-performance medicine: the convergence of human and artificial intelligence". In: *Nature medicine* 25.1 (2019), pp. 44–56 (cit. on p. 2).

[161] ACS. *What Are Clinical Trial Phases?* 2020. URL: https://www.cancer.org/cancer/managing-cancer/making-treatment-decisions/clinical-trials/what-you-need-to-know/phases-of-clinical-trials.html (cit. on p. 108).

[162] Alekh Agarwal, Sham Kakade, Akshay Krishnamurthy, and Wen Sun. "Flambe: Structural complexity and representation learning of low rank mdps". In: *Advances in neural information processing systems* 33 (2020), pp. 20095–20107 (cit. on pp. 198, 199).

[163] Anup Agarwal, Aparna Mukherjee, Gunjan Kumar, Pranab Chatterjee, Tarun Bhatnagar, and Pankaj Malhotra. "Convalescent plasma in the management of moderate covid-19 in adults in India: open label phase II multicentre randomised controlled trial (PLACID Trial)". In: *bmj* 371 (2020) (cit. on p. 19).

[164] R. Ahmed, V. Sreeram, Y. Mishra, and M.D. Arif. "A review and evaluation of the state-of-the-art in PV solar power forecasting: Techniques and optimization". In: *Renewable and Sustainable Energy Reviews* 124 (2020), p. 109792. ISSN: 1364-0321. DOI: https://doi.

org/10.1016/j.rser.2020.109792. URL: https://www.sciencedirect.com/science/article/pii/S1364032120300885 (cit. on p. 77).

[165] Waleed Alhazzani, Morten Hylander Møller, and et al. “Surviving Sepsis Campaign: guidelines on the management of critically ill adults with Coronavirus Disease 2019 (COVID-19)”. In: *Intensive Care Medicine* (2020), pp. 1–34 (cit. on p. 9).

[166] American College of Cardiology. *ACC’s COVID-19 Hub*. 2020 (cit. on p. 8).

[167] Lindsey R Baden, Hana M El Sahly, Brandon Essink, Karen Kotloff, Sharon Frey, Rick Novak, David Diemert, Stephen A Spector, Nadine Rouphael, C Buddy Creech, et al. “Efficacy and safety of the mRNA-1273 SARS-CoV-2 vaccine”. In: *New England journal of medicine* (2020) (cit. on p. 19).

[168] John H Beigel, Kay M Tomashek, Lori E Dodd, Aneesh K Mehta, Barry S Zingman, Andre C Kalil, Elizabeth Hohmann, Helen Y Chu, Annie Luetkemeyer, Susan Kline, et al. “Remdesivir for the treatment of Covid-19—preliminary report”. In: *New England Journal of Medicine* 383.19 (2020), pp. 1813–1836 (cit. on pp. 18, 19).

[169] Jessa Bekker and Jesse Davis. “Learning from positive and unlabeled data: A survey”. In: *Machine Learning* 109.4 (2020), pp. 719–760 (cit. on pp. 48, 52).

[170] Jessa Bekker and Jesse Davis. “Learning from positive and unlabeled data: A survey”. In: *Machine Learning* 109.4 (2020), pp. 719–760 (cit. on p. 61).

[171] Soorajnath Boominathan, Michael Oberst, Helen Zhou, Sanjat Kanjilal, and David Sontag. “Treatment Policy Learning in Multiobjective Settings with Fully Observed Outcomes”. In: *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. KDD ’20. Association for Computing Machinery, 2020 (cit. on p. 32).

[172] CDC, COVID-19 Response. *COVID-19 Case Surveillance Data Access, Summary, and Limitations*. 2020. (Visited on 12/31/2020) (cit. on pp. 21, 30, 34, 127).

[173] Vitor Cerqueira, Luis Torgo, and Igor Mozetič. “Evaluating time series forecasting models: An empirical study on performance estimation methods”. In: *Machine Learning* 109.11 (2020), pp. 1997–2028 (cit. on p. 32).

[174] Lakshay Chauhan, John Alberg, and Zachary Lipton. “Uncertainty-Aware Lookahead Factor Models for Quantitative Investing”. In: *Proceedings of the 37th International Conference on Machine Learning*. Ed. by Hal Daumé III and Aarti Singh. Vol. 119. Proceedings of Machine Learning Research. PMLR, July 2020, pp. 1489–1499. URL: https://proceedings.mlr.press/v119/chauhan20a.html (cit. on p. 33).

[175] Pieter Cohen, Joann G Elmore, Jessamyn Blau, and Allyson Bloom. “Coronavirus disease 2019 (COVID-19): Outpatient evaluation and management in adults”. In: *UpToDate*. UpToDate, 2020 (cit. on p. 18).

[176] Will Dabney, Georg Ostrovski, and André Barreto. “Temporally-extended {\epsilon}-greedy exploration”. In: *arXiv preprint arXiv:2006.01782* (2020) (cit. on p. 96).

[177] John M Dennis, Andrew P McGovern, Sebastian J Vollmer, and Bilal A Mateen. “Improving COVID-19 critical care mortality over time in England: A national cohort study, March to June 2020”. In: *MedRxiv* (2020) (cit. on pp. 18–20).

[178] Annemarie B Docherty, Ewen M Harrison, Christopher A Green, Hayley E Hardwick, Riinu Pius, Lisa Norman, Karl A Holden, Jonathan M Read, Frank Dondelinger, Gail Carson, et al. “Features of 20 133 UK patients in hospital with covid-19 using the ISARIC

WHO Clinical Characterisation Protocol: prospective observational cohort study". In: *bmj* 369 (2020) (cit. on pp. 47, 58).

[179] Jianzhun Du, Joseph Futoma, and Finale Doshi-Velez. "Model-based reinforcement learning for semi-markov decision processes with neural odes". In: *Advances in Neural Information Processing Systems* 33 (2020), pp. 19805–19816 (cit. on p. 96).

[180] Owen Dyer. *Covid-19: Eli Lilly pauses antibody trial for safety reasons*. 2020 (cit. on p. 19).

[181] Guihong Fan, Zhichun Yang, Qianying Lin, Shi Zhao, Lin Yang, and Daihai He. "Decreased Case Fatality Rate of COVID-19 in the Second Wave: A study in 53 countries or regions". In: *Transboundary and Emerging Diseases* (2020) (cit. on pp. 18, 19).

[182] Florida Department of Health. *Florida Case Line Data*. 2020. URL: https://www.arcgis.com/home/item.html?id=4cc62b3a510949c7a8167f6baa3e069d (visited on 09/07/2020) (cit. on p. 21).

[183] James B Galloway, Sam Norton, Richard D Barker, Andrew Brookes, Ivana Carey, Benjamin D Clarke, Raeesa Jina, Carole Reid, Mark D Russell, Ruth Sneep, et al. "A clinical risk score to identify patients with COVID-19 at high risk of critical care admission or death: an observational cohort study". In: *Journal of Infection* 81.2 (2020), pp. 282–288 (cit. on pp. 47, 52, 58).

[184] Saurabh Garg, Yifan Wu, Sivaraman Balakrishnan, and Zachary Lipton. "A unified view of label shift estimation". In: *Advances in Neural Information Processing Systems* 33 (2020), pp. 3290–3300 (cit. on pp. 2, 31, 61).

[185] Andrew Gelman, Jennifer Hill, and Aki Vehtari. *Regression and other stories*. Cambridge University Press, 2020 (cit. on p. 60).

[186] Jiao Gong, Jingyi Ou, and et al. "Multicenter Development and Validation of a Novel Risk Nomogram for Early Prediction of Severe 2019-Novel Coronavirus Pneumonia". In: *Available at SSRN 3551365* (2020) (cit. on pp. 9, 16).

[187] Jiao Gong, Jingyi Ou, Xueping Qiu, Yusheng Jie, Yaqiong Chen, Lianxiong Yuan, Jing Cao, Mingkai Tan, Wenxiong Xu, Fang Zheng, et al. "A tool to early predict severe 2019-novel coronavirus pneumonia (COVID-19): a multicenter study using the risk nomogram in Wuhan and Guangdong, China". In: *medRxiv* (2020) (cit. on pp. 9, 12, 14).

[188] RECOVERY Collaborative Group. "Dexamethasone in hospitalized patients with COVID-19". In: *NEJM* (2020) (cit. on pp. 18, 19).

[189] Humza Haider, Bret Hoehn, Sarah Davis, and Russell Greiner. "Effective Ways to Build and Evaluate Individual Survival Distributions." In: *J. Mach. Learn. Res.* 21.85 (2020), pp. 1–63 (cit. on pp. xii, 52, 57).

[190] Brandon Michael Henry. "COVID-19, ECMO, and lymphopenia: a word of caution". In: *The Lancet Respiratory Medicine* (2020) (cit. on p. 9).

[191] Brandon Michael Henry, Maria Helena Santos De Oliveira, Stefanie Benoit, Mario Plebani, and Giuseppe Lippi. "Hematologic, biochemical and immune biomarker abnormalities associated with severe illness and mortality in coronavirus disease 2019 (COVID-19): a meta-analysis". In: *Clinical Chemistry and Laboratory Medicine (CCLM)* 58.7 (2020), pp. 1021–1028 (cit. on p. 47).

[192] Brandon Michael Henry and Giuseppe Lippi. “Poor survival with extracorporeal membrane oxygenation in acute respiratory distress syndrome due to coronavirus disease 2019: Pooled analysis of early reports”. In: *Journal of Critical Care* (2020) (cit. on p. 9).

[193] Peter Horby, Marion Mafham, Louise Linsell, Jennifer L Bell, Natalie Staplin, Jonathan R Emberson, Martin Wiselka, Andrew Ustianowski, Einas Elmahi, Benjamin Prudon, et al. “Effect of Hydroxychloroquine in Hospitalized Patients with COVID-19: Preliminary results from a multi-centre, randomized, controlled trial”. In: *medRxiv* (2020) (cit. on p. 19).

[194] Leora Horwitz, Simon A Jones, Robert J Cerfolio, Fritz Francois, Joseph Greco, Bret Rudy, and Christopher M Petrilli. “Trends in Covid-19 risk-adjusted mortality rates in a single health system”. In: (2020) (cit. on pp. 18–20).

[195] Xiangao Jiang, Megan Coffee, and et al. “Towards an Artificial Intelligence Framework for Data-Driven Prediction of Coronavirus Clinical Severity”. In: (2020) (cit. on p. 9).

[196] Sanjat Kanjilal, Michael Oberst, Sooraj Boominathan, Helen Zhou, David C Hooper, and David Sontag. “A decision algorithm to promote outpatient antimicrobial stewardship for uncomplicated urinary tract infection”. In: *Science translational medicine* 12.568 (2020), eaay5067 (cit. on pp. 1, 32).

[197] Joseph T King Jr, James S Yoon, Christopher T Rentsch, Janet P Tate, Lesley S Park, Farah Kidwai-Khan, Melissa Skanderson, Ronald G Hauser, Daniel A Jacobson, Joseph Erdos, et al. “Development and validation of a 30-day mortality index based on pre-existing medical administrative data from 13,323 COVID-19 patients: The Veterans Health Administration COVID-19 (VACO) Index”. In: *PLoS One* 15.11 (2020), e0241825 (cit. on p. 47).

[198] Marine Le Morvan, Julie Josse, Thomas Moreau, Erwan Scornet, and Gaël Varoquaux. “NeuMiss networks: differentiable programming for supervised learning with missing values.” In: *Advances in Neural Information Processing Systems* 33 (2020), pp. 5980–5990 (cit. on p. 61).

[199] Marine Le Morvan, Nicolas Prost, Julie Josse, Erwan Scornet, and Gaël Varoquaux. “Linear predictor on linearly-generated data with missing values: non consistency and solutions”. In: *International Conference on Artificial Intelligence and Statistics*. PMLR. 2020, pp. 3165–3174 (cit. on p. 74).

[200] Steven Levy. *Bill Gates on Covid: Most US Tests Are ‘Completely Garbage’*. 2020. URL: https://www.wired.com/story/bill-gates-on-covid-most-us-tests-are-completely-garbage/ (visited on 08/27/2020) (cit. on pp. 18, 28).

[201] Kunhua Li, Jiong Wu, Faqi Wu, Dajing Guo, Linli Chen, Zheng Fang, and Chuanming Li. “The clinical and chest CT features associated with severe and critical COVID-19 pneumonia”. In: *Investigative radiology* (2020) (cit. on p. 48).

[202] Tingbo Liang et al. “Handbook of COVID-19 prevention and treatment”. In: *The First Affiliated Hospital, Zhejiang University School of Medicine. Compiled According to Clinical Experience* (2020) (cit. on pp. 8, 9).

[203] Wenhua Liang, Hengrui Liang, Limin Ou, Binfeng Chen, Ailan Chen, Caichen Li, Yimin Li, Weijie Guan, Ling Sang, Jiatao Lu, et al. “Development and validation of a clinical risk score to predict the occurrence of critical illness in hospitalized patients with COVID-19”. In: *JAMA internal medicine* 180.8 (2020), pp. 1081–1089 (cit. on pp. 47, 58).

[204] Siqi Liu, Kay Choong See, Kee Yuan Ngiam, Leo Anthony Celi, Xingzhi Sun, and Mengling Feng. “Reinforcement learning for clinical decision support in critical care: comprehensive review”. In: *Journal of medical Internet research* 22.7 (2020), e18477 (cit. on p. 2).

[205] Alexis Madrigal. *A Second Coronavirus Death Surge Is Coming*. 2020. URL: https://www.theatlantic.com/health/archive/2020/07/second-coronavirus-death-surge/614122/ (visited on 08/27/2020) (cit. on pp. 18, 19, 22).

[206] Alexis Madrigal and Whet Moser. *How Many Americans Are About to Die?* 2020. (Visited on 11/19/2020) (cit. on pp. 18, 19).

[207] Lone Wulff Madsen et al. “Remdesivir for the Treatment of Covid-19-Final Report”. In: *NEJM* (2020) (cit. on p. 19).

[208] Spyros Makridakis, Evangelos Spiliotis, and Vassilios Assimakopoulos. “The M4 Competition: 100,000 time series and 61 forecasting methods”. In: *International Journal of Forecasting* 36.1 (2020). M4 Competition, pp. 54–74. ISSN: 0169-2070. DOI: https://doi.org/10.1016/j.ijforecast.2019.04.014. URL: https://www.sciencedirect.com/science/article/pii/S0169207019301128 (cit. on p. 80).

[209] M.A. Matthay, J.M. Aldrich, and J. Gotts. “Treatment for severe ARDS from COVID-19”. In: *The Lancet Respiratory Medicine* (2020) (cit. on p. 9).

[210] Vikas Mehta, Sanjay Goel, Rafi Kabarriti, Daniel Cole, Mendel Goldfinger, Ana Acuna-Villaorduna, Kith Pradhan, Raja Thota, Stan Reissman, Joseph A Sparano, et al. “Case Fatality Rate of Cancer Patients with COVID-19 in a New York Hospital SystemCase Fatality Rate of Cancer Patients with COVID-19”. In: *Cancer discovery* 10.7 (2020), pp. 935–941 (cit. on pp. 20, 26).

[211] Thomas M Melhuish, Ruan Vlok, Christopher Thang, Judith Askew, and Leigh White. “Outcomes of extracorporeal membrane oxygenation support for patients with COVID-19: A pooled analysis of 331 cases”. In: *Am J Emerg Med* 29 (2020), S0735–6757 (cit. on p. 9).

[212] Robinson Meyer and Alexis Madrigal. 2020. URL: https://covidtracking.com/ (cit. on pp. xi, 22, 24).

[213] Surveillance Research Program National Cancer Institute DCCPS. *SEER Research Data, 9 Registries, Nov 2020 Sub (1975-2018) - Linked To County Attributes - Time Dependent (1990-2019) Income/Rurality, 1969-2020 Counties.* Nov. 2020 (cit. on pp. 30, 34, 118).

[214] Graziano Onder, Giovanni Rezza, and Silvio Brusaferro. “Case-fatality rate and characteristics of patients dying in relation to COVID-19 in Italy”. In: *JAMA* 323.18 (2020), pp. 1775–1776 (cit. on p. 19).

[215] Organ Procurement and Transplantation Network. *About data: OPTN.* 2020. URL: https://optn.transplant.hrsa.gov/data/about-data/ (cit. on pp. 30, 145).

[216] Afshin Oroojlooyjadid, Lawrence V Snyder, and Martin Takáč. “Applying deep learning to the newsvendor problem”. In: *IISE Transactions* 52.4 (2020), pp. 444–463 (cit. on p. 79).

[217] Maria Pachetti, Bruna Marini, Fabiola Giudici, Francesca Benedetti, Silvia Angeletti, Massimo Ciccozzi, Claudio Masciovecchio, Rudy Ippodrino, and Davide Zella. “Impact of lockdown on Covid-19 case fatality rate and viral mutations spread in 7 countries in Europe and North America”. In: *Journal of Translational Medicine* 18.1 (2020), pp. 1–7 (cit. on pp. 18, 20).

[218] Jason Phua, Li Weng, Lowell Ling, Moritoki Egi, Chae-Man Lim, Jigeeshu Vasishtha Divatia, Babu Raja Shrestha, Yaseen M Arabi, Jensen Ng, Charles D Gomersall, et al. "Intensive care management of coronavirus disease 2019 (COVID-19): challenges and recommendations". In: *The lancet respiratory medicine* 8.5 (2020), pp. 506–517 (cit. on p. 18).

[219] Oleg S Pianykh, Georg Langs, Marc Dewey, Dieter R Enzmann, Christian J Herold, Stefan O Schoenberg, and James A Brink. "Continuous learning AI in radiology: implementation principles and early applications". In: *Radiology* 297.1 (2020), pp. 6–14 (cit. on p. 32).

[220] C Piubelli, M Deiana, E Pomari, R Silva, Z Bisoffi, F Formenti, F Perandin, F Gobbi, and D Buonfrate. "Overall decrease of SARS-CoV-2 viral load and reduction of clinical burden: the experience of a Northern Italy hospital." In: *Clinical Microbiology and Infection: the Official Publication of the European Society of Clinical Microbiology and Infectious Diseases* (2020) (cit. on pp. 18, 20).

[221] Fernando P Polack, Stephen J Thomas, Nicholas Kitchin, Judith Absalon, Alejandra Gurtman, Stephen Lockhart, John L Perez, Gonzalo Pérez Marc, Edson D Moreira, Cristiano Zerbini, et al. "Safety and efficacy of the BNT162b2 mRNA Covid-19 vaccine". In: *New England journal of medicine* (2020) (cit. on p. 19).

[222] CMU Delphi Project. 2020. URL: https://cmu-delphi.github.io/delphi-epidata/api/covidcast.html (visited on 08/27/2020) (cit. on pp. 17, 29).

[223] Kollengode Ramanathan, David Antognini, and et al. "Planning and provision of ECMO services for severe ARDS during the COVID-19 pandemic and other outbreaks of emerging infectious diseases." In: *The Lancet Respiratory Medicine* (2020). ISSN: 2213-2619 (cit. on p. 9).

[224] Regeneron. "Regeneron's Casirivimab and Imdevimab Antibody Cocktail for COVID-19 is First Combination Therapy to Receive FDA Emergency Use Authorization". In: (2020). URL: https://investor.regeneron.com/news-releases/news-release-details/regenerons-regen-cov2-first-antibody-cocktail-covid-19-receive (cit. on p. 19).

[225] Fausto Salaffi, Marina Carotti, Marika Tardella, Alessandra Borgheresi, Andrea Agostini, Davide Minorati, Daniela Marotto, Marco Di Carlo, Massimo Galli, Andrea Giovagnoni, et al. "The role of a chest computed tomography severity score in coronavirus disease 2019 pneumonia". In: *Medicine* 99.42 (2020) (cit. on p. 48).

[226] David Salinas, Valentin Flunkert, Jan Gasthaus, and Tim Januschowski. "DeepAR: Probabilistic forecasting with autoregressive recurrent networks". In: *International Journal of Forecasting* 36.3 (2020), pp. 1181–1191. ISSN: 0169-2070. DOI: https://doi.org/10.1016/j.ijforecast.2019.07.001. URL: https://www.sciencedirect.com/science/article/pii/S0169207019301888 (cit. on p. 80).

[227] Wesley H Self, Matthew W Semler, Lindsay M Leither, Jonathan D Casey, Derek C Angus, Roy G Brower, Steven Y Chang, Sean P Collins, John C Eppensteiner, Michael R Filbin, et al. "Effect of hydroxychloroquine on clinical status at 14 days in hospitalized patients with COVID-19: a randomized clinical trial". In: *Jama* 324.21 (2020), pp. 2165–2176 (cit. on pp. 18, 19).

[228] Mark P Sendak, William Ratliff, Dina Sarro, Elizabeth Alderton, Joseph Futoma, Michael Gao, Marshall Nichols, Mike Revoir, Faraz Yashar, Corinne Miller, et al. "Real-world

integration of a sepsis deep learning technology into routine clinical care: implementation study". In: *JMIR medical informatics* 8.7 (2020), e15182 (cit. on p. 1).

[229] Laleh Seyyed-Kalantari, Guanxiong Liu, Matthew McDermott, Irene Y Chen, and Marzyeh Ghassemi. "CheXclusion: Fairness gaps in deep chest X-ray classifiers". In: *Biocomputing 2021: Proceedings of the Pacific Symposium*. World Scientific. 2020, pp. 232–243 (cit. on p. 148).

[230] Piotr Spychalski, Agata Błażyńska-Spychalska, and Jarek Kobiela. "Estimating case fatality rates of COVID-19". In: *The Lancet Infectious Diseases* 20.7 (2020), pp. 774–775 (cit. on pp. 18, 19, 22).

[231] Shengpu Tang, Parmida Davarmanesh, Yanmeng Song, Danai Koutra, Michael W Sjoding, and Jenna Wiens. "Democratizing EHR analyses with FIDDLE: a flexible data-driven preprocessing pipeline for structured clinical data". In: *Journal of the American Medical Informatics Association* 27.12 (2020), pp. 1921–1934 (cit. on pp. 73, 189).

[232] Derek Thompson. *COVID-19 Cases Are Rising, So Why Are Deaths Flatlining?* 2020. URL: https://www.theatlantic.com/ideas/archive/2020/07/why-covid-death-rate-down/613945/ (visited on 07/09/2020) (cit. on pp. 18, 19, 22).

[233] Donald Trump. *Remarks by President Trump in Press Briefing on COVID-19*. July 2020. URL: https://www.whitehouse.gov/briefings-statements/remarks-president-trump-press-briefing-covid-19/ (cit. on pp. 17, 28).

[234] U.S. FDA. "COVID-19 Update: FDA Authorizes Monoclonal Antibodies for Treatment of COVID-19". In: (2020). URL: https://www.fda.gov/news-events/press-announcements/coronavirus-covid-19-update-fda-authorizes-monoclonal-antibodies-treatment-covid-19 (cit. on p. 19).

[235] Dawei Wang, Bo Hu, and et al. "Clinical Characteristics of 138 Hospitalized Patients with 2019 Novel Coronavirus-Infected Pneumonia in Wuhan, China". In: *JAMA - Journal of the American Medical Association* 323.11 (2020), pp. 1061–1069. ISSN: 15383598 (cit. on p. 9).

[236] M. Whet. *Why Changing COVID-19 Demographics in the US Make Death Trends Harder to Understand*. 2020. URL: https://covidtracking.com/blog/why-changing-covid-19-demographics-in-the-us-make-death-trends-harder-to (visited on 08/27/2020) (cit. on pp. 18, 19).

[237] World Health Organization et al. *Clinical management of severe acute respiratory infection (SARI) when COVID-19 disease is suspected: interim guidance, 13 March 2020*. Tech. rep. 2020 (cit. on p. 9).

[238] Katherine J. Wu. *'It's Like Groundhog Day': Coronavirus Testing Labs Again Lack Key Supplies*. 2020. URL: %7Bhttps://www.nytimes.com/2020/07/23/health/coronavirus-testing-supply-shortage.html%7D (visited on 08/27/2020) (cit. on p. 17).

[239] Xiaobo Yang, Yuan Yu, and et al. "Clinical course and outcomes of critically ill patients with SARS-CoV-2 pneumonia in Wuhan, China: a single-centered, retrospective, observational study". In: *The Lancet Respiratory Medicine* (2020) (cit. on p. 9).

[240] Alexandra L Young, Felix JS Bragman, Bojidar Rangelov, MeiLan K Han, Craig J Galbán, David A Lynch, David J Hawkes, Daniel C Alexander, and John R Hurst. "Disease

progression modeling in chronic obstructive pulmonary disease". In: *American journal of respiratory and critical care medicine* 201.3 (2020), pp. 294–302 (cit. on p. 93).

[241] Said El Zein, Nivine El-Hor, Omar Chehab, Samer Alkassis, Tushar Mishra, Vichar Trivedi, Hossein Salimnia, and Pranatharthi Chandrasekar. "Declining Trend in the Initial SARS-CoV-2 Viral Load During the Pandemic: Preliminary Observations from Detroit, Michigan". In: *medRxiv* (2020). DOI: 10.1101/2020.11.16.20231597. URL: https://www.medrxiv.org/content/early/2020/11/18/2020.11.16.20231597 (cit. on p. 18).

[242] Fei Zhou, Ting Yu, and et al. "Clinical course and risk factors for mortality of adult inpatients with COVID-19 in Wuhan, China: a retrospective cohort study". In: *The Lancet* (2020) (cit. on p. 9).

[243] Helen Zhou, Cheng Cheng, Zachary C Lipton, George H Chen, and Jeremy C Weiss. "Mortality Risk Score for Critically Ill Patients with Viral or Unspecified Pneumonia: Assisting Clinicians with COVID-19 ECMO Planning". In: *International Conference on Artificial Intelligence in Medicine*. Springer. 2020, pp. 336–347 (cit. on p. 5).

[244] Helen Zhou, Cheng Cheng, Zachary C Lipton, George H Chen, and Jeremy C Weiss. "Mortality Risk Score for Critically Ill Patients with Viral or Unspecified Pneumonia: Assisting Clinicians with COVID-19 ECMO Planning". In: *International Conference on Artificial Intelligence in Medicine*. Springer. 2020, pp. 336–347 (cit. on p. 47).

[245] Marie-Lise Bats, Benoit Rucheton, Tara Fleur, Arthur Orieux, Clément Chemin, Sébastien Rubin, Brigitte Colombies, Arnaud Desclaux, Claire Rivoisy, Etienne Mériglier, et al. "Covichem: A biochemical severity risk score of COVID-19 upon hospital admission". In: *PloS one* 16.5 (2021), e0250956 (cit. on pp. 47, 52).

[246] André Biedenkapp, Raghu Rajan, Frank Hutter, and Marius Lindauer. "TempoRL: Learning when to act". In: *International Conference on Machine Learning*. PMLR. 2021, pp. 914–924 (cit. on p. 96).

[247] Jonathon Byrd, Sivaraman Balakrishnan, Xiaoqian Jiang, and Zachary C Lipton. "Predicting mortality in liver transplant candidates". In: *Explainable AI in Healthcare and Medicine*. Springer, 2021, pp. 321–333 (cit. on pp. 32, 35, 145).

[248] CDC. "Science Brief: Emerging SARS-CoV-2 Variants". In: *cdc.gov* (2021). URL: https://www.cdc.gov/coronavirus/2019-ncov/more/science-and-research/scientific-brief-emerging-variants.html (cit. on p. 20).

[249] Peter Chen, Ajay Nirula, Barry Heller, Robert L Gottlieb, Joseph Boscia, Jason Morris, Gregory Huhn, Jose Cardona, Bharat Mocherla, Valentina Stosor, et al. "SARS-CoV-2 neutralizing antibody LY-CoV555 in outpatients with Covid-19". In: *New England Journal of Medicine* 384.3 (2021), pp. 229–237 (cit. on p. 19).

[250] Cheng Cheng, Helen Zhou, Jeremy C Weiss, and Zachary C Lipton. "Unpacking the Drop in COVID-19 Case Fatality Rates: A Study of National and Florida Line-Level Data". In: *AMIA Annual Symposium Proceedings*. Vol. 2021. American Medical Informatics Association. 2021, p. 285 (cit. on pp. 5, 7, 31, 77).

[251] Hanjun Dai, Yuan Xue, Zia Syed, Dale Schuurmans, and Bo Dai. "Neural stochastic dual dynamic programming". In: *arXiv preprint arXiv:2112.00874* (2021) (cit. on p. 80).

[252] Priya L Donti and J Zico Kolter. "Machine learning for sustainable energy systems". In: *Annual Review of Environment and Resources* 46 (2021), pp. 719–747 (cit. on p. 77).

[253] FDA. *Artificial Intelligence and Machine Learning in Software as a Medical Device*. 2021. URL: https://www.fda.gov/medical-devices/software-medical-device-samd/artificial-intelligence-and-machine-learning-software-medical-device (cit. on p. 1).

[254] Samuel G. Finlayson, Adarsh Subbaswamy, Karandeep Singh, John Bowers, Annabel Kupke, Jonathan Zittrain, Isaac S. Kohane, and Suchi Saria. "The Clinician and Dataset Shift in Artificial Intelligence". In: *New England Journal of Medicine* 385.3 (2021). PMID: 34260843, pp. 283–286. DOI: 10.1056/NEJMc2104626. eprint: https://doi.org/10.1056/NEJMc2104626. URL: https://doi.org/10.1056/NEJMc2104626 (cit. on pp. 1, 2).

[255] The U.S. Food and Drug Administration (FDA). *Moderna COVID-19 Vaccine*. 2021. URL: https://www.fda.gov/emergency-preparedness-and-response/coronavirus-disease-2019-covid-19/moderna-covid-19-vaccine (visited on 01/06/2021) (cit. on p. 19).

[256] The U.S. Food and Drug Administration (FDA). *Pfizer-BioNTech COVID-19 Vaccine*. 2021. URL: https://www.fda.gov/emergency-preparedness-and-response/coronavirus-disease-2019-covid-19/pfizer-biontech-covid-19-vaccine (visited on 01/12/2021) (cit. on p. 19).

[257] Saurabh Garg, Yifan Wu, Alexander J Smola, Sivaraman Balakrishnan, and Zachary Lipton. "Mixture Proportion Estimation and PU Learning: A Modern Approach". In: *Advances in Neural Information Processing Systems (NeurIPS)* 34 (2021) (cit. on pp. 61, 179).

[258] HHS. "HHS Launches Web-Based Locator for COVID-19 Outpatient Treatment Sites for Monoclonal Antibodies". In: (2021). URL: https://www.hhs.gov/about/news/2021/01/11/hhs-launches-web-based-locator-for-covid-19-outpatient-treatment-sites-for-monoclonal-antibodies.html (cit. on p. 19).

[259] Alistair Johnson, Lucas Bulgarelli, Tom Pollard, Steven Horng, Leo Anthony Celi, and Roger Mark. *MIMIC-IV*. 2021. URL: https://physionet.org/content/mimiciv/1.0/ (cit. on pp. 30, 35, 138).

[260] Marine Le Morvan, Julie Josse, Erwan Scornet, and Gael Varoquaux. "What's a good imputation to predict with missing values?" In: *Advances in Neural Information Processing Systems*. Ed. by M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan. Vol. 34. Curran Associates, Inc., 2021, pp. 11530–11540. URL: https://proceedings.neurips.cc/paper/2021/file/5fe8fdc79ce292c39c5f209d734b7206-Paper.pdf (cit. on p. 61).

[261] Bryan Lim, Sercan O. Arık, Nicolas Loeff, and Tomas Pfister. "Temporal Fusion Transformers for interpretable multi-horizon time series forecasting". In: *International Journal of Forecasting* 37.4 (2021), pp. 1748–1764. ISSN: 0169-2070. DOI: https://doi.org/10.1016/j.ijforecast.2021.03.012. URL: https://www.sciencedirect.com/science/article/pii/S0169207021000637 (cit. on pp. 80, 86, 195).

[262] Karthika Mohan and Judea Pearl. "Graphical models for processing missing data". In: *Journal of the American Statistical Association* 116.534 (2021), pp. 1023–1037 (cit. on p. 64).

[263] Dewi Rahardja and Dean M. Young. "Confidence Intervals for the Risk Ratio Using Double Sampling with Misclassified Binomial Data". In: *Journal of Data Science* 9.4 (2021), pp. 529–548 (cit. on p. 61).

[264] Laila Rasmy, Yang Xiang, Ziqian Xie, Cui Tao, and Degui Zhi. "Med-BERT: pretrained contextualized embeddings on large-scale structured electronic health records for disease prediction". In: *NPJ digital medicine* 4.1 (2021), pp. 1–13 (cit. on p. 48).

[265] Alex Reinhart, Logan Brooks, Maria Jahja, Aaron Rumack, Jingjing Tang, Sumit Agrawal, Wael Al Saeed, Taylor Arnold, Amartya Basu, Jacob Bien, et al. "An open repository of real-time COVID-19 indicators". In: *Proceedings of the National Academy of Sciences* 118.51 (2021), e2111452118 (cit. on p. 1).

[266] Kristen A Severson, Lana M Chahine, Luba A Smolensky, Murtaza Dhuliawala, Mark Frasier, Kenney Ng, Soumya Ghosh, and Jianying Hu. "Discovery of Parkinson's disease states and disease progression modelling: a longitudinal data study using machine learning". In: *The Lancet Digital Health* 3.9 (2021), e555–e564 (cit. on p. 1).

[267] Kai Wu, Anne P Werner, Matthew Koch, Angela Choi, Elisabeth Narayanan, Guillaume BE Stewart-Jones, Tonya Colpitts, Hamilton Bennett, Seyhan Boyoglu-Barnum, Wei Shi, et al. "Serum neutralizing activity elicited by mRNA-1273 vaccine". In: *New England Journal of Medicine* 384.15 (2021), pp. 1468–1470 (cit. on p. 20).

[268] Wan Xu, Nan-Nan Sun, Hai-Nv Gao, Zhi-Yuan Chen, Ya Yang, Bin Ju, and Ling-Ling Tang. "Risk factors analysis of COVID-19 patients with ARDS and prediction based on machine learning". In: *Scientific reports* 11.1 (2021), pp. 1–12 (cit. on p. 47).

[269] Chao Yu, Jiming Liu, Shamim Nemati, and Guosheng Yin. "Reinforcement Learning in Healthcare: A Survey". In: *ACM Comput. Surv.* 55.1 (2021). ISSN: 0360-0300. DOI: 10.1145/3477600. URL: https://doi.org/10.1145/3477600 (cit. on p. 2).

[270] CDC. *Covid-19 vaccinations in the United States,jurisdiction.* Oct. 2022. URL: https://data.cdc.gov/Vaccinations/COVID-19-Vaccinations-in-the-United-States-Jurisdi/unsk-b7fc (cit. on p. 59).

[271] Cristian Challu, Kin G Olivares, Boris N Oreshkin, Federico Garza, Max Mergenthaler-Canseco, and Artur Dubrawski. "N-hits: Neural hierarchical interpolation for time series forecasting". In: *arXiv preprint arXiv:2201.12886* (2022) (cit. on p. 80).

[272] Li C Cheung, Paul S Albert, Shrutikona Das, and Richard J Cook. "Multistate models for the natural history of cancer progression". In: *British Journal of Cancer* 127.7 (2022), pp. 1279–1288 (cit. on p. 93).

[273] Alexander D'Amour, Katherine Heller, Dan Moldovan, Ben Adlam, Babak Alipanahi, Alex Beutel, Christina Chen, Jonathan Deaton, Jacob Eisenstein, Matthew D. Hoffman, Farhad Hormozdiari, Neil Houlsby, Shaobo Hou, Ghassen Jerfel, Alan Karthikesalingam, Mario Lucic, Yian Ma, Cory McLean, Diana Mincu, Akinori Mitani, Andrea Montanari, Zachary Nado, Vivek Natarajan, Christopher Nielson, Thomas F. Osborne, Rajiv Raman, Kim Ramasamy, Rory Sayres, Jessica Schrouff, Martin Seneviratne, Shannon Sequeira, Harini Suresh, Victor Veitch, Max Vladymyrov, Xuezhi Wang, Kellie Webster, Steve Yadlowsky, Taedong Yun, Xiaohua Zhai, and D. Sculley. "Underspecification Presents Challenges for Credibility in Modern Machine Learning". In: *Journal of Machine Learning Research* 23.226 (2022), pp. 1–61. URL: http://jmlr.org/papers/v23/20-1335.html (cit. on p. 2).

[274] Alexander D'Amour, Katherine Heller, Dan Moldovan, Ben Adlam, Babak Alipanahi, Alex Beutel, Christina Chen, Jonathan Deaton, Jacob Eisenstein, Matthew D. Hoffman, Farhad Hormozdiari, Neil Houlsby, Shaobo Hou, Ghassen Jerfel, Alan Karthikesalingam, Mario Lucic, Yian Ma, Cory McLean, Diana Mincu, Akinori Mitani, Andrea Montanari, Zachary Nado, Vivek Natarajan, Christopher Nielson, Thomas F. Osborne, Rajiv Raman,

Kim Ramasamy, Rory Sayres, Jessica Schrouff, Martin Seneviratne, Shannon Sequeira, Harini Suresh, Victor Veitch, Max Vladymyrov, Xuezhi Wang, Kellie Webster, Steve Yadlowsky, Taedong Yun, Xiaohua Zhai, and D. Sculley. "Underspecification Presents Challenges for Credibility in Modern Machine Learning". In: *Journal of Machine Learning Research* 23.226 (2022), pp. 1–61 (cit. on p. 32).

[275] Robert Fildes, Shaohui Ma, and Stephan Kolassa. "Retail forecasting: Research and practice". In: *International Journal of Forecasting* 38.4 (2022), pp. 1283–1318 (cit. on p. 80).

[276] Danijar Hafner, Kuang-Huei Lee, Ian Fischer, and Pieter Abbeel. "Deep Hierarchical Planning from Pixels". In: *Advances in Neural Information Processing Systems*. Ed. by S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh. Vol. 35. Curran Associates, Inc., 2022, pp. 26091–26104. URL: https://proceedings.neurips.cc/paper_files/paper/2022/file/a766f56d2da42cae20b5652970ec04ef-Paper-Conference.pdf (cit. on p. 95).

[277] Qinghua Liu, Alan Chung, Csaba Szepesvári, and Chi Jin. "When is partially observable reinforcement learning not scary?" In: *Conference on Learning Theory*. PMLR. 2022, pp. 5175–5220 (cit. on p. 198).

[278] Spyros Makridakis, Evangelos Spiliotis, and Vassilios Assimakopoulos. "M5 accuracy competition: Results, findings, and conclusions". In: *International Journal of Forecasting* 38.4 (2022). Special Issue: M5 competition, pp. 1346–1364. ISSN: 0169-2070. DOI: https://doi.org/10.1016/j.ijforecast.2021.11.013. URL: https://www.sciencedirect.com/science/article/pii/S0169207021001874 (cit. on p. 80).

[279] Wenhui Ren, Mingyang Chen, Youlin Qiao, and Fanghui Zhao. "Global guidelines for breast cancer screening: a systematic review". In: *The Breast* 64 (2022), pp. 85–99 (cit. on p. 93).

[280] Helen Zhou, Sivaraman Balakrishnan, and Zachary C Lipton. "Domain Adaptation under Missingness Shift". In: *arXiv preprint arXiv:2211.02093* (2022) (cit. on p. 31).

[281] Helen Zhou, Cheng Cheng, Kelly J Shields, Gursimran Kochhar, Tariq Cheema, Zachary C Lipton, and Jeremy C Weiss. "Learning Clinical Concepts for Predicting Risk of Progression to Severe COVID-19". In: *AMIA Annual Symposium Proceedings*. Vol. 2022. American Medical Informatics Association. 2022, p. 1257 (cit. on p. 5).

[282] Helen Zhou, Cheng Cheng, Kelly J. Shields, Gursimran Kochhar, Tariq Cheema, Zachary C. Lipton, and Jeremy C. Weiss. "Learning Clinical Concepts for Predicting Risk of Progression to Severe COVID-19". In: *AMIA 2022, American Medical Informatics Association Annual Symposium, Washington, DC, USA, November 5-9, 2022*. AMIA, 2022 (cit. on p. 32).

[283] CDC. Mar. 2023. URL: https://www.cdc.gov/museum/timeline/covid19.html (cit. on pp. xi, 8).

[284] Audrey Huang, Jinglin Chen, and Nan Jiang. "Reinforcement Learning in Low-Rank MDPs with Density Features". In: *arXiv preprint arXiv:2302.02252* (2023) (cit. on p. 198).

[285] Qinghua Liu, Praneeth Netrapalli, Csaba Szepesvari, and Chi Jin. "Optimistic MLE: A Generic Model-Based Algorithm for Partially Observable Sequential Decision Making". In: *Proceedings of the 55th Annual ACM Symposium on Theory of Computing*. 2023, pp. 363–376 (cit. on p. 198).

[286] Marlos C Machado, Andre Barreto, Doina Precup, and Michael Bowling. "Temporal abstraction in reinforcement learning with the successor representation". In: *Journal of Machine Learning Research* 24.80 (2023), pp. 1–69 (cit. on p. 95).

[287] Meng Qi, Yuanyuan Shi, Yongzhi Qi, Chenxin Ma, Rong Yuan, Di Wu, and Zuo-Jun Shen. "A practical end-to-end inventory management model with deep learning". In: *Management Science* 69.2 (2023), pp. 759–773 (cit. on p. 80).

[288] Helen Zhou, Sercan O Arik, and Jingtao Wang. "Business Metric-Aware Forecasting for Inventory Management". In: *arXiv preprint arXiv:2308.13118* (2023) (cit. on p. 5).

[289] Helen Zhou, Sivaraman Balakrishnan, and Zachary Lipton. "Domain adaptation under missingness shift". In: *International Conference on Artificial Intelligence and Statistics*. PMLR. 2023, pp. 9577–9606 (cit. on p. 5).

[290] Helen Zhou, Yuwen Chen, and Zachary Lipton. "Evaluating Model Performance in Medical Datasets Over Time". In: *Conference on Health, Inference, and Learning*. PMLR. 2023, pp. 498–508 (cit. on p. 5).

[291] Helen Zhou, Audrey Huang, Kamyar Azizzadenesheli, David Childers, and Zachary Lipton. "Timing as an Action: Learning When to Observe and Act". In: *under submission* (2023) (cit. on p. 5).

[292] Yang Liu, Jianying Liu, Hongjie Xia, Xianwen Zhang, Camila R Fontes-Garfias, Kena A Swanson, Hui Cai, Ritu Sarkar, Wei Chen, Mark Cutler, et al. "Neutralizing Activity of BNT162b2-Elicited Serum". In: *The New England Journal of Medicine* () (cit. on p. 20).