Strategies for Selecting and Adapting Machine Learning Systems to Support Different Types of Experts

Alex Okeson

A dissertation

submitted in partial fulfillment of the

requirements for the degree of

Doctor of Philosophy

University of Washington

2022

Reading Committee:

James Fogarty, Chair

Christopher Althoff

Sean Munson

Program Authorized to Offer Degree:

Paul G. Allen School of Computer Science & Engineering

University of Washington

Abstract

Strategies for Selecting and Adapting Machine Learning Systems

to Support Different Types of Experts

Alex Okeson

Chair of the Supervisory Committee:

Professor James Fogarty

Paul G. Allen School of Computer Science & Engineering

Machine learning prediction and explanation systems offer the ability to learn meaningful representations and patterns in otherwise messy or complex data. These representations can then be used to predict outcomes, to improve human understanding, or to complete tasks more efficiently. As these systems become more prevalent, they are used by a wider variety of people for a wider variety of tasks, meaning more adaptation is required to ensure these systems can be utilized accurately and efficiently. Additionally, these systems are complex and are therefore easy to misuse and misinterpret, particularly when applied to new contexts by individuals without a deep background in the underlying algorithms used by the learning system.

My dissertation explores strategies for selecting and adapting machine learning prediction and explanation systems to help support users of varying sets of expertise in utilizing these systems. These strategies are: aligning systems with user goals, retaining nuance in explanations, and imparting an appropriate level of trust of system outputs. I have explored these strategies through four projects: Problems and an Alternative to Single Explanation Aggregations, Predicting Blood Glucose Test Accuracy in ICU Patients, Predicting and Explaining an Imminent Dementia Diagnosis with Limited Data, and Flexible System for Efficient Goal-Directed Self-Tracking Analysis. Finally, I discuss how these strategies and lessons learned from the highly contextualized projects compare and contrast to guidelines set forth by existing frameworks to guide human-AI interaction and system design.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGEMENTS

First and foremost, I would like to thank my advisor, James Fogarty, for his mentorship and support over the course of my PhD. I would also like to thank all of the other mentors I have had at various points along this journey: Sean Munson, Tim Althoff, Amy Ko, Jenn Wortman Vaughn, Hanna Wallach, Scott Lundberg, Bing Brunton, and Su-In Lee. Finally, I would like to thank my undergraduate research advisor, Sriram Sankaranarayanan, and undergraduate professors Elizabeth Bradley and David Knox for encouraging me to apply to PhD programs.

I have also been fortunate enough to have great lab mates who have supported me through many brainstorming sessions, paper drafts, and practice talks: Shaan Chopra, Raymond Fok, Ravi Karkar, Tae Jones, Susanne Kirchner, Aaleyah Lewis, Richard Li, Kelly Mack, Jesse Martinez, Anant Mittal, Anne Spencer Ross, Jessica Schroeder, Yasaman Sefidgar, Jina Suh, Amanda Swearngin, and Mingyuan Zhong. I would also like to thank many other grad students within CSE and UW for your support, collaboration, and friendship: Nicasia Beebe-Wang, Jennifer Brennan, Hugh Chen, Ayse Dincer, Gabe Erion, Lee Organick, Erin Wilson, and many others who would make this list entirely too long.

Finally, I would like to thank my family, Pam Okeson, Phil Okeson, and Makenna Okeson, who have supported me at every step along the way and provided outside perspective and input when I was too deep in the weeds to see it. I would also like to thank my friends outside of grad school who made me truly think about why I wanted to do a PhD, reminded me of those reasons when I forgot, and provided the emotional support to actually get me to the finish line: Nicholas Carter, Kevin Counts, Thierry Shimoda, and Daniel Tsvankin.

This work has been generously funded by many agencies, grants, and fellowships; without whose support this PhD would not have been financially possible: the Bill & Melinda Gates Foundation under grant INV-004841; the National Institutes of Health under grants R01 LM012810, R01 MH125179, UL1 TR002319; the National Science Foundation under grants IIS-1813675, IIS-1553167, and IIS-1901386; and the Paul G. Allen School of Computer Science & Engineering 1st year Research Fellowship.

Chapter 1. INTRODUCTION

Machine learning systems have drastically increased in popularity in recent years and are being developed and used by people with different levels of expertise and experience and in a wider range of contexts. Often, out-of-the box machine learning systems are applied to new contexts and domains without much adaptation. This has led to a variety of prediction systems which boast high prediction accuracy statistics, but can rely on signal from artifacts in the dataset that are not realistically present in the real world settings these systems are designed to optimize for [30] or are not representative of the underlying phenomena they try to model [67]. This can lead to costly and adverse errors when deployed and utilized for deciding how to respond to the predictions in the real world. This is particularly true in healthcare settings, where it can be important to know the reasons behind a prediction, both to double check the validity of the prediction and to help decide which next steps to take in patient care [68]. It is therefore important to align machine learning system choices and adaptations with the usage requirements of the system so that the system correctly facilitates the downstream usage.

As more complex, accurate machine learning systems have been developed, it has become harder to understand how or why they are making the predictions for any given data sample or data set. Unlike linear regression (where the user can examine linear coefficients of all input features) or single decision trees (where the whole system can be read like a flow chart), machine learning systems that utilize neural networks, deep learning, and gradient boosted decision trees are typically considered black boxes: once the system makes a prediction there is no telling why that decision was made. To address this tradeoff between accurate systems and explainable systems, explainability systems have been developed to provide an after-the-fact interpretation of

how the machine learning system made its decision [66,73,87]. However, even experienced machine learning system developers misuse and over trust explainability systems [51]. Additionally, explainability systems are often not developed with specific use cases in mind, meaning they are often not optimized for various downstream tasks for which they are later employed.

Prior work has examined the how machine learning and explainability systems are used in real world contexts [43,109] and the pitfalls of these systems when they are used in real world contexts [51]. I have researched how these systems can be chosen and adapted to account for user needs and user tested these solutions. This dissertation examines the following thesis statement:

Selecting and adapting machine learning prediction and explanation systems to align with user goals, to retain nuance in explanations, and to impart an appropriate level of trust can support people of varying expertise in leveraging these systems.

I examine this thesis statement across different contexts and amongst user groups with varying expertise through four projects:

1. Problems and an Alternative to Single Explanation Aggregations, which explored how explanation systems can be selected to align with user goals, adapted to retain nuance in explanations, and to impart an appropriate level of trust to support machine learning system developers.
2. Predicting Blood Glucose Test Accuracy in ICU Patients, which explored how prediction systems can be selected to align with user goals and to impart an appropriate level of trust to support clinicians and medical science.

3. Predicting and Explaining an Imminent Dementia Diagnosis with Limited Data, which explored how prediction and explanation systems can be selected and adapted to align with user goals and to retain nuance in explanations to support clinicians.
4. Flexible System for Efficient Goal-Directed Self-Tracking Analysis, which explored how prediction and explanation systems can be selected and adapted to align with user goals, to retain nuance in explanations, and to impart an appropriate level of trust to support patients.

I then discuss how the lessons learned on how to effectively design machine learning systems from these four projects compare to existing guidelines for designing human-AI interaction [4,5,36].

Chapter 2. RELATED WORK

I will first discuss the body of related work that is relevant for the general insights found across all four of the projects. In particular, I discuss the prior work in creating Human-AI interaction guidelines and frameworks to help in the design and implementation of any machine learning prediction and explanation system. I will then touch upon the prior work for each project's specific application domain.

2.1 HUMAN-AI INTERACTION FRAMEWORKS

Three of the major tech companies, Microsoft [4], Apple [5], and Google [36], have each presented their own guidelines or frameworks on human-AI interaction in machine learning systems. All three sets of guidelines have significant overlap, but they were all developed in slightly different manners with slightly different goals. I use the term framework to distinguish between a full set of guidelines introduced by each company and an individual guideline that is stated within each company's full guideline frameworks. I discuss each of the frameworks individually and then discuss a comparative analysis of all three sets of guidelines.

Microsoft presents "Guidelines for Human-AI Interaction" (Table 2.1) that start by compiling the large body of past work on human-AI interaction and design guidelines. They then perform qualitative data analysis to synthesize this past work into a single set of common themes and guidelines. Finally, they ran a heuristic evaluation user study with HCI practitioners to clarify, modify, and validate the set of guidelines. This work was published as an academic paper at the CHI conference in 2019 [4]. This method of generating guidelines means that Microsoft's own guidelines take a more general and academic tone, with less concrete suggestions for how to execute these guidelines in practice.

Table 2.1. Microsoft's Guidelines for Human-AI Interaction [4]

	AI Design Guidelines	Example Applications of Guidelines
Initially	**Make clear what the system can do.** Help the user understand what the AI system is capable of doing.	[Activity Trackers, Product #1] "Displays all the metrics that it tracks and explains how. Metrics include movement metrics such as steps, distance traveled, length of time exercised, and all-day calorie burn, for a day."
	Make clear how well the system can do what it can do. Help the user understand how often the AI system may make mistakes.	[Music Recommenders, Product #1] "A little bit of hedging language: 'we think you'll like'."
During Interaction	**Time services based on context.** Time when to act or interrupt based on the user's current task and environment.	[Navigation, Product #1] "In my experience using the app, it seems to provide timely route guidance. Because the map updates regularly with your actual location, the guidance is timely."
	Show contextually relevant information. Display information relevant to the user's current task and environment.	[Web Search, Product #2] "Searching a movie title returns show times in near my location for today's date"
	Match relevant social norms. Ensure the experience is delivered in a way that users would expect, given their social and cultural context.	[Voice Assistants, Product #1] "[The assistant] uses a semiformal voice to talk to you - spells out "okay" and asks further questions."
	Mitigate social biases. Ensure the AI system's language and behaviors do not reinforce undesirable and unfair stereotypes and biases.	[Autocomplete, Product #2] "The autocomplete feature clearly suggests both genders [him, her] without any bias while suggesting the text to complete."
When Wrong	**Support efficient invocation.** Make it easy to invoke or request the AI system's services when needed.	[Voice Assistants, Product #1] "I can say [wake command] to initiate."
	Support efficient dismissal. Make it easy to dismiss or ignore undesired AI system services.	[E-commerce, Product #2] "Feature is unobtrusive, below the fold, and easy to scroll past… Easy to ignore."
	Support efficient correction. Make it easy to edit, refine, or recover when the AI system is wrong.	[Voice Assistants, Product #2] "Once my request for a reminder was processed I saw the ability to edit my reminder in the UI that was displayed. Small text underneath stated 'Tap to Edit' with a chevron indicating something would happen if I selected this text."

	Scope services when in doubt. Engage in disambiguation or gracefully degrade the AI system's services when uncertain about a user's goals.	[Autocomplete, Product #1] "It usually provides 3-4 suggestions instead of directly auto completing it for you"
	Make clear why the system did what it did. Enable the user to access an explanation of why the AI system behaved as it did.	[Navigation, Product #2] "The route chosen by the app was made based on the Fastest Route, which is shown in the subtext."
Over Time	**Remember recent interactions.** Maintain short term memory and allow the user to make efficient references to that memory.	[Web Search, Product #1] "[The search engine] remembers the context of certain queries, with certain phrasing, so that it can continue the thread of the search (e.g., 'who is he married to' after a search that surfaces Benjamin Bratt)"
	Learn from user behavior. Personalize the user's experience by learning from their actions over time.	[Music Recommenders, Product #2] "I think this is applied because every action to add a song to the list triggers new recommendations."
	Update and adapt cautiously. Limit disruptive changes when updating and adapting the AI system's behaviors.	[Music Recommenders, Product #2] "Once we select a song they update the immediate song list below but keeps the above one constant."
	Encourage granular feedback. Enable the user to provide feedback indicating their preferences during regular interaction with the AI system.	[Email, Product #1] "The user can directly mark something as important, when the AI hadn't marked it as that previously"
	Convey the consequences of user actions. Immediately update or convey how user actions will impact future behaviors of the AI system.	[Social Networks, Product #2] "[The product] communicates that hiding an Ad will adjust the relevance of future ads."
	Provide global controls. Allow the user to globally customize what the AI system monitors and how it behaves.	[Photo Organizers, Product #1] "[The product] allows users to turn on your location history so the AI can group photos by where you have been."
	Notify users about changes. Inform the user when the AI system adds or updates its capabilities.	[Navigation, Product #2] "[The product] does provide small in app teaching callouts for important new features. New features that require my explicit attention are pop-ups."

Microsoft presents 18 guidelines split into 4 categories, as seen in Table 2.1, according to when the guideline will most likely apply in the machine learning system development lifecycle. The four categories of when guidelines might be applied are “initially”, “during interaction”, “when wrong”, and “over time”. Three of the specific guidelines presented that are particularly relevant are “make clear how well the system can do what it can do” in the “initially” category, “make clear why the system did what it did” in the “when wrong” category, and the “show contextually relevant information” in the “during interaction” category.

Google presents a “People + AI Guidebook” (Table 2.2) that is based both on academic literature and internal machine learning and AI guidelines from product teams. This guidebook is presented as a webpage and associated case studies and workshop resources and was first published online in 2019 [36]. This mix of academic references and product application knowledge means that the guidelines are both more numerous and more specific compared to Microsoft’s. They also include more examples of how these guidelines might be put into practice.

Table 2.2. The chapters and subsections of Google's People + AI Guidebook [36]

Chapter Title	Subsection
User Needs + Defining Success	**Find the intersection of user needs & AI strengths.** Solve a real problem in ways in which AI adds unique value.
	Assess automation vs. augmentation. Automate tasks that are difficult, unpleasant, or where there's a need for scale; and ideally ones where people who currently do it can agree on the "correct" way to do it.
	Design & evaluate the reward function. The "reward function" is how an AI defines successes and failures. Deliberately design this function with a cross-functional team, optimizing for long-term user benefits by imagining the downstream effects of your product. Share this function with users when possible.
Data Collection + Evaluation	**Plan to gather high-quality data from the start.** Data is critical to AI, but more time and resources are often invested in model development than data quality. You'll need to plan ahead as you gather and prepare data, to avoid the effects of poor data choices further downstream in the AI development cycle.
	Translate user needs into data needs. Determine the type of data needed to train your model. You'll need to consider predictive power, relevance, fairness, privacy, and security.
	Source your data responsibly. Whether using pre-labeled data or collecting your own, it's critical to evaluate your data and their collection method to ensure they're appropriate for your project.
	Prepare and document your data. Prepare your dataset for AI, and document its contents and the decisions that you made while gathering and processing the data.
	Design for labelers & labeling. For supervised learning, having accurate data labels is crucial to getting useful output from your model. Thoughtful design of labeler instructions and UI flows will help yield better quality labels and therefore better output.
	Tune your model. Once your model is running, interpret the AI output to ensure it's aligned with product goals and user needs. If it's not, then troubleshoot: explore potential issues with your data.
Mental Models	**Set expectations for adaptation.** AI allows for more systems to adapt, optimize, and personalize to users, and probability-based user experiences have become more common over time. Building on the familiarity of existing mental models can help users feel comfortable.
	Onboard in stages. When introducing users to an AI-powered product, explain what it can do, what it can't do, how it may change, and how to improve it.

	Plan for co-learning. People will give feedback to AI products, which will adjust the models and change how people interact with them — which will change the machine learning models further. Users' mental models will similarly change over time.
	Account for user expectations of human-like interaction. People are more likely to have unachievable expectations for products that they assume have human-like capabilities. It's important to communicate the algorithmic nature and limits of these products to set realistic user expectations and avoid unintended deception.
Explainability + Trust	**Help users calibrate their trust.** Because AI products are based on statistics and probability, the user shouldn't trust the system completely. Rather, based on system explanations, the user should know when to trust the system's predictions and when to apply their own judgement.
	Plan for trust calibration throughout the product experience. Establishing the right level of trust takes time. AI can change and adapt over time, and so will the user's relationship with the product.
	Optimize for understanding. In some cases, there may be no explicit, comprehensive explanation for the output of a complex algorithm. Even the developers of the AI may not know precisely how it works. In other cases, the reasoning behind a prediction may be knowable, but difficult to explain to users in terms they will understand.
	Manage influence on user decisions. AI systems often generate output that the user needs to act on. If, when, and how the system calculates and shows confidence levels can be critical in informing the user's decision making and calibrating their trust.
Feedback + Control	**Align feedback with model improvement.** Clarify the differences between implicit and explicit feedback, and ask useful questions at the right level of detail.
	Communicate value & time to impact. Understand why people give feedback so you can set expectations for how and when it will improve their user experience.
	Balance control & automation. Give users control over certain aspects of the experience and allow them to easily opt out of giving feedback.
Errors + Graceful Failure	**Define "errors" & "failure".** What the user considers an error is deeply connected to their expectations of the AI system. For example, a recommendations system that's useful 60% of the time could be seen as a failure or a success, depending on the user and the purpose of the system. How these interactions are handled establishes or corrects mental models and calibrates user trust.
	Identify error sources. With AI systems, errors can come from many places, be harder to identify, and appear to the user and to system creators in non-intuitive ways.
	Provide paths forward from failure. AI capabilities can change over time. Creating paths for users to take action in response to the errors they encounter encourages patience with the system, keeps the user-AI relationship going, and supports a better overall experience.

Google's guidebook is divided into 6 chapters: "user needs and defining success", "data collection and evaluation", "mental models", "explainability and trust", "feedback and control", and "errors and graceful failure". Within each chapter is a set of key considerations and implementation suggestions for some of those considerations. The chapters of particular interest are "user needs and defining success", "mental models", and "explainability and trust".

Apple's "Human Interface Guidelines for Machine Learning" (Table 2.3) are the least academic focused and most design focused. They were developed entirely based on human-AI interaction and machine learning guidelines that exist within Apple [5]. These guidelines were first presented at the Apple Worldwide Developer Conference in 2019 and are now presented as a series of webpages. Given this focus on internal design guidelines, Apple's guidelines are the most product focused and are the most specific and sometimes least generalizable of three sets of guidelines.

Table 2.3. Categories, sections, and subsections from Apple's Human Interface Guidelines for Machine Learning [5]

Category	Section	Subsection
Machine Learning	Machine Learning Roles	Critical or complementary
		Private or public
		Proactive or reactive
		Visible or invisible
		Dynamic or static
Inputs	Explicit Feedback	Request explicit feedback only when necessary
		Always make providing explicit feedback a voluntary task
		Don't ask for both positive and negative feedback
		Use simple, direct language to describe each explicit feedback option and its consequences
		Add iconography to an option description if it helps people understand it
		Consider offering multiple options when requesting explicit feedback
		Act immediately when you receive explicit feedback and persist the resulting changes
		Consider using explicit feedback to help improve when and where you show results
	Implicit Feedback	Always secure people's information
		Help people control their information
		Don't let implicit feedback decrease people's opportunities to explore
		When possible, use multiple feedback signals to improve suggestions and mitigate mistakes
		Consider withholding private or sensitive suggestions
		Prioritize recent feedback
		Use feedback to update predictions on a cadence that matches the user's mental model of the feature
		Be prepared for changes in implicit feedback when you make changes to your app's UI
		Beware of confirmation bias
	Calibration	Always secure people's information
		Be clear about why you need people's information
		Collect only the most essential information
		Avoid asking people to participate in calibration more than once
		Make calibration quick and easy
		Make sure people know how to perform calibration successfully
		Immediately provide assistance if progress stalls
		Confirm success
		Let people cancel calibration at any time
		Give people a way to update or remove information they provided during calibration

	Corrections	Give people familiar, easy ways to make corrections
		Provide immediate value when people make a correction
		Let people correct their corrections
		Always balance the benefits of a feature with the effort required to make a correction
		Never rely on corrections to make up for low-quality results
		Learn from corrections when it makes sense
		When possible, use guided corrections instead of freeform corrections
Outputs	Mistakes	Understand the significance of a mistake's consequences
		Make it easy for people to correct frequent or predictable mistakes
		Continuously update your feature to reflect people's evolving interests and preferences
		When possible, address mistakes without complicating the UI
		Be especially careful to avoid mistakes in proactive features
		As you work on reducing mistakes in one area, always consider the effect your work has on other areas and overall accuracy
	Multiple Options	Prefer diverse options
		In general, avoid providing too many options
		List the most likely option first
		Make options easy to distinguish and choose
		Learn from selections when it makes sense
	Confidence	Know what your confidence values mean before you decide how to present them
		In general, translate confidence values into concepts that people already understand
		In situations where attributions aren't helpful, consider ranking or ordering the results in a way that implies confidence levels
		In scenarios where people expect statistical or numerical information, display confidence values that help them interpret the results
		Whenever possible, help people make decisions by conveying confidence in terms of actionable suggestions
		Consider changing how you present results based on different confidence thresholds
		When you know that confidence values correspond to result quality, you generally want to avoid showing results when confidence is low
	Attribution	Consider using attributions to help people distinguish among results
		Avoid being too specific or too general
		Keep attributions factual and based on objective analysis
		In general, avoid technical or statistical jargon
	Limitations	Help people establish realistic expectations
		Demonstrate how to get the best results
		Explain how limitations can cause unsatisfactory results
		Consider telling people when limitations are resolved

Apple's guidelines are broken down into categories, sections within each category, and then subsections which contain specific recommendations and suggested considerations. Apple's initial category of "machine learning roles" discusses different paradigms a machine learning system might adopt. Of particular interest in the "inputs" section is the "implicit feedback" category. In the "outputs" section, the "attribution", "confidence", and "limitations" categories are particularly applicable to the projects presented in this dissertation.

Wright et. al. performed a systematic review of these three human-AI interaction frameworks to come up with a unified guideline structure that includes all of the guidelines present in each of the three distinct frameworks [105]. They find that the three frameworks can be categorized into 4 higher level categories, each with its own sub-categories.

Figure 2.1. Wright et al.'s Unified Guideline Structure [105]

Interestingly, not every framework has a guideline in every sub-category and some framework's guidelines are more heavily represented in certain overarching categories than others. For example, Google focuses about a quarter of their guidelines on model considerations while Apple only has one guideline in the model category and Microsoft has none. As another example,

Google's framework is the only one that includes guidelines in the sub-category of "value of AI". I focus most of the rest of this dissertation discussing themes in the "value of AI" subcategory within the "initial" category and the "mental models", "explainability", and "confidence" subcategories within the "interface" category.

2.2 Other Human-AI Interaction Research

The three frameworks discussed above relied heavily on prior work in the human-AI interaction space. Many prior works have suggested various design suggestions and guidelines to take into account when designing more specific human-AI systems [44,45,75,95]. Other work has focused on examining how and why human-AI interaction is difficult to design for [106]. Wang et al. presented a framework for building human-centered explainable AI systems which focuses on how understanding people should inform explanations of AI [102]. These prior works are generally subsumed by the three sets of human-AI interaction guidelines discussed above and are the most inclusive and application driven of human-AI interaction prior work.

Other prior work has provided examples of deploying individual machine learning solutions to be used by people of varying sets of expertise. Yang et al. deployed a clinical decision support tool to be used by clinicians in a hospital doing artificial heart implants and found it was necessary to 1) embed the decision support tool into the current workflow of clinicians and 2) slow down decision-making only when necessary [107]. The takeaways presented in this study add further evidence to the findings presented in this dissertation.

2.3 Domain Specific Related Work

Given that each of the four projects is grounded in a very different, specialized subfields, further related work for each individual project will be discussed within the individual chapters of

each project. For the Problems and an Alternative to Single Explanation Aggregations project, Section 3.1 discusses prior work in machine learning local explainability systems, their documented pitfalls, and work related to human-AI interaction in the context of explainability systems. Related work for Predicting Blood Glucose Test Accuracy in ICU Patients is discussed in Section 4.1 and covers the use of point-of-care blood glucose tests in ICU settings, point-of-care blood glucose test accuracy, and potential sources of error for point-of-care blood glucose tests. Section 5.2 covers related work for the Predicting and Explaining an Imminent Dementia Diagnosis with Limited Data project including state-of-the-art dementia diagnosis, basic dementia risk factors, modeling cognitive trajectories, and diagnosing cognition status. Related work for the Flexible System for Efficient Goal-Directed Self-Tracking Analysis project can be found in Section 6.2 and covers self-tracing, Bayesian analysis and network learning, and Bayesian analysis within self-tracking.

Chapter 3. PROBLEMS AND AN ALTERNATIVE TO SINGLE EXPLANATION AGGREGATIONS

In this work I explored the problems with the existing design of a single aggregation of local explanations for dataset level analysis done by machine learning system developers. I also explored how an alternative design that focuses on retaining nuance in explanations and imparting an appropriate level of skepticism can help developers more accurately leverage the explanation systems for a variety of different goals. This work was done with senior advisors Rich Caruana, Nick Craswell, Kori Inkpen, Scott M. Lundberg, Harsha Nori, Hanna Wallach, Jennifer Wortman Vaughan and was published in the IEEE Data Engineering Bulletin on Responsible AI and Human-AI Interaction [78].

3.1 INTRODUCTION

Machine learning is used in a wide range of domains, including medicine, finance, and education. Applications of machine learning impact people's day-to-day lives and livelihoods, yet the behavior of popular models like neural networks is often too complex to fully understand or communicate. In order for stakeholders of systems that rely on machine learning—including machine learning developers, domain experts, and those impacted by such systems—to reason about their behavior, the models involved must be interpretable. Interpretability can support knowledge discovery, enable stakeholders to surface problematic model behavior, enhance stakeholder ability to communicate what their models have learned, and provide stakeholders with a way to calibrate their trust in models [43,104].

There are two common approaches to achieving model interpretability. The first is to train simple and transparent glass-box models that are intended to be interpretable by design. Common

examples include decision trees [85], point systems [48,108], and generalized additive models [16,37]. By examining the internals of a glass-box model, it is possible to obtain an accurate global view of that model's behavior.

In contrast, local interpretability methods provide (generally post-hoc) explanations of a model's predictions for individual data points. Local explanations can take several different forms. Some explain predictions in terms of the most influential training data points (e.g., [55]). Others provide counterfactual explanations, describing how data points could be modified to obtain different predictions (e.g., [90,101]). Perhaps most often, local explanations take the form of feature attribution scores, which capture some notion of how "important" each feature is to each prediction, as shown in the top panel of Figure 3.1. For example, SHAP (Shapley Additive Explanations) divides "credit" for a model's prediction across all of its features using the concept of Shapley values from cooperative game theory [66]. In contrast, LIME (Local Interpretable Model-Agnostic Explanations) generates feature attribution scores by learning a local linear approximation of a model around each data point [87]. Because these explanations are tailored to individual data points, local interpretability methods may be appropriate when stakeholders need, want, or are owed individualized explanations of a model's predictions, such as in personalized medical contexts (e.g., to explain a patient's predicted diagnosis or prognosis) or financial contexts (e.g., to explain an applicant's predicted likelihood of paying back a loan). Although such explanations do not perfectly reflect what the underlying model is doing [89,103], they have the advantage that they can be generated even for complex black-box models, such as neural networks, random forests, or ensemble methods.

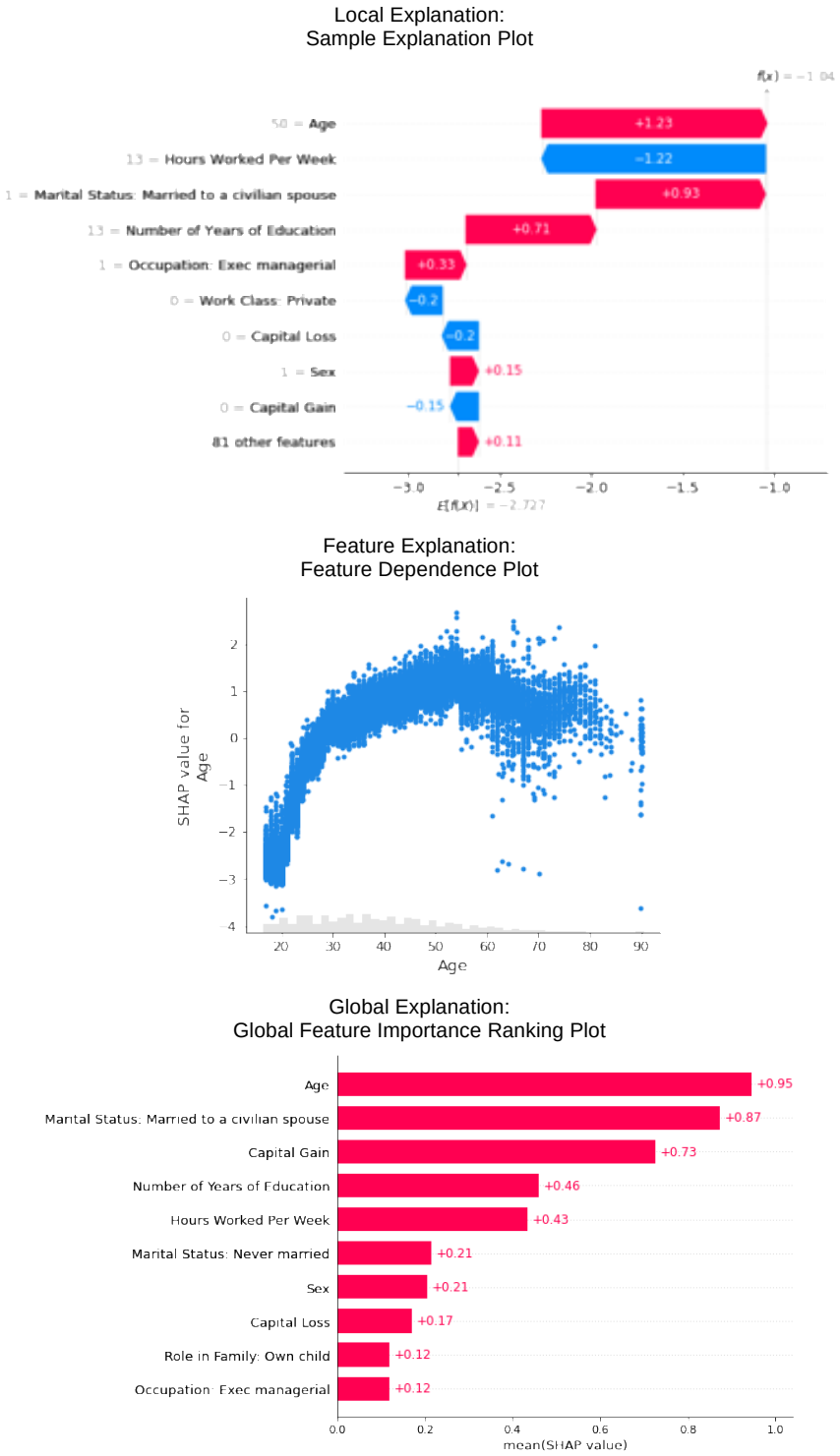

Figure 3.1. Explanations provided by the SHAP Python package. Top: A local explanation for a single data point. Each bar represents a single feature's attribution score for that data point. Middle: A single feature's attribution scores for all training data points, with each data point represented by a dot. Bottom: Global feature attributions obtained by taking the mean absolute value of each feature's absolute attribution scores across all training data points and then ranking the features by their average scores. [79]

Despite their popularity, local interpretability methods are not designed to provide a global view of a model's behavior. However, many common interpretability tools, including the SHAP Python package [65] and InterpretML [74], offer makeshift global feature attributions obtained by taking the mean absolute value of each feature's attribution scores across all training data points and then ranking the features by their average scores, as shown in the bottom panel of Figure 3.1. Such global feature attributions can give a sense of which features a model uses most "on average" across its training dataset. This kind of concise overview of a model is valued by machine learning developers—indeed, in a preliminary study that we ran in order to understand the current practices of experienced users of interpretability tools (described in Section 3.2), we found that machine learning developers commonly rely on these global feature attributions to get an overall sense of what their models have learned, to communicate this information to other stakeholders, and to perform other tasks in their workflows.

However, simply averaging feature attribution scores may not always be appropriate. Reducing a distribution to a single summary statistic loses information, and it is well known that the (arithmetic) mean is susceptible to outliers. Relying on a single summary statistic to make inferences about individuals can lead to ecological fallacies [88]. Furthermore, it may also obscure potentially harmful behavior exhibited by a model for the data points associated with a particular group of people—for example, in a medical context, older patients or patients with certain preexisting conditions. Indeed, with society's increased emphasis on mitigating unfairness caused by systems that rely on machine learning, there has been a push to move away from overreliance on averages and to instead take a more holistic view of model behavior (e.g., [8,76]). Since interpretability is often framed as a way to promote fairness, overreliance on averages may be especially problematic in this context.

In Section 3.3, using models trained on the Adult [56] and NHANES [17] datasets as case studies, we explore the ramifications of averaging feature attribution scores. For each model, we compare the global feature attributions obtained using the status quo approach—that is, by taking the mean absolute value of each feature's attribution scores across all training data points and then ranking the features by their average scores—with a suite of global feature attributions obtained by supplementing the mean absolute value with other summary statistics. We find that the status quo approach can yield overly simplistic global views, as well as overlooking important aspects of model behavior that are present only for a subset of the training data points. We show that using other summary statistics in place of the mean absolute value can help derive different, complementary insights into a model's predictions and may be better suited for different tasks. We therefore propose giving machine learning developers the opportunity to compare and contrast different global feature attributions obtained by ranking features by other summary statistics of their attribution scores, potentially enabling them to obtain a more nuanced global view of model behavior.

To explore whether machine learning developers would benefit from being able to compare and contrast different global feature attributions, we ran an artifact-based interview study with seven participants who had experience with interpretability tools. Participants were first shown the usual global feature attributions provided by SHAP and asked some questions about the underlying model. They were then shown a suite of global feature attributions obtained by ranking features by four different summary statistics of their attribution scores—what we refer to as a global feature attribution suite—and asked to reconsider their answers. We note that we do not view the global feature attribution suite itself as a contribution, but rather as an artifact for exploring machine

learning developer perceptions, needs, and challenges around global feature attributions. Our study addresses the following research questions:

- How do machine learning developers make sense of and use global feature attributions obtained by ranking features by different summary statistics of their attribution scores?
- Does the ability to compare and contrast different global feature attributions allow machine learning developers to better understand the nuanced behavior of models?
- What challenges do machine learning developers face when comparing and contrasting global feature attributions?

We found that machine learning developers are able to use different global feature attributions to achieve tasks and objectives including communicating what their models have learned and identifying next steps for debugging their models. Viewing the global feature attribution suite increased participant uncertainty in their understanding of the underlying model (compared with viewing the usual global feature attributions alone) as they became more aware of the intricacies of the model's behavior. However, they expressed a tension between the benefits obtained by using tools like SHAP to quickly get a sense of what a model has learned and the time it would take to compare and contrast different global feature attributions. This tension might limit machine learning developer willingness to use a global feature attribution suite in their own workflows, echoing observations from prior work about the need to balance the benefits of quick understanding and slower, nuanced understanding when designing interpretability tools [51].

This paper contributed to a recent line of research exploring human-centered approaches to interpretability. Much of this research focuses on how stakeholders use and understand interpretability tools [1,3,14,15,42,43,51,58,63,83,104,109]. Within this research, Kaur et al. found that even experienced machine learning developers tend to misuse and place too much trust

in interpretability tools [51]. They therefore suggested designing interpretability tools that explicitly highlight the nuanced behavior of models, as well as methods that counterbalance the bias toward simple—and potentially misleading—explanations. We see our work as a first exploration of how one might facilitate deeper understanding by enhancing overly simplistic global views of a model.

3.2 Preliminary Study

To better ground our research, we ran a small preliminary study during the summer of 2020 to help us understand the current practices of experienced users of interpretability tools. We conducted semi-structured interviews with ten machine learning developers (e.g., data scientists, research scientists, PhD students) across a variety of domains (e.g., medicine, finance, retail). Participants were recruited through a combination of posts to relevant email lists and message boards at our institution, direct emails to individuals who had written blog posts or made contributions to either the SHAP Python package or InterpretML, and snowball sampling. Each participant had experience using at least one common interpretability tool, and nine had experience specifically with SHAP. Table 3.1 contains additional information about the participants.

Table 3.1. Descriptions of the participants in our studies

ID	Job Description	Years in ML	Types of Data Worked With	Interpretability Tools or Methods Used	Study 2 Dataset
P1	ML PhD Student	2	Medical	SHAP, self-made visualizations	NHANES
P2	ML PhD Student	1	Medical	InterpretML, SHAP, LIME, GAMs, self-made visualizations	NHANES
P3	ML Practitioner	2	Remote Sensing, Retail, Banking	SHAP, self-made visualizations	N/A
P4	Environmental Sci. PhD Student	3	Environmental, Geospatial	SHAP, GAMs, self-made visualizations	N/A
P5	Data Scientist	2	Retail	SHAP	Adult
P6	MD and ML PhD Student	4	Medical	SHAP, GAMs, self-made visualizations	NHANES
P7	Research Scientist	6	Technology, Medical	SHAP, LIME, GAMs, self-made visualizations	Adult
P8	Data Scientist	4	Retail	AzureML, SHAP, LIME	Adult
P9	Data Scientist, Program Manager	7	Retail, Financial, User Behavior	SHAP	N/A
P10	ML Practitioner	3	Medical, Financial	InterpretML, AzureML, LIME, self-made visualizations	Adult

During the interviews, we first asked participants about their background and experience with both machine learning in general and interpretability tools in particular. Next, we asked them to describe the tasks and objectives they use interpretability tools to achieve, both alone and with collaborators. Participants were asked to walk through examples of specific times they had used interpretability tools to accomplish those tasks and objectives and were led through a series of open-ended questions intended to uncover the strategies they had used, including what had worked well and what had not. Finally, participants were asked if they had any wishes for a potential new interpretability tool or for new functionality for an existing interpretability tool. All interviews were conducted virtually on a video conferencing platform due to the COVID-19 pandemic. Audio from the interviews was recorded and transcribed by a third-party service, after which the audio transcripts were reviewed for accuracy and anonymized. The first author then coded the transcripts

using a bottom-up approach and four authors conducted a thematic analysis. The study was approved by our institution's IRB. Participation was voluntary and participants received up to $75 in compensation for their participation.

Participants described using interpretability tools for tasks and objectives including model debugging, improving model performance, communication and collaboration (including building collaborator trust in models), and knowledge discovery. These tasks and objectives are consistent with those identified by Hong et al. [43]. In total, participants mentioned more than forty different strategies for accomplishing these tasks and objectives, such as looking for patterns, outliers, and anomalies in scatter plots of feature attribution scores for all training data points (as in the middle panel of Figure 3.1); comparing observed patterns with prior knowledge; and turning to domain experts when some aspect of an explanation was unclear.

Strikingly, although our preliminary study was not specifically designed to explore the use of global feature attributions, all ten participants said that they use global feature attributions (obtained using the status quo approach of taking the mean absolute value of each feature's attribution scores across all training data points and then ranking the features by their average scores) somewhere in their workflow. Participants mentioned using global feature attributions to get an overall sense of what their models have learned (e.g., for debugging or for determining the overall credibility of their models), to check that the "most important" features match their expectations, to determine which features to prioritize for in-depth analysis, and to communicate what their models have learned to other stakeholders.

However, participants also brought up several pain points around their use of global feature attributions. They were aware that using the mean absolute value could be problematic. As P2 said, *"ranking of feature importance is, you know, a very– somewhat arbitrary way to do things. You*

know, there's so many different importance measures. But, at least looking at something can tell us if our model has– is relying on reasonable features." Some participants mentioned that using the mean absolute value fails to account for relatively rare features that have a large influence when they are present. Participants also brought up the difficulty of communicating about the global behavior of models at a level that is more in-depth than the bar plots that common interpretability tools provide (see the bottom panel of Figure 3.1, for example).

Although it was not our original focus when we first set out to conduct this preliminary study, observing the overwhelming participant use of global feature attributions obtained using the status quo approach in their workflows—despite being aware of some of the pitfalls—motivated us to question whether machine learning developers would benefit from a more nuanced global view of model behavior. That is the question we address in this work. Other needs that emerged from the study include ways to explore and address feature correlation and confounding; less time-consuming ways to analyze individual features; ways to aggregate related features to understand their combined influence; ways to determine the reliability of explanations; ways to validate insights found using explanations; more customizable visualizations; and increased documentation for interpretability tools, including documentation aimed at expert users.

3.3 Benefits and Drawbacks of Different Summary Statistics

In this section, we review the way in which global feature attributions are most commonly obtained from feature attribution scores, describe some alternative approaches to doing this, and discuss the benefits and drawbacks of each. Although most of this discussion is applicable to any local interpretability method that generates feature attribution scores, we focus both here and in the rest of this paper on SHAP [66] for concreteness. SHAP's feature attribution scores, which are motivated by Shapley values from cooperative game theory, can be viewed as a way of dividing

the “credit” for a model’s prediction across all of its features. The sum of the feature attribution scores is equal to the expected value of the prediction for the data point in question. SHAP is widely used in practice—as of November 2021, the SHAP Python package had close to 15k stars on GitHub, and nine of the ten participants in our preliminary study had experience with SHAP.

We illustrate the benefits and drawbacks of different approaches to obtaining global feature attributions from feature attribution scores through case studies using models trained on two widely used open-source datasets: the Adult dataset [56] and the NHANES dataset [17]. The Adult dataset is based on 1994 US Census data and each data point corresponds to a person. The features include age, employment type, education, marital status, occupation, race, and sex, among others. The model that we trained on this dataset predicts whether or not a person makes at least $50k per year (the equivalent of about $92.5k in 2021 when adjusted for inflation). The NHANES dataset is a survival dataset from a longitudinal health and wellness study. Again, each data point corresponds to a person. The features include age, race, sex, poverty index, BMI, lab blood test results, and blood pressure measurements. The model that we trained on this dataset is a Cox proportional hazards model that predicts the differential risk of a person dying versus the typical log hazard background risk.

Although SHAP was designed to offer only local explanations, the SHAP Python package additionally constructs makeshift global feature attributions as follows: First, for each feature, take the mean absolute value of that feature’s attribution scores across all training data points. Next, rank the features by their average scores. The resulting global feature attributions for the models trained on the Adult and NHANES datasets, respectively, can be seen in the top row of Figure 3.2. For example, according to these global feature attributions, age is the most important feature for both models. This is intuitive in the Adult dataset since people who are older and therefore later in

their career tend to earn more money. It is also intuitive in the NHANES dataset because age is highly correlated with how likely someone is to die in the near future. But this does not tell the whole story. For each of these models, does age play an equal role in the model's predictions for all training data points, or is it more important for some data points than for others? Are there groups of data points for which the model relies on completely different features? Are there outlier data points for which the model relies on features that it should not? If the goal is to debug the model, what should the next step be?

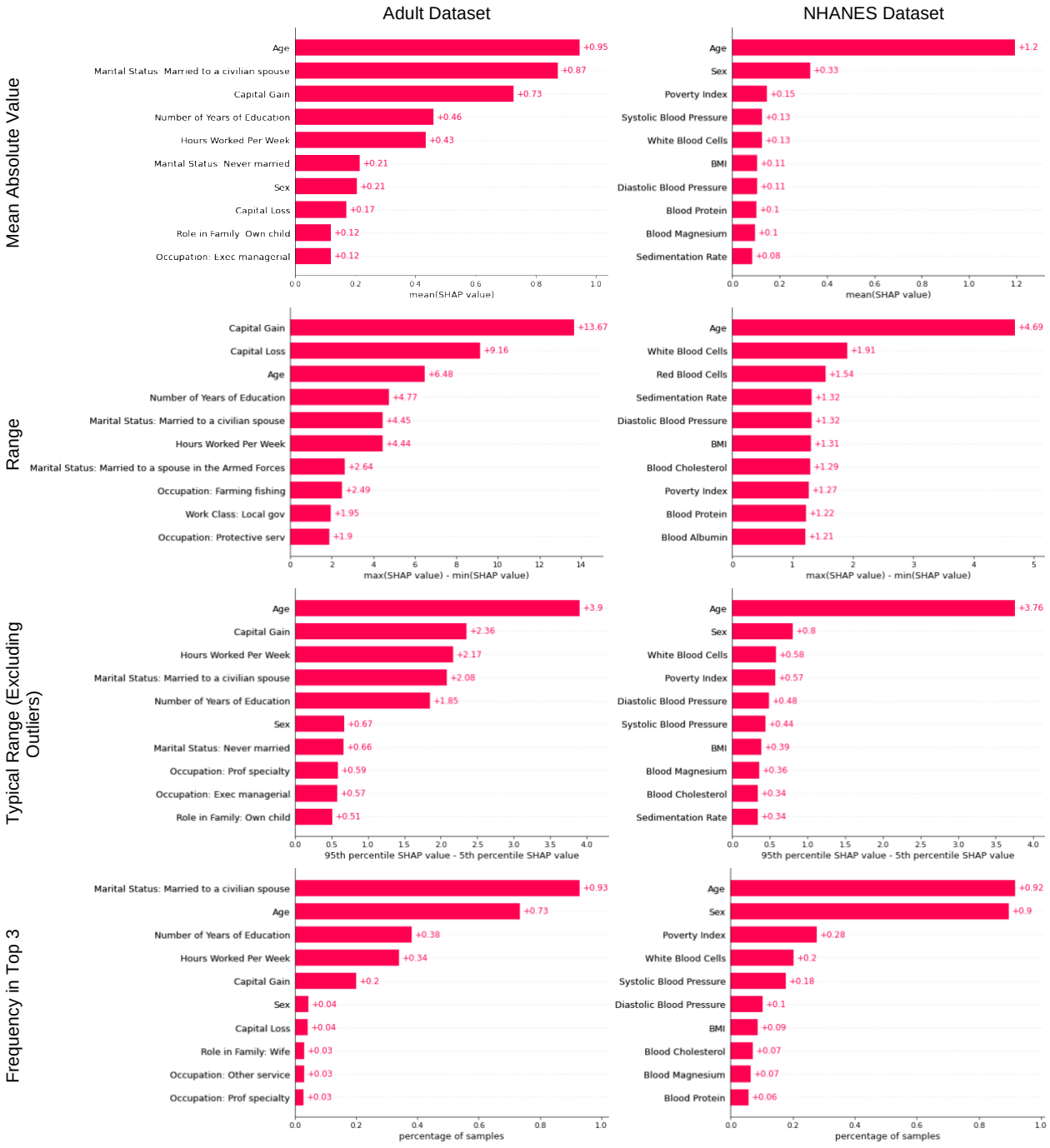

Figure 3.2. Global feature attributions obtained by ranking features by different summary statistics of their attribution scores. Each row corresponds to a summary statistic: the mean absolute value, the range, the typical range, and the frequency in the top three. The column on the left contains global feature attributions for the model trained on the Adult dataset; the column on the right contains global feature attributions for the model trained on the NHANES dataset. [79]

To answer these questions, an machine learning developer could turn to a visualization of a particular feature's attribution score across all training data points, such as the type of scatter plot shown in the middle panel of Figure 3.1 or the beeswarm plots available in the SHAP Python package, both of which provide a more detailed view of a feature's influence. However, for models with hundreds or even thousands of features, it is too burdensome to explore and compare all such plots—indeed, this is why developers turn to summary statistics in the first place. And even with a small number of features, comparing plots across multiple features is not easy.

Instead, we consider supplementing the mean absolute value with other summary statistics. Using different summary statistics yields different rankings of the features and, as we show below, substantively different takeaways. Although in principle any summary statistic could be used, we propose a few alternatives that capture different aspects of the distribution of a model's feature attribution scores across all training data points.

We first consider the range of a feature's attribution scores—that is, the difference between the maximum attribution score for that feature across all training data points and the minimum attribution score for that feature across all training data points (not taking absolute values). Features with a large range of attribution scores are highly influential on at least some data points. Outlier data points can be found by examining scatter plots for features with a large range. This is useful both for understanding a model's behavior on unusual data points and for identifying bugs. As we can see from the second row of Figure 3.2, when predicting whether someone makes over $50k a year using the Adult dataset, capital loss is only the eighth-highest ranked feature when using the mean absolute value, but the second-highest ranked feature in terms of range. This is due to extreme outliers— specifically, atypically high capital loss values—in the training dataset. The prominence of capital loss in this alternative ranking might help draw a developer's attention to

this issue so they can investigate whether it stems from a bug that needs fixing or whether it reflects a true phenomenon in the underlying population.

In some cases, the range may be too susceptible to outliers. Even a single data point with an extreme feature attribution score can boost a feature's range. This can be problematic if the goal is not to identify individual outlier data points, but to identify larger groups of data points for which a feature is highly influential. As a result, for this task, it may be more appropriate to use a censored version of the range. We define the typical range of a feature's attribution scores to be the difference between the feature's ninety-fifth-percentile feature attribution score and its fifth percentile feature attribution score across all training data points. The choice of the 95^{th} and 5^{th} percentiles are somewhat arbitrary, and other percentiles could be used. We thought that this choice would balance the ability to identify groups of data points with robustness to extreme outliers. Ranking features by their typical range can reveal features that are influential not just for a handful of data points, but for a more substantial subset of data points. It can therefore be used to identify groups of data points for which the model behaves similarly. Examining the second row of Figure 3.2, we can see that both red blood cell count and white blood cell count have a large range for the NHANES model. However, examining the third row, we can see that only white blood cell count ranks highly in terms of the typical range. This suggests that white blood cell count is an important feature for a larger subset of the training data points than red blood cell count, for which the large range may be due to outliers.

The final summary statistic that we consider enables us to get a sense of which features are influential for a large proportion of the training data points without worrying about the specific values of their attribution scores. We define the frequency in the top three to be the fraction of the training data points for which the feature in question ranks among the top three in terms of its

absolute feature attribution scores. We can think of this as letting every data point vote for its top three most important features and then tallying up the votes across the training dataset. Again, the choice of 3 votes per data point is arbitrary and other values could be used. Compared with the mean absolute value, the frequency in the top three provides a way to control for high variance in the feature attribution scores. When predicting whether someone will make over $50k a year using the Adult Dataset, the capital gain feature ranks third in terms of the mean absolute value, and one might therefore assume it is important for all data points. However, examining the final row of Figure 3.2, we can see that capital gain is one of the top three most important features for only 20% of the training data points. In contrast, hours worked per week is in the top three for 34% of the training data points, while its mean absolute feature attribution score is significantly lower (0.43, compared with 0.73 for capital gain).

Different summary statistics will yield different global feature attributions that can be used to derive different— and often complementary—insights. We therefore propose that a more accurate global view of a model's behavior might be achieved by allowing machine learning developers to compare and contrast different global feature attributions. In the next section, we describe a study that we designed to explore this idea.

3.4 Main Study

To explore whether machine learning developers would benefit from being able to compare and contrast different global feature attributions, we ran a study in which participants were asked to answer questions about a model before and after seeing global feature attributions obtained by ranking features by four different summary statistics of their attribution scores, as described in Section 3.3. We refer to this as a global feature attribution suite and use it as an artifact for

exploring machine learning developer perceptions, needs, and challenges around global feature attributions.

3.4.1 *Methods*

For this study, which we conducted during the summer of 2020, we recruited seven participants, all of whom had participated in our preliminary study (see Section 3.2) and had agreed to be contacted for follow-up research; the remaining three participants declined to participate. The study was approved by Microosoft's IRB. Each interview lasted approximately one hour and participants received a $50 gift card for their participation.

The study consisted of semi-structured interviews in which participants were shown two different static (HTML file) Jupyter notebooks. Both notebooks contained a model, a textual description of the dataset used to train the model, a beeswarm plot visualizing the distribution of attribution scores for each feature, and a feature dependence scatter plot for each feature (as in the middle panel of Figure 3.1). In the first notebook, we included a bar plot showing global feature attributions obtained using the status quo approach—that is, by taking the mean absolute value of each feature's attribution scores across all training data points and then ranking the features by their average scores, as in the top row of Figure 3.2—as well as a description of how these global feature attributions were obtained and a brief list of potential uses. In the second notebook, we additionally included bar plots showing global feature attributions obtained using other summary statistics (specifically, the range, the typical range, and the frequency in the top three) in addition to the mean absolute value, as shown in Figure 3.2. We described how these global feature attributions were obtained and listed potential uses for each, using the wording in Table 3.2. All participants were shown the first notebook before the second notebook. We chose to show the notebooks sequentially, as opposed to using a counterbalanced design, so that we could first

observe how participants made use of the usual global feature attributions provided by SHAP, and then see whether and how their perspectives changed when they were shown the global feature attribution suite.

Table 3.2. The descriptions of the summary statistics that were shown to participants

Statistic	How it is Calculated	Potential Uses
Mean Absolute Value	Mean over all samples in the training data set of the absolute value of each sample's model attribution score.	Gives a sense of what the model is learning overall. Currently the default global feature importance ranking in SHAP.
Range	Difference between the maximum model attribution score and the minimum model attribution score of the given feature over the training data set.	Identifies features that are heavily influential on at least a small number of samples in the data. Can also help find extreme outliers in the data.
Typical Range (Excluding Outliers)	Difference between the 95th percentile model attribution score and the 5th percentile model attribution score of the given feature over the training data set.	Identifies features that are heavily influential for at least a substantial subset of samples within the data. More robust to outliers than the Range. Can also help find subsets within the data.
Frequency in the Top Three	Fraction of samples in the training data set for which the given feature was ranked in the top three in terms of absolute attribution scores.	Gives a sense of which features most commonly have heavy influence on individual sample predictions. Can also help to get an understanding without needing to understand the model attribution score.

To avoid over-indexing on a single dataset or model, we generated versions of these notebooks for both of the models described in Section 3.3—that is, the model trained on the Adult dataset and the model trained on the NHANES dataset. We assigned the model trained on the NHANES dataset to the three participants who most regularly work with medical data and would therefore likely be more comfortable with both the task and the features; we assigned the model trained on the Adult dataset to the remaining four participants. These assignments are listed in the rightmost column of Table 3.1.

All interviews were conducted virtually on a video conferencing platform due to the COVID-19 pandemic. During each interview, the participant and the interviewer viewed the notebooks

together, one at a time, via screen sharing. The participant had control of the screen to click, scroll, and explore. Participants were first asked to think aloud while they familiarized themselves with each notebook. They were then asked how they would go about accomplishing three of the tasks and objectives for which participants in our preliminary study had reported using interpretability tools. Specifically, we asked participants to describe 1) what they thought the model had learned overall, 2) how they would explain what the model had learned to someone who was not a machine learning developer, and 3) what their next steps would be if they were to go about debugging the model. After completing this sequence with the first notebook, and then completing it again with the additional information provided in the second notebook, participants were asked to share their likes and dislikes for each of the different global feature attributions, as well as their critical feedback, the value they gained from using the global feature attribution suite, and whether they would use a global feature attribution suite in their own workflows. The complete notebooks and the interview protocol can be found at https://github.com/aokeson/Aggregated-Explainability-Ranking-Alternatives.

Both audio and video from the interviews was recorded. Audio was transcribed by a third-party service, after which the audio transcripts were reviewed for accuracy and anonymized. The first author then annotated each transcript with information about the visualizations that the participant viewed at different points in time based on the corresponding video recording. The annotated transcripts were coded by the first author in three distinct passes: 1) coding differences in how participants answered our questions when viewing the first notebook compared with the second notebook, 2) coding potential uses mentioned by participants for the different global feature attributions, and finally 3) coding feedback (both positive and negative) on the global feature attribution suite. All authors then participated in a thematic analysis using the three types of codes.

3.4.2 *Results*

As we describe in this section, participants found the global feature attribution suite useful for communicating what the model had learned and identifying next steps for debugging the model. They also found that it increased their uncertainty in their understanding of the model (compared with viewing the usual global feature attributions alone) and helped them become more aware of the nuances of the model's behavior. However, they expressed concerns that the time it would take to compare and contrast different global feature attributions might affect the extent to which they would use a global feature attribution suite in their own workflows.

With our small sample size, we did not see clear differences between participants who were shown the model trained on the NHANES dataset and participants who were shown the model trained on the Adult datasets, so we do not attempt to make distinctions between the two.

Strategies for using different global feature attributions: Participants used the global feature attributions in a variety of different ways, exploring them individually as well as comparing and contrasting different global feature attributions.

Three participants (P5, P6, P7) checked for agreement between the different global feature attributions in order to pull out specific features that were influential across more than one of them. This gave them more confidence that these features were genuinely influential. For example, P5, who saw the model trained on the Adult dataset, had named age as being important to the model's predictions when they viewed the usual global feature attributions provided by SHAP in the first notebook. After seeing that age was also highly ranked according to the global feature attributions provided in the second notebook, they were more confident in their assessment of what the model had learned and in how to communicate what the model had learned to other stakeholders, stating *"I would feel rather confident that the clearest learning from the model is [...] around age."* P6

and P7 both independently described this process as trying to "flatten" the different global feature attributions back to a single list of the most influential features by extracting features that were highly ranked according to all of the global feature attributions. *"Maybe you want to start by listing the features that are sort of robustly important across an array of these different metrics."* –P6

One of the most common strategies for using the global feature attribution suite was to identify where the different global feature attributions disagreed and to explore the cause of this disagreement. Five of the seven participants (P1, P2, P6, P7, and P8) discussed using this strategy either to uncover new insights into the model's predictions or as a first step for debugging the model. P7 described going through each feature to check if it was consistently important, unimportant, or both across the different global feature attributions: *"Consistently important variables, great. Consistently not important variables, great. But variables where some trick like that could move you around a lot maybe is indicating something. Exactly what, I don't know. But that's why I would have to go explore."* As P6 described, *"It seems potentially very useful to come up with several different orderings of the features and then try to figure out why those orderings disagree in cases where they disagree. That seems like a very potentially fruitful way to find either interesting behavior or problems with your model."* P8, who saw the model trained on the Adult dataset, also used this strategy. When looking at the first notebook, P8 included capital gain in a list of influential features, because it was among the top three features according to the global feature attributions obtained using the status quo approach. However, while exploring the second notebook, P8 found that the different global feature attributions differed in their rankings of capital gain and capital loss and decided to explore this further. They were able to use this observation to jump start the debugging process by identifying outliers in the training dataset: *"I think that probably the [range] or [typical range] here helps to explain why capital gains appears on the*

[ranking by mean absolute value] but rather not in the [ranking by frequency in top three], probably because [capital gains] has very high variance and there are some outliers in the data, which drags this mean absolute value here. So, the outliers are the main cause that drag this capital gain to be the top three in [ranking by mean], rather than the [ranking by frequency in the top three]."

There was no general consensus among participants about which of the global feature attributions was most appropriate for each of the three tasks and objectives. In general, participants followed the brief guidance that we had provided in the notebooks about potential uses. For example, P2, who saw the model trained on the NHANES dataset, used the global feature attributions obtained by ranking features by their range to identify outliers in the training dataset, saying *"This range of the blood cell value, so I would want to verify that that's a realistic effect, that we're not just picking up individuals that have bad values for the white blood cell count."* In some cases, participants also came up with their own uses for the different global feature attributions, either deliberately or by chance. P2, for example, identified a potential bug in the NHANES dataset after examining the global feature attributions obtained by ranking features by their frequency in the top three and then deciding to dig more deeply into the diastolic blood pressure feature. *"For instance, there is a group of patients here with diastolic blood pressure less than 20. That hardly seems realistic. So this is a group of patients for whom either the value is missing or it was input wrong."* –P2

Increased uncertainty about the model's behavior: Our hope was that providing machine learning developers with different global feature attributions to compare and contrast would lessen their confidence in the overly simplistic global feature attributions usually provided by SHAP and instead enable them to obtain a more nuanced global view of model behavior. When interacting

with the first notebook, most participants focused their descriptions of what the model had learned on a few features that were highly ranked according to the usual global feature attributions provided by SHAP. As a result, participants tended to focus their exploration of the model on a few (typically three to five) features. However, when exploring the second notebook, participants began to doubt the simple answers they had given previously. For example, P7 questioned their initial interpretation of what the model had learned, saying *"Now I'm a little hesitant, because I'm not sure. I guess there's now four plots, and they are kind of equivalent. [...] So now I'm a little confused. I'm not sure which one to trust and to use to answer this question."* Participants also commented that their confidence had changed: *"I think it's just sort of broadened my confidence intervals on how important each feature is."* –P6. As desired, participants felt that the global feature attribution suite provided a more nuanced global view of the model's behavior than the global feature attributions obtained using the status-quo approach: *"I mean, it takes you from [...] a scalar importance to a distribution of importance. It really helps you get that new understanding of how the importance of a feature can change over the different samples and the mean will not tell you that."* –P2 Lastly, participants noted that some of the information available in the second notebook could be inferred from other visualizations, such as SHAP's beeswarm plots, but that the new plots made it easier to digest and interpret the information: *"I mean, that's similar information for what's in this summary plot, but it's condensed in a way that it's much easier to read."* –P2 Indeed, although the distribution of attribution scores for each feature was available in other plots, this information was not salient enough to mitigate participant overconfidence.

Required time investment and constraints: The most common challenge raised by participants was that it might be too time consuming to compare and contrast different global feature attributions. P7 articulated a tension between the pressures of real-world time constraints

and the benefits of rigorously examining multiple global feature attributions: *"And if you are really strapped for time, which in the industry you frequently are, then it might be easy to just not explore these other things. [...] It makes me think that, going forward, I should be a little more vigilant about this stuff, but, honestly, it really depends on time."* Participants were concerned about whether a global feature attribution suite would help them accomplish their tasks and objectives more quickly or instead be yet another time sink. Given that participants generally used the mean global feature attribution plot because it was quick and easy, presenting a solution that is no longer as quick and easy may drive users to once again only use the mean plot over the metrics suite. Participants may have been overly pessimistic about the time it would take to compare and contrast different global feature attributions because they were seeing them for the first time. However, before implementing a global feature attribution suite in common interpretability tools, more research is needed to understand how to present different global feature attributions in the most efficient way possible.

3.5 Discussion

We presented an artifact-based interview study intended to investigate whether machine learning developers would benefit from being able to compare and contrast different global feature attributions. This study extends a recent line of research exploring human-centered approaches to interpretability and, in particular, how stakeholders use and understand interpretability tools [1,3,14,15,42,43,51,58,63,83,104,109]; however, our focus is on an aspect of interpretability tools that has been overlooked to date—namely, the summary statistics used to generate global feature attributions. Participants were first shown the usual global feature attributions provided by SHAP and asked some questions about the underlying model. They were then shown a suite of global feature attributions obtained by ranking features by four different summary statistics of their

attribution scores—what we refer to as a global feature attribution suite—and asked to reconsider their answers. Our hope was that providing machine learning developers with different global feature attributions to compare and contrast would lessen their confidence in the global feature attributions usually provided by SHAP, which we found to be overly simplistic, and instead enable them to obtain a more nuanced global view of model behavior.

We found that participants were able to use the global feature attribution suite to communicate what the model had learned and to identify next steps for debugging the model. As desired, we also found that viewing the global feature attribution suite increased their uncertainty in their understanding of the underlying model as they became more aware of the intricacies of the model's behavior. However, they also expressed a tension between the benefits obtained by using tools like SHAP to quickly get a sense of what a model has learned and the time it would take to compare and contrast different global feature attributions, noting that this might affect the extent to which they would use a global feature attribution suite in their own workflows. Of course, participants were seeing the global feature attributions for the first time and they only used the global feature attribution suite for less than an hour. It is possible that with adequate training and practice, this tension would be reduced or even overcome. More generally, though, this finding echoes observations from prior work about the need to balance the benefits of quick understanding and slower, nuanced understanding when designing interpretability tools [51].

Like any study, ours has limitations. In addition to the short timescale over which it was conducted, we only recruited seven participants. We wanted to be able to conduct an in-depth interview with each participant about their experiences using the global feature attribution suite, but this necessarily limits the type of conclusions that we are able to draw. Furthermore, we focused only on experienced users of interpretability tools, which further limits the extent to which

we can generalize to the broader machine learning developer community. We also limited our scope to models trained on two datasets, so more research is needed to investigate whether our findings would change if different datasets were used—for example, datasets with orders of magnitude more features.

3.6 Conclusion

We see this work as a first step toward designing interpretability tools that explicitly highlight the nuanced behavior of models, as advocated for by Kaur et al. [51]. Future work should explore ways for machine learning developers to use a global feature attribution suite to quickly get a sense of what a model has learned without placing undue confidence in the corresponding global feature attributions. This will require carefully balancing the cognitive burden involved in understanding the global feature attributions with the amount of information that they can convey. It will also require investigation into which summary statistics to use and which other information to incorporate. One could imagine, for example, additionally including other notions of global feature importance, such as those obtained by applying the concept of Shapley values directly to global quantities like the variance explained [80] and the loss [25] rather than summarizing (local) feature attribution scores. Doing this well will also require research into how to present different global feature attributions in the most efficient way possible.

3.7 Summary

In this project, we explored how adapting a machine learning explanation system (SHAP) can support machine learning model developers and data scientists in leveraging the system. In order to achieve this, we employed three key strategies: (1) We aligned with user goals by identifying how the users were employing the existing explanation system and adapted that system to better

support debugging, understanding, and explaining tasks. (2) We retained nuance in explanations by providing more aggregations so there was less data artifact loss, while still providing a high level and quick look into the data. (3) We imparted an appropriate level of trust by providing multiple visualizations where there had previously been one, to avoid over trusting a single, simplistic, understanding of the explanations and increasing the user's confidence in their more nuanced understandings and explanations of the explanations.

Chapter 4. PREDICTING BLOOD GLUCOSE TEST ACCURACY IN ICU PATIENTS

In this work we explored the potential for predicting whether or not a point-of-care blood glucose test will be sufficiently accurate in critically ill hospital patients. We aimed to do this while aligning with the clinician goal of finding clinically meaningful indicators for test inaccuracy. This work was done in collaboration with Hannah Burkhardt, Brent Wisse, Tim Althoff, and James Fogarty. Hannah and I both contributed to the writing and editing of the manuscript and the statistical analyses. I additionally contributed the initial data cleaning and the machine learning experiments. It is currently being prepared for a submission to a medical journal in mid-June.

4.1 INTRODUCTION

Due to cost, sample size, and convenience [24,62], millions of POC BG tests are performed annually in US Intensive Care Units (ICUs), most with glucose meters that are not FDA approved for use in critically ill patients. Recently, the FDA (via CLIA and CMS) considered stricter guidelines to govern the use of POC BG meters for ICU patients, but eventually delayed that decision. The concern related to POC BG in ICU patients is not arbitrary as POC BG are less accurate in critically ill patients due to a variety of common clinical factors [46,111]. Importantly, hyperglycemia is common in critically ill patients, and as many as 25% require insulin therapy, a medication commonly associated with adverse events in hospitalized patients [38], increasing the need for accurate and consistent BG monitoring. This tension between accuracy and convenience means that the use of point-of-care (POC) blood glucose (BG) meters in critically ill patients remains controversial [33].

Critical illness is often associated with impaired regulation of glucose balance. Comorbidities such as diabetes mellitus and obesity can cause hyperglycemia in the setting of any critical illness. Additionally, the metabolic stress of surgery, sepsis, or trauma contributes to hyperglycemia in patients without diabetes mellitus. Common treatments, including vasopressors and antibiotics, also contribute to hyperglycemia. Conversely, critical conditions causing severe liver and kidney failure can contribute to hypoglycemia. Consequently, many critically ill patients require frequent BG monitoring as part of their care. Causes of inaccurate POC BG values include precision of the instrument/test strip, sampling errors, and patient factors. Some of the variables contributing to POC BG error have been more thoroughly studied than others, but often using different brands of meters, sometimes with conflicting findings [46].

BG can be measured from various sources and by different devices each with advantages and disadvantages [54,59]. A meta-analysis of studies in critically ill patients concluded that current BG monitoring technology has not reached a sufficient degree of accuracy and reliability to lead to appropriate glucose control in critically ill patients [46]. This is unlikely to change with the current trend moving towards continuous glucose monitoring devices that sample interstitial fluid glucose.

Currently, few hospitals have a viable alternative to performing POC BG tests. Data suggest that the vast majority of BG measurements in ICU patients are performed using a POC glucose meter and capillary blood [24]. This is allowable only because laboratory medicine adopts an extremely narrow definition of “critically ill”, which is currently permitted by the FDA regulations.

Therefore, before the results of POC testing can be trusted in ICU settings, further study of device performance in critically ill patients is needed. Prior research has studied the usage and accuracy of POC BG testing devices in general hospital settings (e.g., [24,54,86,91]), but few have

analyzed how these devices are used in ICUs. Of these studies, attempts to understand the accuracy of these tests have been limited to analysis of relatively small datasets of several hundred paired glucose measurements (i.e., results from POC BG testing and laboratory analysis as ground truth) [33,82]. In addition, these past studies have not explored whether POC BG testing inaccuracy in ICUs can be predicted.

In light of prior reports suggesting that certain patient factors may cause systematic errors in these readings, we hypothesized that patients can be divided into two groups based on their individual and clinical characteristics: a group for which POC BG is acceptable, and a group for which the superior laboratory BG test is indispensable. Accordingly, our institution initiated a quality improvement (QI) project to evaluate the accuracy of POC BG values in ICU patients. For this project, registered nurses (RNs) performed a daily determination of capillary (CAP) and arterial/venous (AV) POC BG relative to venous laboratory glucose values in specific critically ill patients to help guide clinical decision making.

4.2 METHODS

4.2.1 *QI Project and Data Collection*

This retrospective QI project was conducted at two ~400-bed urban tertiary care centers that are part of a regional academic medical center in the Pacific Northwest. The project institution exclusively uses the Accu-Chek Inform II glucose meter (Roche Diagnostics, Indianapolis, Indiana). Inform meter data are automatically downloaded through Remote Automated Laboratory System-Plus (Medical Automation Systems, Charlottesville, Virginia), administered by laboratory medicine, which then uploads (POC) BG values into the electronic health record (EHR). BG was measured at two different body sites: arterial/venous and capillary (finger) indicated by the bedside Registered Nurse (RN). Expected accuracy for this BG meter is ± 12.5% for BG values >100mg/dL

based on manufacturer data. This corresponds closely with previous data using this BG meter which showed that the Clark Error grid was optimized at a POC BG divergence of ≤12% [23,77]. The Roche meter is optimized to perform despite multiple interfering substances and uses AC impedance to correct for decreased red cell volume within a wide range of hematocrit (10-65%).

A QI protocol was implemented measuring venous BG (via laboratory test) as well as capillary and arterial/venous POC once every 24 hours in ICU patients receiving insulin therapy and not on an oral diet. Based on the difference between the POC values and the "ground truth" laboratory measurement, patients were considered to have acceptable (≤12% difference) or unacceptable (>12% difference) POC measurements for that 24h-period. With IT support, an EHR app was built that displayed this patient status. If no coordinated testing was done in the past 24h, this information would be displayed along with a reminder to perform the testing. Results of the testing were displayed within the EHR for any provider to see. Teams were encouraged to take the results of POC BG testing into account and make decisions related to insulin management and glycemic control on an individual basis.

The QI project yielded a dataset of arterial/venous and capillary POC BG readings and associated "gold standard" laboratory readings. Each reading was further associated with patient age, weight, and body mass index (BMI), as well as measurements of blood urea nitrogen (BUN), creatinine, hematocrit, pH, albumin, triglycerides, body temperature, mean arterial pressure, and oxygen saturation (SpO2). Additional variables collected included the patient's race (white vs. non-white), service location and unit; type of insulin therapy (subcutaneous, IV); diet order (enteral, NPO, or parenteral); oxygen delivery mode (oxygen supplementation vs. not); whether the patient was being dialyzed (yes/no); whether or not the patient was on any vasopressors (yes/no); whether or not the patient had been given acetaminophen (yes/no); and whether or not

the patient had been given ascorbic acid supplementation (yes/no). A small number of triplicate BG validations were performed by RNs on ICU patients already receiving oral diets or not receiving insulin and these BG values were included in the analysis. The selection of evaluated variables was largely informed by extant research and literature [33,46,49,82,111].

4.2.2 *Statistical Analysis*

Arterial/venous and capillary measurements were analyzed separately.

Association: We first sought to validate prior reports of patient factors affecting POC BG accuracy. For this purpose, we conducted statistical analyses intended to capture associations between the continuous error magnitude, measured as the percentage difference between the POC BG reading and the "ground truth" laboratory measurement. Prior work has shown that patient factors such as hematocrit levels, vasopressor medications, and mean arterial pressure are associated with POC BG accuracy. We investigated if the reported effects are apparent in our dataset using multiple linear regression, with a significance level of 0.05, using only the first measurement for each patient to avoid data dependency issues. p-values were obtained using ANOVA, such that significance indicates that the variables are statistically significantly associated with the outcome when all other predictors are controlled for. For regression, variables with more than 20% missing values were excluded from the regression, and missing values in variables with few missing values were imputed using the sample mean. Two regressions were run, one for the AV error, and one for the CAP error.

Prediction: Additionally, we aimed to assess whether the acceptability of BG measurement using POC devices is predictable from patient factors. Here, we are interested in the binary determination of whether POC BG readings are acceptable as a basis to inform care decisions. An inherent level of inaccuracy or uncertainty is associated with almost all evidence that is used in

clinical practice. However, whether a given metric is "good enough" (defined in the hospital as ≤12% error), in other words, if the expected amount of error is acceptable, is the ultimate determinant of whether the metric can be used. We thus binarized the POC BG error into acceptable (≤12%) and unacceptable (>12%) error and for this analysis.

Our main objective was to determine which patient features have predictive power for the magnitude of error exhibited by POC BG meters. To this end, we employed machine learning models to assess the utility of each feature to predict the extent of error. We used gradient boosted decision trees, logistic regression models, and single decision trees with the binary prediction task of whether or not a sample was within the acceptable 12% accuracy range. We trained the models using 5-fold cross validation (CV). Because some of our samples come from the same patient over the duration of their ICU stay, we created the 5 CV splits and the test set such that all samples from a single patient were in only one of the CV folds to ensure independence of samples between each CV split. We also standardized all continuous features to have a mean of 0 and a standard deviation of 1. The trained decision tree model was used to both find important features and to find important groups of samples and features based on successive splits in the tree.

4.3 RESULTS

4.3.1 *Association*

Data from 1,608 patients were included in the project. All patients had at least one paired AV measurement, and 1,123 patients had at least one paired CAP measurement. Table 4.1 shows patient and measurement characteristics for the two subsets of patients, using the first measurement pair for each patient only to avoid dependency issues. Serum triglyceride level was the only variable missing frequently enough to warrant excluding from multiple linear regression.

Remaining predictors included age, sex, race, weight, BMI, hospital type, unit, service, diet, acetaminophen treatment, ascorbic acid supplementation, vasopressor treatment, O2 supplementation, dialysis treatment, IV vs SC insulin treatment, albumin levels, blood urea nitrogen levels, creatinine levels, hematocrit levels, mean arterial pressure, blood pH, oxygen saturation, and body temperature. Results showed that POC error rates (both AV and CAP) were significantly ($p<0.0005$) different between groups with different hematocrit levels when all other predictors were controlled for. CAP error rates also differed ($p<0.05$) based on BMI, hospital, ascorbic acid supplementation, and dialysis treatment. AV error rates differed ($p<0.05$) based on the patient's service classification and ascorbic acid supplementation.

10,225 paired AV measurements and 6,439 paired CAP measurements were collected in total. On average, there were 6.4 (SD 7.4) AV measurement pairs per patient and 5.7 (SD 6.5) CAP measurement pairs per patient. Across all of these measurements, the average error of AV measurements was 6.1% (SD 9.1%) and CAP measurements were on average off by 8.5% (SD 19.4%). 89.8% of AV readings and 80.0% of CAP readings were clinically acceptable (<12% different from laboratory BG value). AV measurements had a Pearson correlation coefficient with laboratory values of 0.969 and CAP measurements had a correlation with laboratory values of 0.904.

Table 4.1. Description of cohort. All 1,608 patients had AV measurements at baseline, but only 1,123 also had CAP measurements at baseline. Mean (SD) is shown for continuous variables and counts for categorical variables. p-Values were calculated via multiple linear regression and indicate if the variable was significantly associated with the % error in the respective sample (CAP or AV) when controlling for all other predictors. (* $p<0.05$, ** $p<0.005$, *** $p<0.0005$ for statistical tests)

		CAP (N=1123)			AV (N=1608)		
			p	missing		p	missing
Demographics							
Age		57.9 (17.0)		0	57.1 (17.3)		0
Sex: male	False	403.0 (35.9%)		0	601.0 (37.4%)		0
	True	720.0 (64.1%)			1007.0 (62.6%)		
Race: white	False	326.0 (29.0%)		0	457.0 (28.9%)		24 (1.5%)
	True	797.0 (71.0%)			1127.0 (71.1%)		
Weight		86.7 (30.1)		0	86.2 (29.5)		3 (0.2%)
BMI		29.3 (17.0)	*	1 (0.1%)	29.0 (15.0)		37 (2.3%)
Circumstances							
Hospital	Hospital B	134.0 (11.9%)	*	0	370.0 (23.0%)		0
	Hospital A	989.0 (88.1%)			1238.0 (77.0%)		
Unit	Burns/Plastics ICU	88.0 (7.8%)		0	123.0 (7.6%)		0
	Medical Coronary ICU	395.0 (35.2%)			487.0 (30.3%)		
	Medical Oncology ICU	112.0 (10.0%)			268.0 (16.7%)		
	Neuro ICU	286.0 (25.5%)			328.0 (20.4%)		
	ICU Stepdown	22.0 (2.0%)			102.0 (6.3%)		
	Trauma Surgical ICU	220.0 (19.6%)			300.0 (18.7%)		
Service	Medicine	480.0 (42.7%)		0	685.0 (42.6%)	*	0
	Neuro	273.0 (24.3%)			317.0 (19.7%)		
	Oncology	41.0 (3.7%)			89.0 (5.5%)		
	Other	29.0 (2.6%)			48.0 (3.0%)		
	Surgery	300.0 (26.7%)			469.0 (29.2%)		

Treatment							
Diet	Enteral	360.0 (32.1%)		0	501.0 (31.2%)		0
	NPO	407.0 (36.2%)			552.0 (34.3%)		
	Other	356.0 (31.7%)			555.0 (34.5%)		
Acetaminophen Rx	False	641.0 (57.1%)		0	908.0 (56.5%)		0
	True	482.0 (42.9%)			700.0 (43.5%)		
Ascorbic Acid Rx	False	1006.0 (89.6%)	*	0	1443.0 (89.7%)	*	0
	True	117.0 (10.4%)			165.0 (10.3%)		
Vasopressor Rx	False	767.0 (68.3%)		0	1070.0 (66.5%)		0
	True	356.0 (31.7%)			538.0 (33.5%)		
O2 Supplementation	False	197.0 (17.5%)		0	317.0 (19.7%)		0
	True	926.0 (82.5%)			1291.0 (80.3%)		
Dialysis	False	1031.0 (91.8%)	*	0	1450.0 (90.2%)		0
	True	92.0 (8.2%)			158.0 (9.8%)		
Insulin	IV	419.0 (37.3%)		0	609.0 (37.9%)		0
	None	236.0 (21.0%)			373.0 (23.2%)		
	Subcutaneous	468.0 (41.7%)			626.0 (38.9%)		
Vitals & Lab Values							
Albumin		3.0 (0.8)		17 (1.5%)	2.9 (0.8)		146 (9.1%)
Blood Urea Nitrogen		32.2 (26.5)		0	32.4 (26.3)		0
Creatinine		1.7 (1.9)		0	1.8 (2.0)		2 (0.1%)
Hematocrit		31.9 (7.2)	***	0	31.6 (7.1)	***	0
Mean Arterial Pressure		84.5 (18.7)		0	83.3 (18.8)		1 (0.1%)
pH		7.4 (0.1)		9 (0.8%)	7.4 (0.1)		104 (6.5%)
SpO2		97.5 (3.3)		0	97.3 (3.6)		1 (0.1%)
Temperature		36.7 (1.0)		0	36.7 (1.0)		1 (0.1%)
Triglycerides		209.4 (189.5)		610 (54.3%)	208.3 (189.8)		1030 (64.1%)
Lab BG Value		186 (85.9)		0	170.6 (82.0)		0
POC BG Values							
Arterial BG Value		182.1 (82.2)		0	167.0 (78.4)		0
Arterial % Error ≥ 12%	False	995.0 (88.6%)		0	1420.0 (88.3%)		0
	True	128.0 (11.4%)			188.0 (11.7%)		
Arterial % Error		6.3 (7.3)		0	6.5 (8.4)		0
Capillary BG Value		177.5 (82.4)		0	177.5 (82.4)		485 (30.2%)
Capillary % Error ≥ 12%	False	840.0 (74.8%)		0	840.0 (52.2%)		0
	True	283.0 (25.2%)			768.0 (47.8%)		
Capillary % Error		9.5 (13.2)		0	9.5 (13.2)		485 (30.2%)

4.3.2 *Prediction*

Multiple machine learning models were also trained on all of the data to determine if any combination of variables would allow for improved prediction of the percent error of a POC BG being acceptable or unacceptable. Receiver operating characteristic (ROC) curves were generated to evaluate the results. None of the models allowed for meaningful improvement in predicting the likelihood of a POC BG value being acceptable or unacceptable compared to chance (Figure 4.1).

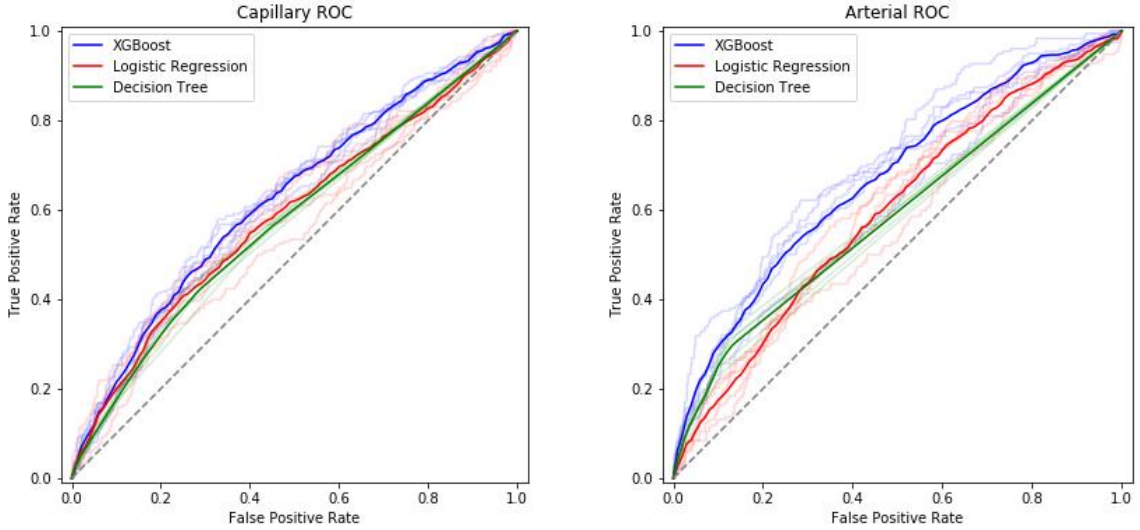

Figure 4.1. ROC curve with each CV (shown in opaque lines) and the average of all of a model's CVs (shown in dark lines) for capillary and arterial/venous POC tests. The grey dashed line indicates the performance of making predictions by random guessing.

Evaluating optimized decision trees based on machine learning models: The optimized decision trees representing models for predicting acceptable POC BG values are shown in Figure 4.2. Model performance is shown as the green lines in Figure 4.1. In each of the final decision trees, none of the leaf nodes (bottom layer of the tree) is sufficiently large or pure, i.e. having a

significantly higher or lower accuracy than chance, to find a subpopulation of samples that might be candidates for further evaluation as a predictor of tests being acceptable or unacceptable.

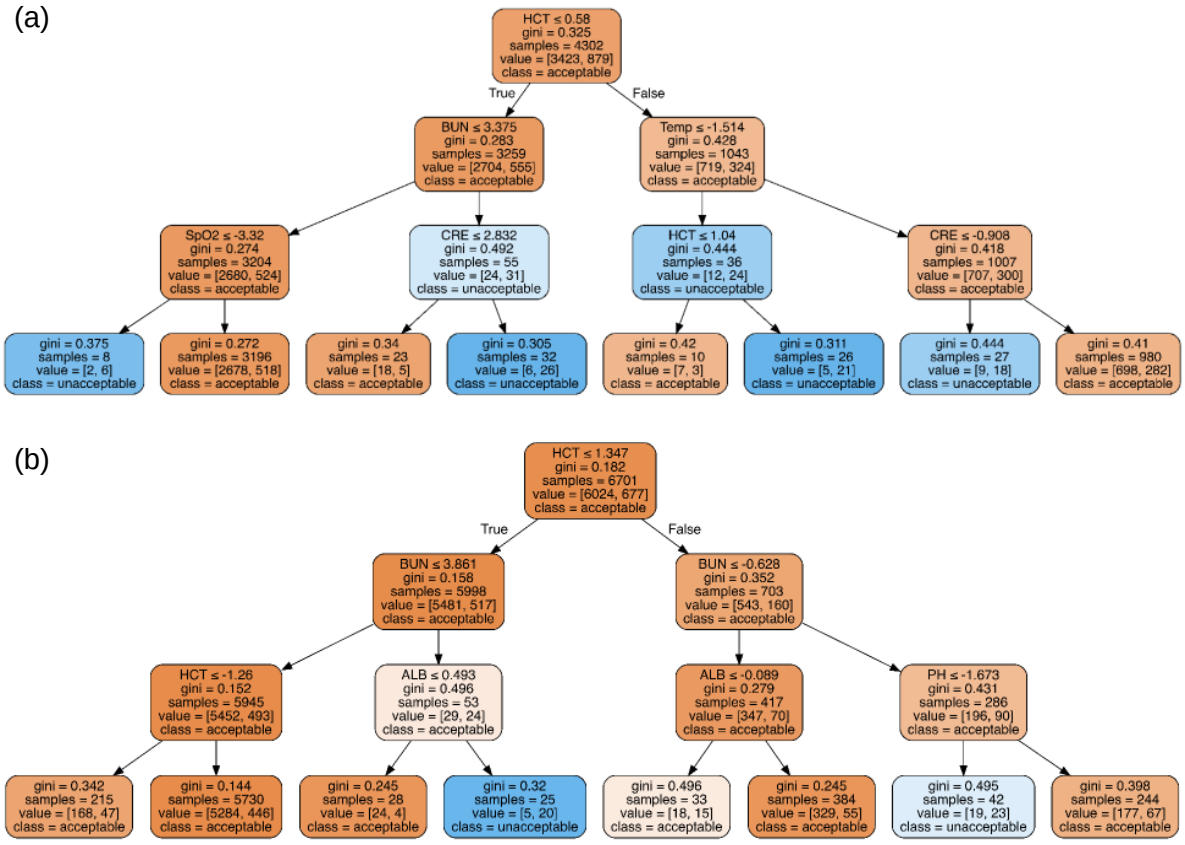

Figure 4.2. Final decision trees for the (a) CAP POC and (b) AV POC tests. Split points reflect the normalized feature values.

4.3.3 *Ad-hoc Hematocrit Analysis*

Despite glucose meter AC impedance correction, the most significant association of POC BG percentage error was found for hematocrit levels. Hematocrit levels were also found to be one of the top features indicative of acceptable versus unacceptable results in all of the machine learning models (shown for decision trees in Figure 4.2). Therefore, we further investigated the distribution of the percent difference between laboratory and CAP/AV BG values across different values of hematocrit (Figure 4.3), split by sex, using sex specific normal ranges (40-50% for men, 35-43%

for women). Horizontal lines show Loess-smoother lines representing a locally weighted average of the POC percent difference from LAB BG along the range of hematocrit values. The vertical black lines indicate the value of hematocrit where the POC BG error is smallest, on average. This value is not within the normal range for hematocrit, suggesting that the glucose meter's built-in hematocrit correction may not be correctly calibrated for ICU patients. This effect is most pronounced for capillary POC measurements in male patients.

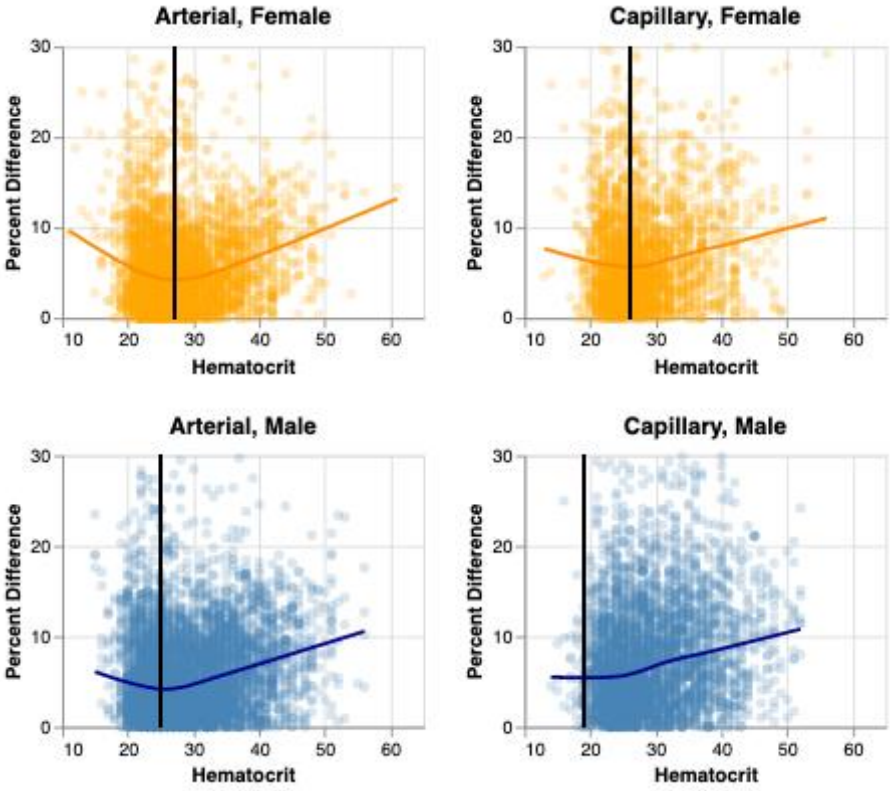

Figure 4.3. Hematocrit vs. Percent Difference between POC measurement and gold standard measurement. Minimum mean error occurred at black lines.

4.4 Discussion

This project evaluated the validity of point-of-care (POC) blood glucose (BG) values in critically ill patients. We analyzed over 16,000 individual POC BG and matched laboratory venous BG measurements to determine which patient factors correlate with differences between POC BG and laboratory venous BG in critically ill patients. We determined the frequency of acceptable

(<12%) and unacceptable (≥12%) difference between POC BG and laboratory venous BG values, aiming to identify variables that influenced POC BG validity. We were unable to identify any individual, or combination of, patient factors that systematically predicted the likelihood of a POC BG measurement having either a clinically acceptable or unacceptable difference from the laboratory venous BG.

Data from this large project demonstrate that in critically ill patients POC BG values differ from the laboratory BG frequently enough to be a concern. The rates were different depending on the sample source: 10.2% of POC measurements taken from arterial or venous blood and 20.0% of measurements from capillary blood differed from the laboratory venous BG by ≥12%. The higher validity rate of > 90% for arterial POC BG measurements indicates that factors other than merely the blood glucose meter and strips are responsible. This could include errors in obtaining the sample (dilution with interstitial fluid, cleaning the skin, etc.) or patient factors related to capillary blood (low MAP, peripheral edema, vasopressors, etc.)

Consistent with prior work, we found that several clinical variables associated with greater POC BG difference from laboratory venous BG: hematocrit levels, BMI, ascorbic acid supplementation, requirement for hemodialysis, and hospital type. Other variables including patient factors, demographic data, and administered medications demonstrated only weak relationships with the rate of POC BG values being unacceptable. Previous studies have shown decreased accuracy of POC BG values as patient hematocrit decreases. Data from this project demonstrates a different trend. We found a U-shaped distribution of error magnitude with respect to hematocrit levels. A hematocrit range of 25-27% showed the lowest POC BG difference from laboratory venous BG, and the difference between the values increased for hematocrit levels both higher and lower than this range. Our finding is most likely explained by the fact that the common

blood glucose meter used attempts to correct for hematocrit levels, using an impedance-based calculation. Our data suggests that the blood glucose meter does not fully correct for extremely low hematocrit readings while over-correcting for hematocrit readings in the normal range.

Despite apparent associations of patient factors with error rates, our statistical analysis, which included current machine learning strategies, was unable to determine any combination of evaluated variables (demographic, clinical, administered medications) that was meaningfully predictive of the acceptability of POC BG values in ICU patients. This may be the case for a number of reasons, including the strength of association or because of insufficient effect sizes. This finding has important implications. Previous work as well as package inserts report evidence for interference of a number of patient factors and blood constituents; yet, none of the previous studies evaluated the potential application of the knowledge of these interferences to selecting candidates for POC BG testing. While further work is needed to confirm factors influencing the magnitude of error, our results provide evidence that error may be largely unpredictable in practice.

The strength of this project lay in the very large number of duplicate and triplicate BG values all performed within 5 minutes, and in the careful documentation of twenty-seven important continuous or categorical variables in this ICU patient population that would be likely to cause POC BG error. However, this work also had limitations. First, we evaluated ICU patients only and thus our results have limited generalizability to other patient populations. Second, all POC measurements were performed using the Accu-Chek Inform II point-of-care blood glucose measurement device; thus, our results may not be generalizable to other POC BG devices. Importantly, some devices (including the Accu-Chek Inform II) perform corrections in real-time, potentially muddling what influences are possible to detect, and changing the patient subpopulations for whom results are or are not reliable. Third, the set of variables collected as part

of this project was informed by prior work. Accordingly, there may be specific factors in critically ill patients that predict POC BG validity but that were not included in this project.

4.4.1 *Future Work*

The results of our analysis are largely consistent with prior work showing associations of patient factors with POC BG value acceptability. However, our large project found only weak associations and was not able to demonstrate all previously reported interferences. Thus, future work might explore the specific circumstances under which certain associations are consistently reproducible.

Further, prior work has not explored how knowledge of interferences can be applied to use POC BG testing on selected patient subpopulations. This work was not able to provide strong evidence that such an application is feasible, but further studies are needed. The successful implementation of predictive modeling and clinical decision support involves many considerations beyond classification performance. Even a model with low AUROC may be able to bring clinical value for a small subset of patients where predictions can be made with confidence, even if this is not true for the overall patient population. Operationalizing predictive models in clinical care involves selecting a decision threshold that balances specificity and sensitivity favorably for a particular clinical goal. For example, when flagging patients as exhibiting unreliable POC blood glucose measurements, it may be desirable to maximize sensitivity, even if specificity is considerably compromised as a result. In other words, to minimize the number of patients who will not receive necessary laboratory tests, we may accept that many patients who receive a laboratory test may actually be able to do without it. Currently, 100% of patients have to undergo laboratory tests in addition to POC BG tests, corresponding to 0 specificity. The impact that can be made on the cost of care even by a small reduction of 5-10% may be meaningful: clinicians

may be able to forego the expensive and slow laboratory blood glucose test, reducing delays and cost of care. While we did not find any factors strongly predictive of POC validity, we were able to achieve classification performance only slightly beyond that of the baseline classifier (random guessing); that is, our classifier is able to make accurate predictions for some subset of patients, though not the majority. While we made initial attempts to find subsets of patients for whom accurate predictions of POC BG test accuracy or inaccuracy can be made, we were unsuccessful at finding a sufficiently large subset such that we could make statistically significant claims about any of those subsets. Future work may thus aim to further explore how such a classifier can be useful under considerations specific to the clinical use case.

4.5 CONCLUSION

This project demonstrated that an unacceptable (≥ 12%) difference from laboratory venous BG occurs with up to 20% of POC BG measurements in ICU patients. Consistent with prior work, this project identified specific patient factors associated with POC BG error. We additionally found that BG meter strategies to correct for a common factor associated with POC BG error (low hematocrit) can also introduce unexpected POC BG error. However, we were not able to demonstrate that POC BG error is systematically predictable from patient factors previously reported to interfere with POC testing, thus we did not identify any groups of ICU patients for which POC BG either should or should not be used. Thus, the variables we evaluated are not suitable for evidence-based decision support to determine safe POC BG usage in ICU patients. Nonetheless, there may be avenues for future decision support projects to determine appropriate POC BG use in specific ICU patients.

Unfortunately, this work does not change the current status quo. RNs will continue to perform millions of POC BG tests for ICU patients. Clinical decisions related to insulin administration and

hypoglycemia treatment will continue to be made based on POC BG values. Hospitals currently have few reasonable alternatives, laboratory venous BG tests for all BG measurements are not timely and are cost prohibitive. Ongoing studies using continuous glucose monitoring from interstitial fluid offers the promise to improve this dilemma, but it is still unclear if this strategy will decrease the rate of BG error in critically ill patients as many of the same patient factors may be making evaluation of BG in interstitial fluid less accurate.

In light of our results, POC BG testing, while inexpensive and rapid, should continue to be used in conjunction with laboratory BG testing based on clinical judgment. Improved methods and accuracy of POC BG testing in critically ill patients is likely to be a continuing source of debate and innovation in the years ahead.

4.6 SUMMARY

In this project, we explored how selecting machine learning prediction and explanation systems can support clinicians in leveraging the systems. In order to achieve this, we employed two key strategies: (1) We aligned with user goals by utilizing many different machine learning prediction and explanation techniques to show that these systems might not meet user accuracy and application needs. (2) We imparted an appropriate level of trust by presenting results alongside baselines for random guessing, supporting statistical analyses that were more familiar to the user group, and presenting visualizations to show why it seems there is not a medically meaningful subgroup to make accurate predictions on.

Chapter 5. PREDICTING AND EXPLAINING AN IMMINENT DEMENTIA DIAGNOSIS WITH LIMITED DATA

In this work, we explored different machine learning systems to accurately predict an imminent dementia diagnosis while fulfilling the clinician goal of using data that is less onerous and time consuming to collect than current approaches. We also explored how to do this while retaining nuance in the explanations so that known biological phenomenon are retained and clinicians can potentially use the explanations for care decisions. This work was completed with Nicasia Beebe-Wang, Tim Althoff, and Su-In Lee and was published in the IEEE Journal of Biomedical and Health Informatics Special Issue on Explainable AI for Clinical and Population Health Informatics [9]. Nicasia Beebe-Wang and I collaborated together on all aspects of this project, with input from senior authors Tim Althoff and Su-In Lee.

5.1 INTRODUCTION

Alzheimer's disease (AD), a degenerative brain condition, affects an estimated 5.8 million Americans. As the world's older population grows at an unprecedented rate, the number of individuals with dementia is projected to more than double, making it an increasingly pressing health concern [110]. Significant advances in diagnostic predictions are essential to curb the devastating effects of dementia worldwide. We believe these advances will be enabled by large-scale aging cohort studies and machine learning innovations.

Although no currently known treatment can cure or slow AD progression, identifying AD cases before severe neurological damage ensues is crucial. Predicting onset can promote treatment efficacy once successful interventions are developed and swiftly identify individuals who may benefit from drug trials. It will also help families plan for patient care and help patients to receive

resources to help make personal decisions about their care before they lose the autonomy to do so [84].

Although studies have demonstrated the possibility of identifying individuals who already have dementia [32], such diagnoses occur beyond the critical window for effective interventions or end-of-life planning [84]. Other studies have predicted the onset of dementia in advance of a clinical diagnosis, but often involve costly data collection using neuroimaging or in-depth neuropsychological batteries over multiple years [7,26,47,60,98,99]. The use of repeated cognitive testing may help to model and predict an individual's cognitive decline [60]; however, given that only 16% of American seniors receive regular cognitive assessments in primary care settings [34], this approach may be impractical for the general population. Furthermore, decreasing the required window of repeated testing would enable earlier diagnostic predictions because predictions would be made using fewer (and therefore earlier) observations.

Our goal was to find a balance between accurate but costly tests and efficient but relatively inaccurate predictions. In particular, we assessed and explained an individual's risk for dementia multiple years into the future using relatively easy-to-collect measures (see examples in Table 5.3) that may scale well to large aging populations. To this end, we addressed the following three research questions (RQs), encapsulated in Figure 5.1 and linked to in-text discussion:

- RQ1: Using longitudinal clinical and cognitive data from an aging cohort study, can we effectively predict whether an individual will develop dementia?
- RQ2: To what extent can we reduce the need for burdensome data collection while still maintaining predictive performance? We explore this question with respect to both repeated cognitive testing over multiple years and the number of required tests.

- RQ3: Using complex models that learn interactions among features and risks, can we leverage interpretability methods to provide personalized dementia risk explanations?

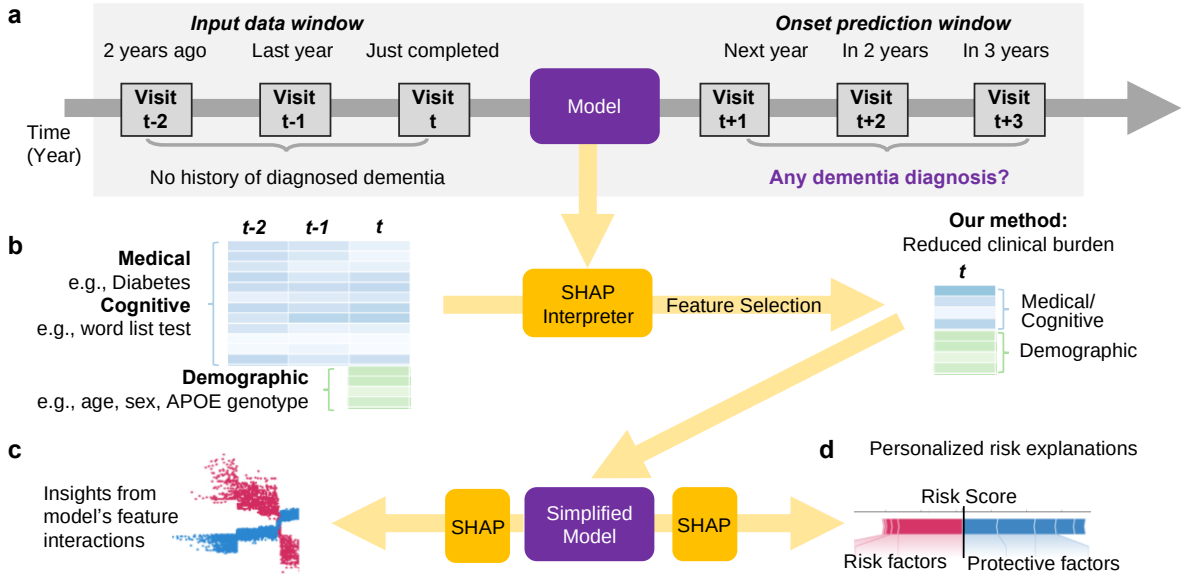

Figure 5.1. Overview of our approach to producing efficient and explainable dementia onset risk predictions. We link figure components to research questions (RQs) and in-text discussion. (a) RQ1: Sections 5.3.1 and 5.3.2. (b) RQ2: Sections 5.3.3 and 5.4.4. (c) RQ3: Section 5.4.5. [9]

Our approach made several noteworthy contributions. First, by exploring multiple classes of machine learning models, we find that dementia onset (within three years) can be predicted robustly and requires cognitive measurements from only a single session. Second, by using an interpretability method to measure sample-level feature importance, we identified a small subset of tests that provide similar predictive value to a standard dementia diagnosis battery, while only taking one fifth of the time to administer. Third, each dementia risk prediction estimate is accompanied by individual explanations of risk, which may aid clinicians in tailoring care to their patients.

5.2 Related Work

5.2.1 *State-of-the-Art Dementia Diagnosis*

Many studies have sought to predict the presence of dementia based on brain scans and other metrics. For example, deep learning has improved AD classification using both magnetic resonance imaging and positron emission tomography scans [26,98,99]. Adding lifestyle and cognitive factors has additionally improved prediction performance for AD onset [7]. Although these studies have shown success in AD diagnosis and risk prediction, neuroimaging data requires significant amounts of time and funding, making it intractable for widespread use. In contrast, we develop imminent dementia predictions based on inexpensive measures. Our approach also complements current approaches by highlighting high-risk individuals who might benefit most from further medical care.

5.2.2 *Basic Dementia Risk Factors*

Without expensive brain imaging, it is common to predict the onset of dementia from age, sex, education, and genetic factors [11,22,71]. In particular, variations in APOE, the gene encoding Apolipoprotein E protein, are thought to be the main genetic factor impacting AD risk [22,64]. However, using only these basic risk factors produces non-robust predictions [96]. Here, we augment these primary risk predictors by adding cognitive and medical variables.

5.2.3 *Modeling Cognitive Trajectories*

Because dementia is characterized by a rapid decline in cognitive functioning, studies have used cognitive variables to predict its onset [60]. Johnson et al. characterized cognitive trajectories for elderly individuals with and without AD and found that precipitous drops in cognition tend to occur between one and three years prior to dementia diagnosis [47]. Based on this result, we use

up to three years of past data to predict imminent dementia onset. Unlike these longitudinal cognition studies, however, we evaluate the need for repeated testing and attempt to reduce the burden on both clinicians and participants of required study visits to achieve accurate, but efficient predictions of dementia onset.

5.2.4 *Diagnosing Cognition Status*

Some research has focused on assessing whether an individual already has dementia [6] or mild cognitive impairment (MCI) [100] via short questionnaires. Multiple cognitive assessments have been developed to efficiently diagnose MCI [32,72], such as the Mini-Mental State Examination (MMSE). Further studies have used MCI diagnoses made by clinicians [13] and MMSE test scores [41] to predict future dementia onset. Building on these successes, we highlight a set of easily administered tests (see Table 5.3) that significantly outperform the sole use of these clinical tests.

5.3 RESULTS

We used data from the Religious Orders Study and Rush Memory and Aging Project (together known as ROSMAP) [11,12], two longitudinal aging cohort studies, to build dementia onset risk prediction models (Section 5.4.1). During each yearly visit, individuals provide medical information and undergo extensive cognitive testing (Table 5.6). We generate samples with at least three years of consecutive visits and no dementia history and then build models to predict imminent dementia onset (i.e., a diagnosis within the next three years). Results described below are based on 9,103 samples from 1,597 individuals, split into stratified training and test sets.

5.3.1 *Preliminary Analyses Reveal Feature Interactions*

Preliminary data exploration comparing imminent dementia and control cases reveals many significant differences in the outcome variable among demographic and cognitive variables (Table 5.6). Additionally, strong correlations are seen between many features and the outcome variable, as well as among features themselves. This is expected since many of the cognitive tests assess the same cognitive domains. Together, these observations suggest that we could train an effective imminent dementia classifier from the available features. Furthermore, the high inter-relatedness of features indicates that some may provide redundant information and may therefore be reduced.

We also explore non-linear and interaction effects in our data to identify appropriate model classes. From these analyses, we observe two notable complex interactions, shown in Figure 5.2. First, having a single APOE e4 allele seems to modulate dementia risk in particular groups: males (Figure 5.2a), people under 85 (Figure 5.2b), and relatively low cognition-scorers (Figure 5.2c). We observe similar modulation among carriers of two APOE e4 alleles (e.g., females), although they represent less than 2% of our sample (Table 5.6). Second, we see a strong interaction between overall cognition and many demographic features. Having a high cognition score may buffer dementia risk regardless of demographic factors, while demographic features might confer more information about risk when they coincide with low cognition. For example, APOE e4-carriers (Figure 5.2c), females (Figure 5.2d), older individuals (Figure 5.2e), and highly educated individuals (Figure 5.2f) seem to exhibit especially high risk if they are also low cognition-scorers. Due to such non-linear effects among our features, a complex model may be useful for capturing interactions among features and risk.

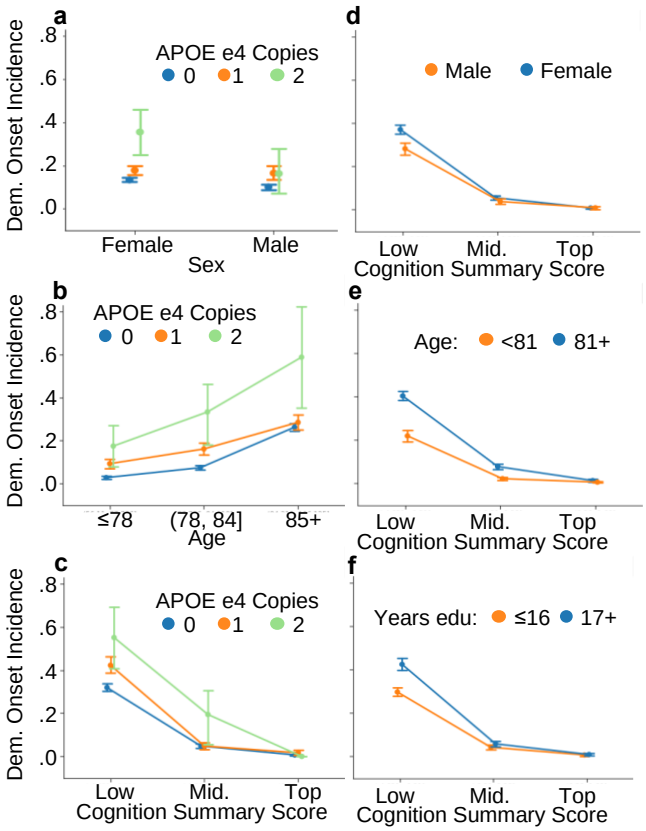

Figure 5.2. Average imminent dementia onset rates (with 95% confidence intervals) by demographic and cognitive factors, highlighting non-linear and interaction effects. [9]

5.3.2 *Multivariate Models Enable Dementia Risk Prediction*

To answer our first research question, we initially aim to build a machine learning model that can accurately predict dementia onset. To do so, we evaluate the prediction performance of multiple model classes and techniques to address class imbalance and time-series data using stratified cross validation (CV) within our training set. Due to class imbalance in our dataset (13.7% rate of dementia onset), we consider various downsampling options. Due to the data's longitudinal nature, we explore the use of time encodings to pre-process input data (e.g., moving averages; described in Section 5.4.4). We find that models trained without downsampling or specialized time encodings had similar or better CV accuracy, AUROC, and AUPRC scores across

all model classes described below, and thus proceed with these selections for all subsequent model tuning.

For our prediction task, we compare the performance of four classes of machine learning models: (1) regularized logistic regression (LR), (2) XGBoost (XGB), (3) multi-layer perceptron (MLP), and (4) long short-term memory network (LSTM). For each model class, we perform extensive hyperparamter selection across five stratified cross-validation (CV) splits (within the training set). Table 5.1 shows the top-performing models in each class (Section 5.4.4 relates tuning procedure details).

Table 5.1. Average cross-validation (CV) performance statistics for each model (± standard error)

Model	CV Accuracy	CV AUROC	CV AUPRC
XGB	**0.9046** ± 0.0045	0.9163 ± 0.0044	0.6763 ± 0.0132
LR	0.9045 ± 0.0048	**0.9205** ± 0.0044	**0.6893** ± 0.0110
MLP	0.9036 ± 0.0056	0.9186 ± 0.0050	0.6694 ± 0.0144
LSTM	0.9021 ± 0.0050	0.9047 ± 0.0168	0.6691 ± 0.0189

In general, we find that many of the model classes achieve similar predictive performance. MLP, LR, and XGB models perform similarly (within the standard error ranges) with respect to AUROC and AUPRC. Among complex model classes, we chose the XGB model because the neural network methods (MLP and LSTM) exhibit unstable performance, as shown by their large error bars in Figure 5.3 (particularly when trained on a single year of data). We opt for an XGB final model over a linear (LR) one because: (1) Unlike linear models, XGB is able to learn non-linear and interaction effects like those found in our data. Prior meta-analyses of dementia risk prediction suggest that the linearity assumption does not hold for critical risk factors (consistent with our observations in Figure 5.2) [35]. Another study found that, even when producing equally accurate predictions, linear methods applied to non-linear data sets tend rely on irrelevant features [67]. Thus, XGB may learn a richer representation of the true complex relationships among

features. (2) Due to the non-linear and interaction effects learned by XGB, we can obtain personalized risk explanations for each individual via interpretability methods (e.g., SHAP, Section 5.4.5), whereas linear models place the same importance on each feature across individuals. Thus, we elect to perform final analyses with an XGB model but compare these results to a linear model for completeness.

5.3.3 *Recent, not Cumulative, Observations are Needed for Effective Dementia Onset Prediction*

We answered our first research question by successfully predicting future dementia onset from three years of consecutive ROSMAP study visits. However, the use of repeated visits may be unrealistic for predicting dementia onset in the general population since only 16% of American seniors receive regular cognitive assessments [34]. Therefore, we turn to RQ2 to evaluate whether we can reduce the burden of repeated cognitive testing (i.e., do we need multiple years of cognitive measurements to make an accurate prediction?). To that end, we evaluate our model's CV AUROC when we reduce the number of consecutive visits in the inputs (Section 5.4.4). As we reduce the number of cumulative years the model sees during training (Figure 5.3, circular markers), we find no major changes in model performance across all four model classes. This suggests that requiring multiple years of consecutive data is not necessary for accurate predictions, which may reduce the burden of regimented follow-up testing in the clinical setting.

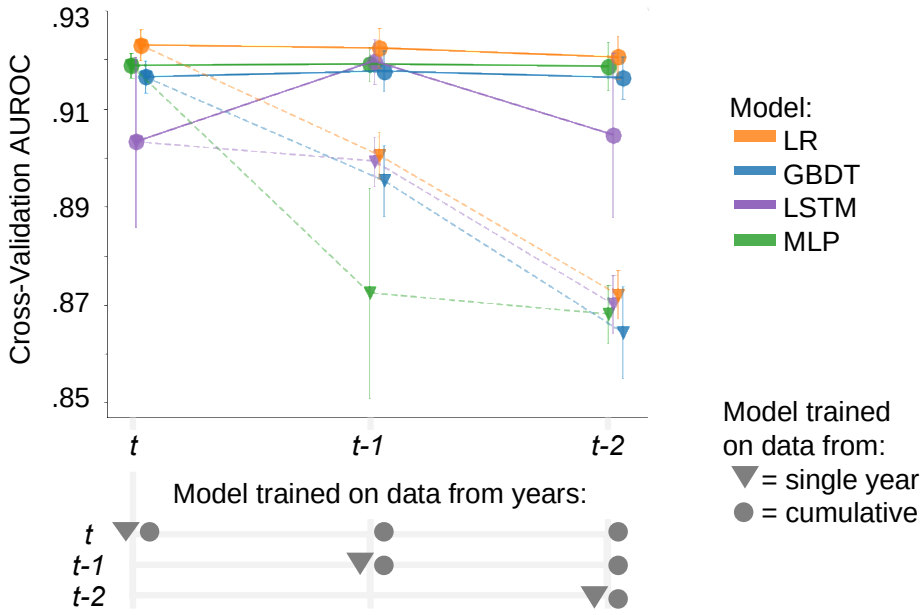

Figure 5.3. Average cross-validation area under the receiver operating curve (AUROC) for our four models trained on different combinations of yearly visits. Circle marks show that cumulative data has limited value, while triangle marks highlight the importance of recent data. [9]

Next, we identify the relative importance of recent data by evaluating the model trained on a single year of past data alone. As expected, we see a decline in prediction performance for models trained on older data (Figure 5.3, triangular markers). Although the most effective prediction models were trained with the most recent year of observations (*t*), we evaluated the stability of our final conclusions by repeating analyses shown in Figure 5.5 and Table 5.2 using data from *t-1* and data from *t-2* (the same data shown by triangle markers in Figure 5.3). In both cases, models show slight drops in all performance metrics, but our models outperform baselines by similar margins on time *t-1* and *t-2* data compared to *t* data. Together, these results imply that repeated testing is not needed for accurate dementia onset prediction since recent data may supersede outdated cognitive information obtained in past years.

Finally, we apply SHAP [66], a local feature attribution method, to our XGB model trained on all three years of consecutive data to ascertain whether the model relies on previously collected data (see Sections 5.3.4 and 5.4.5). We find that the model's top ten features consist of demographic data or tests from the most recent year: even when provided access to measurements from prior years, our model still tends to focus on more recent data. Based on these CV results, we decided to train the final models using only the current year (*t*) of data, a decision that enables earlier, more efficient predictions that need not wait for additional years of cognitive tests before generating a dementia prediction.

5.3.4 *Efficient and Effective Dementia Onset Predictions can be Made with a Small Subset of Features*

After extensive cross-validation experiments, we settle on a final XGB model using the hyperparameters selected based on CV performance. This final "All Features" XGB model is trained on all training data using all available features from year *t* only. Table 5.2 shows held-out test set performance metrics for this model. To drive further insights, we use SHAP local feature explanations [66] to interpret the final model. To see which features our XGB model relies on, we aggregate the local explanations of our training samples to obtain global insights (Section 5.4.5). Figure 5.4 shows the top 20 most important features (ranked by their average SHAP importance magnitude across all samples).

Table 5.2. Test performance of final models (± standard error from bootstrap re-sampling)

Model		Test Accuracy	Test AUROC	Test AUPRC	Relative IDI (Simplified with APOE vs. row)
Final Models	All Features (XGB)	**0.8975** ± 0.0002	**0.8977** ± 0.0003	**0.6387** ± 0.0010	-0.0571
	Simplified (with APOE) (XGB)	0.8947 ± 0.0002	0.8903 ± 0.0003	0.6236 ± 0.0010	–
	Simplified (no APOE) (XGB)	0.8964 ± 0.0002	0.8896 ± 0.0003	0.6184 ± 0.0010	0.0084
Baseline models in our study	Linear Selected Features (LR)	0.8825 ± 0.0002	0.8224 ± 0.0004	0.4907 ± 0.0011	0.6012
	Linear Selected Features (XGB)	0.8781 ± 0.0002	0.8050 ± 0.0005	0.4771 ± 0.0011	0.7432
Baseline feature sets in the literature	Demographics + MCI (XGB) [13]	0.8770 ± 0.0002	0.8203 ± 0.0005	0.4449 ± 0.0011	0.5058
	Normalized Cognitive Features Sum (LR)	0.8737 ± 0.0002	0.8128 ± 0.0005	0.4473 ± 0.0011	0.9804
	Demographics + MMSE30 (XGB) [41]	0.8748 ± 0.0002	0.8124 ± 0.0005	0.4273 ± 0.0011	0.6707
	Demographics (XGB) [11]	0.8593 ± 0.0003	0.7215 ± 0.0005	0.2660 ± 0.0008	2.9291

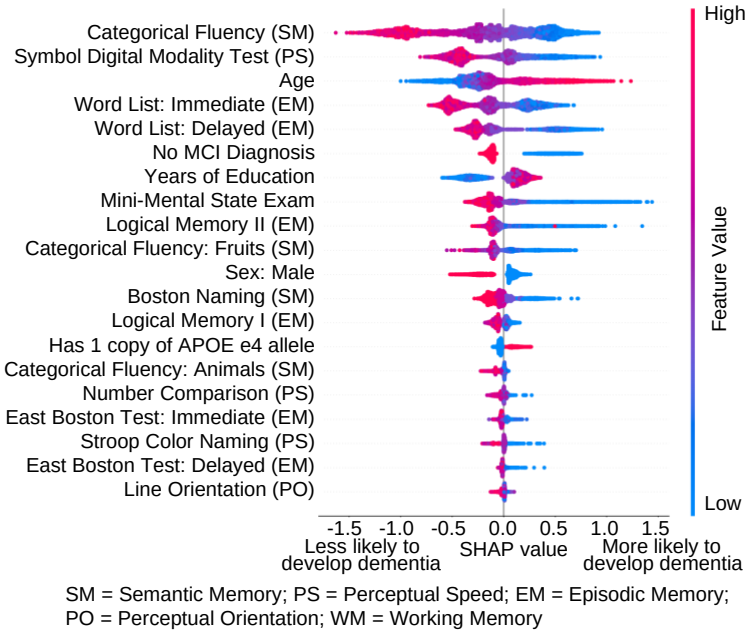

Figure 5.4. SHAP summary plot: violin plot of the 20 most informative features of the XGBoost current year model, ordered by importance. Each point is a training sample colored by its feature value. The point's x-axis position is the feature's contribution to the final risk prediction. [9]

First, we note that the feature attributions are consistent with findings in the literature, validating our modeling approach and SHAP interpretations. For example, nearly all previous work [11] has found females, older individuals, and carriers of an APOE e4 allele to be at higher risk of dementia, consistent with Figure 5.4 SHAP explanations. Similarly, as expected, low performance on all cognitive tests contributes to a higher risk score. In contrast, our years of education feature attributions are not consistent with the literature (which find negative associations between high education and dementia incidence). We discuss this result further in Section 5.5 (Limitations).

As we move down the list of top-ranked features, we see a dramatic drop in the magnitude of SHAP values (i.e., relative influence of a given feature on the final prediction). We therefore hypothesized that future dementia onset can be predicted using only the most informative features. For evaluation, we choose the top four demographic features and top four cognitive tests and use them to train a simplified prediction model. The top demographic features (age, sex, education,

and APOE genotype) are widely cited as being important [11] and are simple to measure. The four top cognitive tests chosen are: categorical fluency (Cat Flu; 2 minutes; semantic memory); symbol digit modality test (SDMT, ≤ 5 minutes, perceptual speed); word list test (WL, 3 minutes, episodic memory), and mini-mental state exam (MMSE30, 5- 10 minutes, general cognition); Table 5.3 describes these tests. Interestingly, each test lies in a different cognitive domain in Table 5.6, indicating that the model relies on diverse and nonredundant cognitive attributes. From our simplified feature set, we train two "simplified" final models on our full training set: one including and one excluding the APOE genotype (which, though commonly used in prior studies, is not always available in clinical settings). Although cognitive diagnostic status (MCI diagnosis) was ranked among the top influential features, we excluded it from our simplified models: it is very time consuming to obtain in the ROSMAP study (since it is based on all cognitive tests and a clinician examination), and it may be difficult to obtain in the general aging population.

Table 5.3. Selected cognitive tests from XGBoost (XGB) and linear regression (LR) models (cognitive domains shown in Table 5.6). Full cognitive battery: 98 minutes

Test (Domain)		Time (min)	Description
Selected cognitive tests from XGB model	Categorical fluency (SM)	2	Subject names as many items in a category as they can in a minute (Rounds: animals, fruits)
	Symbol digit modality (PS)	≤ 5	Subject learns a symbol-to-digit mapping, then must substitute digits when symbols are shown
	Word list (EM)	3	Subject hears a list of 10 words, then is tested on immediate recall, delayed recall, and recognition (selecting correct words from distractors)
	Mini-mental state exam	≤ 10	Short diagnostic general cognition test for dementia
Selected cognitive tests from LR model	Digits forward (WM)	5	Given a list of numbers, subject repeats them in the same order as given
	Digit ordering (WM)	5	Given a list of numbers, subject repeats them in numerical order
	East Boston test (EM)	6	After hearing a short story, subject recalls story units immediately and after distractor-filled delay
	Digits backward (WM)	5	Given a list of numbers, subject repeats them in the reverse order as given

Unlike the above use of SHAP-based feature selection from our XGB model, feature selection for linear models involves choosing those with the highest magnitude regression coefficients. For comparison with our SHAP selection method above, we use standard feature selection based on the final LR model's coefficient magnitudes. For consistency, we use the same four demographic features as above and then select the cognitive features with the highest-magnitude regression coefficients: digits forward, digits backward, digits ordering (all working memory), and the East Boston Test (episodic memory) (21 minutes total; See Table 5.3).

We compare our final simplified XGB models to the XGB model trained on the full feature set, an LR and XGB model trained using the features from linear feature selection, and a baseline XGB trained on multiple commonly used clinical baseline feature sets (Section 5.4.6). Figure 5.5 shows the held-out test set's receiver operating curves for all models, highlighting the sensitivity and specificity based on all decision boundaries on the test set. Using a decision cut-off of 0.5, we report true negatives, false positives, false negatives, and true positives for our top models and the top performing baseline model in Table 5.4. For each model, Table 5.2 lists the area under the receiver operating curves (AUROC), precision recall curves (AUPRC), and the accuracy at a 0.5 decision cut-off point. Additionally, we calculate the relative integrated discrimination improvement (IDI), comparing the "Simplified (with APOE)" model's discrimination ability to every other model [97].

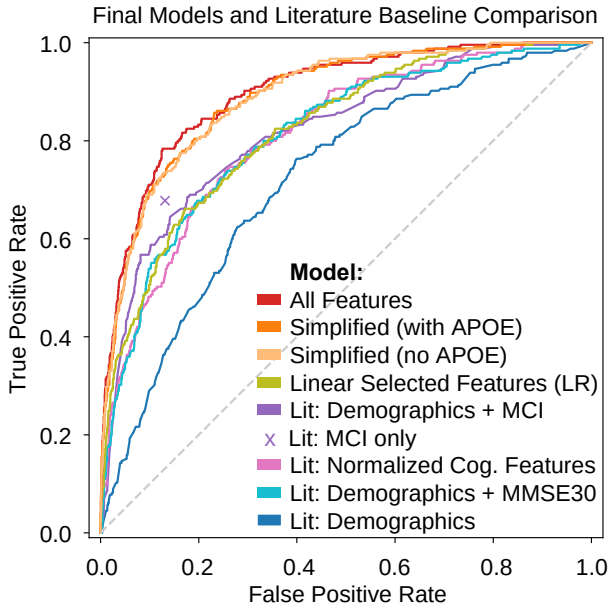

Figure 5.5. Receiver operating curves: final models and baselines from the literature (Lit). Area statistics in Table 5.2. [9]

Table 5.4. Using a 0.5 decision cutoff, we report the number of true negatives (TN), false positives (FP), false negatives (FN), and true positives (TP) in the test set

Model	TN	FP	FN	TP
All Features (XGB)	1510	50	135	110
Simplified (with APOE) (XGB)	1513	47	143	102
Demographics + MCI (XGB) [13]	1479	81	141	104

Figure 5.5 and Table 5.2 show that our methods significantly outperform XGB models trained with more restricted feature sets documented in the literature [13,41]. Notably, our simplified dementia onset predictor is only slightly less accurate than the model trained using all features (with the non-APOE model showing only a slight performance drop compared to the APOE model). Additionally, our SHAP approach selects a more effective set of features than the classic linear feature selection approach, further supporting our choice of using a non-linear model (i.e., XGBoost).

Together, these results show that computing SHAP feature importances for our XGB model allows us to identify measures that are particularly useful in our model and thus dramatically improve prediction performance over more basic clinical baselines by including a few short cognitive tests. These tests are standardized and simple to administer; any primary care physician or assistant could conduct them during a patient's annual physical exam (taking a total of 15-20 minutes to administer compared to the 98 minutes required for all tests in the ROSMAP neuropsychological battery in Table 5.3).

Cross-cohort generalizability: Due to differences in study design and measured features, it is uncommon for dementia prediction studies to validate findings with external datasets [35,96]. While our data is comprised of pooled ROS and MAP samples, the studies recruit participants from different groups (clergy from Catholic religious organizations across the US and individuals in retirement facilities throughout northern Illinois, respectively) [11,12], and these studies differ in demographic and lifestyle factors and outcomes (see Section 5.4.3). Thus, we seek to evaluate the cross-study generalizability of our final models. Using our previously defined training and test splits, we retrain our Simplified (with APOE) model separately for ROS and MAP training samples and evaluate each model's performance separately for ROS and MAP held-out test samples. In both cases, the "external" and "internal" test set AUROCs are within 0.01 of each other (Table 5.5). Furthermore, similar tests would be selected if we were to perform feature selection based on models trained separately from each cohort (the same top four tests for ROS, and three of four top tests–with the number comparison test replacing MMSE–for MAP). Together, these findings indicate that the model generalizes stably and effectively across cohorts, both in terms of predictive performance and selected features.

Table 5.5. Cross-study test set performance for ROS vs. MAP models

		AUROC for Test Set Samples	
		ROS (N=1156)	MAP (N=649)
Training Samples	ROS (N=4506)	0.8848	0.8948
	MAP (N=2792)	0.8792	0.8851

Missing data experiments: As shown in Table 5.6, many features have missing values and are imputed for all analyses (Section 5.4.3). In particular, some features are missing at significantly different rates for control versus dementia onset cases. To ensure that our promising results were not driven by a confounding effect of imputing features at different rates between case and control groups, we experiment with removing potentially confounded samples and features as follows. We first exclude features with one-fifth of samples missing (Stroop color naming and Stroop word reading tests). We next exclude all samples with a missing observation for any of the remaining 10 features with significantly different rates of missingness between control and dementia onset cases, resulting in a new dataset with 8,392 samples (92% of the original dataset). First, we note that final model performance on this filtered dataset (test AUROC=0.8952) is very similar to performance from the full dataset (test AUROC=0.8977). Importantly, our SHAP feature rankings (generated via average SHAP importance magnitude) result in the same top four selected cognitive tests as the original dataset. Further, we observe similar performance for the final simplified model (test AUROCs of 0.8865 and 0.8903, respectively, for filtered and original datasets). Together, these experiments indicate that imputing missing features had little effect on our final models.

Table 5.6. Between-group baseline (time *t*) statistics. We provide summary statistics for each group (including missingness rates and indicators of significantly higher rates of missingness for one group). (* p<0.05, ** p<0.01, *** p<0.001 for statistical tests)

	All Samples: Between-group test statistic	Summary Statistics (% samples missing and associated significance)	
		Controls: No impending dementia (N=7866)	Dementia onset within 3 years (N=1244)
Demographics			
Age	$t = -29.60$***	80.12±6.77 (0%)	86.19±6.37 (0%)
Sex: % male	$\chi^2 = 15.43$***	29.5% (0%)	24.0% (0%)
Years of education	$t = 1.45$	16.91±3.61 (0.1%)	16.75±3.56 (0.2%)
Race (White/Black/Native American/Asian)	$\chi^2 = 12.72$**	94.0% / 5.5% / 0.3% / 0.2% (0%)	94.2% / 4.9% / 0.2% / 0.6% (0%)
Ethnicity: % Hispanic	$\chi^2 = 0.10$	3.1% (0%)	3.3% (0%)
# APOE e4 copies (0/1/2)	$\chi^2 = 54.59$***	78.6% / 20.3% / 1.1% (1.5%)	70.3% / 27.0% / 2.8% (1.4%)
Episodic Memory (EM)			
Word list: immediate (1min)	$t = 40.10$***	20.57±4.55 (2.4%)	15.01±4.08 (2.3%)
Word list: delayed (1min)	$t = 45.43$***	6.76±2.16 (2.4%)	13.69±2.34 (2.4%)
Word list: recognition (1min)	$t = 32.82$***	9.85±0.56 (2.3%)	9.02±1.74 (2.7%)
East Boston test: delayed (3min)	$t = 26.25$***	9.89±1.73 (0.3%)	8.46±2.03 (1.0%**)
East Boston test: immediate (3min)	$t = 34.59$***	9.64±1.90 (0.5%)	7.41±3.07 (1.2%**)
Logical memory I (3min)	$t = 37.73$***	14.34±4.10 (2.3%)	9.51±4.44 (2.1%)
Logical memory II (3min)	$t = 40.47$***	13.23±4.45 (2.4%)	7.60±4.81 (2.4%)
Perceptual Orientation (PO)			
Line orientation (15min)	$t = 13.54$***	10.59±2.97 (3.8%)	9.32±3.02 (6.1%***)
Progressive matrices (20min)	$t = 22.82$***	11.65±2.82 (5.1%)	9.61±2.79 (8.2%***)
Perceptual Speed (PS)			
Symbol digits modality test (5min)	$t = 37.91$***	41.77±10.09 (3.9%)	29.72±9.87 (7.2%***)
Number comparison (3min)	$t = 26.12$***	26.22±7.23 (3.7%)	20.29±7.12 (6.2%***)
Stroop color naming (3min)	$t = 22.30$***	20.19±7.34 (65.2%***)	12.34±6.55 (60.0%)
Stroop word reading (3min)	$t = 13.66$***	48.87±13.53 (65.3%***)	39.74±14.55 (60.1%)
Semantic Memory (SM)			
Boston naming (5min)	$t = 29.15$***	14.19±0.98 (3.0%)	13.22±1.51 (4.0%)
Categorical fluency: animals (1min)	$t = 33.21$***	18.25±5.45 (0.1%)	12.90±3.96 (0.4%)
Categorical fluency: fruits (1min)	$t = 37.98$***	18.26±5.13 (0.2%)	12.44±4.15 (0.6%*)
Categorical fluency (combined)	$t = 40.00$***	36.51±9.42 (0.1%)	25.33±7.08 (0.4%)
National adult reading test (2min)	$t = 5.15$***	8.49±1.94 (3.6%)	8.17±2.14 (6.7%***)
Working Memory (WM)			
Digits backward (5min)	$t = 16.94$***	6.61±2.05 (0.4%)	5.56±1.82 (0.9%*)
Digits forward (5min)	$t = 11.92$***	8.43±1.98 (0.2%)	7.70±1.99 (0.6%)
Digit ordering (5min)	$t = 21.30$***	7.60±1.56 (1.0%)	6.57±1.67 (2.4%***)

Global Cognition			
Mini-mental state exam (5-10min)	$t = 45.16$***	28.59±1.51 (2.2%)	26.20±2.71 (1.4%)
Medical history/lifestyle factors			
MCI (No/Yes/Yes-other)	$\chi^2 = 1685.26$***	86.9% / 12.8% / 0.3% (0%)	37.1% / 60.7% / 2.3% (0%)
Medical conditions sum	$t = -3.77$***	1.68±1.16 (2.0%)	1.82±1.21 (2.0%)
Vascular disease burden	$t = -7.02$***	0.45±0.66 (2.0%)	0.59±0.75 (2.0%)
Vascular disease risk	$t = -1.31$	0.87±0.81 (1.2%)	0.90±0.77 (1.0%)
Any history of:			
cancer	$\chi^2 = 2.48$	40.2% (2.0%)	37.8% (2.0%)
claudication	$\chi^2 = 19.62$***	22.4% (2.0%)	28.2% (2.0%)
diabetes	$\chi^2 = 0.44$	11.6% (2.0%)	12.3% (2.1%)
diabetes medication	$\chi^2 = 1.97$	15.6% (1.2%)	17.2% (1.0%)
head injury with loss of consc.	$\chi^2 = 0.03$	9.7% (2.0%)	9.8% (2.0%)
heart disease	$\chi^2 = 10.04$**	12.7% (2.0%)	16.0% (2.0%)
hypertension	$\chi^2 = 7.24$**	56.9% (2.0%)	61.0% (2.0%)
stroke	$\chi^2 = 36.05$***	9.7% (0.9%)	15.3% (0.5%)
thyroid disease	$\chi^2 = 1.51$	24.1% (2.0%)	25.8% (2.0%)

5.3.5 *SHAP Provides Personalized Risk Explanations*

We turn now to our third research question, which addresses personalized dementia risk explanations. Because XGB learns complex relationships among features (unlike linear models), we examine SHAP interaction values among pairs of features (Section 5.4.7). For example, according to XGB interactions, having one copy of the APOE e4 allele impacts an individual's XGB risk prediction, particularly if he or she has a low cognition score (Figure 5.6a, consistent with Figure 5.2c) or is younger (Figure 5.6b, consistent with Figure 5.2b). Finally, males, especially those younger than 80, are at particularly low risk for developing dementia (Figure 5.6c, consistent with earlier findings [11]). Thus, by using SHAP to interpret our simplified XGB model, we find that aggregating non-linear feature effects across samples reveals relevant interactions learned by the model, and that these interactions are consistent with our data's structure (Figure 5.2) and prior literature.

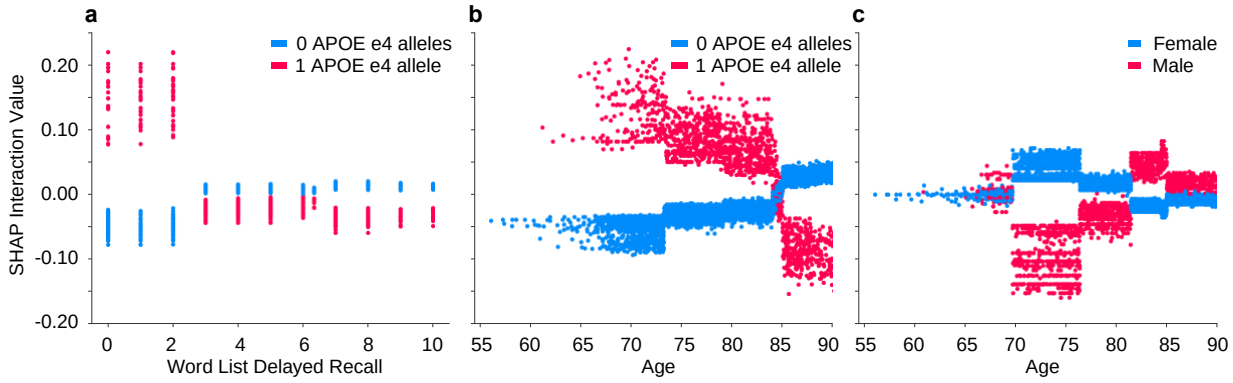

Figure 5.6. SHAP interaction values for selected pairs of features in our final Simplified (with APOE) XGBoost model. [9]

Beyond receiving a risk score, using XGB and SHAP feature attributions gives patients and their medical practitioners a personalized explanation of risk (i.e., how particular features drive the XGB's prediction). To illustrate how this benefit works, we generate a synthetic sample that represents the "typical sample" in our dataset (with mode- or average-valued features) and display the risk score and explanation in Figure 5.7a. We show perturbations to single features of APOE (where we change the APOE e4 allele count from zero to one) in Figure 5.7b and the word list delayed recall (WLDR) score from the average value to two standard deviations below the average in Figure 5.7c. In both examples, we see that the perturbed variable becomes the primary risk factor driving up the risk score compared to the "typical individual."

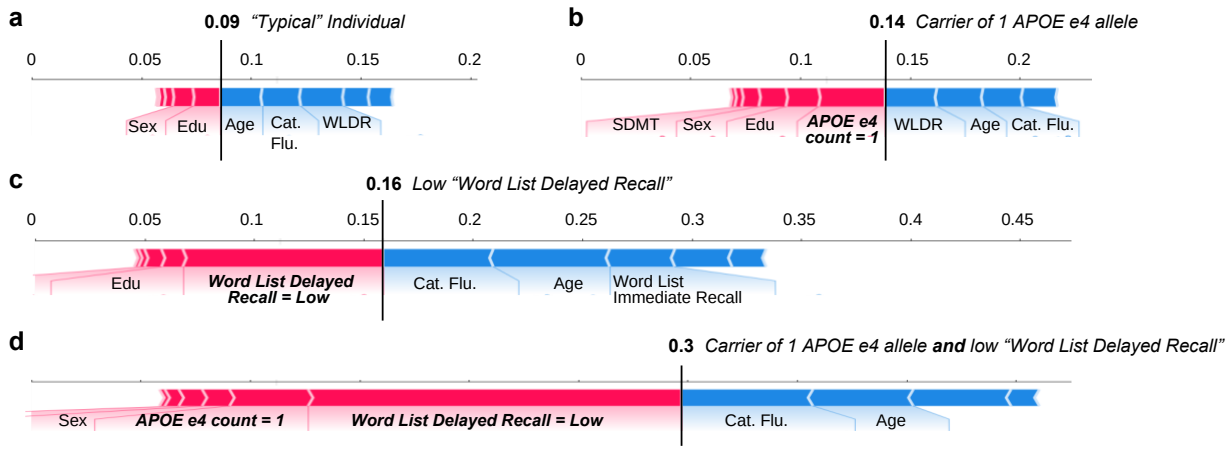

Figure 5.7. Feature explanations for synthetic samples: (a) risk and explanations for a "typical individual" in the ROSMAP data, (b, c) perturbations to single features (bolded), (d) the combined effects of both risk factors. [9]

Finally, in Figure 5.7d, the effect of both risk factors from parts b and c shows that the combined risk of having one APOE e4 allele and a low WLDR score substantially increases risk. In particular, the jump in risk from both risk factors (a 0.21 increase over the "typical individual") far exceeds the additive effects of each single risk score alone (0.05 and 0.07 increase, respectively). A linear model, in contrast, would have produced additive predictive importance values and therefore would have failed to identify a compounding effect of these features. This example highlights the ability of our XGB model with SHAP interpretations to provide personalized risk explanations based on a combination of feature values. This ability may prove to be powerful in clinical settings because it would help clinicians discuss the unique configuration of risk factors relevant to individual patients. For example, a clinician might discuss or recommend different actions to a patient whose risk is driven by demographic factors compared to a patient whose risk is driven by low cognitive testing scores.

5.4 METHODS

We now describe in detail how we produced the results described in this paper. Additionally, our code for reproducing these results is available at: https://github.com/suinleelab/EEDRP.

5.4.1 *Dataset*

The Religious Orders Study (ROS) [11] and Memory Aging Project (MAP) [12] are complementary epidemiological studies that each enroll persons without dementia who agree to annual evaluations and eventual organ donation. ROS enrolls clergy living communally from 40 Catholic groups across the US (primarily employed or retired nuns, priests, and brothers). This study group was selected because communal living provided both high follow-up rates and relative consistency in life experiences and socioeconomic factors. However, as a volunteer cohort of Catholic clergy, the samples are not representative of a wider population of elderly individuals [11]. As a complementary study, MAP recruited participants from a wider range of life experiences throughout northeastern Illinois. Participants are primarily enrolled from continuous care retirement communities (ranging in care levels from independent living to nursing on campus). To reduce participant burden and facilitate high follow-up rates, data was collected via home visits. Clinical data collection procedures were consistent between both studies to allow the data to be merged for analyses [12]. Due to their recruitment strategies, follow-up rates of survivors reached around 95% for both studies. Compared to ROS samples, MAP samples were obtained from relatively fewer males (23.6% vs 31.9%) and from individuals who were older (83 vs 80 years on average) and less educated (15 vs 18 years on average). MAP samples also had higher rates of MCI (21.2% vs 19%) and a higher incidence of dementia onset within three years (15.3% vs. 12.7%).

Upon entering the study, participants share demographic information (e.g., sex, age) and blood samples for genotyping. At each yearly visit, they provide updated medical information and undergo a battery of cognitive tests, resulting in repeatedly measured variables. We predict dementia onset from 41 separate variables (per time point; note that categorical variables were one-hot encoded, leading to 49 total features), which we list in Table 5.6. In total, the data contains 3,194 individuals with one to 23 annual visits. Of all participants with at least two years of visit data and no original dementia diagnosis, 619 (23.7%) were eventually diagnosed with dementia.

5.4.2 *Data Processing: Generating Samples*

Our prediction task (Figure 5.1) is: using data from their three most recent practitioner visits, does an individual with no history of dementia experience dementia onset within the next three years? In particular, our selected time-frame was based on prior findings that a precipitous drop in cognitive abilities is usually observed one-to-three years prior to a dementia diagnosis [47]. To construct the appropriate dataset, we narrow our analyses from the 3,194 existing participants to 1,597 individuals for whom we have enough observations.

Many participants had more than six consecutive yearly visits, so we applied a sliding window of six years over their available consecutive visits, thereby generating at least one sample, but often more, per participant. Each sample is split into an input window (consisting of the first three consecutive visits *t-2*, *t-1*, and *t*) and onset prediction window (consisting of the next three consecutive visits: *t+1*, *t+2*, and *t+3*), as illustrated by positive (dementia onset) and negative (no dementia onset) examples in Figure 5.8. Because the goal is to predict future dementia onset in individuals who do not yet have dementia, we exclude all samples in which dementia is already present during visits *t-2*, *t-1*, and *t* (e.g., Figure 5.8, Example Participant A, samples 2 and 3). Finally, we applied sliding windows of four and five years to identify any additional positive onset

cases (e.g., Figure 5.8, Participant B, Samples 4 and 5), which helped to mitigate our class imbalance. This procedure could not be used to find negative dementia onset samples because all three future years must be known to definitively rule out a dementia diagnosis.

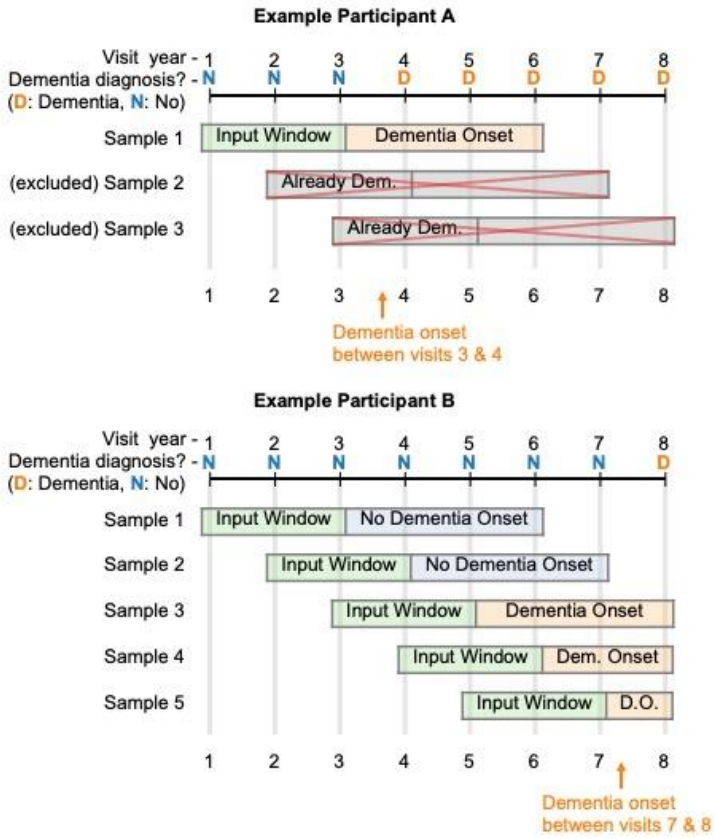

Figure 5.8. Examples of samples from sliding windows. Our samples have no history of dementia during the first 3 years, and either no onset for all of the next 3 years (negative case) or a dementia diagnosis in any of the next 3 years (positive case). [9]

5.4.3 *Data Processing: Pre-processing for all Models*

After combining all valid six-year windows (and four-year and five-year windows where appropriate), we have a sample size of 9,103 samples, of which 13.7% were labeled as positive dementia onset cases (derived from 1,597 individuals, of which 521 developed dementia). For each model next described, our model inputs consist of variables obtained during the first three visits

(*t-2*, *t-1*, and/or *t*), called the input data window, and the outputs are a prediction of whether the individual was diagnosed with dementia at any of visits *t+1*, *t+2*, or *t+3*. Table 5.6 shows all demographic, cognitive, and medical features from our 9,103 samples (at time *t*), split by dementia onset label, and associated between-group differences.

Since some downstream analyses require variables to be on the same scale, we standardize all continuous variables for our input data and use z-scores as features for all models. To maintain consistent scores across time points, z-scores are calculated based on time *t* observations (and the same rescaling procedure based on time *t* is applied to observations at *t-1* and *t-2*). For categorical variables, we apply one-hot encoding. We note in Table 5.6 that most variables have some missing observations across our samples. We impute all missing samples using the mean for continuous variables and the mode for categorical variables (across all samples). Using chi-square tests of independence, we find that some cognitive tests have significantly different missingness rates between dementia onset and control groups, although the rates tend to be low (between 0.2% and 6%, except for Stroop test variables). Additional analyses described in Section 5.3.4 confirm that the effects of imputing values did not impact our final results (compared with filtering out the affected cases).

We next describe model selection with cross-validation and then evaluation on a test set. For each analysis, we use the same stratified training and test sets. To avoid contaminating our test set with training examples, we split our data by participants so that all samples from a single individual fall into the training set or test set, but not both. Of our 1,597 participants, we assigned one fifth of them to the test set (1,805 associated samples) and the remaining individuals (7,298 associated samples) to our training set. Next, we randomly divide our training set participants into five

stratified cross-validation splits. All splits were performed in a stratified manner to maintain consistent ratios of AD to control cases.

5.4.4 *Building and Evaluating Prediction Models*

We evaluated modeling options under several domains: sampling techniques to address class imbalance, time encoding techniques, and model class. Our modeling choices were based on average accuracy, areas under the receiver operating curve (AUROC), and areas under the precision recall curve (AUPRC) across five cross-validation (CV) folds.

Downsampling: The dataset has a class imbalance of 13.7% positive labels since few individuals experience dementia onset in any given 3-year window. Therefore, we experimented with four different downsampling techniques: (1) no downsampling, (2) class re-weighting (incorporated into loss functions during training), (3) random downsampling (randomly selecting as many negative as positive samples), and (4) matched pairs downsampling. In (4), for each positive sample, we select the closest negative sample based on sex, age, and education (greedily, without replacement). Due to equal or better prediction performance across five-fold CV, all final models are trained with no downsampling (see Section 5.3.2).

Time-series encoding: Because of the longitudinal nature of many features, we evaluated methods for incorporating repeatedly observed variables: (1) all data (no special encoding), (2) moving averages, and (3) slopes (see Table 5.7). Per Section 5.3.2, training with all data yielded similar or better CV performance, and thus was used for all subsequent models.

Table 5.7. Encoding methods used for time-series features

Name	Description
All data	Unaltered data from *t, t-1, t-2*
Moving averages	(1) Unaltered features from *t*, (2) one simple moving average feature derived from *t, t-1, t-2* features, and (3) three exponential moving average features with half-life values of 1, 2, 3 years derived from *t-2* features
Slopes	(1) Unaltered features from *t*, (2) the change in features from each year to the next (i.e., ν_t - ν_{t-1} and ν_{t-1} - ν_{t-2} for variable ν), and (3) the overall change in scores from the earliest year to the current year (i.e., ν_t - ν_{t-2})

Model Type: We compared the performance of four classes of machine learning models: (1) logistic regression (LR; implemented with Scikit-Learn [81]), (2) gradient-boosted decision trees via the XGBoost algorithm (XGB; known for handling mixed feature types and medical data well [18]), (3) multi-layer perceptrons (MLP; deep learning approach), and (4) long short-term memory networks (LSTM; time series aware deep learning approach). Both deep learning approaches were implemented in Keras [21] and TensorFlow [69]. For each model class, we evaluated several hyperparameter settings and selected the setting with the highest average CV AUROC (reported in Table 5.1). We share our final hyperparameters, along with average CV performance across modeling choices described in this section, in our code repository: https://github.com/suinleelab/EEDRP.

Training with fewer input years: We next evaluate whether we can reduce the burden of repeated cognitive testing (i.e., do we need multiple years of data to accurately predict dementia?). We compare performance of models trained on the last 3 year's visits with models trained on fewer time points: the last 2 years of visits (*t* and *t-1*) and the most recent visit (*t*) (circular markers in Figure 5.3). We also evaluate the importance of recent data for impending dementia predictions: in addition to evaluating the model trained on the most recent visit alone (*t*), we also train models on data from single visits one and two years earlier (*t-1* alone and *t-2* alone) (triangular markers in

Figure 5.3). Results (Section 5.3.3) indicate that recent, but not repeated, measurements are needed for accurate prediction.

To further explore whether the model relies on past data, we perform feature importance analysis using SHAP (Section 5.4.5) on our XGB model trained on the last 3 years of data. The model's top ten features are from time *t* (including demographic features), which provides further evidence that relying on past measurements is not necessary.

5.4.5 *Model Interpretation with SHAP Explanations*

To explore what the model is learning and drive further insights, we use SHAP local feature explanations applied to our XGB model (trained on the full feature set with current year, t, data). To obtain global feature importances, we aggregate local feature attributions across training samples. Features with higher global importances have more impact on model predictions across samples (Figure 5.4). Next, we select a subset of available features based on their global SHAP ranking: the top 4 demographic features (age, sex, education, APOE genotype) and the top 4 cognitive tests (with their subtests; Table 5.3). Our final feature set excludes the variable "No cognitive impairment diagnosis" because it is a cognitive diagnosis that is inefficient to obtain (based on both the full 98-minute neuropsychological battery and a medical review from a physician). Finally, to compare feature selection using SHAP global importances to the more typical global feature selection method in linear models, we use the same demographic features and select the four cognitive tests with the highest magnitude coefficients from the linear model (Table 5.3).

5.4.6 *Measuring Final Model Performance*

First, we compare final test performance of XGB trained on the full feature set compared with 2 simplified feature sets: (1) the top four demographic features and the top 4 cognitive tests, and (2) the same set of features but excluding APOE genotype, which may be expensive to obtain for those without existing genotype data. To compare selected features from SHAP to those from a simple linear method, we also report performance for XGB and LR models trained on the features selected via LR coefficients, described above (Table 5.3).

Finally, for comparison with our methods, we also generate multiple baseline XGB models trained on features commonly used as risk indicators in the literature (Figure 5.5 and Table 5.2): (1) demographic features (above) [11]; (2) MCI diagnosis and demographic variables [13]; (3) the MMSE30 and demographic variables [41]; and (4) the sum over all normalized cognitive test scores controlled for age, sex, and education.

Figure 5.5 displays ROC curves, showing the performance of models at all possible decision cut-off points. We also show confusion matrices for the top-performing baseline and final models using an example cut-off of 0.5 (Table 5.4). Table 5.2 summarizes all performance metrics, including confidence intervals from bootstrap resampling of the test set (repeated 1,000 times). Per Figure 5.5 and Table 5.2, the features selected from the XGB model result in similar AUROCs compared with the full feature set (and outperform the linearly selected features). While the full cognitive battery requires 98 minutes of cognitive testing, we achieve similar predictive value using only four tests that take under 20 minutes.

5.4.7 *Examining SHAP Explanations in the Final Model*

Feature interactions: To explore the complex interactions learned by the XGB model, we examine SHAP interaction values among pairs of features in our final simplified model. For each

sample in our training set, the SHAP interaction value for two features represents the remaining combined feature effect after removing individual main effects of both features.

Figure 5.6 shows feature interactions in the XGB model: each point is a training sample colored by one feature and placed on the x-axis according to its value for the second feature. The y-axis indicates the sample's SHAP interaction value (refer for more detail to [67]). In parts b and c, samples with ages over 90 were censored due to privacy requirements. Higher absolute value y-axis values in these plots indicate that the XGB model makes risk predictions based on feature combinations rather than independently based on single features.

Personalized explanations: For any sample, we can generate a SHAP force plot to explore personalized risk explanations provided by SHAP applied to our final XGB model [68] (e.g., Figure 5.7). These plots indicate both the model's dementia onset risk prediction and the SHAP values for the highest contributing features impacting the prediction (pink arrows for risk factors, and blue arrows for protective ones).

To clarify the variations in explanations in a controlled setting, we generate four synthetic examples. First, we show a SHAP force plot for a "typical individual" in our dataset (i.e., a sample with mean or mode values for all features; Figure 5.7a). A "typical individual" has a low risk of developing dementia in the next three years since the diagnostic rate for dementia is low in any 3-year period. Next, we show perturbations to single feature values for APOE (where we change the APOE e4 allele count from zero to one; Figure 5.7b) and word list delayed recall (WLDR) score (from the mean value to two standard deviations below the mean, i.e., from six words remembered to just one; Figure 5.7c). Finally, we simultaneously perturb both risk factors above and show that the combined risk of having both one APOE e4 allele and a low WLDR score leads to a large, non-linear jump in risk that exceeds the combined single effects of each feature alone (Figure 5.7d).

5.5 DISCUSSION

Comparison with previous findings: Reviews of dementia prediction studies have found that combinations of cognitive tests have aided in the prediction of dementia onset [10,96]. In particular, for predicting conversion from MCI to dementia, combining episodic memory tests with executive functioning or language tests tended to produce high predictive accuracy [10]. A review of community-based aging cohort studies (consistent with our approach) also found that using three or four tests spanning multiple cognitive domains led to improved predictions of dementia onset for 2.5 to 5 year follow-ups [96]. Compared to our results, these studies reported similar or lower AUROCs (ranging from 0.83 to 0.88); however, each study was based on samples from different cohorts (ranging from 478 to 551 total participants) and with different follow-up periods, so direct comparison may not be appropriate. Importantly, despite being performed on a larger cohort (1,597 individuals) and using a non-linear XGB model (unlike the previous studies, which all relied on linear analyses), our approach identified a small number of tests spanning multiple cognitive domains (Table 5.3) as predictors of dementia, consistent with these prior studies.

Longitudinal input data: Curiously, our analyses show the modest value of longitudinal measurements. Because dementia is an acquired condition marked by cognitive decline, one might expect to see gradual changes in cognition prior to dementia onset. In fact, our choice of a three-year input data window was based on observed cognitive changes preceding dementia in prodromal cases [47]. However, because our goal is to predict a future dementia diagnosis (not a current one), changes in cognition scores may be less useful than expected. Our results seem consistent with other studies, which reported limited value for cognitive changes in predicting future dementia onset. In particular, one study found that reliable change indices (RCIs) for MMSE had low predictive accuracy for dementia onset [40]. Furthermore, because longitudinal input data

inherently requires rarer datasets with multiple cognitive assessments, RCI-based studies have often failed to achieve the same predictive accuracy as single-observation studies [96].

Limitations: Our final dataset contained 9,103 samples from 1,597 individuals, of which, 521 developed dementia. Although our study is based on a larger dataset than prior studies mentioned above [96], future studies should replicate our findings in other populations. Because we rely on samples from the ROS and MAP cohorts, our findings are subject to potential bias introduced by each cohort's procedures. In particular, for our ROSMAP samples, approximately three quarters come from females and two thirds from participants with 16 or more years of education. The unusually high education levels in our data may explain why some feature explanations for education level are inconsistent with findings in the literature. Future studies should especially explore sex and education-based dementia risk in a more balanced dataset.

It is uncommon for dementia prediction studies to validate their findings externally due to prohibitive differences in study design, populations, and measured features [96]. According to a recent review [35], less than a quarter of examined machine learning studies externally validated their findings (the majority of which were imaging studies with harmonized measurements). As with many prior studies, we could not directly assess our findings on an external dataset. Nevertheless, despite differences between the ROS and MAP cohorts, our models generalized well between them when trained separately.

Additionally, the Stroop color naming and word reading tests had high levels of missingness in our dataset (Table 5.6), so it is possible that those tests may have been more highly ranked if they were observed in more samples. However, most features had relatively low rates of missingness (Table 5.6), and we found that there was not a significant relationship between feature missing rates and their SHAP importance for our final XGB model (Pearson correlation $r = -0.11$,

p = 0.46). Furthermore, analyses described in Section 5.3.4 showed that our imputation methods did not significantly affect our findings.

Finally, our initial choice of time window (three input years and three years of onset monitoring) limited the samples that were included from the ROSMAP dataset, biasing our sample against individuals with fewer than six yearly visits. We made this decision based on prior work, which suggested at least one-to-three years of cognitive data are useful for modeling cognitive decline in prodromal dementia patients [47]. We also viewed this as a necessary drawback in order to evaluate whether longitudinal data is needed for accurate prediction.

5.6 CONCLUSION

We conducted an in-depth analysis of many machine learning models, sampling techniques, and usages of time-series data to obtain models that predict imminent dementia onset more accurately than basic demographics-based or single-test approaches and more efficiently than prediction from a full neuropsychological battery. Importantly, we can accurately predict imminent dementia diagnoses using data from just one clinical visit consisting of only demographic information and four easily measured cognitive tests that can be conducted in less than 20 minutes (five times shorter than the standard cognitive battery in the ROSMAP study). By using complex non-linear models and leveraging machine learning interpretability methods, we also generate personalized explanations of risk predictions that account for non-linear and interaction effects. These findings may provide substantial clinical value given the growing aging population and low rates of routine medical assessments. Our method could be scaled to explain and highlight at-risk individuals for additional dementia screenings, preventative treatments (when they become available), and enable planning for a potential imminent diagnosis. Our study takes important steps toward using complex models to generate explainable dementia risk predictions from relatively

cheap metrics. While our findings highlight the effectiveness of our approach, more studies are needed to provide further validation for use in clinical practice. Nevertheless, we provide a framework with which others may replicate our experiments and construct models tailored to other cohorts and their measured cognitive tests.

5.7 SUMMARY

In this project, we explored how selecting and adapting machine learning prediction and explanation systems can support clinicians in leveraging the systems. In order to achieve this, we employed three key strategies: (1) We aligned with user goals by predicting a future dementia diagnosis using minimal, inexpensive data. (2) We retained nuance in explanations by providing biologically sound and personalized explanations. (3) We imparted an appropriate level of trust by showing that both the prediction and explanation systems were consistent with known biological underpinnings of dementia.

Chapter 6. FLEXIBLE SYSTEM FOR EFFICIENT GOAL-DIRECTED SELF-TRACKING ANALYSIS

In this work, I explored how a Bayesian network learning framework might support a variety of self-tracking goals more efficiently than other systems. I also design a reflection interface for the learned network and explore whether the interface retains the network nuance and communicates the skepticism in its explanations. This is a summary of work conducted with Sean Munson and James Fogarty. Qualitative data analysis for the user study was done in collaboration with Shaan Chopra and Yasaman Sefidgar.

6.1 INTRODUCTION

Self-tracking has become widely employed in finance, time management, health and other areas. However, the diverse set of domains, uses, and aims to which self-tracking is applied make it difficult to adequately support individuals in their varied pursuits. Additionally, meaningful quantities of self-tracking data are often difficult to obtain because each data point must come from the everyday experiences of a single person. Automated and semi-automated self-tracking tools have made data collection easier [19,53], but a large individual investment is still necessary.

Several models of personal informatics, self-tracking for the purpose of reflection and gaining knowledge, have been developed to help understand, describe, and scaffold self-tracking journeys [31,61,93]. These models reveal a complex ecosystem (shown in Figure 6.1) of self-tracking activities and stages including starting with individually defined goals, the evolution of goals, reflecting on data, acting on what has been learned, and lapses that occur during tracking.

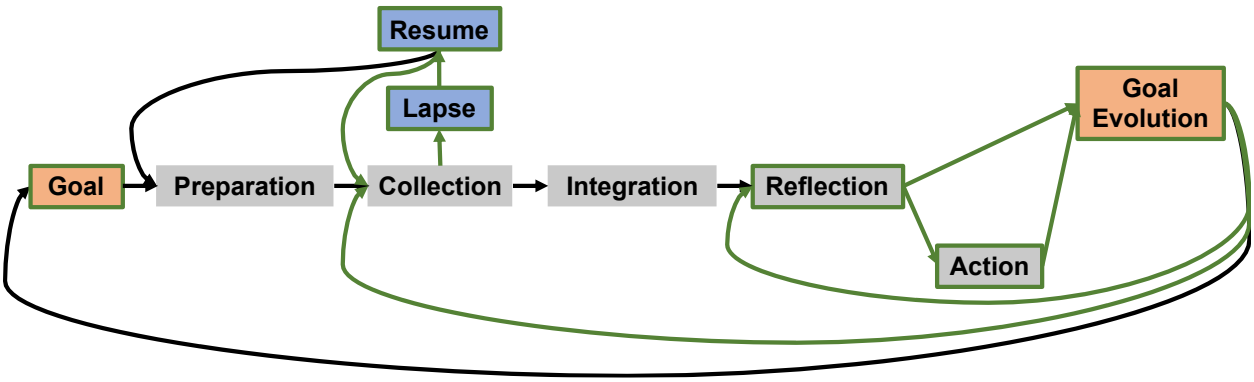

Figure 6.1. An overview of the landscape of the models and scaffolding around self-tracking. Grey boxes indicate Li et al.'s 5 stage model of personal informatics [61]. Blue boxes indicate where Epstein et al.'s lived informatics model [31] added to Li's 5 stage model. Orange boxes are the goal- directed self-tracking scaffolding presented by Schroeder et al. [93]. Green outlines and arrows indicate stages and transition points that the Bayesian network framework and associated reflection interface aim to support.

These models also highlight the potential problems and barriers that can arise within the ecosystem. These problems are particularly prevalent when self-tracking tools are not centered around an individual's goals, especially in n-of-1 studies [70]. Even when someone collects the data they need to answer their questions, they may not have the tools or expertise to conduct the necessary analyses and draw correct inferences. Reflection analyses and tracking regiment designs, even amongst expert self-trackers, commonly lack rigor [20]. Given this diverse ecosystem and the potential for failure in self-tracking when incorrectly or inadequately supported, flexible tools are needed to help people scaffold their diverse self-tracking journeys.

One area of self-tracking that lacks support and scaffolding is supporting reflection, action, and iteration when tracking for the purposes of learning cause-and-effect relationships. This application frequently arises in health contexts because people are often interested in teasing apart what contributes to chronic health condition symptoms, weight fluctuations, and other health related outcomes in order to make informed choices and changes regarding their health and well-being. Bayesian networks have the potential to support this type of tracking and reflection because

they can learn cause-and-effect type relationships, require less data than other artificial intelligence-based solutions, and are inherently interpretable. Additionally, Bayesian analysis techniques better match self-tracking goals and questions than traditional frequentist approaches [92]. Bayesian networks can help tease apart correlations within data sets by modeling each data input as a node and determining the associations present in the data by learning the likelihood of the presence of a directed connection, or arrow, between two nodes in the data.

I designed a Bayesian network framework and reflection interface which supports multiple self-tracking goals and questions, supports goal iteration, is indifferent to many types of lapses in data collection, learns cause-and-effect relationships in a rigorous manner, scaffolds reflection, and supports further action (shown in green outlines and arrows in Figure 6.1). I then conducted a technology probe interview study to see how this framework and interface are able to support people who have pre-existing self-tracked data and questions about the cause-and-effect relationships within that data. I found that the Bayesian network framework and reflection interface were able to support people's needs, allowed for exploration of questions and other learned data relationships, and helped determine potential future actions.

6.2 Related Work

I present related work on self-tracking and its various stages, barriers, usages, and pitfalls. I then present background information on Bayesian analysis and network learning. Finally, I present prior work that has been done to bring Bayesian analysis and network learning into self-tracking contexts.

6.2.1 *Self-Tracking*

Prior work about the process of self-tracking and personal informatics has resulted in several models and ways of scaffolding thinking about the process that people undertake throughout their self-tracking journeys. Li et al. define a 5-stage model (5 stages shown in grey in Figure 6.1) of personal informatics systems which highlights barriers people encounter during each stage [61]. Epstein et al. expand this model in the lived informatics model (expansion from the 5 stage model shown in blue in Figure 6.1), which additionally highlights lapses in tracking, resumption of tracking, and changing goals over the course of tracking [31]. Schroeder et al. provided scaffolding to the models by introducing goal-directed self-tracking (shown in orange in Figure 6.1) where the entire process should be centered around a person's goals and the iteration and evolution of those goals over time [93].

The different models of self-tracking also highlight potential barriers and problems that people might encounter at various stages. Li et al. find that during the reflection stage barriers include challenges presented by sparse data or data that lacks context, data being a poor fit to a person's question, and difficulty in appropriately interpreting or visualizing data. During the action phase one barrier is not having suggestions about specific actions to take [61]. Choe et al. explored the Quantified-Self community, largely considered to be expert self-trackers, and found that their tracking study design, reflections, analyses often lacked scientific rigor [20]. Munson et al. discussed how entire self-tracking regimens can break down and become ineffective when they are not centered around a particular person's goals when conducting n-of-1 studies [70]. Munson et al. focused mainly on design patterns that scaffold what to track and how to support tracking, but did not focus on how to analyze the resulting data [70].

These challenges and barriers have motivated tools that scaffold an end-to-end self-tracking process, effectively guiding people through challenges in the 5-stage model. Several tools support self-experimentation for various health contexts [27,29,50], as well as for generic self-experimentation [28]. Other tools support observational self-tracking: Kim et al. built a tool to support semi-automated self-tracking [53] which helps scaffold the collection and integration stages of self-tracking.

I explore how a Bayesian network framework might better support the following in a single tool: the reflection, action, and lapsing stages; various of goals and their evolution; and a spectrum of ways of tracking ranging from self-experimentation to observation.

6.2.2 *Bayesian Analysis and Network Learning*

Bayesian analysis is a branch of statistics that focuses on predicting the likelihood of outcomes based on priors, or past observed data. Bayesian networks are graphical models that describe probabilistic relationships between different nodes in the network using Bayesian techniques. These networks are defined by nodes and edges, where nodes are inputs to the system representing random variables and edges represent the probabilistic relationship between any two given nodes. The probabilistic relationship indicates the probability of a node taking on various values given that another node has already taken on a specific value. Bayesian network learning is the process of learning the probabilistic relationships between nodes. The input data is a series of observations of the states of the random variables. These observations can take on a variety of different data types: boolean (yes/no), categorical, ordinal (categories with an inherent ordering), and continuous. Individual observations for a random variable can even be unobserved or missing. The confidence in the presence of a probabilistic relationship between two nodes based on the data is known as the strength of the potential edge. If the strength of the arrow is found to be greater than

0.5, the arrow is considered to be present (although the 0.5 threshold can be varied). Bayesian networks are capable of learning complex relationships between nodes, including hierarchical relationships that might appear correlated when using other methods. For example: if getting less sleep causes one to drink caffeine, and drinking caffeine causes one to get headaches, the Bayesian network will model this as an "amount of sleep" node with an arrow to a "caffeine intake" node, which in turn has an arrow to a "headache" node. Since "amount of sleep" and "caffeine intake" occur together in the data, other analysis methods might attribute a headache to "amount of sleep", when the real relationship indicates that "amount of sleep" only indirectly causes headaches. After learning the network and all relationships, likelihood estimations can be made to predict the value of any node(s) given a hypothetical input value(s) of other node(s). These predictions are typically presented as a probability distribution of the values the predicted node may take on [39,57]. There are many implementations of various Bayesian network learning algorithms, one of the more popular ones being the R package bnlearn [94].

Bayesian analysis and network learning have attractive qualities for supporting self-tracking data such as supporting multiple data types, supporting missing data, learning accurate hierarchical relationships, and allowing predictions of the likelihood and size of outcomes. I utilize these qualities to develop a unifying framework to support self-tracking.

6.2.3 *Bayesian Analysis in Self-Tracking*

There has been relatively little work applying Bayesian techniques to self-tracking contexts, despite promising preliminary evidence that it might support self-tracking better than commonly used approaches such as frequentist methods [92]. Schroeder et al. showed that Bayesian analysis can offer better support for and more directly answer nine questions about cause-and-effect relationships commonly asked by self-trackers [92]. The questions break down into five categories

of questions: 1) is there a cause- and-effect relationship, 2) what is the extent of the cause-and-effect relationship, 3) do two or more causes interact to have a greater impact on the cause-and-effect relationship, 4) is there a time delay on experiencing an effect after a cause, and 5) can I predict what effects would be assuming various exposures to causes were. Alemi et al. proposed a Bayesian network framework for self-experimentation which showcased how this type of analysis might support reflection on self-tracked data [2].

Given the potential for breakdowns in the self-tracking journey if goals and goal evolutions are not adequately supported, I explored how a Bayesian network framework might better support these goals and stages while avoiding some common pitfalls and disruptions in self-tracking journeys.

6.3 Framework and Reflection Interface Design

I designed a framework and reflection interface which takes elements of self-tracking, maps them to the Bayesian network environment, and displays the results to users for further reflection. I narrow the scope of this framework and interface to target self- tracking in cases where people are primarily tracking to understand cause-and-effect relationships within their health-related data. I define causes to be anything that may contribute to or result in an effect or a change in effect severity and that can be recorded; I define effects to be anything that affects health or well-being that may be impacted by different causes. I focus on health-related contexts because it is a context where cause-and-effect relationships are widely present, these relationships can vary widely from person to person, and individuals commonly do not know the specific relationships that are true for themselves. This framework and interface are not meant to be a comprehensive account of all the functionalities of Bayesian networks, Bayesian network learning, or self-tracking needs. Instead, the framework seeks to support the reflection, action, goal evolution, and lapsing (shown

in green outlines in Figure 6.1) in order to study the opportunities and challenges present when using a Bayesian network framework to structure these aspects of self-tracking. The framework takes these needs and design challenges identified by prior work and translates them into features, training settings, and data inputs that can be used to learn and interpret a Bayesian network. The reflection interface displays the network and probabilistic relationships between nodes so the results can be explored, interpreted, learned from, and acted upon.

6.3.1 *Framework Design*

The framework is separated into four categories of user-facing functionalities that are supported, along with how that functionality is translated and implemented into the Bayesian network learning algorithm or reflection interface (Table 6.1). Below I discuss how each of the categories and their associated user-facing functionalities may manifest in different self-tracking scenarios.

Table 6.1. Bayesian network learning framework

Category	User-Facing Functionality	Bayesian Network Translation
Different Ways of Tracking	Different data types	Boolean (yes/no), categorical, ordinal (categories with an inherent ordering), and continuous data types are naturally supported so long as any single node has a consistent variable type
	Experiment	All inputs are exactly the same for a specified period of time, while 1 potential cause is varied and effects monitored to gain more confidence in exactly 1 potentially causal relationship
	Quasi-experiment	Continue normal exposure to all other causes but vary 1 potential cause systematically to gain diversity in observations of that cause and associated effects
	Observation	Track as normal, with enough collected, diverse data arcs between causes and effects will be learned
Different Underlying Phenomena	Causes as triggers	Learn arcs from tracked potential causes to tracked potential effects
	Causes as contributors	Create “potential cause” nodes for all possible combinations of tracked potential causes, then model them as individual network nodes

Different Goals/ Different Questions	Observed effect size	Compare predicted probability distribution of different exposures to causes and see how large the difference is
	Interaction effect	Directly compare predicted probability distribution of 2 causes alone and then together to see if the combination results in significantly different effect sizes compared to each cause individually
	Effect prediction	Input hypothetically exposure to causes and observe the predicted probability distribution of effects
	Temporal effect	Compute and create new nodes for time elapsed, time of day, or aggregated data into a time scale of interest
	Quantity/ severity effects	Compare the effect size of different quantities of causes
	Confidence in conclusion	The strength metric of the arc indicates the confidence in the arc's existence and therefore whether or not there is a relationship
Lapsing	Support partial lapse	If some nodes are sporadically not tracked, impute the learned impact based on other data observations
	Support complete lapse and resume	Individual tracking records are considered independent so resumption can begin at any time
Goal Iteration/ Goal Evolution	Prior knowledge	Force an arc to exist (allow list) or not exist (block list) according to whether user knows the relationship between 2 nodes
	New cause/effect	Add new node and relearn relationships either imputing past, untracked values as 0 or based on observed data
	Stop tracking cause/effect	Impute impact based on previously observed data or delete node
	Stop consuming/ doing cause	Automatically impute 0s for that node, or delete node

Different ways of tracking: The system is designed to support different ways of tracking ranging from planned experimentation to observation. This allows people the flexibility to choose whether they are willing to risk experiencing effects to answer questions faster (i.e., experimentation), risk experiencing effects but not undertake the burden of a rigorous controlled self-experiment (i.e., quasi- experiment), or would rather not potentially induce effects and instead learn slower from the natural variation in causes and effects in their day-to-day experiences (i.e.,

observation). Bayesian networks also naturally allow for causes and effects to take on different data types to allow more personalization.

Different underlying phenomena: Many different health conditions multiple types of modeling and tracking underlying health condition phenomena are supported. Two different underlying phenomena are supported: (1) modeling causes as triggers, where a single cause can be enough induce a change in the effect (e.g. in IBS, eating a single type of food or nutrient can cause symptom onset) and (2) modeling causes as contributors, where one cause experienced alone might not be enough to induce a change in the effect, but the sum of several causes may be (e.g. in Migraine, multiple causes may have to compound in order to experience a migraine).

Different goals/conditions: The framework is designed with flexibility to support many different goals and health conditions that people may have. This category includes key types of questions, identified by Schroeder et al. [92], self- trackers might ask about their data which are also supported by Bayesian network analyses. This category also includes a meta-goal of determining how much confidence to place in the answers found for any of the other questions.

Lapsing: Given that self-tracking is often burdensome, people may lapse or stop tracking all or some of the causes and effects and return to tracking them later. In a traditional experimental setup, either a partial or complete lapse would violate the experimental setup assumptions. This framework supports both lapsing and resuming gracefully and can even continue to meaningfully learn through observation in the case of a partial lapse where some, but not all causes and effects, are not tracked for a time.

Goal iteration/evolution: As people build up more understanding about their health condition through lived experience, self-tracking, and/or reflection they have been shown to iterate upon or evolve their goals [93]. The framework supports this by allowing users to enforce prior

knowledge that a cause definitely does or does not impact and effect. It also allows users to begin tracking new causes and effects, stop tracking causes and effects, or note a permanent behavior change in stopping consuming/doing a cause through similar means as supporting tracking lapses.

Each network is modeled as a set of "cause" nodes and a set of "effect" nodes based on the health conditions and goals of each participant. To speed up learning, arrows are restricted to only potentially point from cause nodes to effect nodes and between cause nodes (to assess possible correlations in the input data). Arrows from effect nodes to cause nodes and between effect nodes are not allowed (i.e., put on the block list) because it is assumed that effects should be independent and an effect should never induce a cause. The Bayesian network learning algorithm is implemented in the R package bnlearn [94] using the structural EM algorithm with the hill-climbing optimization function.

This framework provides a means for encoding people's self- tracking needs into a Bayesian network learning algorithm. Once the learning algorithm has been run, the network and associated metrics must be translated back into something that can be understood, explored, and learned from by an end user.

6.3.2 *Reflection Interface Design*

In order to explore the learned network, individuals must have some way of reviewing and interpreting the results. Ideally, this interpretation should be done by the individual in order to build trust in the results and because they are the expert of their own data. This can help because individuals bring knowledge of how their data was tracked, potential confounders, and background knowledge about the specifics of how their condition affects them. However, Bayesian networks and associated properties are notoriously difficult to understand for people who do not have some amount of formal training. Additionally, reflection should support exploration of the entire learned

network, not just specific aspects that answer a single question, because people's questions are often overlapping or iterative and can be answered immediately upon further inspection. Displaying a more holistic view of the network may also help people avoid a confirmation bias in only looking at and believing in conclusions that they already believe. In order to facilitate user-driven reflection and learning, I designed a reflection interface to pair with the Bayesian network framework that has the flexibility to answer many potential questions, can highlight specific aspects for in depth analysis of a single question, and also shows a holistic view of the learned network.

The desktop app interface, shown in Figure 6.2, is implemented in R using the Shiny package and consists of two tabs: a "Scenarios" tab (left) for investigating individual questions via simulated predictions of effects experienced based on different exposures to causes, and an "Overview" tab (right) consisting of a graph of the entire learned network, suggestions of scenarios to look at based on the entire network, and various settings to incorporate prior knowledge.

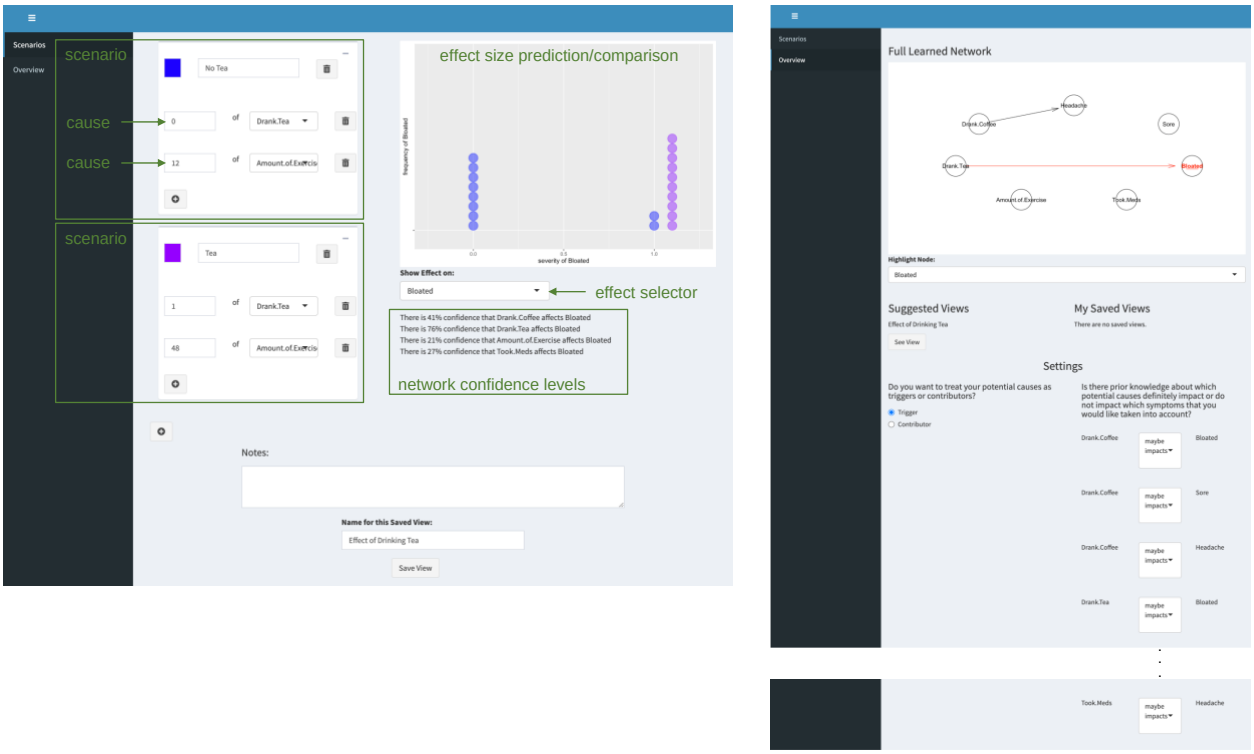

Figure 6.2. The reflection interface with data from P01, however node (i.e., cause and effect) names have been changed for anonymity. The left image is an annotated picture of the Scenarios tab (annotations appear in green) and the right image is a picture of the Overview tab.

Scenarios Tab: This allows the user to explore specifics about the learned network without having to parse through all the details of the full network. On the left side, each white box represents a potential scenario of different causes that an individual might experience. The user specifies which cause(s), and how much of each "cause" they experience in each given scenario. The user also selects the effect nodes on which they would like to see the impact of the scenarios using the drop-down menu on the right side. The network then predicts the probability distribution of the effects the individual might see for each scenario. The probability distribution is displayed in the graph at the top right using quantile dotplots [52]. This graph then allows the user to compare the effect size prediction between the different specified scenarios, shown on the same axis and differentiated by colors specified in each scenario box on the left. Below the effect selector is a notation of the confidence level of a potential arrow between each cause node and the selected

effect node to give the individual an indication of how confident they should be in the above dotplot, based on the full learned network. Much less than 50% indicates that there is high confidence there is no arrow from the cause to the effect, much greater than 50% indicates that there is high confidence there is an arrow from the cause to the effect, and near 50% means the network is unsure of whether there is an arrow from the cause to the effect or not. There is also functionality for users to take notes about any particular set of scenarios and to save the settings and view of the entire tab to be viewed later (accessed from the Overview tab).

Overview Tab: This shows the entire learned network including all nodes that correspond to each column in the data. An arrow between nodes will be present if there is greater than a 50% confidence that one node impacts another node. Below the network diagram is an option to highlight a single node and all the incoming and outgoing arrows from that node in red (currently the "Bloated" effect node is highlighted). Below the learned network are both suggested and saved views. Suggested views are instances of the Scenarios tab that are manually created based on various participant stated goals and the learned network. Saved views are instances of the Scenarios tab that the participant has saved for later viewing. Clicking the "See View" button associated with any of the views in this section will take the user to the Scenarios tab and automatically populate all the fields with associated view. Below the views are settings, which correspond to different user-facing functionalities defined in the toolkit in Table 6.1. On the left side is the option to model the causes as either triggers or contributors. On the right is the option to include prior knowledge or hypotheses into the Bayesian network assumptions for all of the possible cause to effect linkages (cropped and indicated by ellipsis in Figure 6.2). The default is "maybe impacts" which will allow the network to learn whether there is an arrow from the cause to the effect. Selecting "definitely impacts" will enforce the presence of an arrow (i.e., allow list)

while selecting “definitely does not impact” will enforce the absence of an arrow (i.e., block list). Changing any of the selections in the Settings section will automatically trigger the learning algorithm to run again with the updated set of learning parameters.

6.4 TECHNOLOGY PROBE STUDY

I conducted a technology probe study with 8 participants to gain further understanding into how the Bayesian network framework and reflection interface may help people in-situ with answering their own questions based on previously self-tracked data, as well as assess the potential challenges and other opportunities the framework may bring to light.

6.4.1 *Methods*

The technology probe consists of two hour-long semi structured interviews with each participant: one background interview and one technology probe interview. All interviews were conducted over Zoom with audio and video recorded. This study was approved by the University of Washington IRB.

Recruiting: Participants were recruited through a health study recruitment board, quantified-self group message boards, social media groups, and through the personal networks of 3 researchers involved in the preparation of this study. Participants were required to have previously tracked data that included both potential causes and effects for a health condition. We decided to recruit participants who had already tracked data to evaluate what their needs, goals, and challenges were when their tracking was not specifically scaffolded for this study. People with any health condition was eligible to participate. The only exclusion was people whose tracking focus was on a mental health condition, who were excluded under the IRB approval. Individuals who had mental health conditions, but the mental health condition was not the primary focus of

tracking, could participate. Participants also were required to have some method of access to their tracked data. This could have been through written notes, spreadsheets, apps that allowed data to be downloaded or transcribed, or other methods that could be shared with the research team. After verifying participant eligibility, we asked them to share their tracked data with the research team. The participant was given the option to withhold some of their tracked data from the research team for any reason.

Background (1st) interview: This interview had two goals: (1) to gain an understanding of the participant's experience with self- tracking and (2) to go through their tracked data that would be used by the framework and interface. To achieve the first goal, approximately 40 minutes of the interview consisted of the following background questions, each followed by further probing questions as needed: Why they tracked, what they tracked, what they learned from self-tracking, how they had learned through self-tracking, if/what they had used to reflect on the data, what they still wanted to learn from self-tracking, if/how their tracking had changed over time, and what made tracking easier or more difficult for them. To achieve the second goal, the interviewer and participant looked through the participant's tracked data together (either via screen share or synchronously on separate computers). Any data entry that was not clear was clarified, as well as which data fields corresponded to causes and which corresponded to effects.

Data integration: In between the first and second interviews, I took the participant's data and formatted it into an Excel spreadsheet to be fed into the Bayesian network learning algorithm and interface. The formatting consisted of creating a column for each cause and effect, recording each tracking instance (typically one day) in a row with associated cause and effect data in the relevant columns. Distinctions between cause and effect nodes were determined based on information about the particular health condition and different goals provided by the participant in their first

interview. Data was also formatted into the correct types in order to answer the questions the participant discussed in the first interview (e.g., P1 tracked how many ounces of tea they drank each day but was only interested in whether or not drinking any tea affected chronic health condition symptoms. The ounces of tea column was therefore translated into a boolean (yes/no) column where if the number of ounces was greater than zero, the number was entered as a "yes" for did drink tea that day.) If there was the potential for temporal effects in the data, additional cause columns were added to answer those questions or to better scaffold answers (e.g., P3 tracked their amount of core nutrients consumed every day and wanted to know what effected their weight. Cause columns were created to compute the average amount of a particular core nutrient consumed over the previous 7 days to account for the delays and fluctuations seen in weight.) This spreadsheet was then read into R, run through the Bayesian network learning algorithm, and displayed in the reflection interface. After this was complete, I then manually explored each participant's results and developed at least 1 suggested view for each participant based on their goals and results. This was meant to help the participants get familiar with the tool and ground participants who might be confused by the tool, but not to be exhaustive such that there was nothing further for the participant to explore using the tool.

Technology probe (2nd) interview: In this interview the participant was shown the reflection interface pre-prepared with their data and was asked to explore to tool and provide feedback on the tool. The interviewer shared their screen with the participant's data in the reflection interface and gave the participant a basic overview and explanation of the interface. First the scenarios tab was explained, then the suggested scenarios were shown and the first suggested scenario from the overview was shown. Next, the full network diagram from the Overview tab was explained, and finally the settings were explained. The interviewer then gave the participant control of the screen

and mouse and asked the participant to explore the interface while talking aloud about what they were thinking and looking at. The interviewer would question further whenever a participant seemed to discover something new, started exploring a new question or piece of data, or when the participant appeared confused. Further questioning included but was not limited to asking how the participant arrived at a conclusion, why the participant decided to look at a certain thing, and what the participant was trying to accomplish. If the participant had questions about how the interface worked, they were encouraged to try to find the answer themselves. If the participant was still confused or found the wrong answer they were prompted or answered by the interviewer so as to not get too hung up on any one aspect of the interface. Approximately the last 10 minutes of the second interview were reserved for semi structured questions soliciting feedback on the overall interface and experience including: if/what conclusions they derived from the interface, if they were going to change anything about their tracking or exposure to causes based on the conclusions, if/how often they might use this tool on their own, what challenges they might expect if they were to use the tool on their own, how this reflection method compared to previous methods of reflection they had used, three things they liked about the interface, and three things they disliked about the interface.

Data analysis: Automatic audio transcriptions were generated by Zoom and were reviewed for correctness. The data is currently being analyzed and will be written up and submitted after the conclusion of the final analyses. Key insights and findings from the interviews are presented in the results and discussion sections below.

6.4.2 *Results*

The interviews highlight the overall need and appreciation for a many of the functionalities the toolkit and interface provide. They also provide evidence that users have very different ways

of exploring, answering questions, drawing conclusions, and iterating on goals using the reflection interface. These support the hypothesis that flexible tools are needed in order for people to adequately use, explore, and understand self-tracking analysis and reflection tools. However, users also encountered several misconceptions about the underlying network and conclusions that could be drawn, suggesting the need for more scaffolding of the reflection process. Below I present findings from the interviews conducted two participants: P1 and P3.

Overall needs and reactions: Overall, participants were very excited about the reflection interface and the possibilities for using it. In particular they expressed appreciation for a method of reflection that was not just basic associations between causes and effects, which are hard to trust because of correlations, or frequency of causes or effects over time, which motivate continuing a change but do not inform potential future changes to either behavior or tracking. P1 summarized this as: *"This is awesome. Just the fact that this can sort of isolate you know, sort of remove the effects of exercise and look only at coffee is huge. Right now I can't really get any causation from my current ways of tracking that much because the visualizations just plot a bunch of steps with a lot of like moving seven day averages, moving 30 day averages, that type of thing. Which is really great when you're just looking to strictly increase or decrease a behavior because it's very rewarding see, you know, exercise has increased. But it's very hard to get causation, any kind of something affecting something else, from that."* P1 also talked about how the interface relieved the mental burden trying to be more vigilant about tracking via self-experimentation while motivating them to keep tracking via observation: *"I feel like it's very hard for me to plan the controlled experiment in my head. I just like the idea of trying to figure out how to control for all of these things. It [is] so useful to be able to just live my life and know that in living my life I would get enough data points out takes a ton of stress off of me. To be honest, this gives me a lot more faith*

that tracking is actually useful." P3 noted that they have been looking for this type of reflection aid for years and that their current data reflection methods would only reveal something that had a huge effect size: *"[The interface] is mind blowing. Because it's what I've been after for so long. Because [with my current methodology] I can kind of see over a longer period of time how my month ends looked and it really is so diluted that something would have to be so catastrophic for it to show up."*

Exploration and answering questions: After being given a tour of the reflection interface, P1 and P3 each took very different approaches to exploring the interface. P1 spent most of their time exploring various scenarios in the Scenarios tab, mostly exploring answers to their existing questions about their data. They also relied on the confidence notes on the Scenarios tab and occasional looks at the full network on the Overview tab to determine what scenarios might be fruitful to explore. P3 on the other hand spent most of their time exploring the Overview tab and highlighting various nodes to see the learned connections between nodes. P3 started by exploring nodes around which their initial questions were based (highlighting the "weight" and "fat percent" nodes to find potential areas for weight loss) but quickly became interested in getting an overall understanding of the network and not a detailed look at parts of the network. This difference in approaches could be because of personal preference: P3 liked the overall view because it was "easier" to understand than the scenarios. Or it could be due to the nature of their questions: P1 started with more detailed questions about their data and health condition (i.e., does drinking tea make my symptoms worse?) compared to P3's initial questions (i.e., what affects my fat weight?). It could also be because of the differences in the density participants' data and learned network: P1's network had only 7 nodes with 2 arrows while P3 had 29 nodes that were highly interconnected.

Both P1 and P3 were able to find answers to their questions using the reflection interface, as well as discover some unexpected associations in their data. They also both discussed changes they might make to their behaviors and tracking routines based on the reflection interface. P1 wanted to know how water, coffee, and tea intake affected their symptoms. They previously suspected water and coffee to affect symptoms, and was surprised to find that the reflection interface was only highly confident that tea had a impact on symptoms*: "I was kind of surprised that we didn't see a bigger correlation between like coffee and dizziness and I was very surprised by the tea and dizziness thing."* When the reflection interface disagreed with P1's prior hypotheses, they tended to have more skepticism towards the results by suspecting it might be due to lack of variability in the underlying data: *"Okay, in theory, I would have expected to see like a decrease in my dizziness with an increase in water, but again I really feel like that's probably because there was maybe like two days with 48 ounces of water so I give the model a break on that one."* P3 looked at what affected their weight and found both expected and surprising association: *"Well, in this case pretty much everything affects my weight. Except for, I mean it almost looks like sleep doesn't. And not stress, which is a little surprising. Nor does exercise time. Mostly the things that affect my weight are nutritional. What's going in my mouth, rather than physically what I'm doing. And I guess that's a truth that any coach will tell you: weight loss happens in the kitchen. Or weight gain happens in the kitchen. So yeah it's cool to see those things kind of proven out."*

Determining actions: Both P1 and P3 were able to find actionable next steps within the reflection interface. P1's potential actions were to try a quasi-experiment method to collect more varied data, particularly for tea and ounces of water, to confirm or disprove the findings in the tool: *"I mean it'd be interested to see [more about] this tea thing. I'm not trying to call that a conclusion yet, I think that's something that I would need to experiment with more. And just having this*

[interface] at my disposal, I would also try to encourage myself to drink 48 ounces of water more. I mean ideally I'd rather try 32 verses 64 and see if that makes a difference." P3 discussed how they could use this interface to inform a behavior change, but would probably only do so if they felt a change was necessary: *"I would be willing to entertain the notion of change if there was, you know, indication of a reason for doing so."*

Potential misunderstandings: The technology probe also revealed the potential for misunderstanding the results of the tool, particularly in the exact meaning of the presence or absence of an arrow and in the potential for indirect relationships between causes and effects. At various points during the interview, both P1 and P3 asked what the precise meaning of an arrow between nodes in the graph. While both ended up with a good grasp that an edge meant there one node had an effect on another node, this had to be explained and was immediately intuitive to them. This could have become problematic if misunderstandings had persisted and led to unsupported conclusions. For example, P3 had a tendency to interpret an arrow as one node having a positive correlation on another, when the direction (positive or negative) is not inherently specified by the arrow, and instead would need to be unearthed using different scenarios. P3 also had difficulties observing when a cause might be indirectly impacting an effect (i.e., if sleep affects caffeine intake which affects headaches, sleep then indirectly affects headaches). This was particularly a problem for P3 because their learned network was so dense with both nodes and arrows. This could become problematic when trying to determine a change to make, because the full picture of what impacts the effect is not seen. One potential solution to this would be to restructure the network graph to have more of a tree structure instead of a circular structure, so that this type of "flow" would be more easily visualized. It could also be addressed by using another color to highlight second order nodes, thus bringing more attention to their presence.

6.5 DISCUSSION

Throughout the technology probe and data integration into the interface, we of the initial findings were not directly related to the success of the framework and reflection tool but had broader implications for the ecosystem of self-tracking tools. These centered around how people are able to design their goals and determine what data was needed to answer those goals, the need for raw data analysis, and the potential role of confirmation bias in interpretation of results.

Designing goals and tracking regimes: While all of the participants came into the study with goals, none had thought about specific questions (i.e., does drinking coffee cause me to have more symptomatic days? Does drinking coffee cause my symptoms severity to be more extreme?) and how exactly they would try to answer those specific questions. The lack of specificity in questions can be problematic because it can cause people to track more data than is necessary to answer their specific questions. This was particularly evident in the data they had collected. Prior work has shown that self-tracking data is often arduous and has highlighted the need to make tracking easier, particularly by tracking exactly and only what is necessary to answer people's questions [93]. Our participants collected several data "nodes" that were unnecessary to answer their questions or had collected data "nodes" in a more challenging way than was necessary. For example: P1 tracked the number of ounces of tea and coffee that they drank every day, despite only needing to have tracked whether or not they drank tea or coffee that day in order to answer their stated cause-and-effect questions. P1 even mentioned this as a concern when discussing potential challenges with using the framework and interface on their own: *"My biggest concern would be with setting it up. So explaining that coffee is a boolean and here's the format."* In our study design, I mitigated these barriers by being the translator for (1) participant goals or questions as

explained in the first interview and (2) transforming existing participant-tracked data into data types and aggregations that would meaningfully answer their questions.

Analyzing and reflecting on raw data: While the Bayesian network framework was helpful to untangle more complex cause-and-effect relationships, not all questions and goals in self-tracking require such a complex analysis tool. In this study, the focus was specifically on questions that can be answered well by a Bayesian network, however future work should look at how to incorporate and utilize both raw data analysis and Bayesian network analysis in self-tracking tools.

Potential for confirmation bias: As with most instances of data analysis and learning from data, there is a significant possibility that the reflection interface will be used mostly to confirm prior suspicions because of confirmation bias. I attempted to mitigate the potential for confirmation bias in the reflection interface by including the confidence notes in the Scenarios tab and presenting a graph of the full learned network in the Overview tab, however vigilance and further design work is still needed to help people avoid traps of confirmation bias when learning from and interpreting results.

6.5.1 *Limitations and Future Work*

The study inclusion criteria required participants to have not only tracked data in the past, but also for it to potential contain cause-and-effect relationships. While this was a benefit to evaluate how well the Bayesian network framework supported people's existing questions and data, or at least a subset of those questions, it also excluded a lot of people who could potentially derive value from the existence of this framework and reflection interface. It also means that the participants we recruited were already fairly savvy at tracking and evaluating their data, and therefore participants might have an easier time understanding the reflection interface than a novice self-

tracker would. Nonetheless, the Bayesian network framing and reflection shows promise for helping at least some people explore and learn from their data.

The Bayesian network learning algorithm also did not utilize learning priors on the effects or network structures. The use of priors is widely cited in the Bayesian network literature as being extremely helpful for learning faster and more robust conclusions. I chose not to incorporate priors into the framework or learning algorithm in favor of supporting a diverse set of health conditions during this preliminary work. In order to speed up the learning, I did allow participants to specify if they knew there was or was not a relationship between nodes, but this could have been supplemented with incorporating priors for connections between nodes that were still being investigated. Future work should investigate how the use of priors might impact and help this framework within specific health conditions. When investigating a Bayesian network framework within a specific health condition, many aspects of the framework and reflection interface I presented might also be simplified based on the knowledge of underlying health condition. For example, a condition might be known to have causes as contributors (instead of triggers) in which case the framework and interface for that condition could be simplified. Further work is needed to determine the exact needs and specifications for different health conditions.

Finally, initial qualitative data analysis has highlighted specific improvements that could be made to the framework and reflection interface. P1 suggested having histograms of the distribution of data points in their underlying data for each cause and effect. The participant mentioned how this would have helped them know how much confidence to put in conclusions based on scenarios that might be based on limited numbers of data points. Knowing conclusions were based on only a few data points could also have helped this participant know if they should collect more observations with varied amounts of exposure to a particular cause. This could be done either with

a new tab that lists only basic statistics about the participant's data (which might also help to support more goals and questions that are better answered by raw data alone) or with an additional graphic in the Scenarios tab. P3 discussed that the designation between causes and effects might be unnecessary for certain conditions. Since P3 tracked and was interested in general health and nutrition, nodes that might be considered causes for the purposes of one question might be considered effects for the purposes of another goal. From the perspective of the Bayesian network framework, there is no reason why nodes must be classified into the two rigid bins, other than to speed up learning and potentially restrict arrows which would not make sense in some health condition contexts. This should be further explored, particularly in health conditions where the causes and effects are clearly separable from one another.

6.6 CONCLUSION

I designed and built a Bayesian network framework and reflection interface which unifies and supports many self-tracking goals, goal iterations, ways of tracking, underlying health condition phenomena, and potential tracking lapses. A technology probe user study showed that this framework and interface supported people's needs, helped them explore and answer question about their data, and determine actionable ways of iterating on their questions and goals while also highlighting areas for future expansion, development, and refinement of both the framework and interface. This shows progress towards creating a flexible system to support the complex ecosystem of self-tracking and reducing the associated barriers which prevent people from reflecting on, learning from, or even collecting self-tracked data.

6.7 SUMMARY

In this project, I explored how selecting and adapting machine learning prediction and explanation systems can support patients in leveraging the systems. In order to achieve this, I employed three key strategies: (1) I aligned with user goals by matching the prediction system objective function to common, existing self-tracking goals. (2) I retained nuance in explanations by showing holistic visualizations to help users answer questions they did not initially have. (3) I imparted an appropriate level of trust by showing confidence metrics and uncertainty visualizations of the outcomes.

Chapter 7. DISCUSSION

The four projects in this dissertation have explored selecting and adapting machine learning prediction and explanation systems in highly contextualized application domains, while the prior work discussed in Section 2.1 on three human-AI interaction frameworks sets forth more theoretical frameworks for designing these systems [4,5,36]. The two bodies of work offer complementary perspectives on the intricacies of human-AI interaction. I now discuss how the lessons learned from these projects compare and contrast to those three frameworks, in particular the importance of aligning with user goals, retaining nuance in explanations, and imparting an appropriate level of trust.

7.1 Aligning with User Goals

In each of the context-driven projects that I have discussed, selecting the machine learning solution to match the user's needs and end goals was a central focus and was crucial to the success of each project. However, only Google's human-AI guidelines mention these initial considerations. Indeed, Wright et. al.'s systematic review of all three sets of guidelines found that the guidelines classified as "Initial Considerations" represented the smallest proportion of overall guidelines, and most of those were divided into sub-categories of "fairness" and "privacy", with only five of Google's falling into the sub-category of "value of AI" [105]. It is impossible to tell if the omission by Microsoft and Apple is due to oversight, a feeling that designing with a user in mind is a standard that is assumed, or another reason. However, my work showcases that aligning with user goals is more intricate and nuanced when designing systems that need to be used in collaboration with people of varying expertise.

Google's human-AI guidelines begin with the category of "User Needs and Defining Success" within which they stress three distinct considerations. The first consideration is to find the intersection of user needs and AI strengths in which they stress the need to decide if and what the unique value AI is going to bring to the user and how it will effectively deliver that value. In each of my projects, this single point has been a central focus point. In Chapter 3 (Problems and an Alternative to Single Explanation Aggregations), I identified how users were employing an existing machine learning visualization tool and redesigned that tool to better suit the needs of users. This allowed experienced machine learning practitioners and data scientists to complete their regular tasks more efficiently and accurately using the updated tool design. Chapter 4 (Predicting Blood Glucose Test Accuracy in ICU Patients) showed how machine learning can be utilized to provide evidence that a machine learning system might not sufficiently meet prediction accuracy and underlying data signal needs and should therefore not be deployed in a hospital setting. In Chapter 5 (Predicting and Explaining an Imminent Dementia Diagnosis with Limited Data) the choice of both the machine learning system and the explanation system allowed us to produce a prediction system that retained known biological underpinnings of dementia so that explanations could be medically relevant. The choice of explanation system to explain individual samples meant we were able to demonstrate the use case for utilizing this system in individual patient's care. This would not have been possible had we not considered the end user's needs and what added value the machine learning system needed to provide from the outset of system design. Finally, Chapter 6 (Flexible System for Efficient Goal-Directed Self-Tracking Analysis) took user questions over time as a central design tenant and adjusted a machine learning system to match those as close as possible. This enabled users to directly answer their questions using the system instead of answering only tangential questions and then trying to translate those into questionably

accurate answers to their actual questions. It also allowed users to learn and grow in their understanding with assistance from the machine learning system instead of outgrowing the system upon answering a single question and finding no support for evolving questions. In each of these projects, matching user needs to AI's strengths, as Google suggests, was a primary goal and ensured the success of each of the projects.

The second consideration Google mentions when considering user needs and defining success is to assess whether automation or augmentation is more appropriate for the user's needs. In my projects where a machine learning system was produced, I focused entirely on augmentation. In Chapter 5 (Predicting and Explaining an Imminent Dementia Diagnosis with Limited Data) and Chapter 6 (Flexible System for Efficient Goal-Directed Self-Tracking Analysis) this was because while a prediction and explanation was the output of the machine learning systems, the final action was then to use this prediction to determine the next medical steps to take. Focusing on this as the endpoint instead of the prediction itself aided in the successful design of both machine learning prediction and explanation systems. Chapter 3 (Problems and an Alternative to Single Explanation Aggregations) also focused on augmentation, since its primary goal was to provide a better alternative to an explanation system to better facilitate routine data science tasks. While my projects have focused on augmentation, augmentation versus automation is an important consideration, especially considering most machine learning systems are seen at first as automation systems instead of augmentation systems.

The third consideration Google puts forth is to design and evaluate a reward function. They discuss the importance of discussing the choice of reward function and evaluation with various stakeholders including UX, product, and engineering individuals in order to avoid "optimizing for the wrong outcomes". Their practical examples, however, focus more on evaluating false positives

and negatives, evaluating the precision and recall tradeoffs, and accounting and planning how to handle negative impacts as the occur. While each of these pieces of practical advice are important, Google does not discuss in depth about how to consider various rewards functions to as closely as possible match user expectations and questions as a way of avoiding misunderstandings and wrong outcomes. My projects, in particular Chapter 6 (Flexible System for Efficient Goal-Directed Self-Tracking Analysis), showed that choosing an objective function that most closely matches user's existing needs, questions, and mental models can help prevent misunderstandings of what the system output means. Often machine learning systems answer a question tangential to a user's underlying question, which then requires further explanation of the limitations of the system. By matching as close as possible the question the user is asking and the question the machine learning system answers, the system and explanations are both more efficient and accurate.

One of the major advantages of closely matching the machine learning prediction and explanation system to user needs from the outset is that it helps to align a user's existing mental model of the problem, data, output, and explanation to what the machine learning system produces. However, when discussing mental models of the user, only Google's framework discusses setting expectations by "identifying existing mental models" of the user. All three human-AI frameworks discuss how to create and change the user's mental model to become closer to that of the machine learning system and how the machine learning system should learn from and adapt to the user's needs slowly over time. This is good advice for systems that are automation systems or systems running mostly in the background. In the case of each of my projects, the systems were augmentation systems that were, from the outset, highly reliant on input from experts in fields other than machine learning and data science. This resulted in a steep learning curve, which made matching the prediction and explanation systems to the user's existing mental model from the very

beginning, as Google suggests, much more important and valuable. This is also particularly important when users with different expertise need to understand the system in order to correctly utilize it and collaborate with it, but where they do not want to become an expert in the system (i.e., want to use the system only to explore a question and then move on to their actual area of expertise).

The four research projects presented in this dissertation each focus on considering and closely matching user needs from the outset of the project, as well as throughout the design and iteration of those machine learning prediction and explanation systems. While Google's human-AI framework discusses considering user needs from the design stage, neither Microsoft nor Apple's guidelines discuss user needs until the output explanation and system iteration stages. The benefits gained in each these projects by considering user goals early and often suggest that Microsoft and Apple's frameworks could benefit from further guidance on aligning with user goals, particularly earlier in the design and model choice stage of machine learning system development.

7.2 Retaining Nuance in Explanations

In three of the projects I have presented, having nuanced and accurate explanations was crucial for the machine learning prediction and explanation systems to be usable by the various end-user experts. The user population of domain experts is a very different user population than those often designed for by Microsoft, Apple, and Google for their consumer products. These consumer product users are often not utilizing machine learning systems to explore questions in their field of expertise. They are instead looking to automate menial tasks, augment an existing user experience, or recommend new things. As such, the three human-AI interaction frameworks differ in their recommendations from what I have found to be useful in the context of domain expert users. In Microsoft's "during interaction" category, they recommend "showing contextually

relevant information". That is, only showing information that is relevant to their current task. Apple presents a category called "attribution" in which they present two relevant guidelines: to "avoid being too specific or too general" while also "avoiding technical or statistical jargon".

These guidelines seem useful in the context of lessons learned from Chapter 3 (Problems and an Alternative to Single Explanation Aggregations), Chapter 5 (Predicting and Explaining an Imminent Dementia Diagnosis with Limited Data), and Chapter 6 (Flexible System for Efficient Goal-Directed Self-Tracking Analysis). In Chapter 3 (Problems and an Alternative to Single Explanation Aggregations), all of the information that was presented in the multiple aggregated rankings was present in more detailed, low level visualizations. However, those low-level visualizations were too numerous and too time consuming to look through and quickly find relevant information when in the beginning stages of many data science tasks. Chapter 5 (Predicting and Explaining an Imminent Dementia Diagnosis with Limited Data) presented example explanations for individual patients, separate from explanations of the system as a whole. This is so that in the context of an individual patient, a provider can look only at the relevant explanation for that particular individual instead of wading through explanations of the entire patient population to find necessary and relevant information. In Chapter 6 (Flexible System for Efficient Goal-Directed Self-Tracking Analysis), it was necessary to abstract away the underlying Bayesian network learning algorithm and instead show various interactive visualizations of the outputs because Bayesian statistics has proved to be extremely difficult to understand quickly without a lot of background knowledge.

However, showing only contextually relevant information and riding the line between being too specific and too general can also result in designing a system whose use cases are too narrow. Users often co-opt systems when they have a new need or come up with a new way of using the

system. It is imperative, therefore, that systems allow flexibility for the tool to be used for different, evolving, and often unplanned for needs. If they do not, users may try to use the tool for a different purpose but not have the information present to properly use the system, which can lead to misuse and misunderstanding. We found this to be the case in Chapter 3 (Problems and an Alternative to Single Explanation Aggregations) when users co-opted the single aggregated ranking for a host of tasks that it was not nuanced enough to support. This led to inefficient use, misunderstandings, and misplaced trust compared to our intervention of a suite of aggregated rankings designed for multiple, flexible use cases. The explanation system in Chapter 5 (Predicting and Explaining an Imminent Dementia Diagnosis with Limited Data) needed to be chosen and adapted so that it would retain biologically relevant, nuanced explanations even with a small set of input features. Similarly, the success of the reflection interface presented in Chapter 6 (Flexible System for Efficient Goal-Directed Self-Tracking Analysis) was due to the explanations being nuanced and flexible to support individual's evolving understanding and goals. Explanations must be contextualized enough such that they allow users to efficiently use the system but nuanced enough to allow for changing goals and a full understanding of the prediction and explanation systems.

Microsoft's guidelines also suggest "making clear what the system did what it did" in the context of when the system is wrong. Given the context of the three projects discussed above, this guideline seems incredibly important, however it is unnecessarily narrowly scoped to only the case where the system is wrong. Through nuanced explanations in both correct and incorrect cases, a machine learning system can be utilized to help make further decisions based on the output and to help understand how the system works and what it has learned.

Google's guidelines around explanations focus on the intersection of explainability and trust. The guideline specifically about explanations recommends to "optimize for understanding" which

includes further guidance to “describe the system”, “explain the output”, present “example-based explanations”, and have “explanation via interaction”. Given the experiences from three projects discussed above, this guidance seems to be the most relevant and specific when implementing complex, domain specific machine learning systems. The more trust focused guidelines are discussed in the next section.

7.3 Imparting an Appropriate Level of Trust

The projects presented in Chapter 4 (Predicting Blood Glucose Test Accuracy in ICU Patients) and Chapter 6 (Flexible System for Efficient Goal-Directed Self-Tracking Analysis) rely on explanations, confidence metrics, and system performance metrics to impart appropriate levels of trust upon users of these systems. Chapter 3 (Problems and an Alternative to Single Explanation Aggregations) relied solely on explanations because the system explored is entirely explanatory, not predictive. These systems of trust need to strike a balance between gaining trust so that the user willing to believe the system’s predictions and explanations while also imparting a sense of skepticism in places where the system is less accurate or there is ambiguity in the results. I will now discuss how these three facets, trust through confidence metrics, trust through explanations, imparting skepticism where appropriate appear in the three human-AI frameworks.

In Google’s “explainability and trust” section of guidelines, there is a subsection on “managing influence on user decisions”. Google suggests that you should only show confidence metrics where they will help the user gain a better understanding of the system’s capabilities because otherwise these metrics might confuse the user or be unhelpful in determining how to utilize the information. Apple similarly offers guidance in their “confidence” section about choosing carefully how to present confidence metrics so that they are easily and naturally understood by users. Microsoft offers the guideline to “make clear how well the system can do

what it can do" in their "initial" guideline section, but does not specify a way of achieving this. In Chapter 4 (Predicting Blood Glucose Test Accuracy in ICU Patients) we presented prediction system performance metrics alongside baselines for random guessing to calibrate user expectations and show that the system did not perform better than chance. We also chose to present traditional statistical analyses of variable correlations because this type of statistics is more familiar to the clinician expert population we were working with as end users. In Chapter 6 (Flexible System for Efficient Goal-Directed Self-Tracking Analysis) we showed only confidence percentages for the possible links in the Bayesian network graph in order to help users decide how much trust they should put into any "cause" and "effect" node link and analysis. The guidelines set forth, particularly by Apple and Google, seem helpful and insightful in the context of my projects because they focus on the why and how of showing confidence levels, which I also found to be extremely important when working with different expert user groups.

When discussing establishing trust through explanations, Apple mentions confidence values can be translated and conveyed using concepts that people already understand and gives an example of explanations of model decisions as a way of doing this. As discussed above, Microsoft does not give specific guidance on how best to convey trust in the system, only to make sure it is present. Google goes the furthest in discussing the link between explanations and user trust in the machine learning system by discussing ways of "**planning for trust calibration throughout the product experience"**, "optimizing for understanding", and "managing influence on user decisions" with suggestions in each of these subsections around explanation visualizations as a way of completing these objectives. During the Chapter 3 (Problems and an Alternative to Single Explanation Aggregations) work, we presented multiple explanations as a way of calibrating trust to an appropriate level. In Chapter 4 (Predicting Blood Glucose Test Accuracy in ICU Patients)

we used visualizations of the machine learning tree models to show that the model was not making reliable splits and decisions based on known medical phenomenon and instead potentially on noise within the data. In Chapter 6 (Flexible System for Efficient Goal-Directed Self-Tracking Analysis) I showed dotplot visualizations to show the uncertainty and distribution present in different predictions of how a "cause" might impact and "effect" on a sampling of data points. These visualizations gave users both greater insight into what the machine learning system had learned, but also imparted skepticism in the systems where necessary.

When it comes to limiting undue trust in a machine learning system, Apple suggests "avoiding showing results when confidence is low". This is a good suggestion when presenting a ranked-based machine learning system or not showing any of the machine learning system at all is an option. However, in my projects the machine learning system, and its limitations, were central takeaways that needed to be conveyed. Google recommends "managing influence on user decisions" and suggests doing so through confidence metrics, visualizations of confidence intervals, and presenting alternative results. This guideline comes the closest to describing the need to show users where the system might be incorrect and need to be examined further before making a decision. It is possible that the concept of limiting undo trust comes up less in the human-AI frameworks than it does in my work because of the different user groups that are often targeted. Apple and Microsoft often target consumers who are using systems to make decisions whereas my projects focused on showing machine learning systems to expert user groups as a means of generating further knowledge and understanding of both the system and underlying data phenomenon unrelated to the machine learning system. It does seem to me, though, that there is more room to discuss how to calibrate trust, both in trusting a machine learning system more and

in limiting undo trust, through both confidence metrics and explanations in the three human-AI interaction frameworks.

7.4 Other Framework Guidelines

All three of the human-AI frameworks provided more design guidelines on topics which I did encountered less during the course of the four projects presented above. Most of these center around how to handle training data, correcting errors in deployment, calibrating or updating based on feedback. All four of my projects had a set amount of data, did not get fully deployed, and did not continue onto a version 2 to integrate significant user feedback. Exploring all of these aspects are future work for each of the projects. As such, the work covered in this dissertation cannot speak well to these other aspects of the human-AI frameworks, however I hope future work will explore these facets in the context of highly contextualized machine learning prediction and explanation systems.

Chapter 8. CONCLUSION

My dissertation work conducted four independent projects that all applied machine learning prediction and explanation systems into highly specialized domains and use cases. These projects demonstrate my thesis statement:

Selecting and adapting machine learning prediction and explanation systems to align with user goals, to retain nuance in explanations, and to impart an appropriate level of trust can support people of varying expertise in leveraging these systems.

In Chapter 3 (Problems and an Alternative to Single Explanation Aggregations) I adapted a machine learning explanation system to support machine learning model developers and data scientists. In Chapter 4 (Predicting Blood Glucose Test Accuracy in ICU Patients) I selected a machine learning prediction and explanation system to support clinicians. In Chapter 5 (Predicting and Explaining an Imminent Dementia Diagnosis with Limited Data) I selected and adapted machine learning prediction and explanation systems to support clinicians. In Chapter 6 (Flexible System for Efficient Goal-Directed Self-Tracking Analysis) I selected and adapted machine learning prediction and explanation systems to support patients.

The first way I achieved the support for leveraging these systems was by aligning with user goals. In Chapter 3 (Problems and an Alternative to Single Explanation Aggregations) I identified how the users were employing the existing explanation system and adapted that system to better support debugging, understanding, and explaining tasks. In Chapter 4 (Predicting Blood Glucose Test Accuracy in ICU Patients) I utilized many different machine learning prediction and explanation techniques to show that these systems might not meet user accuracy and application

needs. In Chapter 5 (Predicting and Explaining an Imminent Dementia Diagnosis with Limited Data) I predicted a future dementia diagnosis using minimal, inexpensive data. In Chapter 6 (Flexible System for Efficient Goal-Directed Self-Tracking Analysis) I matched the prediction system objective function to common, existing self-tracking goals.

The second way I achieved the support for leveraging these systems was by retaining nuance in explanations. In Chapter 3 (Problems and an Alternative to Single Explanation Aggregations) I provided more aggregations so there was less data artifact loss, while still showing a high level and quick look into the data. In Chapter 5 (Predicting and Explaining an Imminent Dementia Diagnosis with Limited Data) I provided biologically sound and personalized explanations. In Chapter 6 (Flexible System for Efficient Goal-Directed Self-Tracking Analysis) I showed holistic visualizations to help users answer questions they did not initially have.

The third way I achieved the support for leveraging these systems was by imparting an appropriate level of trust. In Chapter 3 (Problems and an Alternative to Single Explanation Aggregations) I provided multiple visualizations where there had previously been one, to avoid over trusting a single, simplistic, understanding of the explanations and increased the user's confidence in their more nuanced understandings and explanations of the explanations. In Chapter 4 (Predicting Blood Glucose Test Accuracy in ICU Patients) I presented results alongside baselines for random guessing, presented supporting statistical analyses that were more familiar to the user group, and presented visualizations to show why it seemed there is not a medically meaningful subgroup to make accurate predictions on. In Chapter 5 (Predicting and Explaining an Imminent Dementia Diagnosis with Limited Data) I showed that both the prediction and explanation systems were consistent with known biological underpinnings of dementia. In Chapter 6 (Flexible System

for Efficient Goal-Directed Self-Tracking Analysis) I showed confidence metrics and uncertainty visualizations of the outcomes.

These takeaways can also be seen in many guidelines set forth in three prominent human-AI frameworks while also suggesting some updates for practical advice while selecting and adapting machine learning prediction and explanation systems to support different types of experts.

BIBLIOGRAPHY

1. Ashraf Abdul, Christian von der Weth, Mohan Kankanhalli, and Brian Y. Lim. 2020. COGAM: Measuring and Moderating Cognitive Load in Machine Learning Model Explanations. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*. Association for Computing Machinery, New York, NY, USA, 1–14.

2. Farrokh Alemi, Shirley Moore, and Heibatollah Baghi. 2008. Self-Experiments and Analytical Relapse Prevention. *Quality Management in Healthcare* 17, 1: 53–65.

3. David Alvarez-Melis, Harmanpreet Kaur, Hal Daumé III, Hanna Wallach, and Jennifer Wortman Vaughan. 2021. From Human Explanation to Model Interpretability: A Framework Based on Weight of Evidence. In *9th AAAI Conference on Human Computation and Crowdsourcing (HCOMP)*.

4. Saleema Amershi, Dan Weld, Mihaela Vorvoreanu, Adam Fourney, Besmira Nushi, Penny Collisson, Jina Suh, Shamsi Iqbal, Paul N. Bennett, Kori Inkpen, Jaime Teevan, Ruth Kikin-Gil, and Eric Horvitz. 2019. Guidelines for Human-AI Interaction. In *Proceedings of the ACM Conference on Human Factors in Computing Systems*.

5. Apple, Inc. Machine Learning Human Interface Guidelines. Retrieved May 14, 2022 from https://developer.apple.com/design/human-interface-guidelines/machine-learning/overview/introduction/

6. Zurraini Arabi, Syed Alwi Syed Abdul Rahman, Helmy Hazmi, and Nazeefah Hamdin. 2016. Reliability and construct validity of the Early Dementia Questionnaire (EDQ). *BMC geriatrics* 16, 1: 202.

7. D. E. Barnes, K. E. Covinsky, R. A. Whitmer, L. H. Kuller, O. L. Lopez, and K. Yaffe. 2009. Predicting risk of dementia in older adults: The late-life dementia risk index. *Neurology* 73, 3: 173–179. https://doi.org/10.1212/WNL.0b013e3181a81636

8. Solon Barocas, Anhong Guo, Ece Kamar, Jacquelyn Krones, Meredith Ringel Morris, Jennifer Wortman Vaughan, W. Duncan Wadsworth, and Hanna Wallach. 2021. Designing Disaggregated Evaluations of AI Systems: Choices, Considerations, and Tradeoffs. In *Proceedings of the 2021 AAAI/ACM Conference on AI, Ethics, and Society*. Association for Computing Machinery, New York, NY, USA, 368–378. Retrieved May 13, 2022 from https://doi.org/10.1145/3461702.3462610

9. Nicasia Beebe-Wang, Alex Okeson, Tim Althoff, and Su-In Lee. 2021. Efficient and Explainable Risk Assessments for Imminent Dementia in an Aging Cohort Study. *IEEE Journal of Biomedical and Health Informatics* 25, 7: 2409–2420. https://doi.org/10.1109/JBHI.2021.3059563

10. Sylvie Belleville and others. 2017. Neuropsychological Measures that Predict Progression from Mild Cognitive Impairment to Alzheimer's type dementia in Older Adults: a Systematic Review and Meta-Analysis. *Neuropsychology Review* 27: 328–353.

11. David Bennett, Julie A Schneider, Zoe Arvanitakis, and Robert S Wilson. 2012. Overview and findings from the religious orders study. *Current Alzheimer Research* 9, 6: 628–645.

12. David Bennett, Julie A Schneider, Aron S Buchman, Lisa L Barnes, Patricia A Boyle, and Robert S Wilson. 2012. Overview and findings from the rush Memory and Aging Project. *Current Alzheimer Research* 9, 6: 646–663.

13. Andrea Bozoki, Bruno Giordani, Judith L. Heidebrink, Stanley Berent, and Norman L. Foster. 2001. Mild Cognitive Impairments Predict Dementia in Nondemented Elderly Patients With Memory Loss. *Archives of Neurology* 58, 3: 411–416. https://doi.org/10.1001/archneur.58.3.411

14. Andrea Bunt, Matthew Lount, and Catherine Lauzon. 2012. Are Explanations Always Important? A Study of Deployed, Low-Cost Intelligent Interactive Systems. In *Proceedings of the 2012 ACM International Conference on Intelligent User Interfaces* (IUI '12), 169–178. https://doi.org/10.1145/2166966.2166996

15. Adrian Bussone, Simone Stumpf, and Dympna O'Sullivan. 2015. The Role of Explanations on Trust and Reliance in Clinical Decision Support Systems. In *2015 International Conference on Healthcare Informatics*, 160–169. https://doi.org/10.1109/ICHI.2015.26

16. Rich Caruana, Yin Lou, Johannes Gehrke, Paul Koch, Marc Sturm, and Noemie Elhadad. 2015. Intelligible Models for HealthCare: Predicting Pneumonia Risk and Hospital 30-day Readmission. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (KDD '15), 1721–1730. https://doi.org/10.1145/2783258.2788613

17. Centers for Disease Control and Prevention. 1974. NHANES Data Set.

18. Tianqi Chen and Carlos Guestrin. 2016. XGBoost: A Scalable Tree Boosting System. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 785–794.

19. Eun Kyoung Choe, Saeed Abdullah, Mashfiqui Rabbi, Edison Thomaz, Daniel A Epstein, Felicia Cordeiro, Matthew Kay, Gregory D Abowd, Tanzeem Choudhury, James Fogarty, and others. 2017. Semi-Automated Tracking: a Balanced Approach for Self-Monitoring Applications. *IEEE Pervasive Computing* 16, 1: 74–84.

20. Eun Kyoung Choe, Nicole B Lee, Bongshin Lee, Wanda Pratt, and Julie A Kientz. 2014. Understanding Quantified-Selfers' Practices in Collecting and Exploring Personal Data. In *Proceedings of the ACM Conference on Human Factors in Computing Systems*. Retrieved from https://doi.acm.org/10.1145/2556288.2557372

21. François Chollet and others. 2015. Keras. Retrieved from \url{https://keras.io}

22. Vincent Chouraki and others. 2016. Evaluation of a genetic risk score to improve risk prediction for Alzheimer's disease. *Journal of Alzheimer's Disease* 53, 3: 921–932.

23. Dawn E. Corl, Tom S. Yin, Michelle E. Mills, Tina L. Spencer, Lucy Greenfield, Erin Beauchemin, Jessica Cochran, Louise D. Suhr, Rachel E. Thompson, and Brent E. Wisse. 2013. Evaluation of Point-of-Care Blood Glucose Measurements in Patients with Diabetic Ketoacidosis or Hyperglycemic Hyperosmolar Syndrome Admitted to a Critical Care Unit. *Journal of Diabetes Science and Technology* 7, 5: 1265–1274. https://doi.org/10.1177/193229681300700516

24. Dawn Corl, Tom Yin, May Ulibarri, Heather Lien, Tracy Tylee, Jing Chao, and Brent E. Wisse. 2018. What Can We Learn From Point-of-Care Blood Glucose Values Deleted and Repeated by Nurses? *Journal of Diabetes Science and Technology* 12, 5: 985–991. https://doi.org/10.1177/1932296818763891

25. Ian Covert, Scott M Lundberg, and Su-In Lee. 2020. Understanding Global Feature Contributions With Additive Importance Measures. In *Advances in Neural Information Processing Systems*, 17212–17223. Retrieved May 13, 2022 from https://proceedings.neurips.cc/paper/2020/hash/c7bf0b7c1a86d5eb3be2c722cf2cf746-Abstract.html

26. Ruoxuan Cui and Manhua Liu. 2019. RNN-based longitudinal analysis for diagnosis of Alzheimer's disease. *Computerized Medical Imaging and Graphics* 73: 1–10. https://doi.org/10.1016/j.compmedimag.2019.01.005

27. Nediyana Daskalova, Karthik Desingh, Jin Young Kim, Lixiang Zhang, Alexandra Papoutsaki, and Jeff Huang. 2017. Lessons Learned from Two Cohorts of Personal Informatics Self-Experiments. In *Proceedings of the ACM Conference on Ubiquitous Computing*. Retrieved from https://doi.org/10.1145/3130911

28. Nediyana Daskalova, Eindra Kyi, Kevin Ouyang, Arthur Borem, Sally Chen, Sung Hyun Park, Nicole Nugent, and Jeff Huang. 2021. Self-E: Smartphone-Supported Guidance for Customizable Self-Experimentation. In *Proceedings of the ACM Conference on Human Factors in Computing Systems*.

29. Nediyana Daskalova, Danaë Metaxa-Kakavouli, Adrienne Tran, Nicole Nugent, Julie Boergers, John McGeary, and Jeff Huang. 2016. SleepCoacher: A Personalized Automated Self-Experimentation System for Sleep Recommendations. In *Proceedings of the ACM Symposium on User Interface Software and Technology*. Retrieved from https://doi.org/bwt4

30. Alex J. DeGrave, Joseph D. Janizek, and Su-In Lee. 2020. *AI for radiographic COVID-19 detection selects shortcuts over signal*. https://doi.org/10.1101/2020.09.13.20193565

31. Daniel A Epstein, An Ping, James Fogarty, and Sean A Munson. 2015. A Lived Informatics Model of Personal Informatics. In *Proceedings of the ACM International Joint Conference on Pervasive and Ubiquitous Computing*. Retrieved from https://doi.org/10.1145/2750858.2804250

32. Marshal F. Folstein, Susan E. Folstein, and Paul R. McHugh. 1975. Mini-mental state: A practical method for grading the cognitive state of patients for the clinician. *Journal of Psychiatric Research* 12, 3: 189–198.

33. Carlo Jan Pati-an Garingarao, Myrna Buenaluz-Sedurante, and Cecilia Alegado Jimeno. 2014. Accuracy of Point-of-Care Blood Glucose Measurements in Critically Ill Patients in Shock. *Journal of Diabetes Science and Technology* 8, 5: 937–944. https://doi.org/10.1177/1932296814538608

34. Joseph Gaugler, Bryan James, Tricia Johnson, Allison Marin, and Jennifer Weuve. 2019. 2019 Alzheimer's disease facts and figures. *Alzheimers & Dementia* 15, 3: 321–387.

35. Jantje Goerdten, Iva Cukic, Samuel O. Danso, Isabelle Carriere, and Graciela Muniz-Terrera. 2019. Statistical methods for dementia risk prediction and recommendations for future work: A systematic review. *Alzheimer's & Dementia: Translational Research & Clinical Interventions* 5: 563–569. https://doi.org/10.1016/j.trci.2019.08.001

36. Google. People + AI Guidebook. Retrieved May 14, 2022 from https://design.google/ai-guidebook

37. T. J. Hastie and R. J. Tibshirani. 2017. *Generalized Additive Models*. Routledge, New York. https://doi.org/10.1201/9780203753781

38. Center for Devices and Radiological Health. 2021. Blood Glucose Monitoring Test Systems for Prescription Point-of-Care Use. *U.S. Food and Drug Administration*. Retrieved November 5, 2021 from https://www.fda.gov/regulatory-information/search-fda-guidance-documents/blood-glucose-monitoring-test-systems-prescription-point-care-use

39. David Heckerman. 1999. A Tutorial on Learning with Bayesian Networks. In *Learning in Graphical Models*, Michael I Jordan (ed.). MIT Press, Cambridge, MA, 301–354.

40. Anke Hensel, Tobias Luck, Melanie Luppa, Heide Glaesmer, Matthias C Angermeyer, and Steffi G Riedel-Heller. 2009. Does a reliable decline in Mini Mental State Examination total score predict dementia? *Dementia and geriatric cognitive disorders* 27, 1: 50–58.

41. David B. Hogan and Erika M. Ebly. 2000. Predicting Who Will Develop Dementia in a Cohort of Canadian Seniors. *Canadian Journal of Neurological Sciences* 27, 1: 18?24. https://doi.org/10.1017/S0317167100051921

42. Fred Hohman, Andrew Head, Rich Caruana, Robert DeLine, and Steven M. Drucker. 2019. Gamut: A Design Probe to Understand How Data Scientists Understand Machine Learning Models. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*. New York, NY, USA, 1–13.

43. Sungsoo Ray Hong, Jessica Hullman, and Enrico Bertini. 2020. Human Factors in Model Interpretability: Industry Practices, Challenges, and Needs. *Proc. ACM Hum.-Comput. Interact.* 4, CSCW1. https://doi.org/10.1145/3392878

44. K. Höök. 2000. Steps to take before intelligent user interfaces become real. *Interacting with Computers* 12, 4: 409–426. https://doi.org/10.1016/S0953-5438(99)00006-5

45. Eric Horvitz. 1999. Principles of mixed-initiative user interfaces. In *Proceedings of the SIGCHI conference on Human Factors in Computing Systems* (CHI '99), 159–166. https://doi.org/10.1145/302979.303030

46. Shigeaki Inoue, Moritoki Egi, Joji Kotani, and Kiyoshi Morita. 2013. Accuracy of blood-glucose measurements using glucose meters and arterial blood gas analyzers in critically ill adult patients: systematic review. *Critical Care* 17, 2: R48. https://doi.org/10.1186/cc12567

47. David K Johnson, Martha Storandt, John C Morris, and James E Galvin. 2009. Longitudinal study of the transition from healthy aging to Alzheimer disease. *Archives of neurology* 66, 10: 1254–1259.

48. Jongbin Jung, Connor Concannon, Ravi Shroff, Sharad Goel, and Daniel G. Goldstein. 2020. Simple rules to guide expert classifications. *Journal of the Royal Statistical Society: Series A (Statistics in Society)* 183, 3: 771–800. https://doi.org/10.1111/rssa.12576

49. Salmaan Kanji, Jennifer Buffie, Brian Hutton, Peter S. Bunting, Avinder Singh, Kevin McDonald, Dean Fergusson, Lauralyn A. McIntyre, and Paul C. Hebert. 2005. Reliability of point-of-care testing for glucose measurement in critically ill adults. *Critical Care Medicine* 33, 12: 2778–2785. https://doi.org/10.1097/01.ccm.0000189939.10881.60

50. Ravi Karkar, Jessica Schroeder, Daniel A Epstein, Laura R Pina, Jeffrey Scofield, James Fogarty, Julie A Kientz, Sean A Munson, Roger Vilardaga, and Jasmine Zia. 2017. TummyTrials: A Feasibility Study of Using Self-Experimentation To Detect Individualized Food Triggers. In *Proceedings of the ACM Conference on Human Factors in Computing Systems*. Retrieved from https://doi.org/10.1145/3025453.3025480

51. Harmanpreet Kaur, Harsha Nori, Samuel Jenkins, Rich Caruana, Hanna Wallach, and Jennifer Wortman Vaughan. 2020. Interpreting Interpretability: Understanding Data Scientists' Use of Interpretability Tools for Machine Learning. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*.

52. Matthew Kay, Tara Kola, Jessica R Hullman, and Sean A Munson. 2016. When (ish) Is My Bus?: User-Centered Visualizations of Uncertainty in Everyday, Mobile Predictive Systems. In *Proceedings of the ACM Conference on Human Factors in Computing Systems*. Retrieved from https://doi.org/10.1145/2858036.2858558

53. Young-Ho Kim, Jae Ho Jeon, Bongshin Lee, Eun Kyoung Choe, and Jinwook Seo. 2017. OmniTrack: A Flexible Self-Tracking Approach Leveraging Semi-Automated Tracking. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 1, 3. Retrieved from https://doi.org/10.1145/3130930

54. David C. Klonoff. 2014. Point-of-Care Blood Glucose Meter Accuracy in the Hospital Setting. *Diabetes Spectrum : A Publication of the American Diabetes Association* 27, 3: 174–179. https://doi.org/10.2337/diaspect.27.3.174

55. Pang Wei Koh and Percy Liang. 2017. Understanding Black-box Predictions via Influence Functions. In *Proceedings of the 34th International Conference on Machine Learning*, 1885–1894. Retrieved May 13, 2022 from https://proceedings.mlr.press/v70/koh17a.html

56. Ronny Kohavi and Barry Becker. 1996. Adult Data Set.

57. Daphne Koller and Nir Friedman. 2009. *Probabilistic Graphical Models: Principles and Techniques*. MIT press. Retrieved from https://books.google.com/books/about/Probabilistic_Graphical_Models.html?id=dOruCwAAQBAJ &hl=en

58. Isaac Lage, Emily Chen, Jeffrey He, Menaka Narayanan, Been Kim, Samuel J. Gershman, and Finale Doshi-Velez. 2019. Human Evaluation of Models Built for Interpretability. 7: 59–67.

59. Huong T. Le, Neil S. Harris, Abby J. Estilong, Arvid Olson, and Mark J. Rice. 2013. Blood Glucose Measurement in the Intensive Care Unit: What is the Best Method? *Journal of Diabetes Science and Technology* 7, 2: 489–499. https://doi.org/10.1177/193229681300700226

60. Seonjoo Lee and others. 2018. Episodic memory performance in a multi-ethnic longitudinal study of 13,037 elderly. *PloS one* 13, 11: e0206803.

61. Ian Li, Anind Dey, and Jodi Forlizzi. 2010. A Stage-Based Model of Personal Informatics Systems. In *Proceedings of the ACM Conference on Human Factors in Computing Systems*. Retrieved from https://doi.acm.org/10.1145/1753326.1753409

62. Courtney H Lias. 2018. Capillary Blood Glucose Testing in Hospital Settings.

63. Brian Y. Lim and Anind K. Dey. 2011. Investigating Intelligibility for Uncertain Context-Aware Applications. In *Proceedings of the 13th International Conference on Ubiquitous Computing* (UbiComp '11), 415–424. https://doi.org/10.1145/2030112.2030168

64. Chia-Chen Liu, Takahisa Kanekiyo, Huaxi Xu, and Guojun Bu. 2013. Apolipoprotein E and Alzheimer disease: risk, mechanisms and therapy. *Nature Reviews Neurology* 9, 2: 106.

65. Scott M. Lundberg. *SHAP*. Retrieved from https://github.com/slundberg/shap

66. Scott M Lundberg and Su-In Lee. 2017. A Unified Approach to Interpreting Model Predictions. In *Advances in Neural Information Processing Systems 30*. 4765–4774.

67. Scott M. Lundberg and others. 2020. From Local Explanations to Global Understanding with Explainable AI for Trees. *Nature Machine Intelligence* 2: 56–67.

68. Scott M Lundberg and and others. 2018. Explainable machine-learning predictions for the prevention of hypoxaemia during surgery. *Nature biomedical engineering* 2, 10: 749–760.

69. Martín Abadi and others. 2015. TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems. Retrieved from https://www.tensorflow.org/

70. Sean A. Munson, Jessica Schroeder, Ravi Karkar, Julie A. Kientz, Chia-Fang Chung, and James Fogarty. 2020. The Importance of Starting With Goals in N-of-1 Studies. *Frontiers in Digital Health* 2: 1–7.

71. Adam C Naj and others. 2014. Age-at-onset in late onset Alzheimer disease is modified by multiple genetic loci. *JAMA neurology* 71, 11: 1394.

72. Ziad S Nasreddine and others. 2005. The Montreal Cognitive Assessment, MoCA: a brief screening tool for mild cognitive impairment. *Journal of the American Geriatrics Society* 53, 4: 695–699.

73. Harsha Nori, Samuel Jenkins, Paul Koch, and Rich Caruana. 2019. InterpretML: A Unified Framework for Machine Learning Interpretability. *arXiv preprint arXiv:1909.09223*.

74. Harsha Nori, Samuel Jenkins, Paul Koch, and Rich Caruana. *InterpretML*. Retrieved from https://interpret.ml

75. Donald A. Norman. 1994. How might people interact with agents. *Communications of the ACM* 37, 7: 68–71. https://doi.org/10.1145/176789.176796

76. Besmira Nushi, Ece Kamar, and Eric Horvitz. 2018. Towards Accountable AI: Hybrid Human-Machine Analyses for Characterizing System Failure. *Proceedings of the AAAI Conference on Human Computation and Crowdsourcing* 6: 126–135.

77. Jean-Jacques Nya-Ngatchou, Dawn Corl, Susan Onstad, Tom Yin, Tracy Tylee, Louise Suhr, Rachel E. Thompson, and Brent E. Wisse. 2015. Point-of-care blood glucose measurement errors overestimate hypoglycaemia rates in critically ill patients. *Diabetes/Metabolism Research and Reviews* 31, 2: 147–154. https://doi.org/10.1002/dmrr.2575

78. Alex Okeson, Rich Caruana, and Nick Craswell. Summarize with Caution: Comparing Global Feature Attributions. 14.

79. Alex Okeson, Rich Caruana, Nick Craswell, Kori Inkpen, Scott M. Lundberg, Harsha Nori, Hanna Wallach, and Jennifer Wortman Vaughan. 2021. Summarize with Caution: Comparing Global Feature Attributions. *Bulletin of the Technical Committee on Data Engineering* 44, 4.

80. Art B. Owen and Clémentine Prieur. 2017. On Shapley Value for Measuring Importance of Dependent Inputs. *SIAM/ASA Journal on Uncertainty Quantification* 5, 1: 986–1002. https://doi.org/10.1137/16M1097717

81. F. Pedregosa and others. 2011. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research* 12: 2825–2830.

82. Heather F. Pidcoke, Charles E. Wade, Elizabeth A. Mann, Jose Salinas, Brian M. Cohee, John B. Holcomb, and Steven E. Wolf. 2010. Anemia Causes Hypoglycemia in ICU Patients

Due to Error in Single-Channel Glucometers: Methods of Reducing Patient Risk. *Critical care medicine* 38, 2: 471–476. https://doi.org/10.1097/CCM.0b013e3181bc826f

83. Forough Poursabzi-Sangdeh, Daniel G Goldstein, Jake M Hofman, Jennifer Wortman Wortman Vaughan, and Hanna Wallach. 2021. Manipulating and Measuring Model Interpretability. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*. Association for Computing Machinery, New York, NY, USA.

84. Martin Prince, Renata Bryce, and Cleusa Ferri. 2011. World Alzheimer Report 2011: The benefits of early diagnosis and intervention. . Alzheimer's Disease International.

85. J. R. Quinlan. 1986. Induction of decision trees. *Machine Learning* 1, 1: 81–106. https://doi.org/10.1007/BF00116251

86. Annette Rebel, Mark A. Rice, and Brenda G. Fahy. 2012. The Accuracy of Point-of-Care Glucose Measurements. *Journal of Diabetes Science and Technology* 6, 2: 396–411. https://doi.org/10.1177/193229681200600228

87. Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. 2016. "Why Should I Trust You?": Explaining the Predictions of Any Classifier. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, August 13-17, 2016*, 1135–1144.

88. WS Robinson. 2009. Ecological Correlations and the Behavior of Individuals*. *International Journal of Epidemiology* 38, 2: 337–341. https://doi.org/10.1093/ije/dyn357

89. Cynthia Rudin. 2019. Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nature Machine Intelligence* 1, 5: 206–215. https://doi.org/10.1038/s42256-019-0048-x

90. Chris Russell. 2019. Efficient Search for Diverse Coherent Explanations. In *Proceedings of the Conference on Fairness, Accountability, and Transparency* (FAT* '19), 20–28. https://doi.org/10.1145/3287560.3287569

91. Brad S. Karon, Laurie Griesmann, Renee Scott, Sandra C. Bryant, Jeffrey A. Dubois, Terry L. Shirey, Steven Presti, and Paula J. Santrach. 2008. Evaluation of the Impact of Hematocrit and Other Interference on the Accuracy of Hospital-Based Glucose Meters. *Diabetes Technology & Therapeutics* 10, 2: 111–120. https://doi.org/10.1089/dia.2007.0257

92. Jessica Schroeder, Ravi Karkar, James Fogarty, Julie A Kientz, Matthew Kay, and Sean A Munson. 2018. A Patient-Centered Proposal for Bayesian Analysis of Self-Experiments for Health. In *Journal of Healthcare Informatics Research*. Retrieved from https://doi.org/10.1007/s41666-018-0033-x

93. Jessica Schroeder, Ravi Karkar, Natalia Murinova, James Fogarty, and Sean A Munson. 2019. Examining Opportunities for Goal-Directed Self-Tracking to Support Chronic Condition Management. In *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 1–26. Retrieved from https://doi.org/10.1145/3369809

94. Marco Scutari. 2007. bnlearn - an R package for Bayesian network learning and inference. Retrieved from https://www.bnlearn.com/

95. Andrew Sears and Julie A. Jacko. 2007. *The Human-Computer Interaction Handbook: Fundamentals, Evolving Technologies and Emerging Applications, Second Edition*. CRC Press.

96. Blossom CM Stephan, Tobias Kurth, Fiona E Matthews, Carol Brayne, and Carole Dufouil. 2010. Dementia risk prediction in the population: are screening models accurate? *Nature Reviews Neurology* 6, 6: 318–326.

97. Ewout W Steyerberg and others. 2010. Assessing the performance of prediction models: a framework for some traditional and novel measures. *Epidemiology* 21, 1: 128.

98. Heung-Il Suk, Seong-Whan Lee, Dinggang Shen, Alzheimer's Disease Neuroimaging Initiative, and others. 2014. Hierarchical feature representation and multimodal fusion with deep learning for AD/MCI diagnosis. *NeuroImage* 101: 569–582.

99. Heung-Il Suk and Dinggang Shen. 2013. Deep learning-based feature representation for AD/MCI classification. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, 583–590.

100. Tom N Tombaugh and Nancy J McIntyre. 1992. The mini-mental state examination: a comprehensive review. *Journal of the American Geriatrics Society* 40, 9: 922–935.

101. Berk Ustun, Alexander Spangher, and Yang Liu. 2019. Actionable Recourse in Linear Classification. In *Proceedings of the Conference on Fairness, Accountability, and Transparency* (FAT* '19), 10–19. https://doi.org/10.1145/3287560.3287566

102. Danding Wang, Qian Yang, Ashraf Abdul, and Brian Y. Lim. 2019. Designing Theory-Driven User-Centric Explainable AI. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems* (CHI '19), 1–15. https://doi.org/10.1145/3290605.3300831

103. Daniel S. Weld and Gagan Bansal. 2019. The challenge of crafting intelligible intelligence. *Communications of the ACM* 62, 6: 70–79. https://doi.org/10.1145/3282486

104. Jennifer Wortman Vaughan and Hanna Wallach. 2021. A Human-Centered Agenda for Intelligible Machine Learning. In *Machines We Trust: Perspectives on Dependable AI*. MIT Press.

105. Austin P. Wright, Zijie J. Wang, Haekyu Park, Grace Guo, Fabian Sperrle, Mennatallah El-Assady, Alex Endert, Daniel Keim, and Duen Horng Chau. 2020. *A Comparative Analysis of Industry Human-AI Interaction Guidelines*. arXiv. https://doi.org/10.48550/arXiv.2010.11761

106. Qian Yang, Aaron Steinfeld, Carolyn Rosé, and John Zimmerman. 2020. Re-examining Whether, Why, and How Human-AI Interaction Is Uniquely Difficult to Design. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*.

Association for Computing Machinery, New York, NY, USA, 1–13. Retrieved May 16, 2022 from https://doi.org/10.1145/3313831.3376301

107. Qian Yang, Aaron Steinfeld, and John Zimmerman. 2019. Unremarkable AI: Fitting Intelligent Decision Support into Critical, Clinical Decision-Making Processes. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems* (CHI '19), 1–11. https://doi.org/10.1145/3290605.3300468

108. Jiaming Zeng, Berk Ustun, and Cynthia Rudin. 2017. Interpretable classification models for recidivism prediction. *Journal of the Royal Statistical Society: Series A (Statistics in Society)* 180, 3: 689–722. https://doi.org/10.1111/rssa.12227

109. Amy X. Zhang, Michael Muller, and Dakuo Wang. 2020. How Do Data Science Workers Collaborate? Roles, Workflows, and Tools. *Proc. ACM Hum.-Comput. Interact.* 4, CSCW1. https://doi.org/10.1145/3392826

110. 2020. 2020 Alzheimer's disease facts and figures. *Alzheimer's & Dementia* 16, 3: 391–460. https://doi.org/10.1002/alz.12068

111. Glucose Measurement: Confounding Issues in Setting Targets for Inpatient Management | Diabetes Care | American Diabetes Association. Retrieved May 13, 2022 from https://diabetesjournals.org/care/article/30/2/403/28440/Glucose-Measurement-Confounding-Issues-in-Setting