Using Expert Interpretation and Reasoning to Guide Model Selection in Machine Learning

by

Jiaxuan Wang

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
(Computer Science and Engineering)
in the University of Michigan
2022

Doctoral Committee:

Associate Professor Jenna Wiens, Chair
Assistant Professor David Fouhey
Assistant Professor Bryan R. Goldsmith
Senior Researcher at Microsoft Research Scott Lundberg

Jiaxuan Wang

jiaxuan@umich.edu

ORCID iD: 0000-0002-3447-383X

Acknowledgements

Firstly, I would like to thank my committee members for their valuable feedback on this dissertation. In particular, I would like to acknowledge my advisor, Professor Jenna Wiens. She has supported every aspect of my PhD journey with enthusiasm. I'm grateful that she let me explore my research interests without worrying about failures and that she cared about my growth as both a person and a researcher. Without her guidance, none of the work in this dissertation would have been possible.

Next, I would like to acknowledge all my collaborators and colleagues. They have helped shape my view on research and have made my graduate school experience fun and enjoyable. In particular, I would like to thank Scott Lundberg for being an awesome mentor during my internship. Brain storm sessions with him were so engaging and focused that I did not notice the hours go by.

Furthermore, I would not have started doing research without the help of Yaoyun Shi and Jia Deng. Yaoyun sparked my interest in research and Jia showed me the excitement of developing machine learning algorithms. I would like to thank both of them for starting me on this rewarding journey.

Finally, I could not have finished this dissertation without the support of friends and family. I am grateful for Hao Yuan, Qi Luo, and Zeyu Zheng for consistently checking in with me throughout the years about both research and life. I am also forever in debt to Jeeheh Oh and my parents for always being there, supporting me emotionally. This dissertation is dedicated to them.

Table of Contents

List of Figures

List of Tables

List of Appendices

Abstract

In using machine learning to train predictive models, training data often under-specify solutions due to limited sample size and/or the lack of diversity in samples. When selecting a solution (or model) for deployment, we must consider metrics beyond statistics calculated in distribution (*i.e.*, statistics on samples available in model learning such as training and validation error) so that we can identify and correct potential pitfalls of the learned model when applied to out of distribution scenarios (*e.g.*, the deployment environment). To address model underspecification, in this thesis, we develop several methods that leverage domain knowledge during model selection.

First, to select among solutions, one must understand the learned model. We demonstrate how one can achieve a holistic understanding by including system level knowledge about the problem. Our approach, *Shapley Flow*, takes a user defined causal graph on the features as input and summarizes the attribution to model prediction along the causal edges. Shapley Flow unifies three widely used Shapley-value based model interpretation methods and elucidates the need to consider the data generation procedure, capturing both the direct and indirect impact of features. Second, domain knowledge is not only useful in model interpretation, it is also crucial in training. If one knows what features experts rely on, one can incorporate such knowledge to avoid using proxy features that are spuriously correlated with the outcome. We propose a novel regularization technique to learn a *credible model*, one that is both accurate and is aligned with domain experts. Using our approach, *Expert Yielded Estimate* (EYE), we demonstrate on two large scale clinical datasets that one can significantly increase alignment with expert knowledge without sacrificing accuracy, while outperforming an approach based on expert knowledge alone. Finally, we connect credible learning with shortcut learning, identifying sufficient assumptions for credible models to eliminate the dependence on spurious correlation. In this process, we extend the EYE penalty to work with nonlinear models and work on tasks with domain knowledge not expressed on the input space. Applied to two benchmark datasets, our approach successfully mitigates shortcut learning, even when assumptions are moderately violated. By leveraging domain knowledge, our proposed approaches help build trustworthy systems that can be safely applied in practice.

Chapter 1

Introduction

Across application domains, machine learning models have demonstrated exceptional performance [1]–[5]. For example, in the game of Go, AlphaGo defeated 18 time world Go champion Lee Sedol in 2016 [1]. In biology, AlphaFold made important breakthrough in protein folding, a problem that has perplexed researchers for more than 50 years [2]. In language modeling, computer vision, and even clinical diagnostics, deep learning models have demonstrated comparable performance to humans on challenging datasets [3]–[5]. However, despite seemingly good performance, practitioners should be wary of adopting machine learning models too quickly. In particular, machine learning models that generalize well in the training distribution can result in very different generalization performance in deployment, a phenomenon known as *underspecification* [6]–[10].

Underspecification arises for many reasons. The training dataset could exhibit selection bias. For example, when classifying dogs versus wolves, a model may rely on features pertaining to the indoors to recognize a dog, resulting in a failure to generalize when dogs appear outside [11]. In settings with a limited sample size or samples lacking diversity, spurious correlations can lead to a model that relies on proxy variables or "shortcuts". The use of deep networks further exacerbates underspecification. Such models are overparametrized (*i.e.*, have more model parameters than what the training samples can afford), which necessarily leads to the existence of multiple solutions. Note that underspecification persists even with low capacity models. During training, a regularized linear model is equally susceptible to using "indoor" features to differentiate a dog from a wolf compared to a deep network, despite the decision process being more transparent. Furthermore, as long as the sampling distribution is biased (*e.g.*, lack of diversity: dog images are consistently taken indoor), gathering more samples may not help

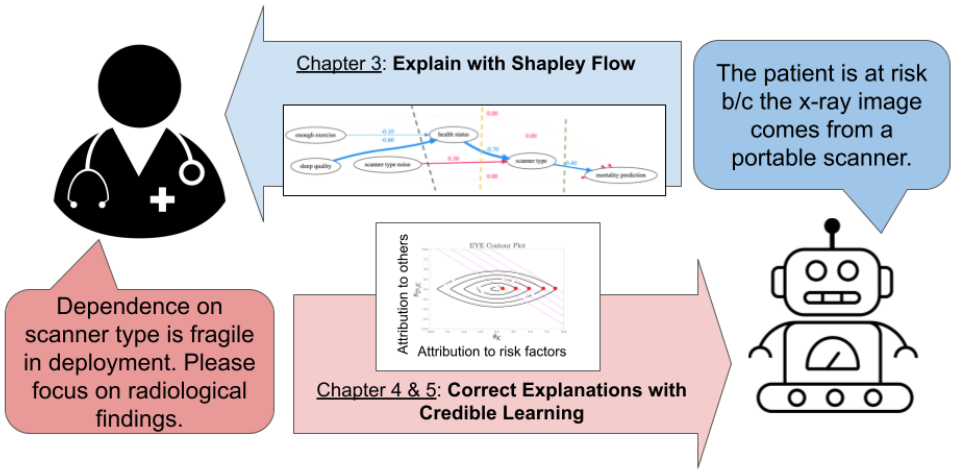


FIGURE 1.1: A schematic picture of the dissertation. We focus on tackling both directions of human model communication. For human to understand a machine's reasoning in detail, we introduce an axiomatic explanation method Shapley Flow in Chapter 3. For machine to take human feedback on explanation, we introduce credible learning in Chapter 4 and 5.

with underspecification. **Since minimizing empirical risk alone does not guarantee robust solutions, we are interested in finding other ways to minimize generalization error in the real world**.

While there are many settings potentially related to underspecification (*e.g.*, transfer learning when we have limited labeled samples from the deployment environment; unsupervised domain adaptation when we have unlabeled deployment data), in this dissertation, we explore settings in which deployment data are unavailable at the time of training and validation. Instead, **we assume access to partial domain knowledge about feature importance within the deployment environment**. This knowledge is available from experts who gain experience in deployment environments to which machines do not have access (*e.g.*, causal knowledge learned from conducting experiments; common sense developed by interacting with the real world). For example, in healthcare, there is a significant body of literature documenting risk and protective factors for diseases. While we may not have access to the experimental data, we can use established scientific knowledge to guide model training and selection.

1.1 Challenges and Opportunities

Incorporating feature level domain knowledge requires a two-way interpretation between machines and humans. First, humans need to understand the machine in order to choose wisely when presented with multiple solutions of the same test performance. This calls for designing model interpretation methods to help identify potential flaws in solutions. Second, the machine needs to incorporate human feedback in order to correct its often fragile reasoning. This calls for designing priors for machine reasoning. As illustrated in **Figure 1.1**, **we propose novel algorithms to tackle both directions**. To help humans understand how machines reason, our edge attribution method [12], Shapley Flow, improves on existing feature attribution methods with a causal graph. To help machines understand how humans reason, our regularization technique [13], expert yielded estimate (EYE), uses expert given feature attributions as priors for the model's feature importance. However, not all domain knowledge can be applied to the input space. For example, concepts like "stripes" that can be used to distinguish between animals in an image are difficult to specify in the input space. We explore methods for incorporating concept level domain knowledge in the final chapter. Below, we summarize each direction.

In Chapter 3, we improve feature attribution [14]–[17], a popular form of model interpretation, by capturing both the direct and indirect effects of features. Existing approaches that incorporate the causal graph on the inputs exhibit clear limitations: either they completely ignore the dependencies across features (features with only indirect influence are given 0 attribution, despite the fact that changing their values significantly affects the output), or they exclusively focus on the independent variables of the graph (non-source features are given 0 importance). Our approach, Shapley Flow, solves the problem by assigning credit to *edges* instead of nodes in a graph, showing both the direct and indirect influence of features. Furthermore, Shapley Flow is the unique solution to a generalization of the Shapley value axioms [18] to directed acyclic graphs. We demonstrate the benefit of using Shapley Flow to reason about the impact of a model's input on its output through case studies on two real datasets. In addition to maintaining insights from existing approaches, Shapley Flow extends the flat, set-based, view prevalent in game theory based explanation methods to a deeper, *graph-based*, view. This graph-based view enables users to understand the flow of importance through a system, and reason

about potential interventions.

In Chapter 4, we look at how to incorporate feature level prior knowledge to tackle model underspecification. Specifically, we ask whether it is possible to learn a model that is not only consistent with the data, but also aligns well with domain knowledge. We refer to such a model as a *credible* model. In solving this problem, we developed a novel regularization technique, EYE, that is both theoretically and empirically sound. EYE encourages selection among highly correlated features to favor a solution that is dense in expert identified features and sparse otherwise. Applied to two large scale patient risk stratification problems, our proposed method is as accurate as all baseline models, but more closely aligns with domain knowledge.

Prior work in credible learning assumes domain knowledge can be directly applied on the input space (*i.e.*, feature level domain knowledge). While this assumption often holds for tabular data, it is hard to justify for complex input modalities such as images and time series. Just imagine how hard it is for a bird expert to pinpoint the exact pixels used in identifying a bird compared to saying that the bird has a red beak (*i.e.*, a "concept" that is derived from the input). Decoupling domain knowledge with input features increases the applicability of credible learning. In Chapter 5, we look at how to incorporate concept level prior knowledge to tackle underspecification. To do that, we propose the concept credible model (CCM), a method that combines EYE regularization with the concept bottleneck model (CBM) [19]. Our approach not only incorporates concept level domain knowledge, but also addresses CBM's known deficiency to deal with incomplete concepts (*i.e.*, the input features provide additional information towards the target given the concepts). In this chapter, we focus on mitigating shortcut learning. That is, we consider scenarios in which the training and validation datasets contain spurious correlations (*i.e.*, shortcuts) unlikely to generalize to the deployment environment. In such cases, we would like to discourage the model from using/taking these shortcuts during training. Our theoretical analysis sheds light on the connection between credible and shortcut learning, identifying sufficient assumptions for a credible model to eliminate the use of shortcut. Empirically, we demonstrate that CCM is more robust to shortcuts compared to baseline approaches, even when the identified sufficient assumptions are moderately violated.

1.2 Contributions

To address model underspecification using domain knowledge, we present several contributions in this dissertation, summarized as follows:

- **Enabling a system-level view of Shapley value based feature attribution**. In Chapter 3, we present Shapley Flow, an explanation method that uniquely satisfies a natural extension of Shapley value axioms to graphs. The resulting method unifies three existing feature attribution methods into a single framework with attractive theoretical properties and strong empirical performance, highlighting both the direct and indirect effects of features [12].

- **Formalizing the idea of credible learning with an approach that leads to accurate linear models with sensible explanations**. In Chapter 4, we present the expert yielded estimates (EYE) penalty, a regularization technique that incorporates feature level domain expertise to tackle underspecification. We formalize the notion of a credible model in the linear setting (*i.e.*, a model that aligns well with expert's reasoning while being consistent with data). The resulting method exhibits desirable theoretical properties and works well on two large scale clinical datasets. [13].

- **Connecting credible and shortcut learning while incorporating non-input level domain knowledge**. In Chapter 5, we decouple domain knowledge from input features, creating concept based credibility. This is achieved through a new model training procedure, concept credible model (CCM), that combines credible models with concept based models. CCM extends the EYE regularization to apply on non-linear models with concept level domain knowledge. We formalize and identify sufficient conditions in which CCM can mitigate shortcut learning. These conditions allow domain knowledge to be incomplete and the shortcut to be perfectly correlated with other features, settings in which many previous works fail. Empirically, we show that CCM leads to better generalization on out of distribution datasets compared to baselines.

Ensuring good performance in the deployment environment is crucial for the adoption of machine learning models in high stake domains such as healthcare. **To that end,**

in this dissertation, we present a variety of methods that use prior knowledge to guide model selection. The rest of the dissertation is organized as follows. The background chapter (Chapter 2) describes relevant concepts used throughout the dissertation. Chapter 3, 4, and 5 describe the technical details of our contributions. And the concluding chapter (Chapter 6) reflects on future directions in relation to the work presented in this dissertation.

Chapter 2

Background

In this chapter, we briefly review important concepts referenced throughout the reminder of the dissertation. First, we introduce Shapley value, which forms the basis of Shapley Flow in Chapter 3. Second, we formalize the notion of causal graphs. Causal graphs are used in Chapter 3 as user defined input to aid model interpretation and used in Chapter 5 to formalize our problem setup. Third, we review interpretability, contextualizing Shapley Flow in the literature. Fourth, we summarize common parameter norm regularization methods. They are the baselines compared to the EYE penalty in Chapter 4. Finally, we formalize the out of distribution generalization problem and contrast it with the standard supervised learning setup, providing background needed for Chapter 5.

2.1 Shapley Value

Shapley value [18] forms the basis of Shapley Flow (Chapter 3). It stems from cooperative game theory [20]. In the context of machine learning, it has been extensively used in feature attribution [14], [21]–[24] and data valuation [25]–[27]. Here we give an overview of Shapley value and its axioms. Its application in feature attribution is summarized in **Section 2.3.4**.

A cooperative game consists of a set of players ($\mathcal{P}$) and a payoff function (v) that assigns a value to every possible subset of players (*i.e.*, $v : 2^{\mathcal{P}} \to \mathbb{R}$). A subset of players is often referred to as a "coalition" (denoted as $\mathcal{C}$). The goal of Shapley value is to assess the contribution of each player in $\mathcal{P}$ in some "fair" way, as formalized by its axioms. Consider a concrete example in which one wishes to measure the contribution of Alice and Bob, when they work together to move a table [28]. Here, the players are $\{\text{Alice}, \text{Bob}\}$,

and the payoff function outputs 1 when the table is successfully moved, otherwise 0. A natural way to quantify Alice's worth in $\mathcal{P}$ is to measure the difference in result with and without her (*i.e.*, $v(\{\text{Alice, Bob}\}) - v(\{\text{Bob}\})$), that is Alice's marginal contribution given the coalition $\{\text{Bob}\}$. If only one person is needed to move the table (two people also work but zero players cannot), then everyone's marginal contribution is 0 (*i.e.*, nobody gets any credit). This cannot be fair since they together get the job done. In fact, only when the coalition grows from zero players to one player does the player gets a marginal contribution of 1. Since the ordering of players is unknown, it is unclear who should get the credit. To solve the problem, Shapley value considers every possible ordering in which a coalition can be formed, and averages the marginal contribution of a player for all orderings. In our case, Alice's Shapley value $\phi_v(\text{Alice}) = 0.5\Big(v(\{\text{Alice, Bob}\}) - v(\{\text{Bob}\})\Big) + 0.5\Big(v(\{\text{Alice}\}) - v(\{\})\Big) = 0.5(1-0) + 0.5(0-0) = 0.5$, averages over the two possible orderings of Bob moves the table first and Alice moves the table first. In this example, Bob would have the same Shapley value as Alice. Note that the sum of Shapley value for all players adds up to the total credit of 1. In general, a player i's Shapley value, $\phi_v(i)$, has the following form:

$$\phi_v(i) = \sum_{\mathcal{C} \subseteq \mathcal{P} \setminus \{i\}} \frac{|\mathcal{C}|!(|\mathcal{P}| - |\mathcal{C}| - 1)!}{|\mathcal{P}|!} \Big(v(\mathcal{C} \cup \{i\}) - v(\mathcal{C})\Big) \tag{2.1}$$

where "$|\cdot|$" denotes the cardinality of a set and "!" denotes factorial. The numerator, $|\mathcal{C}|!(|\mathcal{P}| - |\mathcal{C}| - 1)!$, is the number of ways a coalition $\mathcal{C}$ is followed by i in all orderings. The denominator, $|\mathcal{P}|!$, is the total number of orderings. Together, the expression denotes a player i's marginal contribution when added to a coalition, averaged over all possible orderings in which a coalition could form.

2.1.1 Shapley Value Axioms

Not only does Shapley value have an intuitive explanation (*i.e.*, average of marginal contribution over all orderings), it is also uniquely defined by four simple axioms. This axiomatic nature of Shapley value makes it less arbitrary compared to alternative credit assignment metrics such as the last-on -the-bus value (the value added when a player last

join the group) [28]. The four axioms [18], [23], [28] that Shapley value uniquely satisfies are:

- **Dummy/Null player**: $\phi_v(i) = 0$ when $v(\mathcal{C} \cup \{i\}) = v(\mathcal{C}) \forall \mathcal{C} \subseteq \mathcal{P} \setminus \{i\}$.

  The dummy player axiom states that if a player's marginal contribution is 0 for all orderings of players, it should get 0 credit.

- **Efficiency/Full allocation**: $\sum_{i \in \mathcal{P}} \phi_v(i) = v(\mathcal{P}) - v(\{\})$.

  The efficiency axiom states that the sum of the values for all players equals to the difference of values generated by all players and values generated by no players.

- **Symmetry/Fairness**: $\phi_v(i) = \phi_v(j)$ when $v(\mathcal{C} \cup \{i\}) = v(\mathcal{C} \cup \{j\}) \forall \mathcal{C} \subseteq \mathcal{P} \setminus \{i, j\}$.

  The symmetry axiom states that if two players have the same marginal contribution for any coalition, they should be given equal credit.

- **Linearity**: $\phi_{\alpha u + \beta v}(i) = \alpha \phi_u(i) + \beta \phi_v(i)$ for any payoff function u, v, any $\alpha, \beta \in \mathbb{R}$, and any player $i \in \mathcal{P}$.

  The linearity axiom states that the value assignment function is linear in the payoff functions. In the context of games, the axiom prevents players from splitting up the payoff function or combining two payoff functions to get higher payoff because the credit would have been the same. In the context of model interpretation, it means that the Shapley value for a linear ensemble model is the same as linearly ensemble the Shapley value of individual models.

2.2 Causal Graphs

Causal graphs are formal tools used to encode assumptions about how data are generated. They are graphs associated with structural causal models (SCM), which forms the foundation of causal inference [29], [30]. Related to this dissertation, we use causal graphs to a) help understand the direct and indirect effects of features in Chapter 3 and b) formalize the assumptions on shortcuts in Chapter 5. Here, we first introduce graph notations and then formally define SCM. Our definition follows from Chapter 6 of [30].

2.2.1 Graph Terminology

A graph $\mathcal{G} = (V, E)$ consists of a set of *vertices/nodes* V and a set of *edges* $E \subseteq V \times V$ where each edge is a tuple of two vertices in V. We assume edges are *directed* without loss of generality (*i.e.*, the edge's first vertex, referred to as the source of the edge, points to the second vertex, referred to as the target of the edge). An *undirected* edge between nodes i and j can be represented in a graph by including both (i, j) and (j, i) in E. A *directed path* in a graph is a list of vertices in V such that every pair of adjacent vertices in the list are in E. A node i is called a *parent* of a node j if $(i, j) \in E$ (in which case j is a *child* of i). A node i is called an *ancestor* of a node j if there is a directed path from i to j (in which case j is a *descendant* of i). A graph is called a *directed acyclic graph (DAG)* if for all $i \in V$, i does not have a directed path to itself. A node without any parent is called a *source node*. A node without any child is called a *sink* node. A bijective function $\pi : \{1, \cdots, |V|\} \to \{1, \cdots, |V|\}$ is called a *permutation*. For a DAG, $\mathcal{G}$, denoting the all descendant of a node i as $DE(i, \mathcal{G})$, a permutation π is a *topological/causal ordering* of $\mathcal{G}$ if $\pi(i) < \pi(j)$ whenever $j \in DE(i, \mathcal{G})$.

2.2.2 Structural Causal Model

A structural causal model (SCM), as defined in [30], is a tuple $(S, P_{\boldsymbol{N}})$ consists of a collection S of d structural assignments

$$X_j := f_j(PA(j), N_j), \quad j = 1, \cdots, d \tag{2.2}$$

where $PA(j) \subseteq \{X_1, \cdots, X_d\} \setminus \{X_j\}$ are called *parents* of X_j. $P_{\boldsymbol{N}} = P_{N_1, \cdots, N_d}$ is a joint distribution over noise variables that are assumed to be jointly independent. A *causal graph*, $\mathcal{G} = (V, E)$, is a graphical representation of a SCM. It can be constructed by treating each X_i as a vertex and forming a directed edge from each of X_i's parent to X_i. The causal graph is often assumed to be a DAG [30]. A node i is said to have a direct effect on j if $(i, j) \in E$. A node i is said to have an indirect effect on j if $j \in DE(i, \mathcal{G})$. These notions are extensively used in **Chapter 3**, where a causal graph on the input features to a machine learning model is provided as input for model explanation.

2.3 Interpretability

Interpretability is the research area in which Shapley Flow (Chapter 3) belongs. There are no single definition of interpretability or explanability because of the diverse use cases on what we aim to understand [31]–[33]. For example, we may want to understand how features affect the model to either debug it or extract scientific knowledge from it [11], [14], [34]–[36]. In this scenario, methods such as feature attribution [11], [14] can help us figure out what features are most important for the prediction task. We may also be interested in understanding how the data affect the model behavior [25]–[27], [37]. Methods based on influence functions [37] can identify important training samples that are either noisy or incorrectly labeled. Alternatively, we may want to understand in which population does the model work best. Tools that report model performance on different cohorts[1] or requirements to clearly document the intended use case of a model [38] become essential in order to determine when best to delegate a task to the model [39]. Given this diverse nature of interpretability, we narrow our scope to focus on a branch of commonly used model interpretation methods, feature attribution [14], [15], [40]–[46].

In this section, we first give an overview of the field from the angle of whether the explanation is provided intrinsically by the model or through post-hoc analysis [32], [33], [47]. Then we dive into feature attribution methods, focusing on those with a game theoretic interpretation.

2.3.1 Intrinsic Explanation

Models that provide intrinsic explanations are referred to as inherently "interpretable" models. They includes methods such as sparse linear models (*i.e.*, model weights determines feature importance), low depth decision trees (*i.e.*, how the tree split on features can be visualized as a set of rules), and K Nearest Neighbors (KNN) with small K (*i.e.*, the most influential data point is the data point that is closest to the sample to be explained). These models, albeit interpretable, can be limited in their expressive power to achieve high predictive accuracy. Other inherently interpretable models sidestep the issue by increasing the complexity of the model, often at the cost of reduced interpretability. They

[1] An example tool would be https://erroranalysis.ai/

include Generalized Additive Models (*i.e.*, generalizes linear models to allow per feature non-linearity) [48], [49], interpretable CNN (*i.e.*, confining convolutional neural network to match templates for object parts) [50], [51], prototype based models (*i.e.*, each sample is classified by comparing similarity of the sample to some prototypical examples, often with similarity measured in some deep embedded space) [47], [52]–[54], and attention based models (*i.e.*, attention weights are used as an indicator for feature importance) [55]–[59]. These methods impose strong assumptions on both the form of interpretation and the form of model architecture, making them less flexible compared to post-hoc interpretation methods in which model training and interpretation are separately considered.

2.3.2 Post-hoc Explanation

When a model does not provide intrinsic interpretation or its intrinsic interpretation does not match the intended use (*e.g.*, explaining feature importance of a KNN), it can be explained post-hoc. Post-hoc explanation frees practitioners from juggling interpretation and other metrics at the same time. Most post-hoc explanation methods can be categorized into i) feature/concept attribution methods, that attribute model decisions to salient input features [11], [15], [18], [60] or user defined concepts [51], [61], and ii) sample based explanation methods, that attribute model decisions to salient training data [25], [37]. They usually involve perturbing the input or fitting a proxy inherently interpretable model (locally or globally) to mimic the model to explain [11], [14], [62], [63]. This raises the concern of explanation fragility (*e.g.*, the explanation is not robust to adversarial attack) and fidelity (*e.g.*, the proxy model does not accurately resemble the model to explain), which are active fields of research [32], [64]–[67].

2.3.3 Feature Attribution

Both intrinsic and post-hoc explanations include methods to attribute importance to features. For example, feature importance can be given by the weights of a linear model or by the drop in performance when a feature is randomly permuted [68]. Formally, given a model $f : \mathbb{R}^d \rightarrow \mathbb{R}$ that takes a set of inputs of dimension d to produces an output, and a target sample input $\boldsymbol{x} \in \mathbb{R}^d$ to explain, a feature attribution method quantifies the effect of each input on the output by producing a real valued attribution vector $atr_i(\boldsymbol{x}; f) \in \mathbb{R}$

for $i \in [1 \cdots d]$. The magnitude of each attribution, $|atr_i(\boldsymbol{x}; f)|$, signals the importance of the i^{th} feature from $\boldsymbol{x}$ (denoted as x_i) for the prediction. Our notation follows from [24].

While there are many feature attribution methods such as DeepLIFT [42], Layer-wise Relevance Propagation (LRP) [44], Local Interpretable Model-agnostic Explanations (LIME) [11], and gradient based methods [15], [69], [70], we focus on game theoretic feature attribution [14], [17], [23]. These methods rely on variations of the game theoretic concept, Shapley value [18], to assign importance to features. Game theoretic feature attribution methods are often preferable over other baselines because they are grounded on solid theory with intuitive axioms and are model agnostic (*i.e.*, only assumes the ability to evaluate function output given input). These properties are especially attractive as it is hard to evaluate model interpretation [64], [71]–[73] objectively. For a comprehensive overview of other feature attribution methods and their evaluation, please refer to [74].

2.3.4 Shapley Value applied to Feature Attribution

As introduced in **Section 2.1**, Shapley value can be adapted to measure feature importance [14], [16], [17], [23], [24], [75], [76]. The idea is to view each feature as a player in the cooperative game, and the payoff function can be defined as the model output given which features are present. However, unlike the moving table example where a person is either in $\mathcal{C}$ or not (*i.e.*, binary), features can be continuous or discrete. To resolve this issue, Shapley value based feature attribution methods, along with other popular feature attribution methods such as DeepLift [42] and Integrated Gradient [15], introduce the notion of a background or reference sample. The effect of the input to explain (*i.e.*, the target sample) on the output is measured with respect to the background. For example, in a healthcare setting, we may set the features in the background sample to values that are deemed typical for a disease. Now, the presence of a feature can be binary, in which case a feature can either take on the value of the target sample or the background sample depending on whether it is in $\mathcal{C}$.

Formally, given a target sample input $\boldsymbol{x}$, a background sample input $\boldsymbol{x'}$, and a model $f : \mathbb{R}^d \to \mathbb{R}$, Shapley value based feature attribution methods aim to explain the difference in output *i.e.*, $f(\boldsymbol{x}) - f(\boldsymbol{x'})$. Note that $\boldsymbol{x}$ and $\boldsymbol{x}'$ are of the same dimension d, and each entry can be either discrete or continuous. In the game theoretic language, this means

we are setting $v(\{\}) = f(\boldsymbol{x'})$ and $v(\mathcal{P}) = f(\boldsymbol{x})$. We assume a single background value for notational convenience, but the formalism easily extends to the common scenario of multiple background values or a distribution of background values, P, by defining $v(\{\}) = \mathbb{E}_{\boldsymbol{x'} \sim P} f(\boldsymbol{x'})$. However, the payoff function for a non-empty proper subset of $\mathcal{P}$ can be tricky to define [17], [22]–[24], [77]. We discuss common choices in the next section.

2.3.5 Payoff Function Definition

To define the payoff function based on f, one has to consider what is the model output given a coalition $\mathcal{C}$. One popular approach is to treat features independently [22], [24], [77]. This approach (referred to as *independent SHAP*) defines the value of a coalition, $v(\mathcal{C})$, as $f(X_\mathcal{C} = \boldsymbol{x}_\mathcal{C}, X_{\overline{\mathcal{C}}} = \boldsymbol{x}'_{\overline{\mathcal{C}}})$, where the capitalized letters denote random variables that are arguments to f, $\overline{\mathcal{C}}$ is the complement set of $\mathcal{C}$ (*i.e.*, $\overline{\mathcal{C}} = \mathcal{P} \setminus \mathcal{C}$), and the subscript index into a sample (*e.g.*, $\boldsymbol{x}_\mathcal{C}$ denotes all coordinates of $\boldsymbol{x}$ that corresponds to features in $\mathcal{C}$). Here, features in $\mathcal{C}$ take the target sample's value, and features not in $\mathcal{C}$ take the background sample's value, directly matching the presence and absence of players. This notion, while intuitive, can produce unrealistic or invalid sets of model input because it completely ignores the correlation among features. Consider an example from [23], when input features contains both the marital status and relationship, one can sample $X_\mathcal{C}$ containing "marital status=never married" and $X_{\overline{\mathcal{C}}}$ containing "relationship=husband", producing invalid model input.

Recognizing this deficiency, recent work explore modeling feature correlation for attribution, so that each coalition stays on the data manifold [14], [17], [23]. These methods (referred to as *on-manifold SHAP*) define the value of a coalition as the expected output of f given features in $\mathcal{C}$. That is $v(\mathcal{C}) = \mathbb{E}_{p(X_{\overline{\mathcal{C}}} = \boldsymbol{x}'_{\overline{\mathcal{C}}} | X_\mathcal{C} = \boldsymbol{x}_\mathcal{C})}(f(X_\mathcal{C} = \boldsymbol{x}_\mathcal{C}, X_{\overline{\mathcal{C}}} = \boldsymbol{x}'_{\overline{\mathcal{C}}}) | X_\mathcal{C} = \boldsymbol{x}_\mathcal{C})$. This conditional expectation is estimated with respect to the training dataset. However, this can result in situations in which features not used by the model are given non-zero attribution. For example, given that the presence of dog and frisbee are often correlated, even if f only functionally depends on frisbee (*e.g.*, the weight for dog is 0 when f is linear) for a dog detection task, on-manifold SHAP would attribute equal weights to both features. This is not desirable as a model that actually depends on dog would have the same attribution, despite this latter model would generalize better in real life.

TABLE 2.1: A comparison of relevant regularization penalties.

Method	Formulation	Sparsity	Grouping effect
LASSO [78]	$\|\boldsymbol{\theta}\|_1$	yes	no
l_2 [79]	$\frac{1}{2}\|\boldsymbol{\theta}\|_2^2$	no	yes
elastic net [80]	$\beta\|\boldsymbol{\theta}\|_1 + \frac{1}{2}(1-\beta)\|\boldsymbol{\theta}\|_2^2$	yes	yes
OWL [81]	$\sum_{i=1}^{n} w_i \lvert\boldsymbol{\theta}\rvert_{[i]}$	yes	yes
weighted LASSO [82]	$\|\boldsymbol{w} \odot \boldsymbol{\theta}\|_1$	yes	yes
weighted l_2 [82]	$\frac{1}{2}\|\boldsymbol{w} \odot \boldsymbol{\theta}\|_2^2$	no	no

2.3.6 Removing the Symmetry Axiom

Within on-manifold Shapley value based feature attribution methods, the Asymmetric Shapley Value (ASV) from [23] argues to remove the symmetry axiom. In the author's own words, "when redundancies exist in the data, we might instead seek a sparser explanation of the model's behaviour. Instead of uniformly distributing feature importance over redundant features, we might instead prefer to concentrate the importance on those features we deem more fundamental". Practically, ASV only averages over player orderings that are topological orderings of a causal graph (see **Section 2.2**). In the dog detection example, if we specify a causal edge pointing from the dog to the frisbee, ASV will transfer credits that would have been given to the frisbee in on-manifold SHAP to the dog, leaving the frisbee with 0 attribution.

2.4 Parameter Norm Regularization

Understanding parameter norm regularization is the key to understand the EYE penalty proposed in Chapter 4. Regularization helps reduce overfitting (*i.e.*, discrepancy between training and generalization performance) [83]. While there are many forms of regularization such as data augmentation [84], adversarial training [85], and early stopping [86], we use the term regularization to refer to parameter norm regularization/penalty [83] as it is one of the oldest and most commonly used forms of regularization [78]. Parameter norm regularization can be defined by solving the following optimization problem:

$$\hat{\boldsymbol{\theta}} = \arg\min_{\boldsymbol{\theta}} L(\boldsymbol{\theta}, X, \boldsymbol{y}) + \lambda J(\boldsymbol{\theta}) \tag{2.3}$$

where L is some loss function and J is a regularization term. $\boldsymbol{\theta}$ represents the model parameters. X are $\boldsymbol{y}$ are the input and target of a dataset. $\lambda \in \mathbb{R}_{\geq 0}$ is the tradeoff between loss and the regularization term.

The most common forms of J are the l_1 (LASSO) and l_2 regularization. Their functional forms are summarized in **Table 2.1**. They can be interpreted as placing a prior distribution on feature weights (*i.e.*, isotropic Laplace distribution for l_1 and isotropic Gaussian distribution for l_2) [87]. To understand their properties analytically, consider a simple setting in which L is quadratic in $\boldsymbol{\theta}$ with a diagonal Hessian that is positive definite (*i.e.*, a linear least squares regression problem with an orthogonal design matrix X). Denote the unregularized optimal solution as $\boldsymbol{\theta}^*$ and the i^{th} diagonal entry of the Hessian matrix as $H_{i,i} > 0$, the i^{th} entry of the l_1 regularized solution is $\hat{\theta}_i^{l_1} = \text{sign}(\theta_i^*) \max\left\{|\theta_i^*| - \lambda / H_{i,i}, 0\right\}$ where the sign function output $1/-1/0$ if the input is positive/negative/0, and the i^{th} entry of the l_2 regularized solution is $\hat{\theta}_i^{l_2} = \frac{H_{i,i}}{H_{i,i}+\lambda}\theta_i^*$. A detailed derivation can be found in Chapter 7 of [83]. This means that l_1 regularization can push a solution to exactly 0 while the l_2 regularization only shrinks the solution by a constant factor. The ability to set weights to exactly zero, referred to as sparsity, is desirable when one wants to perform feature selection or increase model interpretability by reducing the number of features presented to human. Thus, many extensions of the LASSO regularization have been proposed, including elastic net [80], ordered weighted LASSO (OWL) [81], adaptive LASSO [87], weighted LASSO [82], [88], [89], and group LASSO [90].

In **Table 2.1**, we summarize properties of several common regularization terms relevant to this dissertation. $\beta \in [0, 1]$ is a hyperparameter that controls the tradeoff between the l_1 and l_2 norms; $\boldsymbol{w}$ is a set of non-negative weights for each feature; $|\boldsymbol{\theta}|_{[i]}$ is the i^{th} largest parameter sorted by magnitude; and $\odot$ is the elementwise product. The sparsity property refers to whether the penalty would push the weights of some parameters to exactly zero. The grouping effect refers to whether correlated features will have similar weights in a linear least squares regression setting [80].

2.5 Out of Distribution Generalization

In order to safely apply a model, one needs to ensure that the model achieves good performance on both the training distribution and on realistic, out of distribution settings. In this section, we formalize the out of distribution (OOD) generalization problem in the supervised learning setting. Then, we will dive into a particular failure mode of OOD generalization: shortcut learning. This will provide the background needed for **Chapter 5**. For a comprehensive review of OOD generalization, please refer to [91].

2.5.1 Formalization of OOD Generalization

Formally, denote the feature space as $\mathcal{X}$ and the label space as $\mathcal{Y}$, X and Y are random variables with support in $\mathcal{X}$ and $\mathcal{Y}$ respectively. Given a dataset $\mathcal{D} = \{(\boldsymbol{x}^{(i)} \in \mathcal{X}, y^{(i)} \in \mathcal{Y})\}_{i=1}^{n}$ of n samples generated from the training distribution $P_{tr}(X, Y)$, the goal of supervised learning is to find an optimal model $f : \mathcal{X} \to \mathcal{Y}$ such that it generalizes well to the testing distribution $P_{te}(X, Y)$:

$$\arg\min_{f \in \mathcal{H}} \mathbb{E}_{X,Y \sim P_{te}} L(f(X), Y) \tag{2.4}$$

where $L : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}$ is a loss function that measures the regret between the predicted label and the ground truth label, and $\mathcal{H}$ is the hypothesis space of f. Traditional learning methods assume that $P_{tr} = P_{te}$ [92]. However, in reality, the distribution that we care about (*i.e.*, P_{te}) often differs from the training distribution. In response, the OOD generalization problem focuses on the more common setting in which $P_{tr} \neq P_{te}$. We will refer to a model that generalizes to P_{te} as a robust model.

2.5.2 Shortcut Learning

One reason that models struggle in the OOD setting is the existence of shortcuts. "Shortcuts are decision rules that perform well on standard benchmarks but fail to transfer to more challenging testing conditions, such as real-world scenarios" [93]. For example,

Berry *et al.* (2018) [94] showed that a model can misclassify a cow if it appears in uncommon locations (*e.g.*, beach instead of grassland). Here, location is a shortcut because it is correlated with the outcome of interest and is easier to learn than recognizing the cow itself. Note that task easiness is related to the inductive bias of a model (*e.g.*, its architecture, optimization procedure, training data, and loss function) and not necessarily aligned with human intuition. For example, while it is more natural for a human to use shape than texture to recognize an object, Geirhos *et al.* (2019) [95] showed that the opposite is true for deep neural networks. Similar examples have been reported in object recognition [95], healthcare [96], image captioning [97], and adversarial training [98]. For a comprehensive review of shortcut learning, please refer to [93].

Approaches to Mitigate Shortcuts

Mitigating shortcuts requires assumptions because all models fail to generalize if P_{te} is allowed to differ from P_{tr} arbitrarily [99]. Shortcut learning approaches place assumptions on how shortcuts relate P_{tr} with P_{te}, usually using a causal graph [9], [100]. Here, we focus on techniques that do not require samples from P_{te}. If samples from P_{te} are available, either with or without labels, one can apply transfer learning [101] or unsupervised domain adaptation techniques [102].

When no domain knowledge on shortcuts is available, one can use causal discovery methods to mine features that are causal ancestors of the output from the data and only use those features for prediction [103], [104]. By assuming that shortcuts utilize features that are not causal ancestors of the target, these methods mitigate the use of shortcuts. The lack of domain knowledge on shortcuts comes with limitations. For example, those methods cannot handle cases when shortcuts are perfectly correlated with other features or when shortcuts are correlated with unobserved confounders.

On the opposite end, when shortcuts are known *a priori* (*e.g.*, through feature attribution and domain knowledge), one can augment the dataset to decorrelate shortcuts with the target [105]–[109], regularize model parameters to not rely on shortcuts [9], [110]–[112], or optimize for the worst case error over a family of distributions induced by changes in shortcuts [113]–[115]. By assuming shortcuts are given, these methods bypass an important challenge of shortcut learning: identifying the shortcuts.

Between the two extremes are methods that exploit indirect knowledge of shortcuts. Our proposed method in **Chapter 5** falls into this category by assuming access to concepts/representations learned from data that are not affected by shortcuts. We defer detailed discussions of this setting to **Chapter 5**.

Chapter 3

Shapley Flow

3.1 Introduction

Explaining a model's predictions by assigning importance to its inputs (*i.e.*, feature attribution) is critical to many applications in which a user interacts with a model to either make decisions or gain a better understanding of a system [14], [15], [40]–[46], [116]. However, correlation among input features presents a challenge when estimating feature importance.

Consider a motivating example adapted from [29], in which we are given a model f that takes as input four features: the season of the year (X_1), whether or not it's raining (X_2), whether the sprinkler is on (X_3), and whether the pavement is wet (X_4) and outputs a prediction $f(\mathbf{x})$, representing the probability that the pavement is slippery (capital X denotes a random variable; lower case $\mathbf{x}$ denotes a particular sample). Assume, the inputs are related through the causal graph in **Figure 3.1**. When assigning feature importance, existing approaches that ignore this causal structure [22], [24], [77] assign zero importance to the season, since it only indirectly affects the outcome through the other input variables. However, such a conclusion may lead a user astray - since changing X_1 would most definitely affect the outcome.

Recognizing this limitation, researchers have recently proposed approaches that leverage the causal structure among the input variables when assigning credit [23], [117]. However, such approaches provide an incomplete picture of a system as they only assign credit to nodes in a graph. For example, the ASV method of [23] solves the earlier problem of ignoring indirect or upstream effects, but it does so by ignoring direct or downstream effects. In our example, season would get all the credit despite the importance of the other

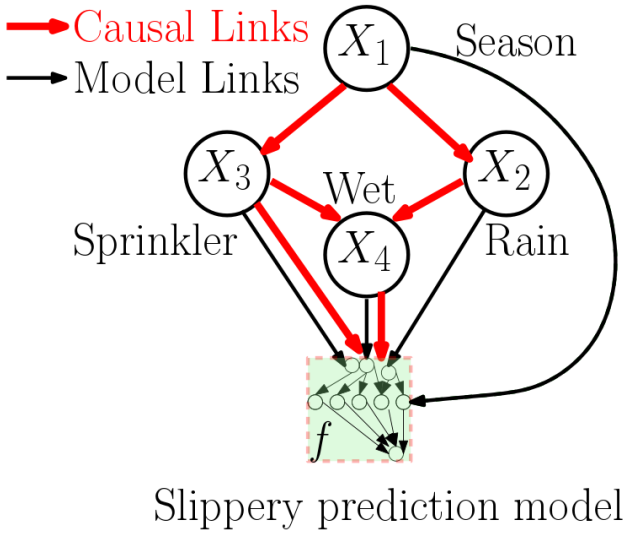


FIGURE 3.1: Causal graph for the sprinkler example from Chapter 1.2 of [29]. The model, f, can be expanded into its own graph. To simplify the exposition, although f takes 4 variables as input, we arbitrarily assumed that it only depends on X_3 and X_4 directly (*i.e.*, $f(X_1, X_2, X_3, X_4) = g(X_3, X_4)$ for some g).

variables. This again may lead a user astray - since intervening on X_3 or X_4 would affect the outcome, yet they are given no credit. The Causal Shapley values of [117] do assign credit to X_3 and X_4, but force this credit to be divided with X_1. This leads to the problem of features being given less importance simply because their downstream variables are also included in the graph.

Our approach. Given that current approaches end up ignoring or dividing either downstream (*i.e.*, direct) or upstream (*i.e.*, indirect) effects, we develop Shapley Flow, a comprehensive approach to interpreting a model (or system) that incorporates the causal relationship among input variables, while accounting for both direct and indirect effects. In contrast to prior work, we accomplish this by reformulating the problem as one related to assigning credit to *edges* in a causal graph, instead of *nodes*. **Figure 3.2** contrasts Shapley Flow with independent SHAP and ASV on the sprinkler example.

Our key contributions are as follows.

- We propose the first (to the best of our knowledge) generalization of Shapley value feature attribution to graphs, providing a complete system-level view of a model.

- Our approach unifies three previous game theoretic approaches to estimating feature importance.
- Through examples on real data, we demonstrate how our approach facilitates understanding feature importance.

In this chapter, we take an axiomatic approach motivated by cooperative game theory, extending Shapley values to graphs. The resulting algorithm, Shapley Flow, generalizes past work on estimating feature importance [14], [23], [118]. The estimates produced by Shapley Flow represent the unique allocation of credit that conforms to several natural axioms. Applied to real-world systems, Shapley Flow can help a user understand both the direct and indirect impact of changing a variable, generating insights beyond current feature attribution methods.

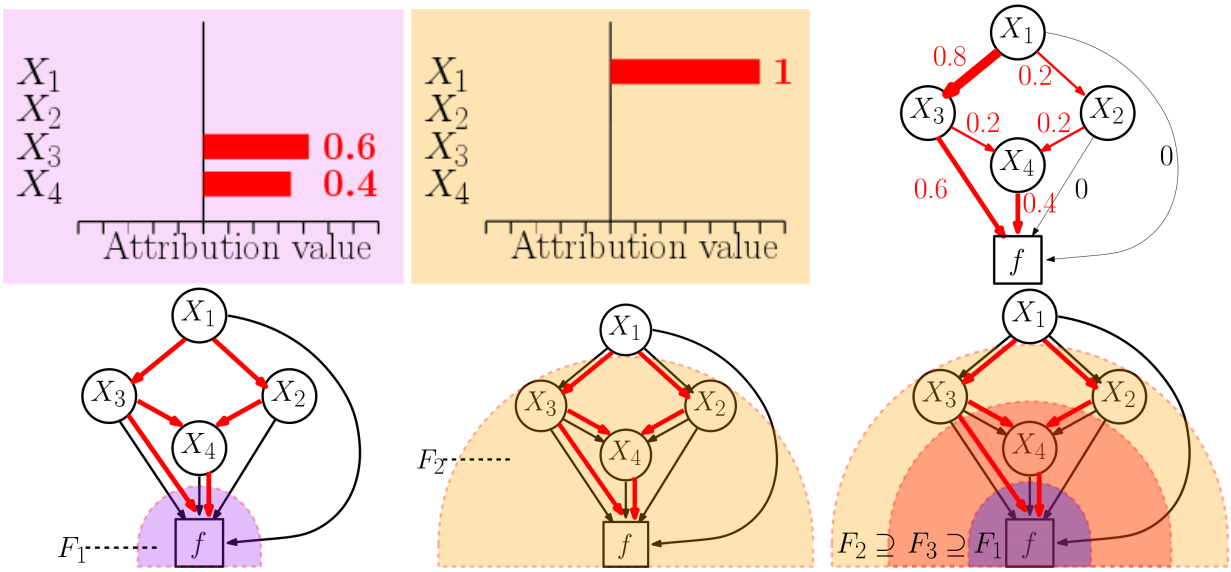
Organization. The rest of the chapter is organized as the following. First, we introduce background for feature attribution with a causal graph. Then, we propose Shapley Flow and show that it is the unique solution to an extension of Shapley value axioms to graphs. Next, we compare Shapley Flow to baseline methods using both linear and nonlinear models, displaying the pitfalls of previous feature attribution methods. Finally, we summarize our contribution and motivate future directions.

3.2 Background & Related Work

Given a model, or more generally a system, that takes a set of inputs and produces an output, we focus on the problem of quantifying the effect of each input on the output. Here, building off previous work, we formalize the problem setting.

3.2.1 Problem Setup

Quantifying the effect of each input on a model's output can be formulated as a credit assignment problem. Formally, given a target sample input $\boldsymbol{x}$, a background sample input $\boldsymbol{x'}$, and a model $f : \mathbb{R}^d \to \mathbb{R}$, we aim to explain the difference in output *i.e.*, $f(\boldsymbol{x}) - f(\boldsymbol{x'})$. We assume $\boldsymbol{x}$ and $\boldsymbol{x}'$ are of the same dimension d, and each entry can be either discrete or continuous.



(A) Independent (B) ASV (C) Shapley Flow

FIGURE 3.2: Top: Output of attribution methods for the example in **Figure 3.1**. Bottom: Causal structure (black edges) and explanation boundaries used by each method. As a reference, we copied the true causal links (red) from **Figure 3.1**. An explanation boundary $\mathcal{B} := (D, F)$ is a cut in the graph that defines a "model" F (nodes in the shaded area in each figure) to be explained. Refer to **Section 3.2.2** for a detailed discussion.

We also assume access to a causal graph, as formally defined in Chapter 6 of [30], over the d input variables. Given this graph, we seek an assignment function ϕ that assigns credit $\phi(e) \in \mathbb{R}$ to each edge e in the causal graph such that they collectively explain the difference $f(\boldsymbol{x}) - f(\boldsymbol{x'})$. In contrast with the classical setting [14]–[17] in which credit is placed on features (*i.e.*, seeking a node assignment function $\psi(i) \in \mathbb{R}$ for $i \in [1 \cdots d]$), our edge-based approach is more flexible because we can recover node i's importance by defining $\psi(i) = \sum_{e \in \text{i's outgoing edges}} \phi(e)$. This exactly matches the classic Shapley axioms [18] when the causal graph is degenerate with a single source node connected directly to all the input features.

Here, the effect of the input on the output is measured with respect to a background sample. For example, in a healthcare setting, we may set the features in the background sample to values that are deemed typical for a disease. We assume a single background value for notational convenience, but the formalism easily extends to the common scenario of multiple background values or a distribution of background values, P, by defining the explanation target to be $f(\boldsymbol{x}) - \mathbb{E}_{\boldsymbol{x'} \sim P} f(\boldsymbol{x'})$.

3.2.2 Feature Attribution with a Causal Graph

In our problem setup, we assume access to a causal graph, which can help in reasoning about the relationship among input variable. However, even with a causal graph, feature attribution remains challenging because it is unclear how to rightfully allocate credit for a prediction among the nodes and/or edges of the graph. Marrying interpretation with causality is an active field (see [119] for a survey). A causal graph in and of itself does not solve feature attribution. While a causal graph can be used to answer a specific question with a specific counterfactual, summarizing many counterfactuals to give a comprehensive picture of the model is nontrivial. Furthermore, each node in a causal graph could be a blackbox model that needs to be explained. To address this challenge, we generalize game theoretic fairness principles to graphs.

Given a graph, $\mathcal{G}$, that consists of a causal graph over the the model of interest f and its inputs, we define the **boundary of explanation** as a cut $\mathcal{B} := (D, F)$ that partitions the input variables and the output of the model (*i.e.*, the nodes of the graph) into D and F where source nodes (nodes with no incoming edges) are in D and sink nodes (nodes with

no outgoing edges) are in F. Note that $\mathcal{G}$ has a single sink, $f(\boldsymbol{x}) \in \mathbb{R}$. A cut set is the set of edges with one endpoint in D and another endpoint in F, denoted as $cut(\mathcal{B})$. It is helpful to think of F as an alternative model definition, where a boundary of explanation (*i.e.*, a model boundary) defines what part of the graph we consider to be the "model". If we collapse F into a single node that subsumes f, then $cut(\mathcal{B})$ represents the direct inputs to this new model.

Depending on the causal graph, multiple boundaries of explanation may exist. Recognizing this multiplicity of choices helps shed light on an ongoing debate in the community regarding feature attribution and whether one should perturb features while staying on the data manifold or perturb them independently [24], [77], [120]. On one side, many argue that perturbing features independently reveals the functional dependence of the model, and is thus *true to the model* [22], [24], [77]. However, independent perturbation of the data can create unrealistic or invalid sets of model input values. Thus, on the other side, researchers argue that one should perturb features while staying on the data manifold, and so be *true to the data* [17], [23]. However, this can result in situations in which features not used by the model are given non-zero attribution. Explanation boundaries help us unify these two viewpoints. As illustrated in **Figure 3.2a**, when we independently perturb features, we assume the causal graph is flat and the explanation boundary lies between $\boldsymbol{x}$ and f (*i.e.*, D contains all of the input variables). In this example, since features are assumed independent all credit is assigned to the features that directly impact the model output, and indirect effects are ignored (no credit is assigned to X_1 and X_2). In contrast, when we perform on-manifold perturbations with a causal structure, as is the case in Asymmetric Shapley Values (ASV) [23], all the credit is assigned to the source node because the source node determines the value of all nodes in the graph (**Figure 3.2b**). This results in a different boundary of explanation, one between the source nodes and the remainder of the graph. Although giving X_1 credit does not reflect the true functional dependence of f, it does for the model defined by F_2 (**Figure 3.2c**). Perturbations that were previously faithful to the data are faithful to a "model", just one that corresponds to a different boundary. See **Section A.1** in the Appendix for how on-manifold perturbation (without a causal graph) can be unified using explanation boundaries.

Beyond the boundary directly adjacent to the model of interest, f, and the boundary directly adjacent to the source nodes, there are other potential boundaries (**Figure 3.2c**) a

user may want to consider. However, simply generating explanations for each possible boundary can quickly overwhelm the user (**Figures 3.2a, 3.2b** in the main text, and **A.1a** in the Appendix). Our approach sidesteps the issue of selecting a single explanation boundary by considering all explanation boundaries simultaneously. This is made possible by assigning credit to the edges in a causal graph (**Figure 3.2c**). Edge attribution is strictly more powerful than feature attribution because we can simultaneously capture the direct and indirect impact of edges. We note that concurrent work by [117] also recognized that existing methods have difficulty capturing the direct and indirect effects simultaneously. Their solution however is node based, so it is forced to split credit between parents and children in the graph.

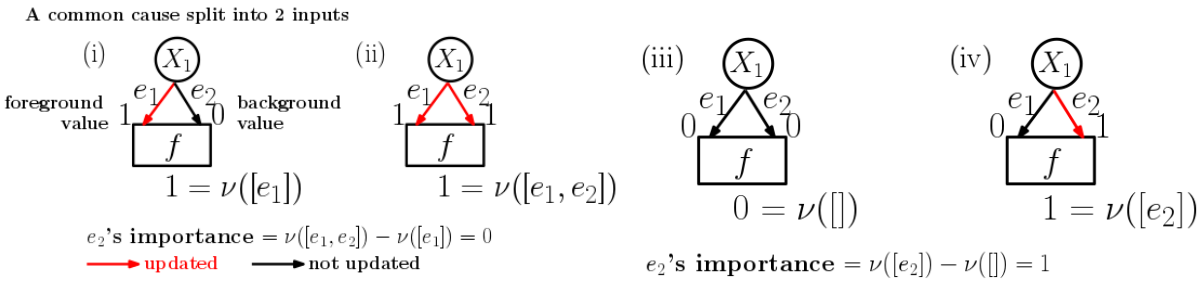
While other approaches to assign credit on a graph exist, (*e.g.*, Conductance from [70] and DeepLift from [121]), they were proposed in the context of understanding internal nodes of a neural network, and depend on implicit linearity and continuity assumptions about the model. We aim to understand the causal structure among the input nodes in a fully model agnostic manner, where discrete variables are allowed, and no differentiability assumption is made. To do this we generalize the widely used Shapley value [23], [24], [75], [77], [120], [122], [123] to graphs.

3.3 Methods

Our proposed approach, Shapley Flow, attributes credit to edges of the causal graph. In this section, we present the intuition behind our approach and then formally show that it uniquely satisfies a generalization of the classic Shapley value axioms, while unifying previously proposed approaches.

3.3.1 Assigning Credit to Edges: Intuition

Given a causal graph defining the relationship among input variables, we re-frame the problem of feature attribution to focus on the edges of a graph rather than nodes. Our approach results in edge credit assignments as shown in **Figure 3.2c**. This eliminates the need for multiple explanations (*i.e.*, bar charts) pertaining to each explanation boundary.



(A) e_2 updates after e_1

(B) e_2 updates before e_1

FIGURE 3.3: Edge importance is measured by the change in output when an edge is added. When a model is non-linear, say $f = OR$, we need to average over all scenarios in which e_2 can be added to gauge its importance. **Section 3.3.1** has a detailed discussion.

Moreover, it allows a user to better understand the nuances of a system by providing information regarding what would happen if a single causal link breaks.

Shapley Flow is the unique assignment of credit to edges such that a relaxation of the classic Shapley value axioms are satisfied for all possible boundaries of explanation. Specifically, we extend the efficiency, dummy, and linearity axioms from [18] and add a new axiom related to boundary consistency. Efficiency states that the attribution of edges on any boundary must add up to $f(\mathbf{x}) - f(\mathbf{x}')$. Linearity states that explaining a linear combination of models is the same as explaining each model, and linearly combining the resulting attributions. Dummy states that if adding an edge does not change the output in any scenarios, the edge should be assigned 0 credit. Boundary consistency states that edges shared by different boundaries need to have the same attribution when explained using either boundary. These concepts are illustrated in **Figure 3.4** and formalized in **Section 3.3.3**.

An edge is important if removing it causes a large change in the model's prediction. However, what does it mean to remove an edge? If we imagine every edge in the graph as a channel that sends its source node's current value to its target node, then removing an edge e simply means messages sent through e fail. In the context of feature attribution, in which we aim to measure the difference between $f(\mathbf{x}) - f(\mathbf{x}')$, this means that e's target node still relies on the source's background value in $\mathbf{x}'$ to update its current value, as opposed to the source node's foreground value in $\mathbf{x}$, as illustrated in **Figure 3.3a**. Note that treating edge removal as replacing the parent node with the background value is

equivalent to the approach advocated by [24], and matches the default behavior of SHAP and related methods. However, we cannot simply toggle edges one at a time. Consider a simple OR function $g(X_1, X_2) = X_1 \vee X_2$, with $x_1 = 1$, $x_2 = 1$, $x_1' = 0$, $x_2' = 0$. Removing either of the edges alone, would not affect the output and both x_1 and x_2 would be (erroneously) assigned 0 credit.

To account for this, we consider all scenarios (or partial histories) in which the edge we care about can be added (see **Figure 3.3b**). Here, ν is a function that takes a list of edges and evaluates the network with edges updated in the order specified by the list. For example, $\nu([e_1])$ corresponds to the evaluation of f when only e_1 is updated. Similarly $\nu([e_1, e_2])$ is the evaluation of f when e_1 is updated followed by e_2. The list $[e_1, e_2]$ is also referred to as a (complete) *history* as it specifies how $\mathbf{x}'$ changes to $\mathbf{x}$.

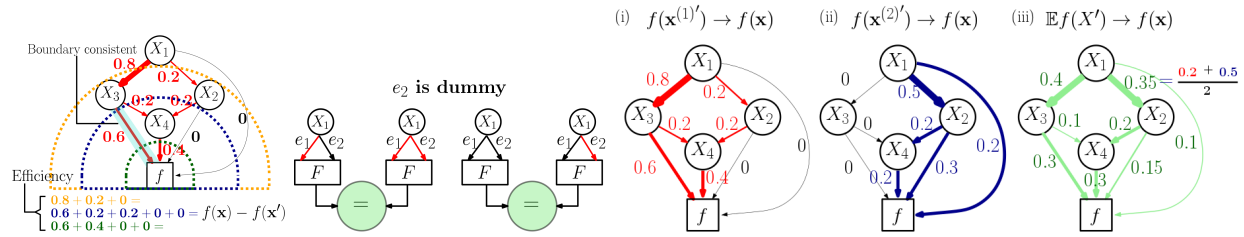
For the same edge, attributions derived from different explanation boundaries should agree, otherwise simply including more details of a model in the causal graph would change upstream credit allocation, even though the model implementation was unchanged. We refer to this property as *boundary consistency*.

3.3.2 Model explanation as value assignments in games

The concept of Shapley value stems from game theory, and has been extensively applied in model interpretability [14], [21]–[24]. Before we formally extend it to the context of graphs, we define the credit assignment problem from a game theoretic perspective.

Given the message passing system in **Section 3.3.1**, we formulate the credit assignment problem as a game specific to an explanation boundary $\mathcal{B} := (D, F)$. The game consists of a set of players $\mathcal{P}_\mathcal{B}$, and a payoff function $\nu_\mathcal{B}$. We model each edge external to F as a player. A *history* is a list of edges detailing the event from $t = 0$ (values being $\mathbf{x}'$) to $t = T$ (values being $\mathbf{x}$). For example, the history $[i, j, i]$ means that the edge i finishes transmitting a message containing its source node's most recent value to its target node, followed by the edge j, and followed by the edge i again. A *coalition* is a partial history from $t = 0$ to any $t \in [0 \cdots T]$. The *payoff function*, ν, associates each coalition with a real number, and is defined in our case as the evaluation of F following the coalition.

This setup is a generalization of a typical cooperative game in which the ordering of players does not matter (only the set of players matters). However, given our message



(A) Effi. + Bound. Consist. (B) Dummy player (C) Linearity

FIGURE 3.4: Illustration of axioms for Shapley Flow. Except for boundary consistency, all axioms stem from Shapley value's axioms [18]. Detailed explanations are included in **Section 3.3.3**.

passing system, history is important. In the following sections, we denote '+' as list concatenation, '[]' as an empty coalition, and $\mathcal{H}_{\mathcal{B}}$ as the set of all possible histories. We denote $\tilde{\mathcal{H}}_{\mathcal{B}} \subseteq \mathcal{H}_{\mathcal{B}}$ as the set of boundary consistent histories. The corresponding coalitions for $\mathcal{H}_{\mathcal{B}}$ and $\tilde{\mathcal{H}}_{\mathcal{B}}$ are denoted as $\mathcal{C}_{\mathcal{B}}$ and $\tilde{\mathcal{C}}_{\mathcal{B}}$ respectively. A sample game setup is illustrated in **Figure 3.3**.

3.3.3 Axioms

We formally extend the classic Shapley value axioms (efficiency, linearity, and dummy) and include one additional axiom, the boundary consistency axiom, that connects all boundaries together.

- Boundary consistency: for any two boundaries $\mathcal{B}_1 = (D_1, F_1)$ and $\mathcal{B}_2 = (D_2, F_2)$, $\phi_{\nu_{\mathcal{B}_1}}(i) = \phi_{\nu_{\mathcal{B}_2}}(i)$ for $i \in cut(\mathcal{B}_1) \cap cut(\mathcal{B}_2)$

  For edges that are shared between boundaries, their attributions must agree. In **Figure 3.4a**, the edge wrapped by a teal band is shared by both the blue and green boundaries, forcing them to give the same attribution to the edge.

In the general setting, not all credit assignments are boundary consistent; different boundaries could result in different attributions for the same edge to demonstrate why considering all histories $\mathcal{H}$ can violate boundary consistency, thus motivating the need to only focus on boundary consistent histories. This occurs when histories associated with different boundaries are inconsistent (**Figure 3.5**). Moving the boundary from $\mathcal{B}$ to $\mathcal{B}^*$

(where $\mathcal{B}^*$ is the boundary with D containing f's inputs), results in a more detailed set of histories. This expansion has 2 constraints. First, any history in the expanded set follows the message passing system in **Section 3.3.1**. Second, when a message passes through the boundary, it immediately reaches the end of computation as F is assumed to be a black-box.

Denoting the history expansion function into $\mathcal{B}^*$ as HE (*i.e.*, HE takes a history h as input and expand it into a set of histories in $\mathcal{B}^*$ as output) and denoting the set of all boundaries as $\mathcal{M}$, a history h is *boundary consistent* if $\exists h_{\mathcal{B}} \in \mathcal{H}_{\mathcal{B}}$ for all $\mathcal{B} \in \mathcal{M}$ such that

$$(\bigcap_{\mathcal{B}\in\mathcal{M}} HE(h_{\mathcal{B}})) \cap HE(h) \neq \emptyset$$

That is h needs to have at least one fully detailed history in which all boundaries can agree on. $\tilde{\mathcal{H}}$ is all histories in $\mathcal{H}$ that are boundary consistent. We rely on this notion of boundary consistency in generalizing the Shapley axioms to any explanation boundary, $\mathcal{B}$:

- Efficiency: $\sum_{i\in cut(\mathcal{B})} \phi_{\nu_{\mathcal{B}}}(i) = f(\mathbf{x}) - f(\mathbf{x}')$.

  In the general case where $\nu_{\mathcal{B}}$ can depend on the ordering of h, the sum is $\sum_{h\in\tilde{\mathcal{H}}_{\mathcal{B}}} \frac{\nu_{\mathcal{B}}(h)}{|\tilde{\mathcal{H}}_{\mathcal{B}}|} - \nu_{\mathcal{B}}([])$. But when the game is defined by a model function f, $\sum_{h\in\tilde{\mathcal{H}}_{\mathcal{B}}} \nu_{\mathcal{B}}(h)/|\tilde{\mathcal{H}}_{\mathcal{B}}| = f(\boldsymbol{x})$ and $\nu_{\mathcal{B}}([]) = f(\boldsymbol{x'})$. An illustration with 3 boundaries is shown in **Figure 3.4a**.

- Linearity: $\phi_{\alpha u+\beta v} = \alpha\phi_u + \beta\phi_v$ for any payoff functions u and v and scalars α and β.

  Linearity enables us to compute a linear ensemble of models by independently explaining each model and then linearly weighting the attributions. Similarly, we can explain $f(\mathbf{x}) - \mathbb{E}(f(X'))$ by independently computing attributions for each background sample $\mathbf{x}^{(\mathbf{i})'}$ and then taking the average of the attributions, without recomputing from scratch whenever the background sample's distribution changes. An illustration with 2 background samples is shown in **Figure 3.4c**.

- Dummy player: $\phi_{\nu_{\mathcal{B}}}(i) = 0$ if $\nu_{\mathcal{B}}(S + [i]) = \nu_{\mathcal{B}}(S)$ for all $S, S + [i] \in \tilde{\mathcal{C}}_{\mathcal{B}}$ for $i \in cut(\mathcal{B})$.

  Dummy player states that if an edge does not change the model's output when added to in all possible coalitions, it should be given 0 attribution. In **Figure 3.4b**, e_2 is a dummy edge because starting from any coalition, adding e_2 wouldn't change the output.

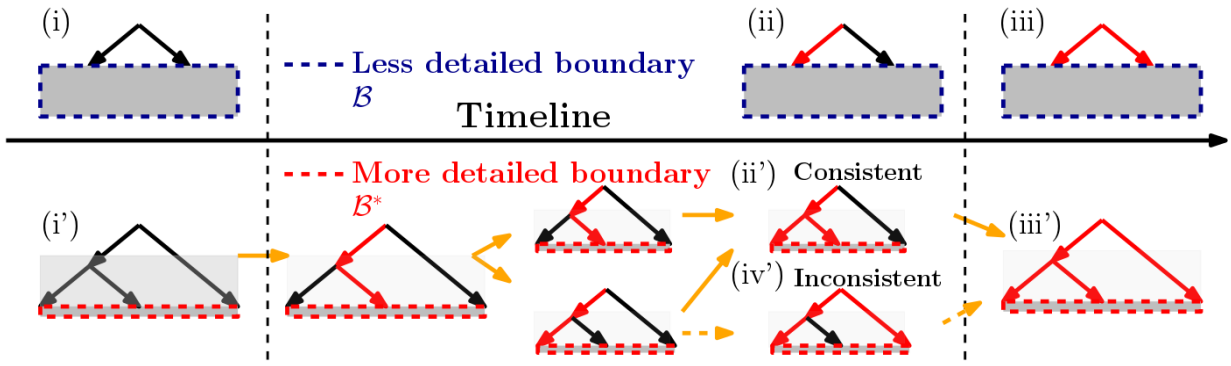


FIGURE 3.5: Boundary Consistency. For the blue boundary (upper), we show one potential history h. When we expand h to the red boundary (lower), h corresponds to multiple histories as long as each history contains states that match (i) (ii) and (iii). (i') matches (i), no messages are received in both states. (ii') matches (ii), the full impact of message transmitted through the left edge is received at the end of computation. (iii') matches (iii), all messages are received. In contrast, the history containing (iv') has no state matching (ii), and thus is inconsistent with h.

These last three axioms are extensions of Shapley's axioms. Note that Shapley value also requires the symmetry axiom because the game is defined on a set of players. For Shapley Flow values this symmetry assumption is encoded through our choice of an ordered history formulation.

3.3.4 Shapley Flow is the unique solution

Shapley Flow uniquely satisfies all axioms from the previous section. Here, we describe the algorithm, show its formulae, and state its properties. Please refer to **Appendix A.2** and **A.3** for the pseudo code[1] and proof.

Description. Define a configuration of a graph as an arbitrary ordering of outgoing edges of a node when it is traversed by depth first search. For each configuration, we run depth first search starting from the source node, processing edges in the order of the configuration. When processing an edge, we update the value of the edge's target node by making the edge's source node value visible to its function. If the edge's target node is the sink node, the difference in the sink node's output is credited to every edge along

[1]code can be found in `https://github.com/nathanwang000/Shapley-Flow`

the search path from source to sink. The final result averages over attributions for all configurations.

Formulae. Denote the attribution of Shapley Flow to a path as $\tilde{\phi}_v$, and the set of all possible orderings of source nodes to a sink path generated by depth first search (DFS) as Π_{dfs}. For each ordering $\pi \in \Pi_{\text{dfs}}$, the inequality of $\pi(j) < \pi(i)$ denotes that path j precedes path i under π. Since v's input is a list of edges, we define $\tilde{v}$ to work on a list of paths. The evaluation of $\tilde{v}$ on a list of paths is the value of v evaluated on the corresponding edge traversal ordering. Then

$$\tilde{\phi}_v(i) = \sum_{\pi \in \Pi_{\text{dfs}}} \frac{\tilde{v}([j : \pi(j) \leq \pi(i)]) - \tilde{v}([j : \pi(j) < \pi(i)])}{|\Pi_{\text{dfs}}|} \tag{3.1}$$

To obtain an edge e's attribution $\phi_v(e)$, we sum the path attributions for all paths that contains e.

$$\phi_v(e) = \sum_{p \in \text{paths in } \mathcal{G}} \mathbb{1}_{p \text{ contains}}(e) \tilde{\phi}_v(p) \tag{3.2}$$

Additional properties. Shapley Flow has the following beneficial properties beyond the axioms.

- Generalization of SHAP: if the graph is flat, the edge attribution is equal to feature attribution from SHAP because each input node is paired with a single edge leading to the model.
- Generalization of ASV: the attribution to the source nodes is the same as in ASV if all the dependencies among features are modeled by the causal graph.
- Generalization of Owen value: if the graph is a tree, the edge attribution for incoming edges to the leaf nodes is the Owen value [118] with a coalition structure defined by the tree.
- Implementation invariance: implementation invariance means that no matter how the function is implemented, so long as the input and output remain unchanged, so does the attribution [15], which directly follows boundary consistency (*i.e.*, knowing f's computational graph or not wouldn't change the upstream attribution).

- Conservation of flow: efficiency and boundary consistency imply that the sum of attributions on a node's incoming edges equals the sum of its outgoing edges.
- Model agnostic: Shapley Flow can explain arbitrary (non-differentiable) machine learning pipelines.

3.4 Experiments & Results

Shapley Flow highlights both the direct and indirect impact of features. In this section, we consider several applications of Shapley Flow. First, in the context of a linear model, we verify that the attributions match our intuition. Second, we show how current feature attribution approaches lead to an incomplete understanding of a system compared to Shapley Flow. In particular, we seek to answer the following questions:

- Question 1: Does Shapley Flow capture the ground truth direct and indirect effects of linear models? (**Section 3.4.3**, **Table 3.1**)
- Question 2: Does Shapley Flow capture the insights of and beyond the baselines on non-linear models? (**Section 3.4.4**, **Figure 3.6a**)
- Question 3: Why are on-manifold explanations misleading? (**Section 3.4.4**, **Figure 3.6**, **Figure 3.7a**, **Figure 3.7b**)

3.4.1 Experimental Setup

We illustrate the application of Shapley Flow to a synthetic and a real dataset. In addition, we include results for a third dataset in the Appendix. Note that our algorithm assumes a causal graph is provided as input. In recent years there has been significant progress in causal graph estimation [30], [124]. However, since our focus is not on causal inference, we make simplifying assumptions in estimating the causal graphs (see **Section A.4.2** of the Appendix for details).

Datasets. *Synthetic*: As a sanity check, we first experiment with synthetic data. We create a random graph dataset with 10 nodes. A node i is randomly connected to node j (with j pointing to i) with 0.5 probability if $i > j$, otherwise 0. The function at each node is linear with weights generated from a standard normal distribution. Sources follow a $N(0, 1)$ distribution. This results in a graph with a single sink node associated with function f (*i.e.*, the 'model' of interest). The remainder of the graph corresponds to the causal structure among the input variables.

National Health and Nutrition Examination Survey: This dataset consists of $9,932$ individuals with 18 demographic and laboratory measurements [125]. We used the same preprocessing as described by [76]. Given these inputs, the model, f, aims to predict survival.

Model training. We train f using an $80/20$ random train/test split. For experiments with linear models, f is trained with linear regression. For experiments with non-linear models, f is fitted by 100 XGBoost trees with a max depth of 3 for up to 1000 epochs, using the Cox loss.

Causal Graph. For the nutrition dataset, we constructed a causal graph (**Figure A.2a**) based on our limited understanding of the causal relationship among input variables. This graph represents an oversimplification of the true underlying causal relationships and is for illustration purposes only. We assigned attributes predetermined at birth (age, race, and sex) as source nodes because they temporally precede all other features. Poverty index depends on age, race, and sex (among other variables captured by the poverty index noise variable) and impacts one's health. Other features pertaining to health depend on age, race, sex, and poverty index. Note that the relationship among some features is deterministic. For example, pulse pressure is the difference between systolic and diastolic blood pressure. We include causal edges to account for such facts. We also account for when features have natural groupings. For example, transferrin saturation (TS), total iron binding capacity (TIBC), and serum iron are all related to blood iron. Serum albumin and serum protein are both blood protein measures. Systolic and diastolic blood pressure can be grouped into blood pressure. Sedimentation rate and white blood cell counts both measure inflammation. We add these higher level grouping concepts as new

latent variables in the graph. To account for noise in modeling the outcome (*i.e.*, the effect of exogenous variables that are not used as input to the model), we add an independent noise node to each node (detailed in **Section A.4.2** in the Appendix). **The resulting causal structure is an oversimplification of the true causal structure; the relationship between source nodes (e.g., race) and biomarkers is far more complex [126]. Nonetheless, it can help in understanding the in/direct effects of input variables on the outcome**.

3.4.2 Baselines

We compare Shapley Flow with other game theoretic feature attribution methods: independent SHAP [14], on-manifold SHAP [17], and ASV [23], covering both independent and on-manifold feature attribution.

Since Shapley value based methods are expensive to compute exactly, we use a Monte Carlo approximation of **Equation 3.1**. In particular, we sample orderings from Π_{dfs} and average across those orderings.We randomly selected a background sample from each dataset and share it across methods so that each uses the same background. A single background sample allows us to ignore differences in methods due to variations in background sampling and is easier to explain the behavior of baselines [127]. To show that our result is not dependent on the particular choice of background sample, we include an example averaged over 100 background samples in **Section A.5.4** in the Appendix (the qualitative results shown with a single background still holds). We sample $10,000$ orderings from each approach to generate the results. Since there's no publicly available implementation for ASV, we show the attribution for source nodes (the noise node associated with each feature) obtained from Shapley Flow (summing attributions of outgoing edges), as they are equivalent given the same causal graph. Since noise node's credit is used, intermediate nodes can report non zero credit in ASV.

For convenience of visual inspection, we show top 10 links used by Shapley Flow (credit measured in absolute value) on the nutrition dataset.

3.4.3 Sanity checks with linear models

To build intuition, we first examine linear models (*i.e.*, $f(\mathbf{x}) = \mathbf{w}^\top \mathbf{x} + b$ where $\mathbf{w} \in \mathbb{R}^d$ and $b \in \mathbb{R}$; the causal dependence inside the graph is also linear). When using a linear

Methods	Nutrition (**D**)	Synthetic (**D**)	Nutrition (**I**)	Synthetic (**I**)
Independent	**0.0** ($\pm$ 0.0)	**0.0** ($\pm$ 0.0)	0.8 ($\pm$ 2.7)	1.1 ($\pm$ 1.4)
On-manifold	1.3 ($\pm$ 2.5)	0.8 ($\pm$ 0.7)	0.9 ($\pm$ 1.6)	1.5 ($\pm$ 1.5)
ASV	1.5 ($\pm$ 3.3)	1.2 ($\pm$ 1.4)	0.6 ($\pm$ 1.9)	1.1 ($\pm$ 1.5)
Shapley Flow	**0.0** ($\pm$ 0.0)	**0.0** ($\pm$ 0.0)	**0.0** ($\pm$ 0.0)	**0.0** ($\pm$ 0.0)

TABLE 3.1: Mean absolute error (std) for all methods on direct (**D**) and indirect (**I**) effect for linear models. Shapley Flow makes no mistake across the board.

model the ground truth direct impact of changing feature X_i is $w_i(x_i - x_i')$ (that is the change in output due to X_i directly), and the ground truth indirect impact is defined as the change in output when an intervention changes x_i' to x_i. Note that when the model is linear, only 1 Monte Carlo sample is sufficient to recover the exact attribution because feature ordering doesn't matter (the output function is linear in any boundary edges, thus only the background and foreground value of a feature matters). This allows us to bypass sampling errors and focus on analyzing the algorithms.

Results for explaining the datasets are included in **Table 3.1**. We report the mean absolute error (and its variance) associated with the estimated attribution (compared against the ground truth attribution), averaged across $1,000$ randomly selected test examples and all graph nodes for both datasets. Note that only Shapley flow results in no error for both direct and indirect effects.

3.4.4 Examples with non-linear models

We demonstrate the benefits of Shapley Flow with non-linear models containing both discrete and continuous variables. As a reminder, the baseline methods are not competing with Shapley Flow as the latter can recover all the baselines given the corresponding causal structure (**Figure 3.2**). Instead, we highlight why a holistic understanding of the system is better.

Independent SHAP ignores the indirect impact of features. Take an example from the nutrition dataset (**Figure 3.6**). Independent SHAP gives lower attribution to age compared to ASV. This happens because age, in addition to its direct impact, indirectly affects

the output through blood pressure, as shown by Shapley Flow (**Figure 3.6a**). Independent SHAP fails to account for the indirect impact of age, leaving the user with a potentially misleading impression that age is less important than it actually is.

On-manifold SHAP provides a misleading interpretation. With the same example as before (**Figure 3.6**), we observe that on-manifold SHAP strongly disagrees with independent SHAP, ASV, and Shapley Flow on the importance of age. Not only does it assign more credit to age, it also flips the sign, suggesting that age is protective. However, **Figure 3.7a** shows that age and earlier mortality are positively correlated; then how could age be protective? **Figure 3.7b** provides an explanation. Since SHAP considers all partial histories regardless of the causal structure, when we focus on serum magnesium and age, there are two cases: serum magnesium updates before or after age. We focus on the first case because it is where on-manifold SHAP differs from other baselines (all baselines already consider the second case as it satisfies the causal ordering). When serum magnesium updates before age, the expected age given serum magnesium is higher than the foreground age (yellow line above the black marker). Therefore when age updates to its foreground value, we observe a decrease in age, leading to a decrease in the output (so age appears to be protective). From both an in/direct impact perspective, on-manifold perturbation can be misleading since it is based not on causal but on observational relationships.

ASV ignores the direct impact of features. As shown in **Figure 3.6**, serum protein appears to be more important in independent SHAP compared to ASV. From Shapley Flow (**Figure 3.6a**), we know serum protein is not given attribution in ASV because its upstream node, blood protein, gets all the credit. However, looking at ASV alone, one fails to understand that intervening on serum protein could have a larger impact on the output.

Shapley Flow shows both direct and indirect impacts of features. Focusing on the attribution given by Shapley Flow (**Figure 3.6a**). We not only observe similar direct impacts in variables compared to independent SHAP, but also can trace those impacts to their source nodes, similar to ASV. Furthermore, Shapley Flow provides more detail compared to other approaches. For example, using Shapley Flow we gain a better understanding

of the ways in which age impacts survival. The same goes for all other features. This is useful because causal links can change (or break) over time. Our method provides a way to reason through the impact of such a change.

More case studies with an additional dataset are included in the Appendix.

3.5 Summary & Conclusions

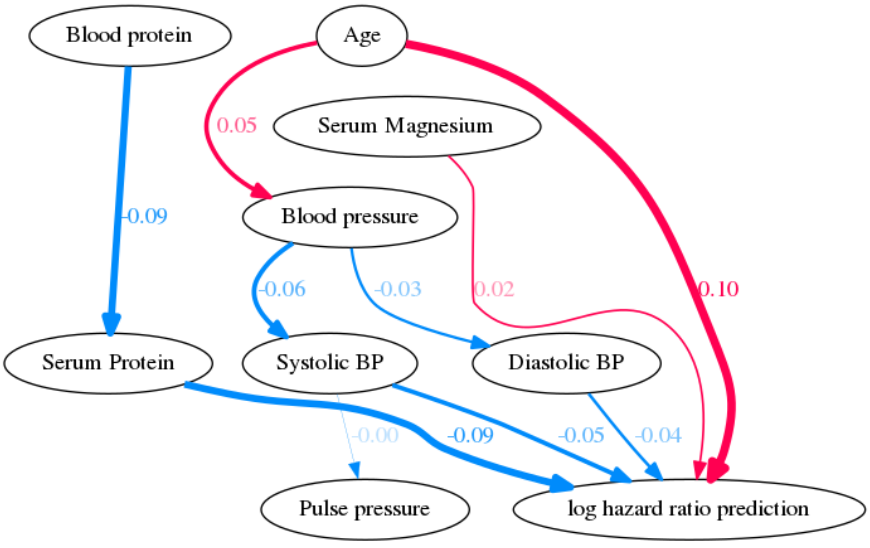
In this chapter, we extend the classic Shapley value axioms to causal graphs, resulting in a unique edge attribution method: Shapley Flow. It unifies three previous Shapley value based feature attribution methods and enables the joint understanding of both the direct and indirect impact of features. This more comprehensive understanding is useful when interpreting any machine learning model, both "black box" methods and "interpretable" methods (such as linear models).

The key message of this chapter is that model interpretation methods should account for the entire machine learning pipeline to understand the impact of features. As we demonstrated through experiments, baseline approaches ignore aspects of a model and can be misleading when accounting for correlation but not causal relationships among input features. In contrast, Shapley Flow generates more insights about the model than baselines.

While our approach relies on access to a complete causal graph, Shapley Flow is still valuable because a) there are well-established causal relationships in domains such as healthcare and ignoring such relationships can produce confusing explanations; b) recent advancements in causal estimation are complementary to our work and make defining these graphs easier; c) finally and most importantly, existing methods already implicitly make causal assumptions, Shapley Flow makes these assumptions explicit (**Figure 3.2**). However, this does open up new research opportunities. Can Shapley Flow work with partially defined causal graphs? How to explore Shapley Flow attribution when the causal graph is complex? How sensitive is Shapley Flow to a wrongly specified causal graph as experts can be wrong? These question are important to answer in order to safely apply our approach. We leave those questions for future work and offer suggestions to tackle them in Chapter 6.

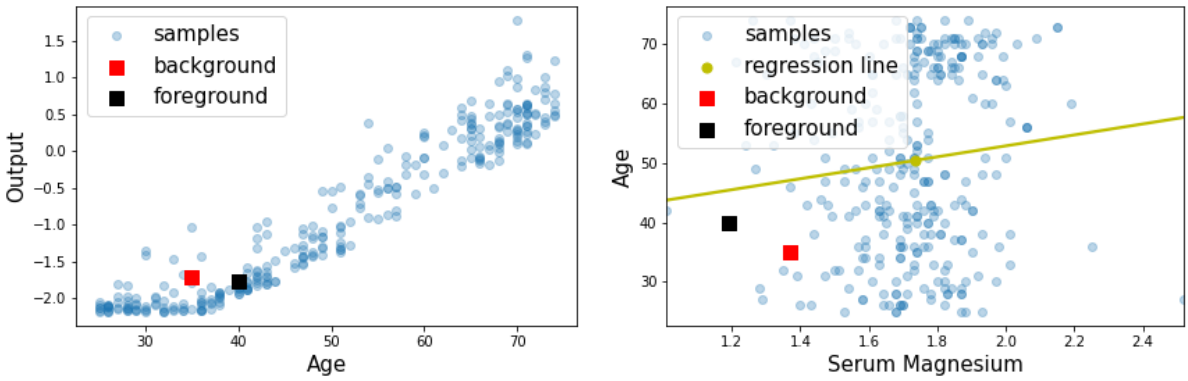
Top features	Age	Serum Magnesium	Serum Protein
Background sample	35	1.37	7.6
Foreground sample	40	1.19	6.5

Attributions	Independent	On-manifold	ASV
Age	0.1	-0.26	0.16
Serum Magnesium	0.02	0.2	0.02
Serum Protein	-0.09	0.07	0.0
Blood pressure	0.0	0.0	-0.14
Systolic BP	-0.05	-0.05	0.0
Diastolic BP	-0.04	-0.07	0.0
Serum Cholesterol	0.0	-0.15	0.0
Serum Albumin	0.0	-0.14	0.0
Blood protein	0.0	0.0	-0.08
White blood cells	0.0	0.11	0.0
Race	0.0	0.09	0.0
BMI	-0.0	0.08	-0.0
TIBC	0.0	0.06	0.0
Sex	0.0	-0.05	0.0
TS	0.0	0.05	0.0
Pulse pressure	0.0	-0.05	0.0
Poverty index	0.0	0.04	0.0
Red blood cells	0.0	0.03	0.0
Serum Iron	0.0	-0.02	0.0
Sedimentation rate	0.0	0.0	0.0
Iron	0.0	0.0	-0.0
Inflamation	0.0	0.0	0.0



(A) Shapley Flow

FIGURE 3.6: Comparison among baselines on a sample (top table) from the nutrition dataset, showing top 10 features/edges.



(A) Age vs. output

(B) Age vs. magnesium

FIGURE 3.7: Age appears to be protective in on-manifold SHAP because it steals credit from other variables.

Chapter 4

Credible Model

4.1 Introduction

When features are highly correlated, there exists multiple solutions that can achieve equally good test performance. Often, however, good performance alone is not enough. These solutions may vary in other desirable aspects not specified in the training distribution (*i.e.*, the underspecification problem). For example, in settings such as healthcare, to be adopted by practitioners, the model must also align at least in part with domain knowledge. In other words, the model needs to be "credible".

Informally, a credible model is a model that i) provides reasons for its predictions that are, at least in part, inline with well-established domain knowledge, and ii) does no worse than other models in terms of predictive performance. While a user is more likely to adopt a model that agrees with well-established domain knowledge, one should not have to sacrifice accuracy to achieve such adoption. That is, the model should only agree with well-established knowledge, if it is consistent with the data. Relying on domain expertise alone would defeat the purpose of data-driven algorithms, and could result in worse performance in practice. Admittedly, the definition of credibility is a subjective matter. In this chapter, we offer a first attempt to formalize the intuition behind a credible model.

Our Approach. To learn a credible model, we propose the Expert Yielded Estimates (EYE) penalty. Our proposed approach leverages domain expertise regarding known relationships between the set of covariates and the outcome. This domain expertise is used

to guide the model in selecting among highly correlated features, while encouraging sparsity. Our proposed framework allows for a form of collaboration between the data-driven learning algorithm and the expert. We prove desirable properties of our approach in the least squares regression setting. Furthermore, we give empirical evidence of these properties on synthetic and real datasets. Applied to two large-scale patient risk stratification tasks, our proposed approach resulted in an accurate model and a feature ranking that, when compared to a set of well-established risk factors, yielded an average precision (AP) an order of magnitude greater than the second most credible model in one task, and twice as large in AP in the other task.

Our key contributions are:

- formalizing the notion of credibility in the linear setting
- proposing a novel regularization term EYE (expert yielded estimates) to achieve this form of credibility.

Organization. The rest of the chapter is organized as follows. First, we review related work on variable selection and interpretability. Then, we define credibility and describe our proposed method in detail. Next, we present experiments and results, demonstrating that it is possible to align well with expert knowledge without sacrificing accuracy. Finally, we summarize the importance of our work and discuss the limitation of the proposed method.

4.2 Background & Related Work

Credibility is closely related to interpretability, which has been actively explored in the literature [11], [21], [128]–[131]. Yet, to the best of our knowledge, credibility has never been formally studied.

Interpretability is often achieved through dimensionality reduction. Common approaches include preprocessing the data to eliminate correlation, or embedding a feature selection criterion into the model's objective function. Embedding a regularization term in the objective function is often preferred over preprocessing techniques since it combines feature selection and training together, often resulting in more accurate models

(which we show in the Appendix). Thus we focus on regularization techniques in this work. A review of popular regularization methods are included in Chapter 2 Table 2.1.

In terms of incorporating additional expert knowledge at training time, Sun *et al.* explore using features identified as relevant during training, along with a subset of other features that yield the greatest improvement in predictive performance [132]. This work differs from ours because they assume expert knowledge as ground truth, a potentially dangerous assumption when experts are wrong. Vapnik *et al.* explore the theory of learning with privileged information [133]. Though similar in setting, they use expert knowledge to accelerate the learning process, not to enforce credibility. Helleputte and Dupont use partially supervised approximation of zero-norm minimization (psAROM) to create a sparse set of relevant features. Much like weighted LASSO, psAROM does not exhibit the grouping effect, thus is unable to retain all known relevant features. Moreover, the non-convex objective function for psAROM makes exact optimization hard [134]. [135] looks at utilizing hierarchical expert information to learn embeddings that help model prediction of rare diseases. While it is an interesting approach, its model's interpretability is questionable. [110] constrains the input gradient of features that are believed not to be relevant in a neural network. In the linear setting, the method simplifies to l_2 regularization on unknown features, which is suboptimal for model interpretability because the learned weights are dense.

Perhaps closest to our proposed approach, and the concept of credibility, is related work in interpretability that focuses on enforcing monotonicity constraints between the covariates and the prediction [136]–[140]. The main idea behind this branch of work is to restrict classifiers to the set of monotone functions. This restriction could be probabilistic [137] or monotone in certain arguments identified by experts [136], [139], [140]. Though similar in aim (having models inline with domain expertise), previous work has focused on rule based systems. Other attempts to enforce monotonicity in nonlinear models [141]–[143] aim to increase performance. Again, relying too heavily on expert knowledge may result in a decrease in performance when experts are wrong. In contrast, we propose a general regularization technique that aims to increase credibility without decreasing performance. Moreover, in the linear setting, credible models satisfy monotonicity and sparsity constraints.

4.3 Methods

In this chapter, we focus on linear models. Within this setting, we start by formally defining credibility in 4.3.1. Then, building off of a naïve approach in 4.3.2, we introduce our proposed approach in 4.3.3. In 4.3.4, we state important properties and theoretical results relevant to our proposed method.

4.3.1 Definition and Notation

Interpretability is a prerequisite for credibility. For linear models, interpretability is often defined as sparsity in the feature weights. Here, we define the set of features as $\mathcal{D}$. We assume that we have some domain expertise that identifies $\mathcal{K} \subseteq \mathcal{D}$, a subset of the features as known (or believed) to be important. Intuitively, among a group correlated features a credible model will select those in $\mathcal{K}$, if the relationship is consistent with the data.

Consider the following unconstrained empirical risk minimization problem.

$$\hat{\boldsymbol{\theta}} = \arg\min_{\boldsymbol{\theta}} L(\boldsymbol{\theta}, X, \boldsymbol{y}) + n\lambda J(\boldsymbol{\theta}, \boldsymbol{r}) \tag{4.1}$$

where L is some loss function and J is a regularization term. X is an n by d design matrix, where row $\boldsymbol{x}$ corresponds to one observation. The corresponding entry in $\boldsymbol{y} \in \mathbb{R}^n$ is the target value for $\boldsymbol{x}$. Let v_i denote the i^{th} entry of a vector $\boldsymbol{v}$. $\lambda \in \mathbb{R}_{\geq 0}$ is the tradeoff between loss and regularization, and $\boldsymbol{r} \in \{0,1\}^d$ is the indicator array where $r_i = 1$ if $i \in \mathcal{K}$ and 0 otherwise. Note that our setting differs from the conventional setting only through the inclusion of $\boldsymbol{r}$ in the regularization term. For theoretical convenience, we prove theorems in the least squares regression setting and denote $\hat{\boldsymbol{\theta}}^{OLS}$ as the ordinary least squares solution. For experiments, we use logistic loss.

We denote $\boldsymbol{\theta}$ as the true underlying parameters. Then $\boldsymbol{\theta}_{\mathcal{K}}$ and $\boldsymbol{\theta}_{\mathcal{D}\setminus\mathcal{K}}$ are the true parameters associated with the subset of known and unknown features, respectively. Throughout the text, vectors are in bold, and estimates are denoted with a hat.

Definition A linear model is *credible* if

1. Within a group of correlated *relevant* features $\mathcal{C} \subseteq D$: $\hat{\boldsymbol{\theta}}_{\mathcal{K}\cap\mathcal{C}}$ is dense, and $\hat{\boldsymbol{\theta}}_{\mathcal{C}\setminus\mathcal{K}}$ is sparse (*structure constraint*).

2. Model performance is comparable with other regularization techniques (*performance constraint*)

Consider the following toy example where $|\mathcal{C}| = 2$ and one of these features has been identified $\in \mathcal{K}$ by the expert, while the other has not. One could arbitrarily select among these two correlated features, including only one in the model. To increase credibility, we encourage the model to select the known feature (*i.e.*, the feature in $\mathcal{K}$)

We stress *relevant* in the definition because we do not care about the structure constraint if the group of variables does not contribute to the predictive performance. We assume expert knowledge is sparse compared to all features; thus a credible model is sparse due to the structure requirement. Credible models will result in dense weights among the known features, if the expert knowledge provided is indeed supported by the data. If experts are incorrect, *i.e.*, the set of features $\mathcal{K}$ are not relevant to the task at hand, then credible models will discard these variables, encouraging sparsity.

4.3.2 A Naïve Approach to Credibility

Intuitively, one may achieve credibility by constraining weights for known important factors with the l_2 norm and weights for other features with the l_1 norm. The l_2 norm will maintain a dense structure in known important factors and the l_1 norm will encourage sparsity on all remaining covariates. Formally, this penalty can be written as $q(\boldsymbol{\theta}) = (1-\beta)\|\boldsymbol{r} \odot \boldsymbol{\theta}\|_2^2 + 2\beta\|(\mathbf{1}-\boldsymbol{r}) \odot \boldsymbol{\theta}\|_1$ where $\boldsymbol{\theta} \in \mathbb{R}^d$, $\beta \in (0,1)$ controls the tradeoff between weights associated with the features in $\mathcal{K}$ and in $\mathcal{D} \setminus \mathcal{K}$.

Unfortunately, q does not encourage sparsity in $\hat{\boldsymbol{\theta}}_{\mathcal{D}\setminus\mathcal{K}}$. **Figure 4.1a** shows its contour plot. For a convex problem, each level set of the contour corresponds to a feasible region associated with a particular λ. A larger level value implies a smaller λ. It is clear from the figure that this penalty is non-homogeneous, that is $f(t\boldsymbol{x}) \neq |t|f(\boldsymbol{x})$. In a two-dimensional setting, when the covariates perfectly correlate with one another, the level curve for the loss function will have a slope of -1 corresponding to the violet dashed lines in **Figure 4.1**.

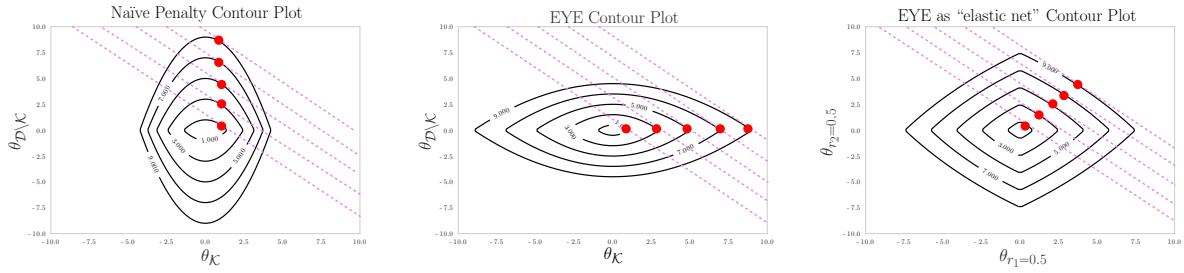
To understand why the slope must be -1, consider the classifier $y = \theta_{\mathcal{K}} x_1 + \theta_{\mathcal{D}\setminus\mathcal{K}} x_2$. Since x_1 and x_2 are perfectly correlated by assumption, we have $y = (\theta_{\mathcal{K}} + \theta_{\mathcal{D}\setminus\mathcal{K}})x_1$. Note that the loss value is fixed as long as $\theta_{\mathcal{K}} + \theta_{\mathcal{D}\setminus\mathcal{K}}$ is fixed, which means that each level curve of the loss function has the form $\theta_{\mathcal{K}} + \theta_{\mathcal{D}\setminus\mathcal{K}} = c$ for some scaler c, *i.e.*, $\theta_{\mathcal{D}\setminus\mathcal{K}} = -\theta_{\mathcal{K}} + c$. Thus, the slope of the violet lines must be -1 in **Figure 4.1**.

By the KKT conditions, with $\lambda > 0$, the optimal solution (red dots for each level curve in **Figure 4.1**) occurs at the boundary of the contour with the same slope ($\lambda = 0$ means the problem is unconstrained, then all methods are equal). We observe that with small λ, the large constraint region forces the model to favor features not in $\mathcal{K}$ because the point on the boundary with slope of -1 occurs near $\theta_{\mathcal{D}\setminus\mathcal{K}}$ axis, leading to a model that is not credible.

4.3.3 The Expert Yielded Estimates (EYE) Penalty

To address this sensitivity to the choice of hyperparameter, we propose the EYE penalty, obtained by fixing a level curve of q and scaling it for different contour levels. The trick is to force the slope of level curve in the positive quadrant to approach -1 as $\theta_{\mathcal{D}\setminus\mathcal{K}}$ approaches 0. Note that since q is symmetric around both axes, we can just focus on one "corner". That is, we want the "corner" on the right of the level curve to have a slope of -1, so that $\hat{\boldsymbol{\theta}}$ hits it in the perfectly correlated case. In fact, as long as $-1 \leq$ the "corner" slope ≤ 0, we achieve the desired feature selection. In the extreme case of slope 0 ($\beta = 1$), we do not penalize $\boldsymbol{\theta}_{\mathcal{K}}$ at all. Using a slope with a magnitude smaller than 1 assumes that features in $\mathcal{K}$ are much more relevant than other features, thus biasing $\hat{\boldsymbol{\theta}}_{\mathcal{K}}$. Since we do not wish to bias $\hat{\boldsymbol{\theta}}_{\mathcal{K}}$ towards larger values, if the solution is inconsistent with the data, we keep the slope as -1. This minimizes the effect of our potential prejudices, while maintaining the desirable feature selection properties. Casting our intuition mathematically yields the EYE penalty:

$$eye(\boldsymbol{x}) = \inf\left\{t > 0 \mid \boldsymbol{x} \in \left\{t\boldsymbol{x} \mid q(\boldsymbol{x}) \leq \frac{\beta^2}{1-\beta}\right\}\right\} \tag{4.2}$$



(A) (B) (C)

FIGURE 4.1: Visualization of selected regularization penalties. Dashed violet lines denote level sets for the loss function when features are perfectly correlated; red dots are the optimal points for each feasible region. A large feasible region (level sets with large labeled values) corresponds to a small λ. **(a)** The naïve penalty ($\beta = 0.5$) favors $\theta_{\mathcal{D}\setminus\mathcal{K}}$ as the feasible region grows. **(b)** EYE consistently favors $\theta_{\mathcal{K}}$. **(c)** When $\boldsymbol{r} = \mathbf{0.5}$, EYE produces a contour plot similar to elastic net. Setting $\boldsymbol{r} = \mathbf{0.5}$ represents a situation in which two features i and j are equally "known" and perfectly correlated. In this setting, $\hat{\theta}_i = \hat{\theta}_j$ (*i.e.*, highly correlated known factors have similar weights)

where t is a scaling factor to make EYE homogeneous and the inner set defines the level curve to fix. Note that β only scales the EYE penalty, thus can rewrite the penalty as:

$$eye(\boldsymbol{\theta}) = \|(\mathbf{1} - \boldsymbol{r}) \odot \boldsymbol{\theta}\|_1 + \sqrt{\|(\mathbf{1} - \boldsymbol{r}) \odot \boldsymbol{\theta}\|_1^2 + \|\boldsymbol{r} \odot \boldsymbol{\theta}\|_2^2} \tag{4.3}$$

Derivations of (4.2) and (4.3) are included in the Appendix. **Figure 4.1b** shows the contour plot of EYE penalty (note that the optimal solution for each level set occurs at the "corner" as desired).

4.3.4 EYE Properties

In this section, we give theoretical results for the proposed EYE penalty. We include detailed proofs in the Appendix. While the first three properties are general, the last three properties are valid in the least squares regression setting, *i.e.*, $Loss(\boldsymbol{\theta}, X, \boldsymbol{y}) = \frac{1}{2}\|\boldsymbol{y} - X\boldsymbol{\theta}\|_2^2$. We focus on the least square regression setting because a closed form solution exists, though our method is applicable to the classification setting as well (demonstrated in section 4.4).

- *EYE is a norm:* This comes for free as **Equation (4.2)** is an atomic norm [144], thus, convex.

- *EYE is β free:* Similar to elastic net and the naïve penalty q, EYE is a combination of the l_1 and l_2 norms, but it omits the extra parameter β. This leads to a quadratic reduction in the hyperparameter search space for EYE compared to elastic net and q.

- *EYE is a generalization of LASSO, l_2 norm, and "elastic net":* Setting $\boldsymbol{r} = \mathbf{1}$ and $\mathbf{0}$, we recover the l_2 norm and LASSO penalties, respectively. Relaxing $\boldsymbol{r}$ from a binary valued vector to a float valued vector, so that $\boldsymbol{r} = \mathbf{0.5}$, we get the elastic net shaped contour (**Figure 4.1c**). Elastic net is in quotes because the contour represents one particular level set, and elastic net is non-homogeneous.

- *EYE promotes sparse models:* Assuming $X^\top X = I$, the solution to EYE penalized least squares regression is sparse.

- *EYE favors a solution that is sparse in $\hat{\boldsymbol{\theta}}_{\mathcal{D}\setminus\mathcal{K}}$ and dense in $\hat{\boldsymbol{\theta}}_{\mathcal{K}}$:* In a setting in which covariates are perfectly correlated, $\hat{\boldsymbol{\theta}}_{\mathcal{D}\setminus\mathcal{K}}$ will be set to exactly zero. Conversely, $\hat{\boldsymbol{\theta}}_{\mathcal{K}}$ has nonzero entries. Moreover, the learned weights will be the same for every entry of $\hat{\boldsymbol{\theta}}_{\mathcal{K}}$ (*e.g.*, **Figure 4.1c**). This verifies the first part of the structure constraint. We also note that when the group of correlated features are all in $\mathcal{D} \setminus \mathcal{K}$, the objective function reverts back to LASSO, so that the weights are sparse, substantiating the second part of the structure constraint.

- *EYE groups highly correlated known factors together:*

  If $\hat{\theta}_i \hat{\theta}_j > 0$ and the design matrix is standardized, then

  $$\frac{|r_i^2 \hat{\theta}_i - r_j^2 \hat{\theta}_j|}{Z} \leq \frac{\sqrt{2(1-\rho)}\|\boldsymbol{y}\|_2}{n\lambda} + |r_i - r_j| \left(1 + \frac{\|(\mathbf{1}-\boldsymbol{r}) \odot \hat{\boldsymbol{\theta}}\|_1}{Z}\right)$$

  where ρ is the sample covariance between x_i and x_j, and

  $$Z = \sqrt{\|(\mathbf{1} - \boldsymbol{r}) \odot \hat{\boldsymbol{\theta}}\|_1^2 + \|\boldsymbol{r} \odot \hat{\boldsymbol{\theta}}\|_2^2}$$

  .

This implies that when $r_i = r_j \neq 0$

$$\frac{|\hat{\theta}_i - \hat{\theta}_j|}{Z} \leq \frac{\sqrt{2(1-\rho)}\|\boldsymbol{y}\|_2}{r_i^2 n\lambda}$$

i.e., the more correlated known important factors are, the more similar their weights will be. This is analogous to the grouping effect.

4.4 Experiments & Results

In this section, we empirically verify EYE's ability to yield credible models through a series of experiments. We compare EYE to a number of other regularization penalties across a range of settings using both synthetic and real data. In particular, we are interested in answering the following questions:

- Question 1: What are limitations of the naïve penalty? (**Section 4.4.3**, **Figure 4.2a**, **Figure 4.2b**)
- Question 2: Are EYE regularized models more credible than baselines under different feature correlations? (**Section 4.4.3**, **Figure 4.3a**)
- Question 3: Are EYE regularized models more credible than baselines under different percentage of known important factors? (**Section 4.4.3**, **Figure 4.3b**)
- Question 4: Are EYE regularized models able to recover from mistakes in known important factors? (**Section 4.4.3** and **Table 4.1** for synthetic dataset, **Table 4.2** second to last line for real datasets)
- Question 5: Are EYE regularized models more credible than baselines on large scale clinical datasets? (**Section 4.4.4**, **Section 4.4.5**, **Table 4.2**)

4.4.1 Measuring Credibility

Criterion (i): density in the set of known relevant features and sparsity in the set of unknown. In a two dimensional setting, we measure $\log|\frac{\theta_{\mathcal{K}}}{\theta_{\mathcal{D}\setminus\mathcal{K}}}|$ as a proxy for desirable weight structure (the higher the better). In a high-dimensional setting, highly correlated covariates form groups. For each group of correlated features, if known factors exist and are indeed important, then the shape of the learned weights should match $\boldsymbol{r}$ in the corresponding groups. *E.g.*, given two correlated features x_1 and x_2 that are associated with the outcome, if $r_1 = 0$ and $r_2 = 1$, then $\theta_1 = 0$ and $\theta_2 \neq 0$. Thus, to measure credibility, we use the symmetric KL divergence, $symKL(\hat{\boldsymbol{\theta}}_g{}', \boldsymbol{r}') = \frac{1}{2}\left(KL(\hat{\boldsymbol{\theta}}_g{}'\|\boldsymbol{r}') + KL(\boldsymbol{r}'\|\hat{\boldsymbol{\theta}}_g{}')\right)$, between the normalized absolute value of learned weights and the normalized $\boldsymbol{r}$ for each group g. For groups of relevant features that do not contain known factors, the learned weights should be sparse (*i.e.*, all weight should be placed on a single feature within the group). Thus, we report $\min_{\boldsymbol{x}\in \textit{one hot vectors}} symKL(\boldsymbol{x}, \hat{\boldsymbol{\theta}}')$ for such groups. As $symKL$ decreases, the credibility of a model increases. Note that $symKL$ only measures the shape of weights within *each group* of correlated features and does not assume expert knowledge is correct (*e.g.*, all weights within a group could be near zero).

In our experiments on real data, we do not know the true underlying $\boldsymbol{\theta}$ and the partition of groups. In this case, we measure credibility by computing the fraction of known important factors in the top n features sorted by the absolute feature weights learned by the model. We sweep n from 1 to d and report the average precision (AP) between $|\hat{\boldsymbol{\theta}}|$ and $\boldsymbol{r}$.

Criterion (ii): maintained classification performance. Recall that we want to learn a credible model without sacrificing model performance. That is, there should be no statistically significant difference in performance between a credible model and the best performing one (in this case, we focus on best linear models learned using other regularization techniques). We measure model performance in terms of the area under the receiver operating characteristic curve (AUC). In our experiments, we split our data into train, validation, and test sets. We train a model for each hyperparameter and bootstrap the validation set 100 times and record performance on each bootstrap sample. We want a model that is both accurate and sparse (measured using the Gini coefficient due to its desirable properties [145]). To ensure accuracy, for each regularization method, we remove models that are significantly worse than the best model in that regularization class

using the validation set bootstrapped 100 times (p value set at .05). From this filtered set, we choose the sparsest model and report criteria (i) and (ii) on the held-out test set.

4.4.2 Experimental Setup and Benchmarks

We compare EYE to the regularization penalties in **Table 2.1** across various settings. We exclude ridge from our comparisons, because it produces a dense model. In addition, we exclude adaptive LASSO because it requires an additional stage of processing.

We set the weights, $\boldsymbol{w}$, in **Table 2.1**, to mimic the effect of the $\boldsymbol{r}$. This gives a subset of the regularization techniques according to the same kind of expert knowledge that our proposed approach uses. In weighted LASSO and weighted ridge, the values in $\boldsymbol{w}_{\mathcal{D}\setminus\mathcal{K}}$ were swept from 1 to 3 times the magnitude of the values in $\boldsymbol{w}_{\mathcal{K}}$ to penalize unknown factors more heavily. For OWL, we set the weights in two ways. In the first case, we only penalize $|\hat{\boldsymbol{\theta}}|_{[1]}$, effectively recovering the l_∞ norm. In the second case, weights for the m largest entries in $\hat{\boldsymbol{\theta}}$ are set to be twice the magnitude of the rest, where m is the number of known important factors. Note that a direct translation from known factors to weights is not possible in OWL, since the weights are determined based on the learned ordering. We implemented all models as a single layer perceptron with a softmax trained using the ADADELTA algorithm [146] minimizing the logistic loss.

4.4.3 Validation on Synthetic Datasets

To test EYE under a range of settings, we construct several synthetic datasets [2]. In all experiments, we generate the data and run logistic regression with EYE and each regularization benchmark. In all of our experiments on synthetic data, we found no statistically significant differences in AUC, thus satisfying the performance constraint. These experiments expose the limitations of the naïve penalty, measure sensitivity to noise and to correlation in covariates, explore different shapes of $\boldsymbol{r}$, and examine the effect of the accuracy of expert knowledge on credibility. In all cases, the EYE penalty leads to the most credible model, validating our theoretical results.

[2] code available at `https://github.com/nathanwang000/credible_learning`

Limitations of the Naïve Penalty: Sensitivity to Hyperparameters

The naïve penalty q appears to be a natural solution for building credible linear models. However, since q is non-homogeneous, as the constraint region grows, the models begin to prefer features *not* in $\mathcal{K}$. Since small λ corresponds to a large constraint region, we vary λ to expose this undesirable behavior.

We sample 100 data points uniformly at random from -2.5 to 1.5 to create $\boldsymbol{v}$. We set $X = [\boldsymbol{v}, \boldsymbol{v}]$ to produce two perfectly correlated features with one known factor. We set $\boldsymbol{\theta} = [1, 1]$ (note that since the two features are perfectly correlated, it doesn't matter how $\boldsymbol{\theta}$ is assigned), and assign the label $\boldsymbol{y}$ as $\mathbb{1}_{\boldsymbol{\theta}^\top \boldsymbol{x} > 0}(\boldsymbol{x})$ for each data point $\boldsymbol{x}$.

Figure 4.2a shows the log ratio for credibility for different settings of λ and β. First note that as λ approaches zero, the log ratio approaches 0 for all methods because the models are effectively unconstrained. With nontrivial λ and large β, both EYE and the naïve penalty result in high credibility. This is expected as a large β will constrain known important factors less, thus placing more weight on them. For β in the lower range, the log ratio is negative because the naïve penalty penalizes known features more. For β in the middle range, the log ratio varies from credible to non-credible, exhibiting the artifact of non-homogeneity (the penalty contour is elongated along $\theta_{\mathcal{K}}$ as λ decreases, thus again favoring $X_{\mathcal{D}\setminus\mathcal{K}}$). Since we want the log ratio> 0 for all nontrivial λ, the naïve penalty with $\beta < 0.8$ fails.

The naïve penalty with large β also fails to produce credible models because the resulting models have worse classification performance. In particular, when $\beta > 0.8$, the naïve penalty overemphasizes the relevancy of known important factors. As shown in **Figure 4.2b**, the naïve penalty with large β performs considerably worse in terms of accuracy than EYE for large λ. On small λ, their performance are comparable. This is expected because EYE introduces less bias towards known important factors.

Varying the Degree of Collinearity

We can show theoretically that EYE results in a credible model when features are highly correlated. However, the robustness of EYE in the presence of noise is unknown. To explore how EYE responds to changes in correlation between features, we conduct an experiment in a high-dimensional setting.

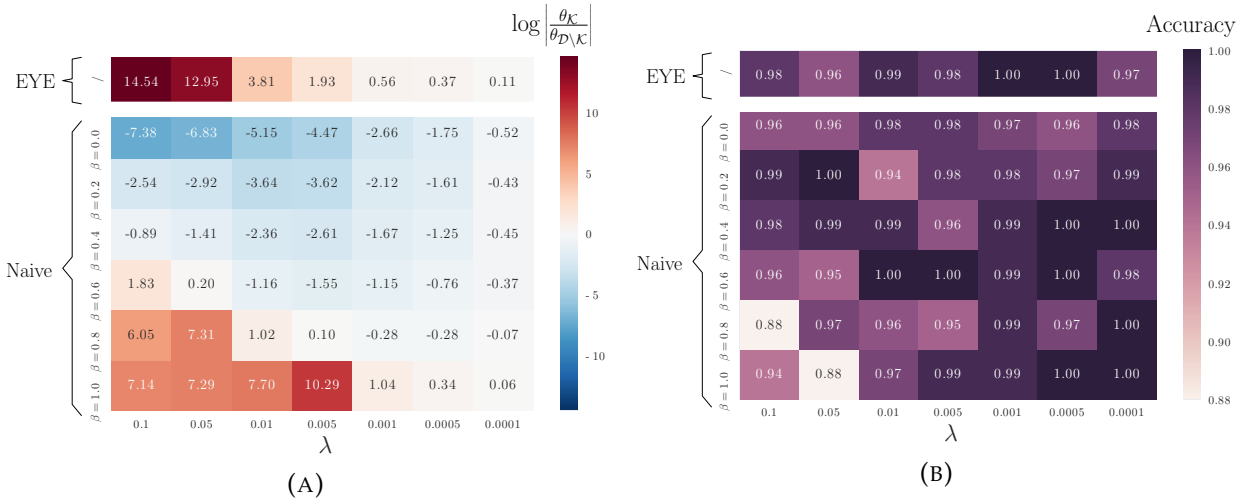


FIGURE 4.2: A comparison of the naïve penalty and EYE. **(a)** EYE meets the structural constraint better than naïve penalty with small and mid-ranged β **(b)** EYE has better performance than naïve Penalty with large β.

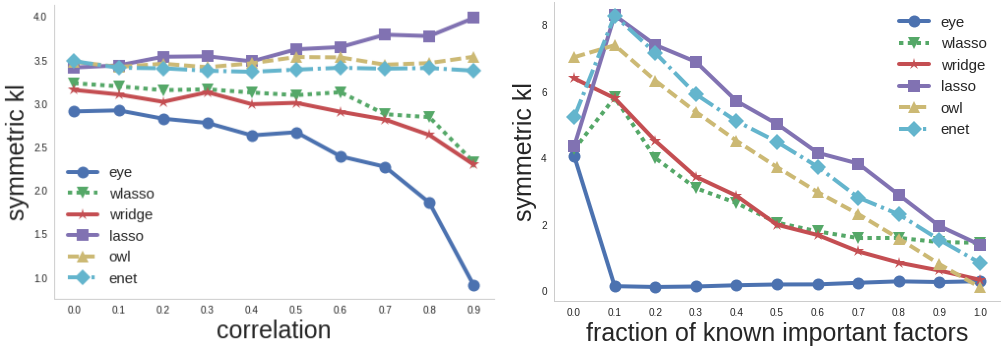


FIGURE 4.3: Comparisons of EYE with other methods under various settings **(a)** EYE leads to the most credible models in all correlations. **(b)** EYE leads to the most credible model for all shapes of $\boldsymbol{r}$.

We generate 10 groups of data, each having 30 features, with 15 in $\mathcal{K}$. We assigned each group a correlation score from 0 to 0.9 (here, we exclude the perfectly correlated case as it will be examined in detail in the next experiment). Intra-group feature correlations are fixed to the group's correlation score, while inter-group feature correlations are 0.

Figure 4.3a plots the *symKL* for each group. Moving from left to right, the correlation increases in step size of 0.1 from 0 to 0.9. As correlation increases, the EYE regularized model achieves the smallest *symKL*, and becomes the most credible model. In comparison, the other approaches do not achieve the same degree of credibility though, weighted LASSO and weighted ridge do exhibit a similar trend. However, since weighted LASSO fails to capture denseness in known important factors and weighted ridge fails to capture sparseness in unknown features, EYE leads to a more credible model. As correlation increases, LASSO actually produces a less credible model (as expected).

Varying Percentage of Known Important Factors

Besides varying correlation, we also vary the percentage of known important factors within a group of correlated features. We observe that EYE is consistently better than other methods.

In this experiment, we generate groups of data $\mathcal{C}_i$ where $i = 0, ..., 10$, each having 10 features. Features in each group are perfectly correlated, and features across groups are independent. Each group has a different number of features in $\mathcal{K}$, *e.g.*, group 0 has 0 known relevant factors and group 10 has 10 known important factors.

Figure 4.3b plots the *symKL* for each group of features. The groups are sorted by $|\mathcal{C}_i \cap \mathcal{K}|$. When $|\mathcal{C}_i \cap \mathcal{K}| = 0$, the model should be sparse. Indeed, for group 0, we observe that EYE, LASSO, and weighted LASSO do equally well (EYE in fact degenerates to LASSO in this case), closely followed by elastic net. Weighted ridge and OWL, on the other hand, do poorly since they encourage dense models. For other groups, EYE penalty achieves the best result (lowest *symKL*). This can be explained by property 4.3.4 as EYE sets the weights the same for correlated features in $\mathcal{K}$ while zeroing out weights in $\mathcal{D} \setminus \mathcal{K}$. Again, LASSO performed the worst overall because it ignores $\boldsymbol{r}$ and is sparse even when $\boldsymbol{r}$ is dense.

Varying Accuracy of Expert Knowledge

The experiments above only test cases where $\boldsymbol{\theta}$ is elementwise positive and where expert knowledge is correct (*i.e.*, the features identified by the expert were indeed relevant). To simulate a more general scenario in which the expert may be wrong, we use the following generative process:

1. Select the number of independent groups, $n \sim \text{Poisson}(10)$
2. For each group i in n groups
   (a) Sample a group weight, $w^{(i)} \sim \text{Normal}(0,1)$
   (b) Sample the number of features, $m^{(i)} \sim \text{Poisson}(20)$
   (c) Sample known important factor indicator array, $\boldsymbol{r}^{(i)} \sim \text{Bernoulli}(0.5)^{m^{(i)}}$
   (d) Assign true relevance $\boldsymbol{\theta}^{(i)} \in \mathbb{R}^{m^{(i)}}$ by distributing $w^{(i)}$ according to $\boldsymbol{r}^{(i)}$ (*e.g.*, if $w^{(i)} = 3$ and $\boldsymbol{r}^{(i)} = [0, 1, 1]$, then $\boldsymbol{\theta}^{(i)} = [0, 1.5, 1.5]$)
3. Generate covariance matrix C such that intra-group feature correlation=0.95 and inter-group feature correlation=0
4. Generate 5000 i.i.d. samples $\boldsymbol{x}_i \in \mathbb{R}^{\sum_{i=1}^{n} m^{(i)}} \sim \text{Normal}(\mathbf{0}, C)$
5. Choose label $y_i \sim \text{Bernoulli}(sigmoid(\boldsymbol{\theta}^\top \boldsymbol{x}_i))$ where $\boldsymbol{\theta}$ is the concatenated array from $\boldsymbol{\theta}^{(i)}$

Generating data this way covers cases where expert knowledge is wrong as feature group relevance and $\boldsymbol{r}$ are independently assigned. It also allows the number of features and weights for each group to be different. **Table 4.1** summarizes performance and credibility for each method averaged across 100 runs. EYE achieves the lowest sum of $symKL$ for each group of correlated features. In terms of AUC, the best models for each penalty are comparable, confirming that EYE is able to recover from the expert's mistakes.

4.4.4 Application to a Real Clinical Prediction Task

After verifying desirable properties in synthetic datasets, we apply EYE to a large-scale clinical classification task. In particular, we consider the task of identifying patients at greatest risk of acquiring an infection during their hospital stay. We selected a task from

TABLE 4.1: EYE leads to the most credible model on a synthetic dataset (mean $\pm$ stdev)

Method	$\sum_{g=1}^{n} symKL_g$	AUC
EYE	**0.442** $\pm$ 0.128	0.900 $\pm$ 0.044
wLASSO	0.929 $\pm$ 0.147	0.898 $\pm$ 0.044
wridge	1.441 $\pm$ 0.241	0.899 $\pm$ 0.045
LASSO	2.483 $\pm$ 0.440	0.898 $\pm$ 0.044
elastic net	2.673 $\pm$ 0.399	0.893 $\pm$ 0.044
OWL	3.125 $\pm$ 0.329	0.900 $\pm$ 0.044

TABLE 4.2: EYE leads to the most credible model on both the *C. difficile* and *PhysioNet Challenge* datasets; it keeps more of the factors identified in the clinical literature, while performing on par with other regularization techniques; it also has very sparse weights, second only to the model that just uses features in the risk factors

	C. difficile			*PhysioNet Challenge*		
Method	AP	AUC	sparsity$^+$	AP	AUC	sparsity$^+$
expert-features-only	1*	0.598	**0.998**	1*	0.754	**0.877**
EYE	**0.204**	0.753	0.980	**0.671**	0.815	0.794
wLASSO	0.033	0.764	0.884	0.300	0.810	0.824
LASSO	0.032	0.760	0.856	0.131	**0.823**	0.779
wridge	0.031	**0.768**	0.755	0.209	0.810	0.069
elastic net	0.031	0.754	0.880	0.153	0.818	0.649
EYE-random-r	0.031	0.748	0.936	0.589	0.792	0.779
OWL	0.028	0.548	0.544	0.108	0.794	0.046

$^+$ percentage of near-zero feature weights, where near-zero is defined as < 0.01 of the largest absolute feature weight

* expert-features-only logistic regression trivially achieves AP of 1 simply because it only uses expert features

healthcare since credibility is critical to ensuring the safe adoption of such models. We focus on predicting which patients will acquire a *Clostridium difficile* infection (CDI), a particularly nasty healthcare-associated infection. Using electronic health record (EHR) data from a large academic US hospital, we aim to learn a credible model that produces accurate *daily* estimates of patient risk for CDI.

The Dataset. We consider all adult hospitalizations between 2010 and 2015. We exclude hospitalizations in which the patient is discharged or diagnosed with CDI before the 3rd calendar day, since we are interested in healthcare-acquired infections (as opposed

to community-acquired). Our final study population consists of 143,602 adult hospitalizations. Cases of CDI are clinically diagnosed by positive laboratory test. We label a hospitalization with a positive laboratory test for CDI as +1, and 0 otherwise. 1.09% of the study population is labeled positive.

The Task. We frame the problem as a prediction task: the goal is to predict whether or not the patient will be clinically diagnosed with CDI at some point in the future during their visit. In lieu of a single prediction at 24 hours, we make predictions every 24 hours. To generate a single AUC given multiple predictions per patient, we classify patients as high-risk if their risk ever exceeds the decision threshold, and low-risk otherwise. By sweeping the decision threshold, we generate a single receiver operating characteristic curve and a single AUC in which each hospitalization is represented exactly once.

Feature Extraction. We use the same feature extraction pipeline as described in [147]. In particular, we extract high-dimensional feature vectors for each day of a patient's admission from the structured contents of the EHR (*e.g.*, medication, procedures, in-hospital locations etc.). Most variables are categorical and are mapped to binary features. Continuous features are either binned by quintiles or well-established reference ranges (*e.g.*, a normal heart rate is 60-100 beats per minute). If a feature is not measured (*e.g.*, missing vital), then we explicitly encode this missingness. Finally, we discard rare features that are not present in more than .05% of the observations. This feature processing resulted in 4,739 binary variables. Of these variables, 264 corresponded to known risk factors. We identified these variables working with experts in infectious disease who identified key factors based on the literature [148]–[150].

Analysis. We train and validate the models on data from the first five years (n=444,184 days), and test on the held-out most recent year (n=217,793 days). Using the training data, we select hyperparameters using a grid search for λ and β from 10^{-10} to 10^{10} and 0 to 1 respectively. The final hyperparameters are selected based on model performance and sparsity as detailed in section 4.4.1.

For each regularization method, we report the AUC on the held-out test set, and the average precision (AP) between $|\hat{\theta}|$ and $\boldsymbol{r}$ (see Section 4.4.1). **Table 4.2** summarizes the results on the test set with various regularizations.

Relative to the other common regularization techniques, EYE achieves an AP that is an order of magnitude higher, while maintaining good predictive performance. Moreover, EYE leads to one of the sparsest models, increasing model interpretability.

For comparison, we include a model based on only the 264 expert features (trained using l_2 regularized logistic regression) "expert-features-only." This baseline trivially achieves AP of 1, since it only uses expert features, but performs poorly relative to the other tasks. This confirms that simply retaining expert features is not enough to solve this task.

In addition, we include a baseline, "EYE-random-r", in which we randomly permuted $\boldsymbol{r}$. This corresponds to the setting where the expert is incorrect and is providing information about features that may be irrelevant. In this setting, EYE achieves a high AUC and low AP. This confirms that EYE is not severely biased by incorrect expert knowledge. Moreover, we believe this to be a feature of the approach, since it can highlight settings in which the data and expert disagree.

4.4.5 Application to PhysioNet Challenge Dataset

To further validate our approach, we turn to a publicly available benchmark dataset from PhysioNet [151]. In this task, the goal is to predict in-hospital mortality using EHR data collected in intensive care units (ICUs). Similar to above using the EYE penalty we trained a model and evaluated it in terms of predictive performance, average precision (AP), and model sparsity.

The Dataset. We use the ICU data provided in the PhysioNet Challenge 2012 [152] to train our model. This challenge utilizes a subset of the MIMIC-III dataset. We focus on this subset rather than using the entire dataset, since the goal is not to achieve state-of-the-art in in-hospital mortality prediction, but simply to evaluate the performance of the EYE penalty. The challenge data consist of three sets, each set containing data for 4000

patients. In our experiments, we use set A, since it is the only publicly labeled subset. We split the data randomly, reserving 25% as the held-out test set.

The Task. Using data collected during the first two days of an ICU stay, we aim to predict which patients survive their hospitalizations, and which patients do not. In contrast to the *C. difficile* task, here, we make a single prediction per patient at 48 hours.

Feature Extraction. The PhysioNet challenge dataset has considerably fewer features relative to the earlier task. In total, for each patient the data contain four general descriptors (*e.g.*, age) and 37 time-varying variables (*e.g.*, glucose, pH, etc.) measured possibly multiple times during the first 48 hours of the patient's ICU stay. We describe our feature extraction process below. Since again the goal was not state-of-the-art prediction on this particular task, we performed standard preprocessing without iteration/optimization.

We represent each patient by a vector containing 130 features. More specifically, for each time-varying variable we compute the maximum, mean, and minimum over the 48 hour window, yielding 111 features. In addition, for each of the 15 time-varying variables used in the Simplified Acute Physiology Score (SAPS-I) [153] we extract the most abnormal value observed within the first 24 hours,based on the SAPS scoring system. We concatenate these 126 features along with the 4 general descriptors producing a final vector of length 130. Out of the 130 variables, we consider the 15 SAPS-I variables along with age as expert knowledge. SAPS-I is a scoring system used to predict ICU mortality in patients greater than the age of 15 and thus corresponds to factors believed to increase patient risk.

Analysis. Using the training data, we select hyperparameters in the same way we did earlier. As with the previous experiment on the *C. difficile* dataset, for each regularization method, we report both AUC and AP on the held-out test set for this task. Again, we compared the model learned using the EYE penalty to the other baselines. **Table 4.2** summarizes our results on the held-out test set.

Overall, we observed a similar trend as to what we observed for the *C. difficile* dataset. Compared to the other common regularization techniques, EYE achieves significantly

higher AP and results in a sparse model. In terms of discriminative performance it performs on par with the other techniques. Again, we see that a model based on the expert features alone (i.e., *expert-features-only*) performs worse than the other regularization techniques. However, the difference in performance is not as striking as it was earlier. This suggests that perhaps the additional features (beyond the 16 SAPS-I features) do not provide much complementary information. Interestingly, the model using randomly permuted $\boldsymbol{r}$ ("EYE-random-r") achieves high AUC and AP. We suspect this may be due to the amount of collinearity present in the data. The non-expert and expert features are highly correlated with one another and thus both subsets are predictive (*i.e.*, supported by the data).

Besides regularization, another way to learn a credible model is to preprocess the input to exclude non-expert identified features that are highly correlated with expert identified features. It thus requires setting a threshold on the correlation between expert features and non-expert features to exclude the latter. However, this approach ignores each feature's relationship with the target variable, and thus may be less accurate compared to an EYE regularized model when the threshold is set to match the latter's level of alignment with experts (*e.g.*, measured by AP). We describe this baseline and include its results in Appendix B.1. As expected, we observe that this baseline is less accurate compared to an EYE regularized model to achieve an AP above 0.5 (AUROC of 0.760 vs 0.815). When we exclude less features by increasing the threshold, the accuracy of this approach increases at the cost of lowered AP (*e.g.*, AUROC of 0.789 and AP of 0.26 when the correlation threshold is set at 0.8).

4.5 Summary & Conclusions

In this chapter, we presented a formal definition of credibility in a linear setting. We proposed a regularization penalty, EYE, that encourages such credibility. Our proposed approach incorporates domain knowledge about which factors are known (or believed) to be important. Our incorporation of expert knowledge results in increased credibility, encouraging model adoption, while maintaining model performance. Through a series of experiments on synthetic data, we showed that sparsity inducing regularization such as LASSO, weighted LASSO, elastic net, and OWL do not always produce credible models.

In contrast, EYE produces a model that is provably credible in the least squares regression setting, and one that is consistently credible across a variety of settings.

Applied to two large-scale patient risk stratification tasks, EYE produced a model that was significantly better at highlighting known important factors, while maintaining predictive performance comparable with other regularization techniques. Moreover, we demonstrated how the proposed approach does not lead to worse performance when the expert is wrong. This is especially important in a clinical setting, where some relationships between variables and the outcome of interest may be less well-established.

This work debunks the notion that credibility comes at the cost of accuracy and provides a tool for researchers to correct confusing model reasoning with domain knowledge. However, there are several important limitations of the proposed approach. For example, we focused on a linear setting and one form of expert knowledge that can be expressed in the input space. In reality, many models are non-linear and not all expert knowledge can be expressed as a binary vector on input features. We address these limitations in part by extending credible learning to neural networks and incorporating non-input level domain knowledge in Chapter 5. Moreover, soliciting inputs from experts can be time consuming, in Chapter 6, we give pointers for future work to address those limitations. Finally, we do not claim EYE to be the optimal approach to yield credibility (we give no proof of that). Compared to other regularization penalties considered in this chapter, EYE introduces the least amount of bias, while striving to attain credibility.

Chapter 5

Concept Credible Model

5.1 Introduction

In addition to credibility (Chapter 4), we also want a model to perform well in out of distribution settings. In practice, machine learning models often fail to generalize under distribution shift despite having good performance in the training distribution [6], [154]–[156]. One of the mechanisms that lead to lack of robustness to distribution shift is shortcut learning [111], [155], [156]. "Shortcut learning occurs when a predictor relies on input features that are easy to represent (*i.e.*, shortcuts) and are predictive of the outcome in the training data, but do not remain predictive when the distribution of inputs changes" [111]. For example, consider building a machine learning model to predict the severity of knee osteoarthritis from X-ray images [19]. If people with mobility problems in the training set are more likely to have an X-ray acquired using a particular type of mobile X-ray scanner, the model may learn to rely on features that arise from the type of scanner to make a prediction, resulting in a failure to generalize when patients are scanned by a different scanner.

More formally, consider the causal graph in **Figure 5.1**, where Y is the target of interest (*e.g.*, diagnosis), S is the shortcut (*e.g.*, scanner type), X is the input (*e.g.*, X-ray image), C and U are representations that can be inferred from X but are not causally affected by S. Here, the dashed bidirectional arrow denotes a spurious correlation that holds during training but not at test time. Solid arrows denote causal relationships that are robust to changes. Note that S only affects the part of the input that is irrelevant for the diagnosis (X'), making it causally irrelevant for the prediction. We will consider both settings where

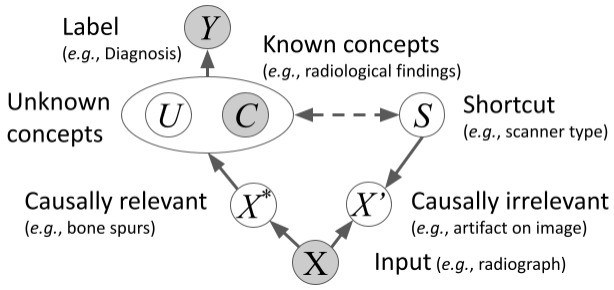


FIGURE 5.1: We formalize shortcut learning with a causal graph: Y is the label (*e.g.*, disease diagnosis) and X is the input (*e.g.*, radiograph). X can be decomposed into causally relevant and irrelevant features (X^* and X'), meaning that changing X^* changes the label whereas changing X' does not. X^* can be further decomposed into known and unknown relevant concepts (C and U). The node surrounding U and C abstracts their interaction (*e.g.*, they can be correlated). A shortcut variable S changes X' and is correlated with U and C. Observed variables are colored in gray. Dashed/solid edges represent correlation that is broken/unaffected under distribution shifts. We aim to eliminate model dependence on S.

S is and is not correlated with U. In our example, C could be known radiological risk factors, and U could be unknown radiological risk factors for the disease.

To mitigate a model's reliance on S, one can use existing tools if S is observed (*e.g.*, through model interpretation) [110], [111]. However, these methods do not apply when shortcuts are unknown prior to the occurrence of distribution shifts. Moreover, such approaches fail when the spurious correlation is strong (*i.e.*, more convincing shortcuts). In such scenarios, we need additional guardrails. **Our approach considers the setting in which we do not have direct knowledge of S, but have access to a representation, C, that is invariant to S** (formally defined in **Section 5.3.1**). It is true that without knowing S, we cannot truly confirm whether C is invariant to it. However, in practice, we can rely on established domain knowledge such as risk factors for disease to not encode shortcuts. By exploiting this representation C, we mitigate the reliance on shortcuts.

Where does C come from? C arises from domain knowledge and can be elicited in a number of different ways. For example, C may be elicited using transfer learning. Using domain knowledge, experts can identify a related source task. Predictive features (*i.e.*, learned representation) from the related source task (*i.e.*, C) can be shared to predict Y [155]. Alternatively, if one has auxiliary concept labels, one can train a model to predict

the presence of concepts and use these predictions as C [19]. In fact, both [19] and [155] have shown that relying on C alone can outperform a standard model (*i.e.*, a state of the art model) in the presence of shortcuts.

However, depending solely on C, referred to as a concept bottleneck model (CBM) [19], ignores potentially unknown concepts (*i.e.*, U). When U contains additional useful information, relying solely on C results in inferior predictive performance.

Our Approach. To tackle this problem, we propose two approaches based on *concept credible models* (CCM). The first approach, CCM RES, while simple, is susceptible to a particular failure case when U is correlated with S. The second approach, CCM EYE, extends the EYE penalty from Chapter 4 to address those issues. The EYE penalty was proposed for linear models as a way to increase the alignment with expert knowledge (in our case C) without sacrificing predictive performance. Here, we hypothesize that the same idea can mitigate the use of shortcuts. We thus extend the EYE penalty to work with non-linear models by applying it to the learned representation/concept space. This is a nontrivial application of the EYE penalty since here the concept space is not equivalent to the input space. Moreover, we identify the conditions in which CCM RES and CCM EYE mitigate learning shortcuts.

Our key contributions are as follows.

- We propose the idea of learning concept credible models (CCM), in which C is not required to be directly represented in the input space. We demonstrate that CCMs are more robust to shortcuts compared to existing approaches.
- Unlike previous work on shortcut learning, we show that our approaches still apply when shortcuts are *perfectly* correlated with other features, and address the limitation of existing methods that rely solely on C in making predictions.
- Theoretically, we identify the sufficient conditions under which a CCM can eliminate shortcuts for the setting considered in **Figure 5.1**.
- Empirically, we demonstrate that our approach can still help mitigate shortcuts even when these conditions are violated.

Organization. The rest of the chapter is organized as follows. First, we review related work on concept bottleneck models, shortcut learning, and credible learning. Then, we define our assumptions and describe our proposed methods in detail. Next, we present experiments and results, demonstrating that our proposed approaches can mitigate shortcuts even when assumptions are violated. Finally, we summarize the importance of our work and suggests potential extensions of the proposed method.

5.2 Background & Related Work

The idea of "concept credible models" is connected to multiple fields in ML.

Connection to shortcut learning. Shortcut learning is a particular failure mode that arises due to distribution shifts [113], [154]. However, to date researchers have typically assumed shortcuts are known *a priori*. Under such settings, one can augment the dataset to decorrelate shortcuts with data [105]–[109] or regularize model parameters to not rely on shortcuts [9], [110], [111]. In contrast, we do not assume that we know S. This change makes approaches such as IRM [113] and REx [114] no longer applicable, because without knowing S, it is hard to specify the family of distributions to which a model should be robust. The above methods also will not work when shortcuts and robust features are perfectly correlated because without prior knowledge, they cannot be separated apart.

Connection to concept bottleneck models. The concept bottleneck model (CBM) was proposed in [19] with the goal of making a model's decision more transparent by only using C for prediction. While this can mitigate shortcuts since the model is forced to rely on C instead of spurious correlations, it ignores unknown concepts, often resulting in lower accuracy compared to a standard model [51]. We address this problem by adding a channel that takes X as input, in addition to predicting Y from C. This added channel enables CCM to learn U, resulting in better accuracy.

Connection to credible learning. Credible models are trained by regularizing a model's feature attribution to be close to expert identified features (*i.e.*, features known to be relevant for the prediction), in addition to being accurate [13], [157]. While credible learning has been shown to work well in a transfer learning setup within natural language processing [157], we are the first to study its applicability to mitigate the effects of shortcut

learning. However, unlike previous work, we do not require domain knowledge (*i.e.*, concepts) to be expressed directly in the input space. This provides us with greater flexibility in exploring different types of inputs in which it may be difficult to collect domain expertise (*e.g.*, the pixel value of images). As a result, our approach does not require models to be linear.

We note that the EYE penalty was originally introduced as a method to encourage credible learning. By using it as a regularizer to discourage shortcut learning, we draw a connection between credible learning and robustness.

5.3 Methods

We formalize the setup of the problem, lay out assumptions, and propose methods to learn concept credible models.

5.3.1 Preliminaries

To simplify the exposition, we illustrate the setup for a regression problem. The setting, however, is easily adaptable to multi-class classification. We capitalize random variables and bold vectors. For example, $\boldsymbol{x}$ denotes an instance of the random vector X. We denote the Pearson correlation between two random variables as $corr(\cdot, \cdot)$.

Setup & Assumptions

Given a dataset $\mathcal{D} = \{(\boldsymbol{x}^{(i)} \in \mathbb{R}^d, y^{(i)} \in \mathbb{R})\}_{i=1}^n$ of n samples generated according to **Figure 5.1** and a function $f_c : \mathbb{R}^d \to \mathbb{R}^c$ such that $C := f_c(X)$, **we aim to learn an accurate prediction from X to Y ($f : \mathbb{R}^d \to \mathbb{R}$) that is invariant to the *unknown* shortcut S**. Here, c and d are dimensionality for C and X respectively.

Although we assumed that f_c is given in this setup, we can also learn it from a related dataset (*e.g.*, predictive features for this related task). Note that this does not require direct knowledge of S. It is sufficient to know, for example, that the shortcut for the target task is unlikely to be a shortcut for the related task.

In this chapter, invariance refers to counterfactual invariance from [9] (**Definition 1.1**). Adapting their notation, let $X(s)$ denote the counterfactual X we would have seen had S

been set to s, leaving all else fixed, f is *counterfactually invariant* to S if $f(X(s)) = f(X(s'))$ almost everywhere, for all s, s' in the sample space of S. This invariance ensures generalization of the model regardless of the shortcut's distribution. Note that invariance is not the same as independence. For example, scanner type does not cause the diagnosis (*i.e.*, diagnosis is invariant to scanner type) yet they can be correlated.

We require two assumptions about C. The first is implied by the causal graph, while the second is not.

- **A1**: C is counterfactually invariant to S

  For example, changing the scanner type does not change the occurrence of a bone spur in an X-ray image. Thus the presence of bone spur is invariant to the scanner type. Without this assumption, even a model that only uses C may indirectly depend on S.

- **A2**: S is redundant given C (*i.e.*, $Y \perp\!\!\!\perp S|C$)

  For example, given bone spur and other radiological findings from an X-ray image, the type of scanner is irrelevant in predicting arthritis severity. Without this assumption, including S improves the accuracy.

Since both **A1** and **A2** are not testable without knowing S, in experiments, we test our methods' sensitivity to each assumption empirically. Note that we do not make any assumption regarding the correlation between S and U.

Existing Approaches

We formally introduce common methods that are typically used in this prediction setting.

- **Standard Model**: Standard model refers to the task specific state-of-the-art model trained with loss L with empirical risk minimization:

$$\arg\min_{f_{\text{STD}}} \sum_{i\in[1,\cdots,n]} L(f_{\text{STD}}(\boldsymbol{x}^{(i)}), y^{(i)})$$

  Such a model cannot distinguish among C, U, and S, thus is vulnerable to rely on shortcuts for prediction.

- **Concept bottleneck model**: Our proposed approach builds off of CBM [19]. With our notation, a CBM's prediction can be written as $f_{\text{CBM}}(X) = f_y(f_c(X))$. Here, f_y maps from C to Y and is trained using empirical risk minimization:

$$\arg\min_{f_y} \sum_{i \in [1,\cdots,n]} L(f_{\text{CBM}}(\boldsymbol{x}^{(i)}), y^{(i)})$$

  When U contains additional information useful in predicting Y given C, CBM is less accurate than a standard model.

A motivating example

To build intuition, consider the following linear regression example in which C and S are perfectly correlated during training (*i.e.*, $C = S$ in $\mathcal{D}$) while C and U are not. Given $X = [C, S, U]$ and $Y = C + U$, a least squares linear regression solution gives a prediction of $\hat{Y} = (1 - t)C + U + tS$ (derived in the Appendix). The free parameter $t \in \mathbb{R}$ results from the spurious correlation between C and S.

The minimum L_2 norm solution of this problem results in $t = 0.5$ and will fail to generalize when the correlation between S and C no longer holds at test time. In contrast, if we only use C for prediction (*i.e.*, CBM), the solution will not achieve a loss of 0 since it ignores U. Furthermore, in cases where C and U are correlated, CBM is asymptotically biased due to omitting the variable U [158]. This means that models that ignore U, such as CBM, cannot recover the true regression coefficients even as the training set size approaches infinity.

5.3.2 Proposed Approaches: Concept Credible Models

We introduce two approaches to learn a concept credible model with the goal of mitigating shortcuts: CCM RES and CCM EYE.

CCM RES

The limitation of CBM stems from its inability to infer U from X. To address this limitation, we design a two stage approach, CCM RES, that first fits a CBM on the dataset,

and then fits a residual model f_x based on the difference between Y and the output of the CBM. This idea is similar in spirit to boosting methods. f_x enables CCM RES to learn U. CCM RES obtains its prediction by adding the output from f_x to the output of the CBM:

$$f_{\text{RES}}(\boldsymbol{x}) = f_{\text{CBM}}(\boldsymbol{x}) + f_x(\boldsymbol{x}) \tag{5.1}$$

When CBM achieves small training loss (*e.g.*, the difference between Y and CBM's prediction is small), CCM RES does not have to rely on information other than C. Otherwise, CCM RES relies on f_x to make up for what C alone cannot learn. We learn CCM RES with empirical risk minimization:

$$\hat{f}_{\text{RES}} = \arg\min_{f_x} \sum_{i \in [1,\cdots,n]} L(f_{\text{RES}}(\boldsymbol{x}^{(i)}), y^{(i)}) \tag{5.2}$$

Applied to **Example 5.3.1**, when U is independent from S in $\mathcal{D}$, the resulting model not only achieves 0 empirical loss, but also has 0 reliance on S, achieving our goal of accuracy without relying on shortcuts. We generalize this motivating example to all linear models below.

To highlight the strength of CCM RES compared to previous approaches, we consider the worst case scenario where $|corr(C, S)| = 1$. In this case, the system is under-specified (*i.e.*, allowing multiple minimum loss solutions), and a standard model may not be consistent with the causal DAG. Previous approaches fail in such scenarios since they cannot distinguish C from S.

Consistency of CCM RES: If f_{RES} is linear (both f_{CBM} and f_x are linear), $X = [C, S, U]$, $Y = aC + bU + \epsilon$ with $a, b \in \mathbb{R}$ and a zero mean error ϵ, $|corr(C, S)| = 1$ on the training distribution P_{XY}, and $U \perp\!\!\!\perp S$,

$$\arg\min_{f_{\text{RES}}} \mathbb{E}_{x,y \sim P_{XY}} (y - f_{\text{RES}}(x))^2$$

recovers the true parameters without relying on S (*i.e.*, weight a for C, b for U, and 0 for S).

Proof. We first show that the residual is independent of S, thus fitting to the residual will

not use S. From $U \perp\!\!\!\perp S$ and $|corr(C, S)| = 1$, we know $U \perp\!\!\!\perp C$. Fitting on infinite data with squared loss simplifies f_{CBM} to $\mathbb{E}(Y|C) = \mathbb{E}(aC + bU + \epsilon|C) = aC + b\mathbb{E}(U|C) = aC + b\mathbb{E}(U)$. The residual, $Y - \mathbb{E}(Y|C) = b(U - \mathbb{E}(U)) + \epsilon$, is independent of S because $U \perp\!\!\!\perp S$. Thus the prediction is $aC + bU$, recovering the true parameters. □

Remark: As the $|corr(S, C)|$ decreases, the system may no longer be under-specified, and both CCM RES as well as a standard model are expected to be consistent. This happens when S is not a linear combination of C and U, in which case the minimum loss solution is unique. If, however, S is a function of U, we cannot distinguish S from U and thus cannot guarantee the consistency of CCM RES.

This result shows that a linear CCM RES, unlike a linear CBM, is a consistent estimator. However, while CCM RES enables learning unknown concepts, it fails when S and U are correlated because the residual can be estimated as a linear combination of U and S, making f_x vulnerable to encoding a shortcut. We address this problem with a second approach.

CCM EYE

In our second approach, we utilize the EYE regularization from [13] to learn a concept credible model (CCM EYE). The EYE penalty penalizes reliance on features that are correlated with C but not in C. We propose to apply EYE regularization on the concept space (*i.e.*, the learned representation space) as follows:

$$f_{\text{EYE}}(\boldsymbol{x}) = \boldsymbol{\theta}_x^\intercal f_x(\boldsymbol{x}) + \boldsymbol{\theta}_c^\intercal f_c(\boldsymbol{x}) \tag{5.3}$$

where f_c computes the known relevant representation C and f_x computes a representation from the last layer of a standard model. This transformation from $\boldsymbol{x}$ to $f_x(\boldsymbol{x})$ allows the model to be non-linear. $\boldsymbol{\theta}_x$ and $\boldsymbol{\theta}_c$ are coefficients for $f_x(\boldsymbol{x})$ and $f_c(\boldsymbol{x})$ respectively. We then apply the EYE regularization on those parameters:

$$\hat{f}_{\text{EYE}} = \arg\min_{\boldsymbol{\theta}_x, \boldsymbol{\theta}_c} \sum_{i \in [1, \cdots, n]} L(f_{\text{EYE}}(\boldsymbol{x}^{(i)}), y^{(i)}) + \lambda J([\boldsymbol{\theta}_x, \boldsymbol{\theta}_c]) \tag{5.4}$$

Here, $J([\boldsymbol{\theta}_x, \boldsymbol{\theta}_c]) = \|\boldsymbol{\theta}_x\|_1 + \sqrt{\|\boldsymbol{\theta}_x\|_1^2 + \|\boldsymbol{\theta}_c\|_2^2}$ is the EYE regularization applied to our setting and $\lambda \in \mathbb{R}_{\geq 0}$ is a hyperparameter that controls the trade-off between regularization and loss. The EYE penalty more strictly penalizes $\boldsymbol{\theta}_x$ compared to $\boldsymbol{\theta}_c$, allowing the norm of $\boldsymbol{\theta}_c$ to be larger and hence encouraging the model to rely on C. Conversely, J discourages the use of X, which include both U and S. If U is in fact important in predicting Y, the minimization of the loss encourages the use of U more than S because of **A2**: U has more predictive power compared to S given C.

We choose λ such that $\boldsymbol{\theta}_x$ is strictly regularized without sacrificing in-distribution performance (*i.e.*, performance under the biased training distribution). We do so by picking the largest λ such that the model's accuracy on the validation set is not statistically worse than that of a standard model. We say that a model has statistically worse performance compared to the standard model if its empirical 95% bootstrapped confidence interval over the performance metric falls below the performance of the standard model. This ensures that CCM EYE maximizes the use of C without sacrificing predictive performance.

Similar to CCM RES, CCM EYE is a consistent estimator even when $|corr(C, S)| = 1$. The remark for CCM RES applies to CCM EYE as well.

Consistency of CCM EYE: If f_{EYE} is linear, $X = [C, S, U]$, $Y = aC + bU + \epsilon$ with $a, b \in \mathbb{R}$ and a zero mean error ϵ, $|corr(C, U)| \neq 1$ and $|corr(C, S)| = 1$ on the training distribution P_{XY},

$$\arg\min_{f_{\text{EYE}}} \mathbb{E}_{x,y \sim P_{XY}} (y - f_{\text{EYE}}(x))^2 + \lambda J([\boldsymbol{\theta}_x, \boldsymbol{\theta}_c])$$

recovers the true parameters (*i.e.*, weight a for C, b for U, and 0 for S) with standardized input, where λ is chosen as described before from $P_{X,Y}$.

Proof. With infinite data, the empirical loss is the in-distribution generalization loss, therefore λ is chosen such that the generalization loss is minimized. In the worst case scenario, the perfect correlation between S and C makes this linear problem underspecified (*i.e.*, multiple solutions), which means λ is non-zero because it is set to be the largest value such that model performance is not statistically worse than a standard model trained on $P_{X,Y}$. Fixing the same loss, the EYE penalty places zero weight on standardized features

(features normalized to zero mean and unit variance) that are perfectly correlated with expert identified features [13]. Treating C as the expert identified feature, the coefficient for S is thus 0. Combined with the fact that C and U are not perfectly correlated, to achieve the minimum loss, the coefficients for C and U must be a and b respectively. □

Remark: Unlike CCM RES, the consistency of CCM EYE no longer requires $U \perp\!\!\!\perp S$. Intuitively, EYE can separate U from S because of **A2**: U is needed in addition to C to be accurate, yet S is not needed given C. Note that $|corr(S, C)| = 1$ does not imply $U \perp\!\!\!\perp S$ because U can be correlated with C, which in turn is correlated with S.

While our theoretical results on linear models are restrictive, they a) serve as a sanity check and b) hint at what we might expect with more complex models as their last layers are often linear (*e.g.*, neural networks). We also note that the additive structure of both CCM RES and CCM EYE does not restrict their expressive power as f_x can be arbitrarily complex, capturing the interactions between C and U.

5.4 Experiments & Results

In this section, we verify CCM's robustness to spurious correlations on three tasks using publicly available datasets. The first is an image classification task similar to the one examined by [19]. This task demonstrates the superior performance of CCM when C is complex and non-linear in X. The second task is the prediction of pulmonary edema from chest radiographs. It demonstrates CCM's effectiveness in a critical domain where accuracy and robustness are needed. We include an additional task to predict in-hospital mortality in the Appendix.

We start with a setting in which our assumptions hold (**Section 5.3.1**), and then relax our assumptions to stress test our methods. We evaluate on both biased and unbiased/clean data (defined in the evaluation section of each task) to explore the effects of a distribution shift caused by the shortcut. All models are trained and selected using only the biased dataset (*e.g.*, the dataset where $corr(S, Y) \neq 0$). We are interested in answering the following questions:

- Question 1: Can CCMs mitigate shortcuts when **A1** and **A2** hold? (**Table 5.1**)
- Question 2: Can CCMs mitigate shortcuts when **A1** breaks? (**Figure 5.2**)
- Question 3: Can CCMs mitigate shortcuts when **A2** breaks? (**Figure 5.3**)
- Question 4: Can CCMs mitigate shortcuts when **A1** and **A2** break? (**Figure 5.4**)

Baselines. We compare CCM with the following methods:

- **STD(X)** is a model trained end-to-end on the biased dataset [159], using X to predict Y. We expect it to learn S because there is a backdoor path from S to Y in **Figure 5.1**.
- **Concept bottleneck model (CBM)** removes STD(X)'s reliance on S by only fitting on C [19]. However, CBM lacks the ability to infer U and thus may sacrifice discriminative performance before and after shortcut induced distribution shifts. Following [19], we train a CBM by fitting a logistic regression model on top of C.
- **STD(C, X)** is a standard model that conditions both on X and C for prediction. On the one hand, we expect this baseline to be more robust than STD(X) when S breaks because it has an easy access to C. On the other hand, conditioning on X gives the baseline the ability to infer U, unlike CBM. However, when S is highly correlated with C, this baseline can still rely on S to make a prediction. While there are many ways to implement STD(C, X), we implement this baseline as a special case of CCM EYE with $\lambda = 0$. This allows us to clearly demonstrate the effect of EYE regularization on model robustness.

5.4.1 Experiments on the CUB dataset

The **Caltech-UCSD Birds-200-2011** dataset (CUB) consists of 11,788 images of birds [160] each belonging to one of 200 species (Y). In addition to the images, the dataset also contains 312 binary attributes/concepts (*e.g.*, beak color) describing birds in each image. Following [19], we filter out concepts with noisy annotations. We then train a standard model to predict those concepts from X, with random Gaussian noise $N(0, 0.1^2)$ added to

the image. The resulting prediction is treated as C. Note that no shortcut is introduced in obtaining C, in order to satisfy **A1**. We will later test breaking this assumption.

The shortcut we consider here is the level of noise, σ, on an image. We will correlate σ with bird species to mimic a setting in reality where some birds have noisier photos than others because they are harder to observe in the wild. Ideally, a model should be able to classify the birds regardless of the noise level.

To introduce σ as a shortcut, we correlate it with bird classes in the training dataset. However, we do not correlate σ with Y directly because it violates **A2** (a setting we explore in the Appendix). Instead, we correlate σ to Y through C, using the function below.

1: **function** BIAS(X)
2: ss $\leftarrow$ LINSPACE(0, 0.1, n_σ) ▷ Create different levels of σ
3: **if** Uniform$(0, 1) < T$ **then**
4: $\sigma \leftarrow$ ss[arg max(CBM(X)) mod n_σ] ▷ Correlate σ with predicted Y
5: **else**
6: $\sigma \leftarrow$ randomChoice(ss)
7: **end if**
8: **return** $X + N(0, \sigma^2)$ ▷ Add class specific noise to input
9: **end function**

The algorithm starts by uniformly spacing n_σ levels of noise between 0 and 0.1 (line 2). Setting n_σ to 200 would be equivalent to each bird species having its own level of noise. To simulate a more realistic setting, we arbitrarily set $n_\sigma = 10$, allowing multiple birds to share a noise level. We show in the Appendix that varying n_σ does not affect our results. We then use modular arithmetic to ensure that each of the 200 bird types gets mapped onto one of the 10 noise levels. Then with probability T, we correlate σ with the bird class predicted from an oracle model CBM_O. CBM_O is similar to the CBM baseline explained above, with the exception that it has oracle access to a noise free dataset at training time (line 3-4) [1]. By using predictions from CBM_O rather than the true labels, we ensure that σ contains information about Y through C. We test breaking this assumption later. Then with probability $1 - T$, we break the correlation between σ and Y by randomly choosing a noise level (line 5-6). Finally we return an image with shortcut.

[1] We note that CBM_O is not a valid baseline as it is trained on the clean dataset, while all baselines are trained on the biased dataset. It simply serves as an approach to satisfy **A2**.

TABLE 5.1: On the CUB dataset, when **A1** and **A2** hold, CCM is no worse than baselines on the biased dataset (column 1), and is better than baselines on the clean dataset (column 2). Empirical 95% confidence intervals are included in parentheses.

Method	Test Acc (biased)	Test Acc (clean)
CCM EYE	76.0 (75.0, 77.2)	75.2 (74.1, 76.5)
CCM RES	75.6 (74.2, 76.9)	76.0 (74.8, 77.2)
STD(X)	75.7 (74.7, 77.0)	55.8 (54.7, 57.3)
CBM	71.6 (70.4, 72.9)	72.8 (71.7, 73.9)
STD(C,X)	76.0 (74.7, 77.2)	69.7 (68.6, 70.8)

The choice of T determines how biased the training dataset is (*i.e.*, the correlation between S and Y), with $T = 1$ being the most biased and $T = 0$ being not biased. To ensure that the shortcut is easy to learn, we keep $T = 1$ for all experiments except when we explore CCM's sensitivity to assumptions, described later.

Model Training. Following [19], all methods use an Inception V3 architecture [159] initialized using the Imagenet dataset [161]. We divide the training set predefined in the CUB dataset into train and validation set with a 80/20 random split, and use the predefined test set for evaluation. We report the performance on this test set as the result for unbiased/clean dataset. We then add class dependent noise described earlier to the train, validation and the test set to form the biased dataset. All methods are trained on the training set of this biased dataset using SGD with learning rate of 0.01, momentum of 0.9, and batch size of 32. We apply 10^{-4} weight decay to each model and decay the learning rate every 15 epochs. For CCM EYE, we tune λ in the range of $[10^{-2}, 10^{-3}, 10^{-4}, 10^{-5}, 10^{-6}]$.

Evaluation. Recall that our goal is to learn an accurate model without using S. We generate the biased and the clean test set as described in the model training section above. Evaluating on the biased test set demonstrates how the model performs when S is correlated with Y. Evaluating on the clean test set demonstrates how the model performs without image noise (*i.e.*, the shortcut no longer exists). A model that does not rely on S should perform similarly on both datasets. We measure performance on the CUB dataset

using accuracy (ACC) as bird classes are balanced. Empirical 95% confidence intervals are reported based on bootstrapped samples from the test set.

Results. We examine the results when **A1** and **A2** hold/break. We also explore varying λ in the Appendix to justify its choice.

How does CCM perform when A1 and A2 are satisfied? The test accuracy results in **Table 5.1** show that both CCM RES and CCM EYE are no worse than baselines when tested on the biased dataset (first column), but are significantly better than baselines when S is removed (second column). In contrast, STD(X) performs well on the biased dataset but underperforms on the clean dataset, indicating its reliance on S. As expected, CBM does not rely on the shortcut as its performance is stable with and without S. However, its inability to utilize U results in a drop in accuracy compared to others. Finally, STD(C, X) is accurate on the biased dataset and improves over STD(X) on the clean dataset because it encourages the model to use C. However, CCM EYE dominates, suggesting that conditioning on both C and X is not enough to remove model reliance on S.

How does CCM perform when A1 is violated? We relax **A1** by learning C on a biased dataset. Specifically, we use the BIAS function introduced in **Section 5.4.1** but vary the probability of correlating S with Y (*i.e.*, varying the parameter T). **Figure 5.2** shows the results of varying T on the test accuracy in the clean data. As before, CCM dominates the other baselines except at $T = 1$, indicating that unless C is extremely corrupted with S, CCM performs well compared to the other models. We note that STD(X) does not change with T because it does not use C. In the Appendix, we show that all methods except CBM perform similarly well on the biased dataset.

How does CCM perform when A2 is violated? We relax **A2** in two ways: a) S contains information outside of C but within U (*i.e.*, $Y \perp\!\!\!\perp S|C, U$), and b) S contains information outside of C and U (*i.e.*, $Y \not\perp\!\!\!\perp S|C, U$).

To violate **A2**, S can no longer be redundant given C. To achieve this, we first introduce a correlation between S and C as before (to satisfy **A1**) and then we randomly replace columns in C with Gaussian noise $N(0, 1)$, but keep S the same. This procedure correlate

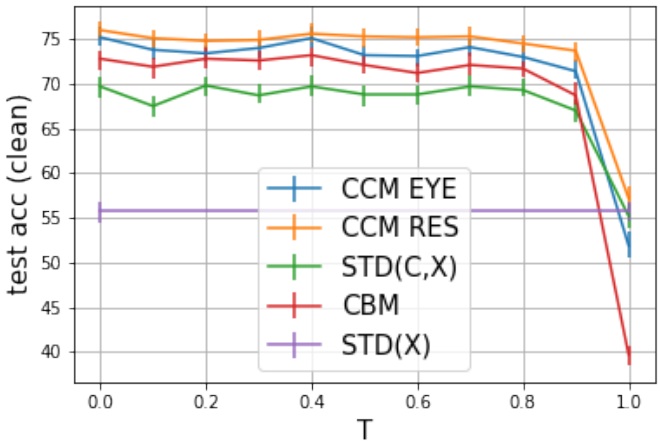


FIGURE 5.2: Model performance under the clean test set when violating **A1**. When C is learned using a biased dataset (sweeping T on the horizontal axis), we violate **A1**. Unless C is extremely corrupted (*e.g.*, $T = 1$), CCM performs relatively well.

S with U because the swapped out information becomes unknown concepts based on which S is generated. The more concepts swapped for noise, the less informative C becomes, increasing the relative value of S in predicting Y. For example, when 100 random dimensions of C are replaced with noise, a linear model trained with (C, S) significantly outperforms a model based on just C.

This concept swapping greatly affects CBM because it relies solely on C, which is corrupted. When S is removed (**Figure 5.3**), both CCM EYE and CCM RES outperform baselines until all concepts are corrupted (in which case CCM performs similarly to a standard model). The performance of $STD(X)$ is constant across settings because it does not rely on C. This experiment also shows that not all dimensions of C need to be relevant to the prediction for CCM to work. This is a desirable property as **expert knowledge with respect to relevant concepts could be flawed**.

In the case where S contains information outside of C and U (*i.e.*, $Y \not\perp\!\!\!\perp S|C,U$), our findings are similar (Appendix). All methods except CBM perform well on the biased dataset.

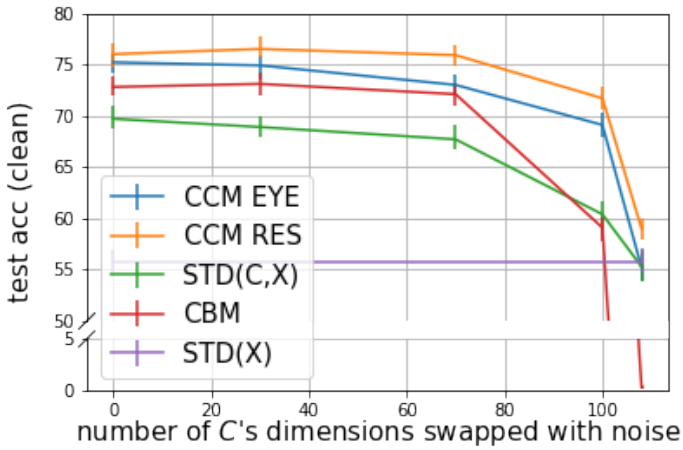


FIGURE 5.3: Model performance under the clean test set when violating **A2**. C becomes less informative when replaced with noise, presenting an advantage to using S and violating assumption **A2**. Despite this, CCM still performs well, even when large portions of C are irrelevant for the prediction.

5.4.2 Experiments on the MIMIC dataset

The **MIMIC-CXR** dataset [162], [163] consists of chest X-rays and corresponding radiology reports. These chest X-rays can be linked to MIMIC-IV [163], [164], which contains clinical data. Each chest X-ray is associated with 14 text-mined radiology report labels corresponding to 14 different radiological findings. Out of the 14 provided tasks, we chose cardiomegaly (enlarged heart) as the source task to learn C and edema (excess fluid in lungs) as the target task. The two tasks are related. Cardiomegaly is a structural abnormality of the heart and a sign of cardiac dysfunction. Patients with cardiac dysfunction are more likely to develop heart failure, and pulmonary edema can develop as a consequence of heart failure. As such, we expect that predictive features of cardiomegaly are useful concepts in diagnosing edema. The 14 provided labels are categorized as positive, negative, uncertain, and not mentioned. Following [165] and [155], we mapped uncertain labels to positive and discard images without labels. After discarding images, the cardiomegaly/edema tasks contained $108,785/107,510$ X-rays.

Instead of introducing synthetic shortcuts such as noise, we opted for a more realistic shortcut based on patient sex, which we extracted from the clinical data contained in

MIMIC-IV. In particular, we increased the correlation between male sex and edema by dropping T proportion of females/males with/without a positive label. In contrast, male sex was only mildly correlated with cardiomegaly without resampling (correlation coefficient of -0.025; Empirical 95% bootstrapped confidence interval of $(-0.031, -0.019)$). To obtain C, we trained an Inception V3 network pretrained on the ImageNet dataset to predict cardiomegaly. Then we used the last layer representation of the network as C (dimension 2048).

Model Training. Similar to the CUB experiment, we used the Inception V3 network initialized using the ImageNet dataset as the prediction model. We divided the chest X-ray datasets into train, validation, and test sets with a 64/16/20 random split. Then, we resampled the edema dataset such that male and edema are correlated. All methods were trained on this biased edema dataset. The hyperparameter search range was the same as the CUB experiments.

Evaluation. Since both S and Y are binary, we can resample the test set to vary the correlation between S and Y to stress test our model under different testing distributions. In particular, we swept the correlation between sex and edema from -1 (reversing the training correlation) to 1 (extremely biased distribution). A model robust to the sex shortcut should do well in all settings.

Results. First, the performance of all methods is significantly affected when the test correlation is decreased to the point of reversal with the correlation in the training set (**Figure 5.4**). This is inevitable as the shortcut provides information to predicting Y given C, violating **A2**. However, across the range of test correlation settings, when trained on the biased distribution (correlation between male and edema is 0.65), CCM EYE performs consistently better than baselines. Compared to CBM, CCM EYE is most effective when the testing correlation is similar to the training correlation. Compared to other baselines, CCM EYE is most effective when the shortcut is negatively correlated with the outcome, completely reversing the relationship observed during training. This demonstrates the robustness of CCM EYE against the sex shortcut. Similar trends hold when we vary the training correlation between male and edema (Appendix).

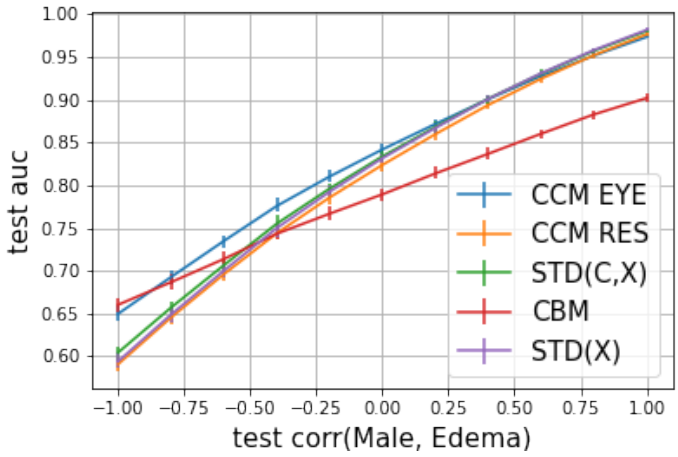


FIGURE 5.4: Result of the MIMIC-CXR experiment. The model is trained on a biased dataset where S and Y has a correlation of 0.65 and tested on sub-sampled dataset with different correlation. The result shows that CCM EYE is consistently better than baselines. The error bars are the 95% confidence intervals bootstrapped on the test set.

5.5 Summary & Conclusions

In this chapter, we proposed two approaches that use domain knowledge C to learn an accurate model while mitigating the use of shortcuts. Our methods do not assume C to be sufficient to make an accurate prediction and apply even to scenarios where $|corr(S, C)| = 1$, settings previous work have not addressed. Between our proposed methods, CCM EYE outperforms CCM RES when U is correlated with S. Applied to two datasets, we show that CCMs successfully reduce shortcut learning without sacrificing accuracy, even when our assumptions that C is invariant to S and S is redundant given C do not hold.

We note that a model may not use S even if S is correlated with Y. This happens either because C and U are more predictive than S, or when C and U are easier to extract (*e.g.*, requires less training iterations to accurately learn) from the input than S [166]. However, we cannot count on those facts to learn a robust model because verifying the predictability or the extractability of S versus C and U requires knowing all three variables in advance, which may not be the case.

This chapter also advances credible learning introduced in Chapter 4 by allowing domain knowledge to be expressed on the concept/representation space and making credible models non-linear. In this setting, other regularizers may also increase credibility (*e.g.*, $\|\boldsymbol{\theta}_x\|_2^2$). Future work could consider the task of finding the optimal credible regularizer to mitigate using shortcuts. Furthermore, while we explored two ways to obtain C (*i.e.*, using auxiliary labels and using a transfer learning setup), we have not compared different sources of C in terms of their robustness to shortcuts and the effort to obtain them, both important for the adoption of our technique. Future work should explore those directions. Going forward, we expect CCM to be a stepping stone towards building robust systems that can be safely applied in practice.

Chapter 6

Conclusion

This dissertation addressed the challenge of underspecification. In this setting, good testing performance does not always translate to good deployment performance because of the gap between the modeling world and the real world [8]. This gap can be the result of biased data collection and/or limited sample size, leading to multiple models with similar test performance but drastically different generalization error in the real world. Resolving underspecification is of particular interest to high stake domains such as healthcare where the cost of making a mistake is high and the data are often limited. Fortunately, beyond training data, experts often have knowledge about feature importance such as risk factors for a disease. Our primary thesis centers around the idea of using domain knowledge to select models that are generalizable beyond the training distribution.

In this dissertation, we presented approaches that address issues around model underspecification by building on work spanning several fields, including model interpretation, regularization and shortcut learning. We summarize challenges and our contributions in each field.

When selecting among models, model interpretation allows domain experts to examine the reasoning of a model (*e.g.*, importance of features) after it is trained to safeguard its deployment. However, as we described in Chapter 3, many existing approaches for estimating feature importance are problematic because they ignore or hide dependencies among features [14], [17], [23]. A causal graph, which encodes the relationships among input variables, can aid in this process. Unfortunately, current approaches that assign credit to nodes in the causal graph fail to explain the entire graph, limiting our understanding on how a feature affects the outcome [23], [120]. In order to understand both the direct and indirect impact of features on the output of a model, **we proposed Shapley Flow**,

a model agnostic explanation method that generalizes three previous game theoretic approaches and uniquely satisfies an extension of Shapley value axioms to graph (**Chapter 3**). By viewing feature attribution from a graph perspective, Shapley Flow allows machine learning practitioners to reason about feature interventions in relation to other features. Furthermore, by highlighting causal graphs, Shapley Flow reminds researchers about the importance/danger of reasoning based on causality/correlation in model interpretation.

Even when the explanation procedure is sound, a model's reasoning can still be confusing (*e.g.*, not conforming to well-established knowledge). This could happen because of underspecification: models pick up features that are correlated with expert identified features as opposed to using the expert features. The usual tool to resolve underspecification is regularization, through which we restrict the hypothesis space. However, as we described in Chapter 4, most regularization approaches do not distinguish between expert identified features and others, again allowing learning models with confusing reasoning [78], [80], [87]. For regularization approaches that do take into account domain knowledge [81], [88], they either densely use non-expert identified features or sparsely use expert identified features. In both cases, the resulting models do not conform to well-established knowledge. To address those issues in the linear setting, **we proposed the Expert Yielded Estimate penalty** (*i.e.*, EYE regularization), a regularization term with provable desirable properties that significantly increased alignment with domain knowledge without sacrificing accuracy when applied on two large scale clinical datasets (**Chapter 4**). This work debunks the notion that credibility comes at the cost of accuracy and provides a tool for researchers to correct confusing model reasoning with domain knowledge.

Finally, it is not clear when and how learning a *credible* model (*i.e.*, an accurate model with sensible reasoning) can help mitigate shortcut learning (*i.e.*, using features that are spuriously correlated with the target). As we described in Chapter 5, most approaches for mitigating shortcut learning assume that shortcuts are known *a priori* [105]–[108], [110], [111], [113], [114]. However, in practice, we might not have direct access to them, rather we have domain knowledge about what input features or functions on those features that are likely to not contain shortcuts. We refer to features that do not contain shortcuts as causally relevant features. The catch is that using those expert features alone prevents the model from learning yet unknown but causally relevant features from the data [19], [155], [167]. To address those issues, we **identified sufficient assumptions** for a credible model

to eliminate using shortcuts while incorporating unknown but causally relevant features. Furthermore, previous credible learning approaches required domain knowledge to be expressed in the input space, which is inconvenient or even infeasible for complex input modalities such as images. To address this limitation, we **proposed learning Concept Credible Models** in which we incorporate domain knowledge in the concept space instead of the input space (Chapter 5). This work provides a valuable tool to safeguard model deployment without requiring deployment data or direct knowledge of shortcuts.

There are several areas touched upon in this dissertation that could be interesting for further examination. Here we outline a few possibilities. First, we examine potential extensions to Shapley Flow. Shapley Flow assumes access to a complete causal graph of the input, which can be expensive if not infeasible to obtain. Future work could examine the implication of working with a partially defined causal graph or a causal graph learned directly from data. Relating the completeness and the quality of the causal graph to the utility of Shapley Flow is important for its real world application. For example, we need to know how sensitive Shapley Flow is to a wrongly specified input causal graph in order to safely apply this technique. Another interesting line of future work involves examining the trade-off made in our design decisions for formalizing graph based explanations. This involves detailed analysis on the definition of edge removal, the payoff function, and the choice of summary statistics used to report the effect of removing edges in different contexts (*e.g.*, coalition), paralleling a recent effort [168] to characterize feature based explanation method. As an example, since the publication of Shapley Flow, Singal *et al.* [169] derived a different unique axiomatic edge attribution method by changing the payoff function. Instead of characterizing the attribution of an edge using the change in output, they focus on the change in attribution to the source node of the edge. This allows them to bypass our need to define the complex notion of boundary consistent history while still satisfying an extension to Shapley value axioms. Despite the equivalence of both approaches when the model and the causal graph are linear, in nonlinear cases, little is known about their differences and the resulting implications. Similarly, while Shapley value is one way to summarize the effect of edge removal under different contexts[1], it is not clear how it compares to other axiomatic summaries such as Banzhaf value [171] and minimal cause motivated summaries [172].

[1] with some limitations exposed in [170]

Second, we examine potential extensions to credible learning. While we focused on credibility, our proposed regularization technique could be extended to other settings in which the user would like to guide variable selection. For example, instead of encoding knowledge pertaining to which variables are known risk factors, $\mathbf{r}$ could encode information about which variables are actionable, resulting in a more *actionable* model. Similarly, one can use $1 - \mathbf{r}$ to encode the cost of obtaining features (assuming that some features go through longer pipelines to obtain than others). This in turn can cut down inference time without sacrificing accuracy. Another limitation of credible learning is that we assumed $\mathbf{r}$ to be binary. In reality, there are other forms of constraints an user might want a model to satisfy. For example, researchers could enforce monotonicity in variables, incorporate known logic rules, or relax $\boldsymbol{r}$ from binary to fractional so as to encode experts' confidence. Those changes require developing new notions of credibility and techniques to achieve them. Interested readers can refer to [173] for further inspirations to encode domain knowledge.

Third, we examine potential extensions to concept credible models (CCM). One area that needs more attention is comparing the robustness of known concepts (C) coming from different sources. In our work, we explored two settings to obtain C, one using auxiliary labels and another using a transfer learning setup. However, there are many more potential sources of C such as legacy code [174] and rule of thumbs that practitioners follow. These sources of C can be less accurate in training but potentially more robust to shortcuts than learning from the data alone. identifying which sources of C fit well with our assumptions would be valuable. Another area that merits attention is marrying concept based interpretation with CCM. While CCM is capable of ruling out shortcuts that are redundant given C, it does not replace the need to carefully interpret and examine concepts picked up by the model, as some shortcuts contain more information. Future work could consider complementing CCM with unknown concept interpretation to rule out other kinds of shortcuts.

Finally, all techniques presented in this dissertation require soliciting inputs from experts. In particular, Shapley Flow requires experts to specify an input causal graph and credible learning requires experts to go through and mark concepts (or features) as important or not. If the graph is dense or the number of concepts is large, applying our techniques can be time consuming. Therefore future work should examine potential ways to

decrease the cognitive burdens for experts to provide their knowledge. For example, for Shapley Flow, one can consider first learning a causal graph from the data and then only ask experts to verify causal links that are uncertain. In presenting the edge attribution, one can either only show the most important edges, or cluster features and aggregate edges between clusters so that there are less interactions for experts to look at. If experts need more details, we can expand the cluster of interest to show them the interaction within the cluster. For credible learning, one can consider first mining relevant features from the literature to create a short list of candidates for experts to consider. For example, one can first apply information retrieval techniques on medical journals to extract out potential risk factors for a disease and then let experts examine and modify the identified risk factors. Furthermore, future work should compare the cognitive cost of different ways to solicit expert knowledge. For example, asking experts to specify a related task may be much easier than asking them to directly specify relevant features.

The main contributions of this dissertation are: 1) enabling a system level view of Shapley value based feature attribution, 2) formalizing the idea of credible learning with an approach that leads to accurate linear models with sensible explanations, and 3) connecting credible and shortcut learning while incorporating non-input level domain knowledge. Going forward, we expect techniques developed in this dissertation to help build robust machine learning models that generalize beyond the training environment.

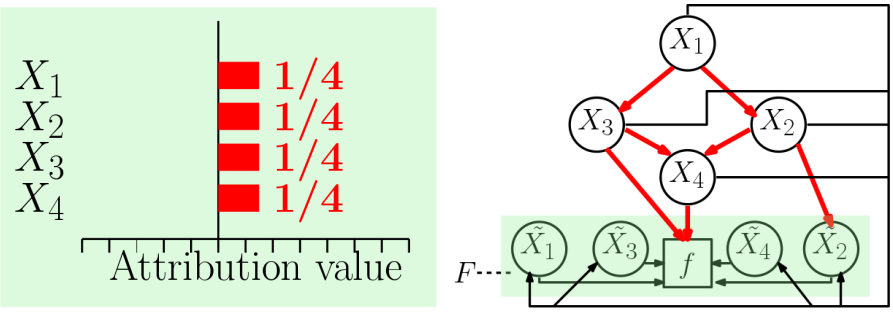
Appendices

Appendix A

Shapley Flow Appendix

A.1 Explanation boundary for on-manifold methods without a causal graph

On-manifold perturbation using conditional expectations can be unified with Shapley Flow using explanation boundaries (**Figure A.1a**). Here we introduce $\tilde{X}_i$ as an auxiliary variable that represent the imputed version of X_i. Perturbing any feature X_i affects all input to the model ($\tilde{X}_1$, $\tilde{X}_2$, $\tilde{X}_3$, $\tilde{X}_4$) so that they respect the correlation in the data after the perturbation. When X_i has not been perturbed, $\tilde{X}_j$ treats it as missing for $i, j \in [1, 2, 3, 4]$ and would sample $\tilde{X}_j$ from the conditional distribution of X_j given non-missing predecessors. The red edges contain causal links from **Figure 3.1**, whereas the black edges are the causal structure used by the on-manifold perturbation method. The credit is equally split among the features because they are all correlated. Again, although giving X_1 and



(A) On-manifold attribution (B) On-manifold boundary

FIGURE A.1: On manifold perturbation methods can be computed using Shapley Flow with a specific explanation boundary.

X_2 credit is not true to f, it is true to the model defined by F.

A.2 The Shapley Flow algorithm

A pseudo code implementation highlighting the main ideas for Shapley Flow is included in **Algorithm 1**. For approximations, instead of trying all edge orderings in line 15 of **Algorithm 1**, one can try random orderings and average over the number of orderings tried.

A.3 Shapley Flow's uniqueness proof

Without loss of generality, we can assume $\mathcal{G}$ has a single source node s. We can do this because every node in a causal graph is associated with an independent noise node [30, Chapter 6]. For deterministic relationships, the function for a node doesn't depend on its noise. Treating those noise nodes as a single node, s, wouldn't have changed any boundaries that already exist in the original graph. Therefore we can assume there is a single source node s.

A.3.1 At most one solution satisfies the axioms

Assuming that a solution exists, we show that it must be unique.

Proof. We adapt the argument from the Shapley value uniqueness proof [1], by defining basis payoff functions as carrier games. Choose any boundary $\mathcal{B}$, we show here that any game defined on the boundary has a unique attribution. We also drop the subscript $\mathcal{B}$ in the proof as there is no ambiguity. Note that since every edge will appear in some boundary, if all boundary edges are uniquely attributed to, all edges have unique attributions. A carrier game associated with coalition (ordered list) O is a game with payoff function v^O such that $v^O(S) = 1(0)$ if coalition S starts with O (otherwise 0). By dummy player, we know that only the last edge e in O gets credit and all other edges in the cut set are

[1] https://ocw.mit.edu/courses/economics/14-126-game-theory-spring-2016/lecture-notes/MIT14_126S16_cooperative.pdf

Algorithm 1 Shapley Flow pseudo code

Input: A computational graph $\mathcal{G}$ (each node i has a function f_i), foreground sample $\mathbf{x}$, background sample $\mathbf{x}'$
Output: Edge attribution $\phi : E \to \mathbb{R}$
Initialization:
$\mathcal{G}$: add an new source node pointing to original source nodes.

function SHAPLEYFLOW($\mathcal{G}$, $\mathbf{x}'$, $\mathbf{x}$)
    INITIALIZE($\mathcal{G}$, $\mathbf{x}'$, $\mathbf{x}$) ▷ Set up game v for any boundary in $\mathcal{G}$
    $s \leftarrow$ SOURCE($\mathcal{G}$) ▷ Obtain the source node
    return DFS(s, $\{\}$, $[]$)
end function

function DFS(s, D, S)
    ▷ s is a node, D is the data side of the current boundary, S is coalition
    ▷ Using Python list slice notation
    Initialize ϕ to output 0 for all edges
    if ISSINKNODE(s) **then**
        ▷ Here we overload D to refer to its boundary
        $\phi(S[-1]) \leftarrow v_D(S) - v_D(S[:-1])$ ▷ Difference in output is attributed to the edge
        return ϕ
    end if

    for $p \leftarrow$ AllOrderings(Children(s)) **do** ▷ Try all orderings/permutations of the node's children
        for $c \leftarrow p$ **do** ▷ Follow the permutation to get the node one by one
            edgeCredit $\leftarrow$ DFS(c, $D \cup \{s\}$, $S + [(s, c)]$) ▷ Recurse downward
            $\phi \leftarrow \phi + \frac{\text{edgeCredit}}{\text{NumChildren}(s)!}$ ▷ Average attribution over number of runs
            $\phi(S[-1]) \leftarrow \phi(S[-1]) + \frac{\text{edgeCredit}(s,c)}{\text{NumChildren}(s)!}$ ▷ Propagate upward
        end for
    end for
    return ϕ
end function

dummy because a coalition is constructed in order (only adding e changes the payoff from 0 to 1). Note that in contrast with the traditional symmetry axiom [18] defined on a set of players, the symmetry axiom is not explicit in our case (it is made implicitly) because not all edges in the carrier game are symmetric with each other (observe that e is different from all other edges, which are dummy), thus we do not need an explicit symmetry axiom to argue for unique attribution in the carrier game. Furthermore, e must be an edge in the boundary to form a valid game because boundary edges are the only edges that are connected to the model defined by the boundary. Therefore we give 0 credit to edges in the cut set other than e (because they are *dummy players*). By the *efficiency axiom*, we give $\sum_{h\in\tilde{\mathcal{H}}} \frac{\nu_{\mathcal{B}}(h)}{|\tilde{\mathcal{H}}|} - \nu_{\mathcal{B}}([])$ credit to e where $\tilde{\mathcal{H}}$ is the set of all possible boundary consistent histories as defined in **Section 3.3.3**. This uniquely attributed the boundary edges for this game.

We show that the set of carrier games associated with every coalition that ends in a boundary edge (denoted as $\hat{\mathcal{C}}$) form basis functions for all payoff functions associated with the system. Recall from **Section 3.3.2** that $\tilde{\mathcal{C}}$ is the set of *boundary consistent coalitions*. We show here that payoff value on coalitions from $\tilde{\mathcal{C}}$ is redundant given $\hat{\mathcal{C}}$. Note that $\tilde{\mathcal{C}} \setminus \hat{\mathcal{C}}$ represents all the coalitions that do not end in a boundary edge. For $c \in \tilde{\mathcal{C}} \setminus \hat{\mathcal{C}}$, $v^O(c) = v^O(c[:-1])$ (using Python's slice notation on list) because only boundary edges are connected to the model defined by the boundary. Therefore it suffices to show that v^O is linearly independent for $O \in \hat{\mathcal{C}}$. For a contradiction, assume for all $c \in \hat{\mathcal{C}}$, $\sum_{O\subseteq\hat{\mathcal{C}}} \alpha^O v^O(c) = 0$, with some non zero $\alpha^O \in \mathbb{R}$ (definition of linear dependence). Let S be a coalition with minimal length such that $\alpha^S \neq 0$. We have $\sum_{O\subseteq\hat{\mathcal{C}}} \alpha^O v^O(S) = \alpha^S$, a contradiction.

Therefore for any ν we have unique α's such that $\nu = \sum_{O\subseteq\hat{\mathcal{C}}} \alpha^O v^O$. Using the *linearity axiom*, we have

$$\phi_\nu = \phi_{\sum_{O\subseteq\hat{\mathcal{C}}} \alpha^O v^O} = \sum_{O\subseteq\hat{\mathcal{C}}} \alpha^O \phi_{v^O}$$

The uniqueness of α and ϕ_{v^O} makes the attribution unique if a solution exists. Axioms used in the proof are italicized.

□

A.3.2 Shapley Flow satisfies the axioms

Proof. We first demonstrate how to generate all boundaries. Then we show that Shapley Flow gives boundary consistent attributions. Following that, we look at the set of histories that can be generated by DFS in boundary $\mathcal{B}$, denoted as $\Pi^{\text{dfs}}_{\mathcal{B}}$. We show that $\Pi^{\text{dfs}}_{\mathcal{B}} = \tilde{\mathcal{H}}_{\mathcal{B}}$. Using this fact, we check the axioms one by one.

- Every boundary can be "grown" one node at a time from $D = \{s\}$ where s is the source node: Since the computational graph $\mathcal{G}$ is a directed acyclic graph (DAG), we can obtain a topological ordering of the nodes in $\mathcal{G}$. Starting by including the first node in the ordering (the source node s), which defines a boundary as $(D = \{s\}, F = \text{Nodes}(\mathcal{G}) \backslash D)$, we grow the boundary by adding nodes to D (removing nodes from F) one by one following the topological ordering. This ordering ensures the corresponding explanation boundary is valid because the cut set only flows from D to F (if that's not true, then one of the dependency nodes is not in D, which violates topological ordering).

  Now we show every boundary can be "grown" in this fashion. In other words, starting from an arbitrary boundary $\mathcal{B}_1 = (D_1, F_1)$, we can "shrink" one node at a time to $D = \{s\}$ by reversing the growing procedure. First note that, D_1 must have a node with outgoing edges only pointing to nodes in F_1 (if that's not the case, we have a cycle in this graph because we can always choose to go through edges internal to D_1 and loop indefinitely). Therefore we can just remove that node to arrive at a new boundary (now its incoming edges are in the cut set). By the same argument, we can keep removing nodes until $D = \{s\}$, completing the proof.

- Shapley Flow gives boundary consistent attributions: We show that every boundary grown has edge attribution consistent with the previous boundary. Therefore all boundaries have consistent edge attribution because the boundary formed by any two boundary's common set of nodes can be grown into those two boundaries using the property above. Let's focus on the newly added node c from one boundary to the next. Note that a property of depth first search is that every time c's value is updated, its outgoing edges are activated in an atomic way (no other activation of edges occur between the activation of c's outgoing edges). Therefore, the change

in output due to the activation of new edges occur together in the view of edges upstream of c, thus not changing their attributions. Also, since c's outgoing edges must point to the model defined by the current boundary (otherwise it cannot be a valid topological ordering), they don't have down stream edges, concluding the proof.

- $\Pi_{\mathcal{B}}^{\text{dfs}} = \tilde{\mathcal{H}}_{\mathcal{B}}$: Since attribution is boundary consistent, we can treat the model as a blackbox and only look at the DFS ordering on the data side. Observe that the edge traversal ordering in DFS is a valid history because a) every edge traversal can be understood as a message received through edge , b) when every message is received, the node's value is updated, and c) the new node's value is sent out through every outgoing edge by the recursive call in DFS. Therefore the two side of the equation are at least holding the same type of object.

  We first show that $\Pi_{\mathcal{B}}^{\text{dfs}} \subseteq \tilde{\mathcal{H}}_{\mathcal{B}}$. Take $h \in \Pi_{\mathcal{B}}^{\text{dfs}}$, we need to find a history h^* in $\mathcal{B}^*$ such that a) h can be expanded into h^* and b) for any boundary, there is a history in that boundary that can be expanded into h^*. Let h^* be any history expanded using DFS that is aligned with h. To show that every boundary can expand into h^*, we just need to show that the boundaries generated through the growing process introduced in the first bullet point can be expanded into h^*. The base case is $D = \{s\}$. There must have an ordering to expand into h^* because h^* is generated by DFS, and that DFS ensures that every edge's impact on the boundary is propagated to the end of computation before another edge in D is traversed. Similarly, for the inductive step, when a new node c is added, we just follow the expansion of its previous boundary to reach h^*.

  Next we show that $\tilde{\mathcal{H}}_{\mathcal{B}} \subseteq \Pi_{\mathcal{B}}^{\text{dfs}}$. First observe that for history h_1 in $\mathcal{B}_1 = (D_1, F_1)$ and history h_2 in $\mathcal{B}_2 = (D_2, F_2)$ with $F_2 \subseteq F_1$, if h_1 cannot be expanded into h_2, then $HE(h_1) \cap HE(h_2) = \emptyset$ because they already have mismatches for histories that doesn't involve passing through $\mathcal{B}_1$. Assume we do have $h \in \tilde{\mathcal{H}}_{\mathcal{B}}$ but $h \notin \Pi_{\mathcal{B}}^{\text{dfs}}$. To derive a contradiction, we shrink the boundary one node at a time from $\mathcal{B}$, again using the procedure described in the first bullet point. We denote the resulting boundary formed by removing n nodes as $\mathcal{B}_{-n}$. Since h is assumed to be boundary consistent, there exist $h_{\mathcal{B}_{-1}} \in \mathcal{H}_{\mathcal{B}_{-1}}$ such that $h_{\mathcal{B}_{-1}}$ must be able to expand into h.

Say the two boundaries differ in node c. Note that any update to c crosses $\mathcal{B}_{-1}$, therefore its impact must be reached by F before another event occurs in D_{-1}. Since all of c's outgoing edges crosses $\mathcal{B}$, any ordering of messages sent through those edges is a DFS ordering from c. This means that if $h_{\mathcal{B}_{-1}}$ can be reached by DFS, so can $h_{\mathcal{B}}$, violating the assumption. Therefore, $h_{\mathcal{B}_{-1}} \notin \Pi^{\text{dfs}}_{\mathcal{B}_{-1}}$ and $h_{\mathcal{B}_{-1}} \in \tilde{\mathcal{H}}_{\mathcal{B}_{-1}}$ (the latter because $h_{\mathcal{B}_{-1}}$ can expand into a history that is consistent with all boundaries by first expanding into h). We run the same argument until $D = \{s\}$. This gives a contradiction because in this boundary, all histories can be produced by DFS.

- Efficiency: Since we are attributing credit by the change in the target node's value following a history h given by DFS, the target for this particular DFS run is thus $v_{\mathcal{B}}(h) - v_{\mathcal{B}}([])$. Average over all DFS runs and noting that $\tilde{\mathcal{H}}_{\mathcal{B}} = \Pi^{\text{dfs}}_{\mathcal{B}}$ gives the target $\sum_{h \in \tilde{\mathcal{H}}_{\mathcal{B}}} v_{\mathcal{B}}(h)/|\tilde{\mathcal{H}}_{\mathcal{B}}| - v_{\mathcal{B}}([])$. Noting that each update in the target node's value must flow through one of the boundary edges. Therefore the sum of boundary edges' attribution equals to the target.

- Linearity: For two games of the same boundary v and u, following any history, the sum of output differences between the two games is the output difference of the sum of the two games, therefore ϕ_{v+u} would not differ from $\phi_v + \phi_u$. It's easy to see that extending addition to any linear combination wouldn't matter.

- Dummy player: Since Shapley Flow is boundary consistent, we can just run DFS up to the boundary (treat F as a blackbox). Since every step in DFS remains in the coalition $\tilde{\mathcal{C}}_{\mathcal{B}}$ because $\Pi^{\text{dfs}}_{\mathcal{B}} \subseteq \tilde{\mathcal{H}}_{\mathcal{B}}$, if an edge is dummy, every time it is traversed through by DFS, the output won't change by definition, thus giving it 0 credit.

□

Therefore Shapley Flow uniquely satisfies the axioms. We note that efficiency requirement simplifies to $f(\boldsymbol{x}) - f(\boldsymbol{x'})$ when applying it to an actual model because all histories from DFS would lead the target node to its target value. We can prove a stronger claim that actually all nodes would reach its target value when DFS finishes. To see that, we do an induction on a topological ordering of the nodes. The source nodes reaches its final value by definition. Assume this holds for the k^{th} node. For the $k+1^{\text{th}}$ node, its parents

achieves target value by induction. Therefore DFS would make the parents' final values visible to this node, thus updating it to the target value.

A.4 Causal graphs

While the nutrition dataset is introduced in the main text, we describe an additional dataset to further demonstrate the usefulness of Shapley Flow. Moreover, we describe in detail how the causal relationship is estimated. Note that the resulting causal graphs are over-simplifications of the true causal graphs; the relationship between source nodes (*e.g.*, race and sex) and other features is far more complex. These causal graphs are used as proof of concepts to show both the direct and indirect effects of features on the prediction output. The causal graphs for the nutrition dataset and the income dataset are visualized in **Figure A.2**.

A.4.1 The Census Income dataset

The Census Income dataset consists of $32,561$ samples with 12 features. The task is to predict whether one's annual income exceeds $50k$. We assume a causal graph, similar to that used by [23] (**Figure A.2b**). Attributes determined at birth e.g., sex, native country, and race act as source nodes. The remaining features (marital status, education, relationship, occupation, capital gain, work hours per week, capital loss, work class) have fully connected edges pointing from their causal ancestors. All features have a directed edge pointing to the model.

A.4.2 Causal Effect Estimation

Given the causal structure described above, we estimate the relationship among variables using XGBoost. More specifically, using an $80/20$ train test split, we use XGBoost to learn the function for each node. If the node value is categorical, we train to minimize cross entropy loss. Otherwise, we minimize mean squared error. Models are fitted by 100 XGBoost trees with a max depth of 3 for up to 1000 epochs. Since features are rarely perfectly determined by their dependency node, we add independent noise nodes to account for

this effect. That is, each non-sink node is pointed to by a unique noise node that account for the residue effect of the prediction.

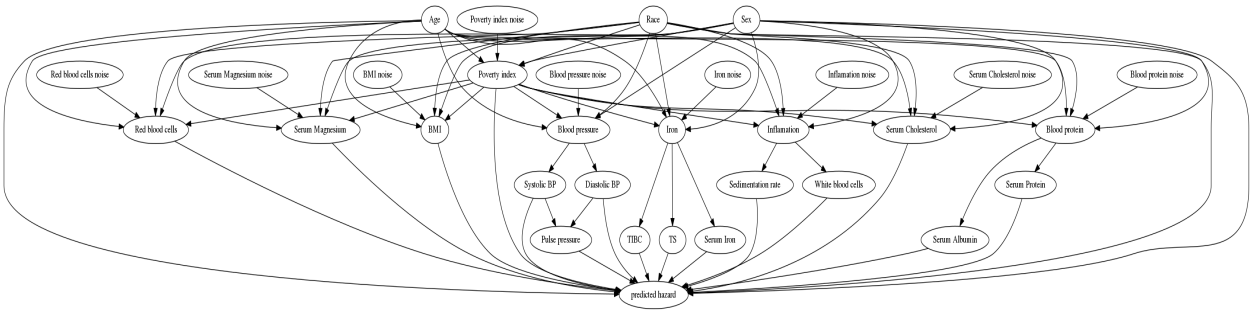
Depending on whether the variable is discrete or continuous, we handle the noise differently. For continuous variables, the noise node's value is the residue between the prediction and the actual value. For discrete variables, we assume the actual value is sampled from the categorical distribution specified by the prediction. Therefore the noise node's value is any possible random number that could result in the actual value. As a concrete example for handling discrete variable, consider a binary variable y, and assume the trained categorical function f gives $f(\boldsymbol{x}) = [0.3, 0.7]$ where $\boldsymbol{x}$ is the foreground value of the input to predict y. We view the data generation as the following. The noise term associated with y is treated as a uniform random variable between 0 and 1. If it lands within 0 to 0.3, y is sampled to be 0, otherwise 1 (matching the categorical function of 70% chance of sampling y to be 1). Now if we observe the foreground value of y to be 0, it means the foreground value of noise must be uniform between 0 to 0.3. Although we cannot infer the exact value of the noise, we can sample the noise from 0 to 0.3 multiple times and average the resulting attribution.

A.5 Additional Results

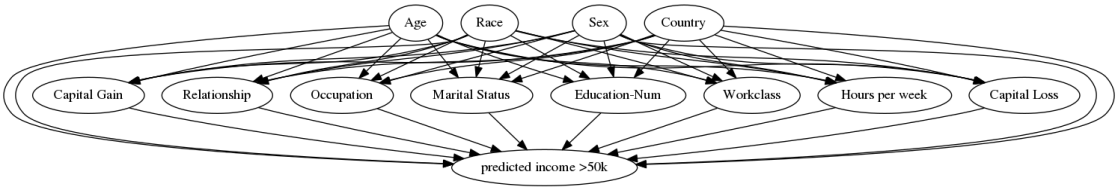
In this section, we first present additional sanity checks with synthetic data. Then we show additional examples from both the nutrition and income datasets to demonstrate how a complete view of boundaries should be preferable over single boundary approaches.

A.5.1 Additional Sanity Checks

We include further sanity check experiments in this section. The first sanity check consists of a chain with 4 variables. Each node along the chain is an identical copy of its predecessor and the function to explain only depends on X_4 (**Figure A.3**). The dataset is created by sampling $X_1 \sim \mathcal{N}(0, 1)$, that is a standard normal distribution, with 1000 samples. We use the first sample as background, and explain the second sample (one can choose arbitrary samples to obtain the same insights). As shown in **Figure A.3**, independent SHAP fails to show the indirect impact of X_1, X_2, and X_3, ASV fails to show the direct impact

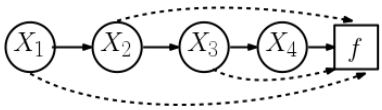


(A) Causal graph for the nutrition dataset



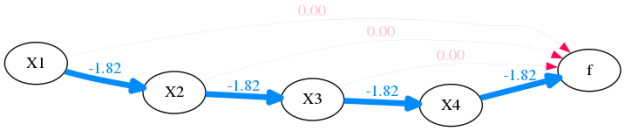
(B) Causal graph for the Census Income dataset

FIGURE A.2: The causal graphs we used for the two real datasets. Note that each node in the causal graph for (a) is given a noise node to account for random effects. The noise nodes are omitted for better readability for (b). The resulting causal structures are over-simplifications of the true causal structures; the relationship between source nodes (*e.g.,* race and sex) and other features is far more complex. These causal graphs are used as proof of concepts to show both the direct and indirect effects of features on the prediction output.



(A) chain dataset

	Independent	On-manifold	ASV
X4	-1.82	-0.45	0.0
X1	0.0	-0.45	-1.82
X3	0.0	-0.45	0.0
X2	0.0	-0.45	0.0



(B) Shapley Flow

FIGURE A.3: **(a)** The chain dataset contains exact copies of nodes. The dashed edges denotes dummy dependencies. **(b)** While Shapley Flow shows the entire path of influence, other baselines fails to capture either direct and indirect effects.

Methods	Income	Nutrition	Synthetic
Independent	**0.0** ($\pm$ 0.0)	**0.0** ($\pm$ 0.0)	**0.0** ($\pm$ 0.0)
On-manifold	0.4 ($\pm$ 0.3)	1.3 ($\pm$ 2.5)	0.8 ($\pm$ 0.7)
ASV	0.4 ($\pm$ 0.6)	1.5 ($\pm$ 3.3)	1.2 ($\pm$ 1.4)
Shapley Flow	**0.0** ($\pm$ 0.0)	**0.0** ($\pm$ 0.0)	**0.0** ($\pm$ 0.0)

TABLE A.1: Shapley Flow and independent SHAP have lower mean absolute error (std) for direct effect of features on linear models.

of X_4, on manifold SHAP fails to fully capture both the direct and indirect importance of any edge.

The second sanity check consists of linear models as described in **Section 3.4.3**. We include the full result with the income dataset added in **Table A.1** and **Table A.2** for direct and indirect effects respectively. The trend for the income dataset algins with the nutrition and synthetic dataset: only Shapley Flow makes no mistake for estimating both direct and indirect impact. Independent Shap only does well for direct effect. ASV only does well for indirect effects (it only reaches zero error when evaluated on source nodes).

Methods	Income	Nutrition	Synthetic
Independent	0.1 ($\pm$ 0.2)	0.8 ($\pm$ 2.7)	1.1 ($\pm$ 1.4)
On-manifold	0.4 ($\pm$ 0.3)	0.9 ($\pm$ 1.6)	1.5 ($\pm$ 1.5)
ASV	0.1 ($\pm$ 0.1)	0.6 ($\pm$ 1.9)	1.1 ($\pm$ 1.5)
Flow	**0.0** ($\pm$ 0.0)	**0.0** ($\pm$ 0.0)	**0.0** ($\pm$ 0.0)

TABLE A.2: Shapley Flow and ASV have lower mean absolute error (std) for indirect effect on linear models.

A.5.2 Additional examples

In this section, we analyze another example from the nutrition dataset (**Figure A.4**) and 3 additional example from the adult censor dataset.

Independent SHAP ignores the indirect impact of features. Take an example from the nutrition dataset (**Figure A.4**). The race feature is given low attribution with independent SHAP, but high importance in ASV. This happens because race, in addition to its direct impact, indirectly affects the output through blood pressure, serum magnesium, and blood protein, as shown by Shapley Flow (**Figure A.4a**). In particular, race partially accounts for the impact of serum magnesium because changing race from Black to White on average increases serum magnesium by 0.07 meg/L in the dataset (thus partially explaining the increase in serum magnesium changing from the background sample to the foreground). Independent SHAP fails to account for the indirect impact of race, leaving the user with a potentially misleading impression that race is irrelevant for the prediction.

On-manifold SHAP provides a misleading interpretation. With the same example (**Figure A.4**), we observe that on-manifold SHAP strongly disagrees with independent SHAP, ASV, and Shapley Flow on the importance of age. Not only does it assign more credit to age, it also flips the sign, suggesting that age is protective. However, **Figure A.5a** shows that age and earlier mortality are positively correlated; then how could age be protective? **Figure A.5b** provides an explanation. Since SHAP considers all partial histories regardless of the causal structure, when we focus on serum magnesium and age, there are two cases: serum magnesium updates before or after age. We focus on the first case because it is where on-manifold SHAP differs from other baselines (all baselines already consider the second case as it satisfies the causal ordering). When serum magnesium updates before age, the expected age given serum magnesium is higher than the foreground age (yellow line above the black marker). Therefore when age updates to its foreground

value, we observe a decrease in age, leading to a decrease in the output (so age appears to be protective). Serum magnesium is just one variable from which age steals credit. Similar logic applies to TIBC, red blood cells, serum iron, serum protein, serum cholesterol, and diastolic BP. From both an in/direct impact perspective, on-manifold perturbation can be misleading since it is based not on causal but on observational relationships.

ASV ignores the direct impact of features. As shown in **Figure A.4**, serum magnesium appears to be more important in independent SHAP compared to ASV. From Shapley Flow (**Figure A.4a**), this difference is explained by race as its edge to serum magnesium has a negative impact. However, looking at ASV alone, one fails to understand that intervening on serum magnesium could have a larger impact on the output.

Shapley Flow shows both direct and indirect impacts of features. Focusing on the attribution given by Shapley Flow (**Figure A.4a**). We not only observe similar direct impacts in variables compared to independent SHAP, but also can trace those impacts to their source nodes, similar to ASV. Furthermore, Shapley Flow provides more detail compared to other approaches. For example, using Shapley Flow we gain a better understanding of the ways in which race impacts survival. The same goes for all other features. This is useful because causal links can change (or break) over time. Our method provides a way to reason through the impact of such a change.

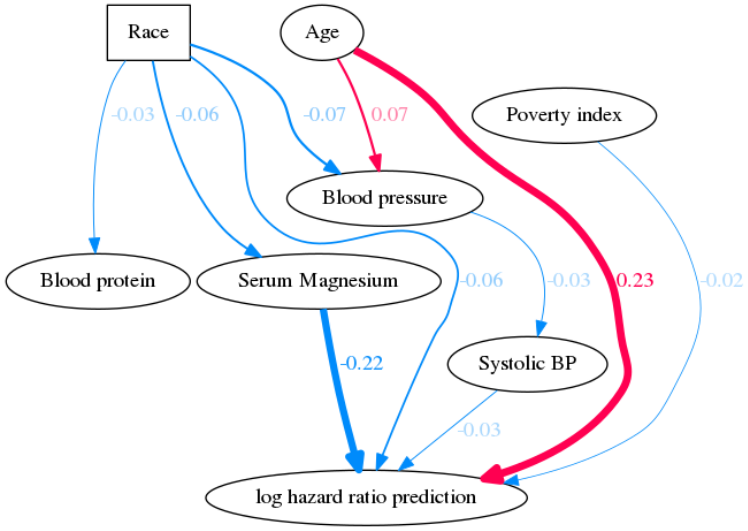
Figure A.6 gives an example of applying Shapley Flow and baselines on the income dataset. Note that the attribution to capital gain drops from independent SHAP to on-manifold SHAP and ASV. From Shapley Flow, we know the decreased attribution is due to age and race. More examples are shown in **Figure A.7** and **A.8**.

A.5.3 A global understanding with Shapley Flow

In addition to explaining a particular example, one can explain an entire dataset with Shapley Flow. Specifically, for multi-class classification problems, we take the average of attributions for the probability predicted for the actual class, in accordance with [23]. A demonstration on the income dataset using 1000 randomly selected examples is included in **Figure A.9**. As before, we use a single shared background sample for explanation.

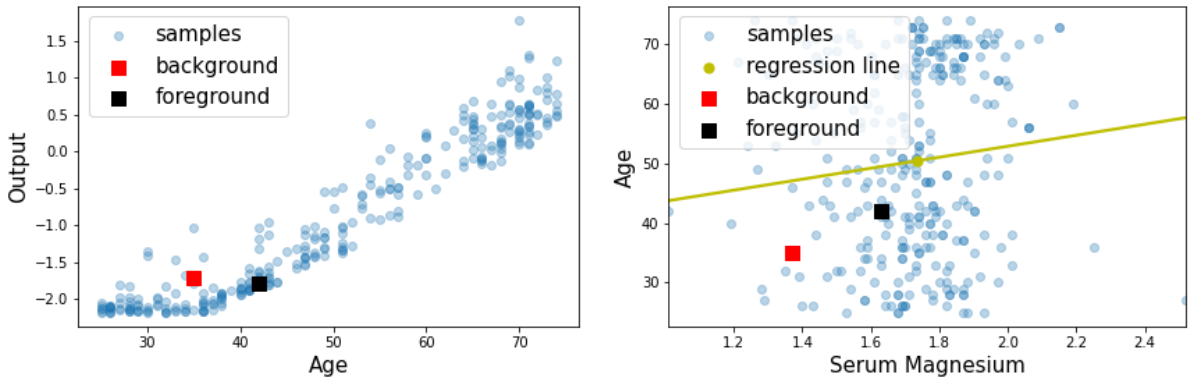
Top features	Age	Serum Magnesium	Race
Background sample	35.0	1.37	Black
Foreground sample	42.0	1.63	white

Attributions	Independent	On-manifold	ASV
Age	0.23	-0.38	0.3
Serum Magnesium	-0.21	-0.02	-0.15
Race	-0.06	0.04	-0.24
Pulse pressure	0.0	-0.08	0.0
Diastolic BP	0.0	0.08	0.0
Serum Cholesterol	0.0	0.07	0.0
Serum Protein	0.01	0.06	0.0
Serum Iron	0.0	0.05	0.0
Poverty index	-0.02	0.01	-0.01
Systolic BP	-0.03	-0.01	0.0
Red blood cells	0.0	0.05	0.0
Blood protein	0.0	0.0	0.04
TIBC	0.0	0.04	0.0
Blood pressure	0.0	0.0	-0.03
TS	0.0	0.03	0.0
BMI	-0.0	-0.03	-0.0
Sex	0.0	0.02	0.0
Serum Albumin	0.0	-0.01	0.0
White blood cells	0.01	-0.01	0.0
Sedimentation rate	0.0	0.01	0.0
Inflamation	0.0	0.0	0.01
Iron	0.0	0.0	0.0



(A) Shapley Flow

FIGURE A.4: Comparison among baselines on a sample (top table) from the nutrition dataset, showing top 10 features/edges. As noted in the main text this graph is an oversimplification and is not necessarily representative of the true underlying causal relationship.



(A) Age vs. output

(B) Age vs. magnesium

FIGURE A.5: Age appears to be protective in on-manifold SHAP because it steals credit from other variables.

Here, we observe that although the relative importance across independent SHAP, on-manifold SHAP, and ASV are similar, age and sex have opposite direct versus indirect impact as shown by Shapley Flow.

A.5.4 Example with multiple background samples

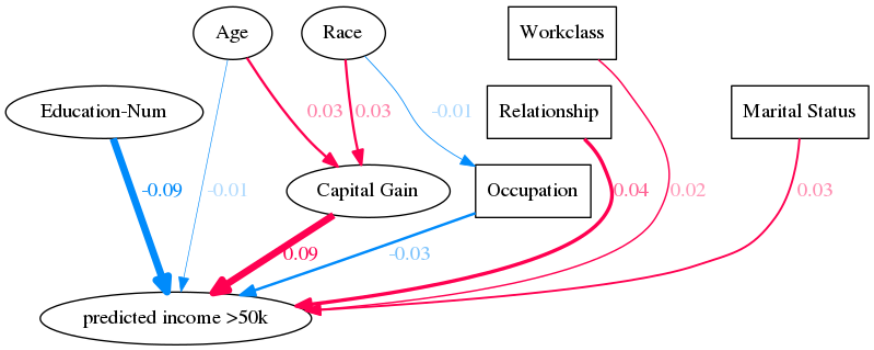
An example with 100 background samples is shown in **Figure A.10**. Shapley Flow shows a holistic picture of feature importance, while other baselines only show part of the picture.

Independent SHAP ignores the indirect impact of features. Take an example from the nutrition dataset (**Figure A.10**). Independent SHAP only considers the direct impact of systolic blood pressure, and ignores its potential impact on pulse pressure (as shown by Shapley Flow in **Figure A.10a**). If the causal graph is correct, independent SHAP would underestimate the effect of intervening on Systolic BP.

On-manifold SHAP provides a misleading interpretation. With the same example (**Figure A.10**), we observe that on-manifold SHAP strongly disagrees with independent SHAP, ASV, and Shapley Flow on the importance of age. In particular, it flips the sign on the importance of age. Since the background age (50) is very close to the foreground age (51), we would not expected age to significantly affect the prediction. **Figure A.11b** provides an explanation. Since SHAP considers all partial histories regardless of the causal

	Background sample	Foreground sample
Age	39	35
Workclass	State-gov	Federal-gov
Education-Num	13	5
Marital Status	Never-married	Married-civ-spouse
Occupation	Adm-clerical	Farming-fishing
Relationship	Not-in-family	Husband
Race	White	Black
Sex	Male	Male
Capital Gain	2174	0
Capital Loss	0	0
Hours per week	40	40
Country	United-States	United-States

	Independent	On-manifold	ASV
Education-Num	-0.12	-0.11	-0.09
Relationship	0.05	0.06	0.04
Capital Gain	0.09	0.01	0.03
Occupation	-0.03	-0.07	-0.02
Marital Status	0.04	0.05	0.03
Workclass	0.02	0.03	0.02
Race	-0.01	-0.03	0.01
Age	-0.01	-0.01	0.02
Capital Loss	0.0	0.03	0.0
Country	0.0	0.03	0.0
Sex	0.0	0.03	0.0
Hours per week	0.0	0.0	0.0

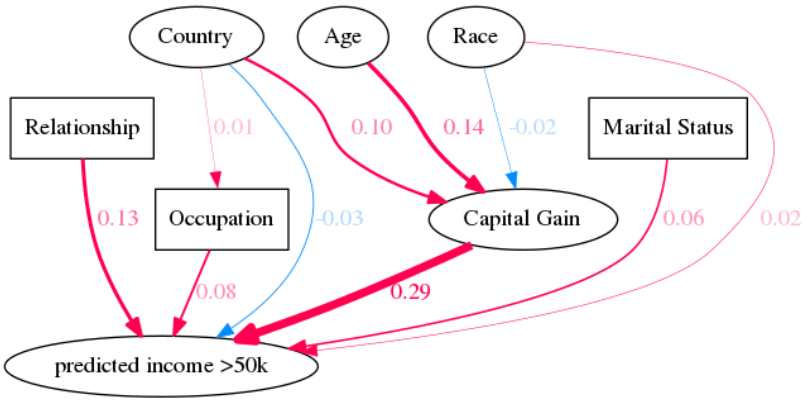


(A) Shapley Flow

FIGURE A.6: Comparison between independent SHAP, on-manifold SHAP, ASV, and Shapley Flow on a sample from the income dataset. Shapley flow shows the top 10 links. The direct impact of capital gain is not represented by on-manifold SHAP. As noted in the text this graph is based on previous work and is not necessarily representative of the true underlying causal relationship.

	Background sample	foreground sample
Age	39	30
Workclass	State-gov	State-gov
Education-Num	13	13
Marital Status	Never-married	Married-civ-spouse
Occupation	Adm-clerical	Prof-specialty
Relationship	Not-in-family	Husband
Race	White	Asian-Pac-Islander
Sex	Male	Male
Capital Gain	2174	0
Capital Loss	0	0
Hours per week	40	40
Country	United-States	India

	Independent	On-manifold	ASV
Relationship	0.17	0.04	0.13
Capital Gain	0.22	0.01	0.07
Occupation	0.1	0.06	0.07
Marital Status	0.08	0.06	0.07
Country	-0.04	0.07	0.07
Age	-0.0	-0.02	0.13
Education-Num	0.0	0.12	0.0
Race	0.02	0.07	0.0
Workclass	0.0	0.06	0.0
Hours per week	0.0	0.03	0.0
Sex	0.0	0.03	0.0
Capital Loss	0.0	0.01	0.0

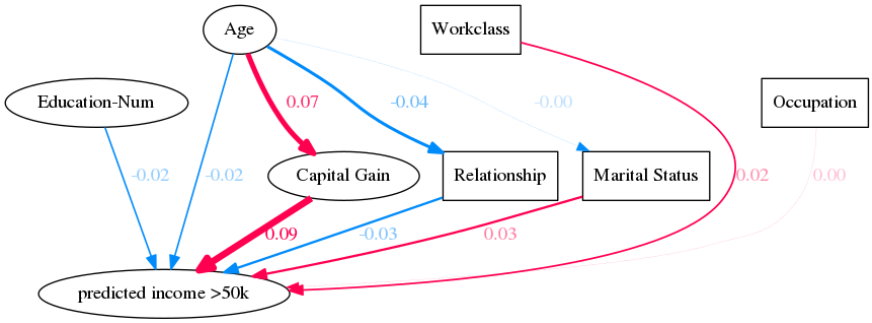


(A) Shapley Flow

FIGURE A.7: Comparison between independent SHAP, on-manifold SHAP, ASV, and Shapley Flow on a sample from the income dataset. Shapley flow shows the top 10 links. The indirect impact of age is only highlighted by Shapley Flow and ASV. As noted in the text this graph is based on previous work and is not necessarily representative of the true underlying causal relationship.

	Background sample	Foreground sample
Age	39	30
Workclass	State-gov	Federal-gov
Education-Num	13	10
Marital Status	Never-married	Married-civ-spouse
Occupation	Adm-clerical	Adm-clerical
Relationship	Not-in-family	Own-child
Race	White	White
Sex	Male	Male
Capital Gain	2174	0
Capital Loss	0	0
Hours per week	40	40
Country	United-States	United-States

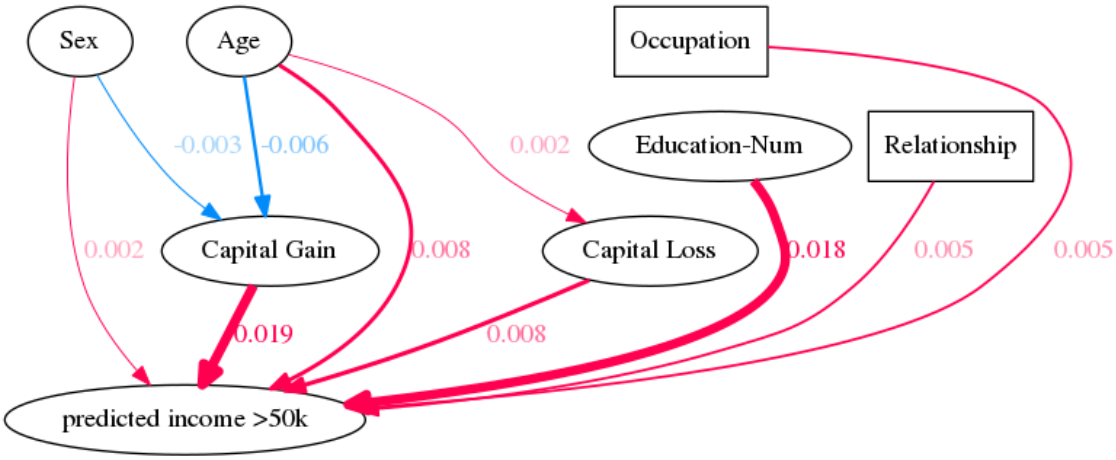
Attributions	Independent	On-manifold	ASV
Marital Status	0.03	0.08	0.03
Capital Gain	0.06	0.02	0.02
Workclass	0.03	0.03	0.02
Relationship	-0.01	-0.11	0.01
Education-Num	-0.02	0.01	-0.02
Age	-0.02	-0.03	0.01
Country	0.0	0.03	0.0
Capital Loss	0.0	0.03	0.0
Occupation	0.0	-0.03	0.0
Sex	0.0	0.03	0.0
Race	0.0	0.02	0.0
Hours per week	0.0	-0.0	0.0



(A) Shapley Flow

FIGURE A.8: Comparison between independent SHAP, on-manifold SHAP, ASV, and Shapley Flow on a sample from the income dataset. Shapley flow shows the top 10 links. Note that although age appears to be not important for all baselines, its impact through different causal edges are opposite as shown by Shapley Flow.

	Independent	On-manifold	ASV
Capital Gain	0.02	0.02	0.03
Education-Num	0.02	0.03	0.02
Age	0.01	0.01	0.01
Occupation	0.0	0.01	0.0
Capital Loss	0.01	-0.0	0.01
Relationship	0.01	0.0	0.0
Hours per week	0.0	0.01	-0.0
Sex	0.0	-0.01	0.0
Country	0.0	-0.01	0.0
Marital Status	-0.0	0.0	-0.0
Race	0.0	-0.01	-0.0
Workclass	0.0	-0.0	-0.0



(A) Shapley Flow

FIGURE A.9: Comparison of global understanding between independent SHAP, on-manifold SHAP, ASV, and Shapley Flow on the income dataset. Showing only the top 10 attributions for Shapley Flow for visual clarity.

structure, when we focus on systolic blood pressure and age, there are two cases: systolic blood pressure updates before or after age. We focus on the first case because it is where on-manifold SHAP differs from other baselines (all baselines already consider the second case as it satisfies the causal ordering). When systolic blood pressure updates before age, the expected age given systolic blood pressure is lower than the foreground age (yellow line below the black marker). Therefore when age updates to its foreground value, we observe a large increase in age, leading to a increase in the output (so age appears to be riskier). from both an in/direct impact perspective, on-manifold perturbation can be misleading since it is based not on causal but on observational relationships.

ASV ignores the direct impact of features. As shown in **Figure A.10**, ASV gives no credit systolic blood pressure because it is an intermediate node. However, it is clear from Shapley Flow that intervening on systolic blood pressure has a large impact on the outcome.

Shapley Flow shows both direct and indirect impacts of features. Focusing on the attribution given by Shapley Flow (**Figure A.10a**). We not only observe similar direct impacts in variables compared to independent SHAP, but also can trace those impacts to their source nodes, similar to ASV.

A.6 Considering all histories could lead to boundary inconsistency

In this section, we give an example of how considering all history $\mathcal{H}$ in the axioms (as opposed to $\tilde{\mathcal{H}}$) could lead to inconsistent attributions across boundaries. Consider two cuts for the same causal graph shown in **Figure A.12**. Note that both the green and the red cut share the edge "a". We have 8 possible message transmission histories ('c', 'b' can be transmitted only after 'd' has been transmitted):

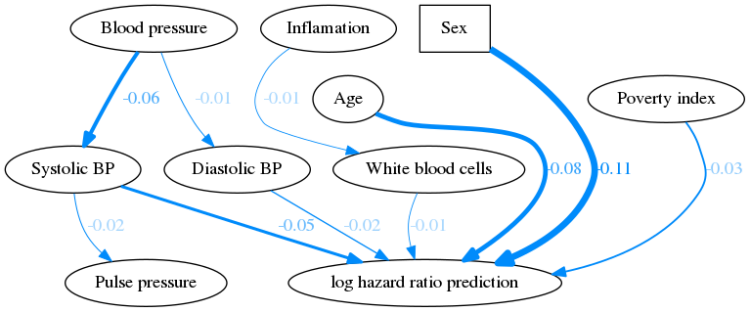
$$\{[a,d,c,b],[a,d,b,c],[d,a,c,b],[d,a,b,c],[d,c,a,b],[d,c,b,a],[d,b,a,c],[d,b,c,a]\}$$

. We use the same notation for carrier games (defined in **Section A.3**) and construct a game as the following:

$v_{red} = v_{red}^{dca} - v_{red}^{dcab} + v_{red}^{dba} - v_{red}^{dbac}$

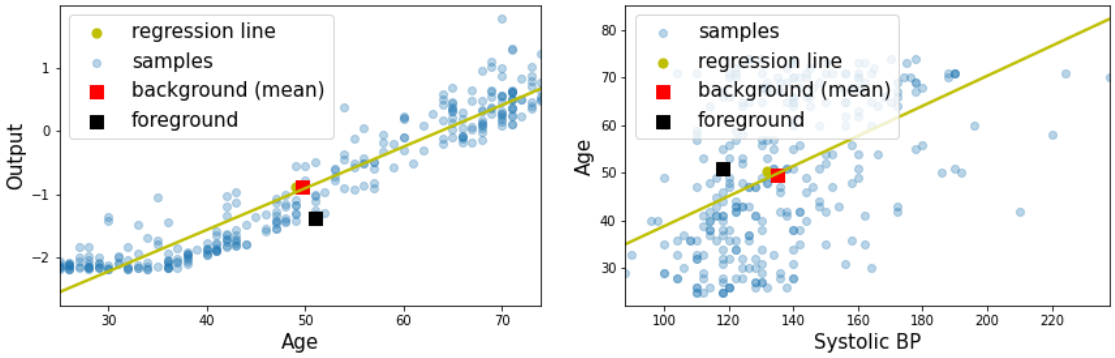
Top features	Sex	Age	Systolic BP
Background mean	NaN	50	135
Foreground sample	Female	51	118

Attributions	Independent	On-manifold	ASV
Sex	-0.11	-0.16	-0.1
Age	-0.07	0.23	-0.08
Systolic BP	-0.05	-0.22	0.0
Poverty index	-0.03	0.09	-0.02
Blood pressure	0.0	0.0	-0.08
TIBC	0.0	-0.16	0.0
Diastolic BP	-0.02	-0.08	0.0
Pulse pressure	-0.01	-0.11	0.0
Serum Iron	0.01	0.07	0.0
BMI	-0.0	-0.05	-0.0
White blood cells	-0.01	0.03	0.0
Serum Protein	-0.0	0.05	0.0
Serum Albumin	-0.0	-0.04	0.0
Inflamation	0.0	0.0	-0.02
Serum Cholesterol	-0.0	0.04	-0.0
Iron	0.0	0.0	0.02
Sedimentation rate	-0.01	-0.01	0.0
Race	-0.0	0.0	-0.01
TS	0.01	0.01	0.0
Serum Magnesium	-0.0	-0.01	-0.0
Blood protein	0.0	0.0	-0.01
Red blood cells	-0.0	0.01	-0.0



(A) Shapley Flow

FIGURE A.10: Comparison among methods on 100 background samples from the nutrition dataset, showing top 10 features/edges.



(A) Age vs. output

(B) Age vs. systolic blood pressure

FIGURE A.11: Age appears to be highly risky in on-manifold SHAP because it steals credit from other variables.

(A) Red cut

(B) Green cut

FIGURE A.12: Two cuts that represent two boundaries for the same causal graph.

Because of the linearity axiom, we have

$$\phi_{v_{red}}(a) > 0, \phi_{v_{red}}(b) < 0, \phi_{v_{red}}(c) < 0, \phi_{v_{red}}(d) = 0$$

.

However, when we consider the green boundary, the ordering *dcab* and *dbac* does not exist because in the green boundary A and Y are assumed to be a black-box. Therefore, $v_{green} = \mathbf{0}$, which means a is now a dummy edge: $\phi_{v_{green}}(a) = 0 \neq \phi_{v_{red}}(a)$. This demonstrate that we cannot consider all histories in $\mathcal{H}$ and being boundary consistent.

Appendix B

Credible Model Appendix

This Appendix includes additional results on Physionet 2012 challenge dataset and proofs for properties in **Section 4.3.4**. We assume $\lambda > 0$ because otherwise the model is not regularized.

B.1 Physionet Results with stage-wise feature selection

Besides regularization, another way to learn a credible model is to preprocess the input to exclude non-expert identified features that are highly correlated with expert identified features. By doing so, we preserve non-expert identified features that are useful for prediction. However, this two stage approach (*e.g.*, feature selection followed by l_2 regularized logistic regression) requires setting a correlation threshold to eliminate features (*i.e.*, non-expert identified features are eliminated if they have a correlation higher than the threshold with any of the expert identified features). If the threshold is set too high, we risk keeping irrelevant non-expert identified features. If this threshold is set too low, we risk throwing away relevant non-expert identified features. We therefore explore using different correlation thresholds. Results for this two stage approach applied to the Physionet 2012 challenge dataset are shown in **Table B.1**. We include the expert feature only baseline and the EYE regularization results from before to compare to this two staged approach. The correlation threshold holds are 0.4, 0.6, and 0.8 respectively. As expected, as the correlation threshold decreases, the alignment with experts (AP) increases because more non-expert features are thrown away. However, the accuracy decreases because more relevant non-expert identified features are thrown away. The two stage approach is less accurate compared to regularization approaches such as EYE because it ignores the

TABLE B.1: Stage-wise feature selection is inaccurate because it ignores the conditional distribution of target given input. It only models correlation in the input

Method	AP	AUC
expert-features-only	1*	0.754
EYE	**0.671**	0.815
Two-stage-0.4	0.541	0.760
Two-stage-0.6	0.365	0.782
Two-stage-0.8	0.260	0.789

conditional distribution of target given input and only models correlation in the input. For example, two features that have the same correlation with an expert identified feature could have different correlation with the target (one more predictive than the expert feature and one similarly predictive as the expert feature), yet they are treated equally in the feature selection stage. This means we cannot throw away one feature but not the other. If we throw away both, the model can be less accurate. Otherwise, keeping both can decrease alignment with experts.

B.2 Derivation of original EYE penalty

First note that $\{\boldsymbol{x} \mid q(\boldsymbol{x}) = c\}$ is the convex contour plot of q for $c \in \mathbb{R}$. We set c so that the slope in the first quadrant between known important factor and unknown feature is -1.

Since we only care about the interaction between known and unknown risk factors and that the contour is symmetric about the origin, without loss of generality, let y be the feature of unknown importance and x be the known important factor and $y \geq 0$, $x \geq 0$.

$$
\begin{aligned}
& 2\beta y + (1-\beta)x^2 = c \\
\Rightarrow \quad & y = \frac{c}{2\beta} - \frac{(1-\beta)x^2}{2\beta} \\
\Rightarrow \quad & y = 0 \Rightarrow x = \sqrt{\frac{c}{1-\beta}} \\
\Rightarrow \quad & f'(x) = -\frac{(1-\beta)}{\beta}x \\
\Rightarrow \quad & f'(\sqrt{\frac{c}{1-\beta}}) = -\frac{1-\beta}{\beta}\sqrt{\frac{c}{(1-\beta)}} = -1 \\
\Rightarrow \quad & c = \frac{\beta^2}{1-\beta} \\
\Rightarrow \quad & 2\beta y + (1-\beta)x^2 = \frac{\beta^2}{1-\beta}
\end{aligned} \tag{B.1}
$$

Thus, we just need $q(\boldsymbol{x}) = \frac{\beta^2}{1-\beta}$. The rest deals with scaling of the level curve. We define EYE penalty as a an atomic norm $\|\cdot\|_A$ introduced in [144]:

$$\|\boldsymbol{x}\|_A := \inf\{t > 0 \mid \boldsymbol{x} \in t\mathit{conv}(A)\}$$

where *conv* is the convex hull operator of its argument set A.

Let $A = \left\{\boldsymbol{x} \mid q(\boldsymbol{x}) \leq \frac{\beta^2}{1-\beta}\right\}$. Using the fact that the sublevel set of q is convex, we have

$$\mathit{eye}(\boldsymbol{x}) = \inf\left\{t > 0 \mid \boldsymbol{x} \in \left\{t\boldsymbol{x} \mid q(\boldsymbol{x}) \leq \frac{\beta^2}{1-\beta}\right\}\right\} \tag{B.2}$$

B.3 EYE has no extra parameter

We show that β conserves the shape of the contour and only controls the size of the contour, which is redudent given λ. Therefore, we can remove β from EYE's formulation.

Proof. Consider the contour $B_1 = \left\{\boldsymbol{x} : \mathit{eye}_{\beta_1}(\boldsymbol{x}) = t\right\}$ and $B_2 = \left\{\boldsymbol{x} : \mathit{eye}_{\beta_2}(\boldsymbol{x}) = t\right\}$, we want to show B_1 is similar to B_2. To do that, let's consider cases in which $t = 0$ and $t \neq 0$.

When $t = 0$, since EYE is a norm, $B_1 = B_2 = \{0\}$. Therefore they are trivially similar to each other. When $t \neq 0$, we can write B_1 as $t\left\{\boldsymbol{x} : \boldsymbol{x} \in \left\{\boldsymbol{x} \mid q_{\beta_1}(\boldsymbol{x}) = \frac{\beta_1^2}{1-\beta_1}\right\}\right\}$, and B_2 as $t\left\{\boldsymbol{x} : \boldsymbol{x} \in \left\{\boldsymbol{x} \mid q_{\beta_2}(\boldsymbol{x}) = \frac{\beta_2^2}{1-\beta_2}\right\}\right\}$. We further drop t as it doesn't affect similarity. Let $B_1' = \left\{\boldsymbol{x} : \boldsymbol{x} \in \left\{\boldsymbol{x} \mid q_{\beta_1}(\boldsymbol{x}) = \frac{\beta_1^2}{1-\beta_1}\right\}\right\}$ and $B_2' = \left\{\boldsymbol{x} : \boldsymbol{x} \in \left\{\boldsymbol{x} \mid q_{\beta_2}(\boldsymbol{x}) = \frac{\beta_2^2}{1-\beta_2}\right\}\right\}$. To show that B_1 is similar to B_2, we just need to show that B_1' is similar to B_2'. In fact, we can show that $B_2' = \frac{\beta_2(1-\beta_1)}{\beta_1(1-\beta_2)} B_1'$ as follows. Take $\boldsymbol{x} \in B_1'$, then $q_{\beta_1}(\boldsymbol{x}) = 2\beta_1\|(\mathbf{1}-\boldsymbol{r}) \odot \boldsymbol{x}\|_1 + (1-\beta_1)\|\boldsymbol{r} \odot \boldsymbol{x}\|_2^2 = \frac{\beta_1^2}{1-\beta_1}$. let $\boldsymbol{x'} = \frac{\beta_2(1-\beta_1)}{\beta_1(1-\beta_2)}\boldsymbol{x}$, then

$$
\begin{aligned}
q_{\beta_2}(\boldsymbol{x'}) &= 2\beta_2\|(\mathbf{1}-\boldsymbol{r}) \odot \boldsymbol{x'}\|_1 + (1-\beta_2)\|\boldsymbol{r} \odot \boldsymbol{x'}\|_2^2 \\
&= \frac{2\beta_2^2(1-\beta_1)}{\beta_1(1-\beta_2)}\|(\mathbf{1}-\boldsymbol{r}) \odot \boldsymbol{x}\|_1 + \frac{\beta_2^2(1-\beta_1)^2}{\beta_1^2(1-\beta_2)}\|\boldsymbol{r} \odot \boldsymbol{x}\|_2^2 \\
&= \frac{\beta_2^2(1-\beta_1)}{\beta_1^2(1-\beta_2)}(2\beta_1\|(\mathbf{1}-\boldsymbol{r}) \odot \boldsymbol{x}\|_1 + (1-\beta_1)\|\boldsymbol{r} \odot \boldsymbol{x}\|_2^2) \\
&= \frac{\beta_2^2(1-\beta_1)}{\beta_1^2(1-\beta_2)}\frac{\beta_1^2}{1-\beta_1} \\
&= \frac{\beta_2^2}{1-\beta_2}
\end{aligned}
$$

This shows that $\boldsymbol{x'} \in B_2'$, that is $\frac{\beta_2(1-\beta_1)}{\beta_1(1-\beta_2)} B_1' \subset B_2'$. The other direction can be similarly proven.

□

B.4 Equivalence with the triangular form of EYE penalty

In this section, we prove **Equation (4.2)** and **(4.3)** are equivalent.

Proof. Since β can be arbitrarily set (B.3), fix β=0.5, then **Equation (4.2)** becomes

$$
eye(\boldsymbol{x}) = \inf\left\{t > 0 \mid \boldsymbol{x} \in t\left\{\boldsymbol{x} \mid 2\|(\mathbf{1}-\boldsymbol{r}) \odot \boldsymbol{x}\|_1 + \|\boldsymbol{r} \odot \boldsymbol{x}\|_2^2 = 1\right\}\right\} \tag{B.3}
$$

Assume $\boldsymbol{x} \neq 0$ and denote $eye(\boldsymbol{x}) := t$, then $\boldsymbol{x} \in t\left\{\boldsymbol{x} \mid 2\|(\mathbf{1}-\boldsymbol{r}) \odot \boldsymbol{x}\|_1 + \|\boldsymbol{r} \odot \boldsymbol{x}\|_2^2 = 1\right\}$, that is $\frac{2\|(\mathbf{1}-\boldsymbol{r})\odot \boldsymbol{x}\|_1}{t} + \frac{\|\boldsymbol{r}\odot \boldsymbol{x}\|_2^2}{t^2} = 1$. As this is a quadratic equation in t and from assumption we know $t > 0$ (EYE being a norm and $\boldsymbol{x} \neq 0$), solving for t yields:

$$t = \|(\mathbf{1}-\boldsymbol{r}) \odot \boldsymbol{x}\|_1 + \sqrt{\|(\mathbf{1}-\boldsymbol{r}) \odot \boldsymbol{x}\|_1^2 + \|\boldsymbol{r} \odot \boldsymbol{x}\|_2^2} \tag{B.4}$$

Note that in the event $\boldsymbol{x} = 0$, $t = 0$, **Equation (B.4)** agrees with the fact that $eye(\mathbf{0}) = 0$. Thus **Equation (4.3)** and **(4.2)** are equivalent. □

B.5 Sparsity with Orthonormal Design Matrix

We consider a special case of regression and orthogonal design matrix ($X^\top X = I$) with EYE regularization. This restriction allows us to obtain a closed form solution so that key features of EYE penalty can be highlighted. With **Equation (4.3)**, we have

$$\min_{\boldsymbol{\theta}} \frac{1}{2}\|\boldsymbol{y} - X\boldsymbol{\theta}\|_2^2 + n\lambda\left(\|(\mathbf{1}-\boldsymbol{r}) \odot \boldsymbol{\theta}\|_1 + \sqrt{\|(\mathbf{1}-\boldsymbol{r}) \odot \boldsymbol{\theta}\|_1^2 + \|\boldsymbol{r} \odot \boldsymbol{\theta}\|_2^2}\right) \tag{B.5}$$

Since the objective is convex, we solve for its subgradient $\boldsymbol{g}$.

$$\boldsymbol{g} = X^\top X\boldsymbol{\theta} - X^\top \boldsymbol{y} + n\lambda(\mathbf{1}-\boldsymbol{r}) \odot \boldsymbol{s} + \frac{n\lambda}{Z}(\|(\mathbf{1}-\boldsymbol{r}) \odot \boldsymbol{\theta}\|_1(\mathbf{1}-\boldsymbol{r}) \odot \boldsymbol{s} + \boldsymbol{r} \odot \boldsymbol{r} \odot \boldsymbol{\theta}) \tag{B.6}$$

where $s_i = sgn(\theta_i)$ if $\theta_i \neq 0$, $s_i \in [-1, 1]$ if $\theta_i = 0$, and $Z = \sqrt{\|(\mathbf{1}-\boldsymbol{r}) \odot \boldsymbol{\theta}\|_1^2 + \|\boldsymbol{r} \odot \boldsymbol{\theta}\|_2^2}$.

By our assumption $X^\top X = I$, and the fact that $\hat{\boldsymbol{\theta}}^{OLS} = (X^\top X)^{-1}X^\top \boldsymbol{y} = X^\top \boldsymbol{y}$ (the solution for oridinary least squares), we simplify (B.6) as

$$\boldsymbol{g} = \boldsymbol{\theta} - \hat{\boldsymbol{\theta}}^{OLS} + n\lambda(\mathbf{1}-\boldsymbol{r}) \odot \boldsymbol{s} + \frac{n\lambda}{Z}(\|(\mathbf{1}-\boldsymbol{r}) \odot \boldsymbol{\theta}\|_1(\mathbf{1}-\boldsymbol{r}) \odot \boldsymbol{s} + \boldsymbol{r} \odot \boldsymbol{r} \odot \boldsymbol{\theta}) \tag{B.7}$$

Setting $\boldsymbol{g}$ to $\mathbf{0}$ we have

$$\hat{\theta}_i = \frac{\hat{\theta}_i^{OLS}}{1 + \frac{n\lambda}{Z} r_i^2} \max\left(0, 1 - \frac{n\lambda(1 - r_i)\left(1 + \frac{\|(\mathbf{1}-\boldsymbol{r})\odot\hat{\boldsymbol{\theta}}\|_1}{Z}\right)}{|\hat{\theta}_i^{OLS}|}\right) \tag{B.8}$$

where $Z = \sqrt{\|(\mathbf{1} - \boldsymbol{r}) \odot \hat{\boldsymbol{\theta}}\|_1^2 + \|\boldsymbol{r} \odot \hat{\boldsymbol{\theta}}\|_2^2}$.

Note that **Equation (B.8)** is still an implicit equation in $\boldsymbol{\theta}$ because Z is a function of $\hat{\boldsymbol{\theta}}$. Also, we implicitly assumed that $Z \neq 0$.

Although this is an implicit equation for θ_i, the max term confirms EYE's ability to set weights to exactly zero in the orthonormal design matrix setting.

What if $Z = 0$? This only happens if $\boldsymbol{\theta} = \mathbf{0}$. However, by the complementary slackness condition in KKT, we know $\lambda > 0$ implies that the solution is on the boundary of the constraint formulation of the problem (for $\lambda = 0$, we are back to ordinary least squares). So long as the optimal solution for the unconstrained problem is not at $\mathbf{0}$, we won't get into trouble unless the constraint is $eye(\boldsymbol{\theta}) \leq 0$, which won't happen in the regression setting as λ is finite. If the optimal solution for the unconstrained problem is $\mathbf{0}$, we are again back to ordinary least squares solutions. So the upshot is we can assume $Z \neq 0$ otherwise it will automatically revert to ordinary least squares.

B.6 Perfect Correlation

Denote the objective function in **Equation (B.5)** as $L(\boldsymbol{\theta})$ and denote $\hat{\boldsymbol{\theta}}$ as the optimal solution, we show that

- $r_i = 1$, $r_j = 0$, $x_i = x_j \implies \hat{\theta}_j = 0$ (EYE penalty prefers known risk factors over unknown risk factors).

  Proof. Assume $r_i = 1$, $r_j = 0$, and consider $\hat{\boldsymbol{\theta}}'$ that only differs from $\hat{\boldsymbol{\theta}}$ at the i^{th} and j^{th} entry such that $\hat{\theta}_i' = \hat{\theta}_i + \hat{\theta}_j$ and $\hat{\theta}_j' = 0$, we have $L(\hat{\boldsymbol{\theta}}) - L(\hat{\boldsymbol{\theta}}') = \frac{1}{2}\|\boldsymbol{y} - X\hat{\boldsymbol{\theta}}\|_2^2 + n\lambda\left(|\hat{\theta}_j| + \sqrt{(C + |\hat{\theta}_j|)^2 + D + \hat{\theta}_i^2}\right) - \frac{1}{2}\|\boldsymbol{y} - X\hat{\boldsymbol{\theta}}'\|_2^2 - n\lambda\left(|\hat{\theta}_j'| + \sqrt{(C + |\hat{\theta}_j'|)^2 + D + \hat{\theta}_i'^2}\right)$ where C and D are non-negative constant involving entries other than i and j. Note that the sum of squared residual is the same for both $\hat{\boldsymbol{\theta}}'$ and $\hat{\boldsymbol{\theta}}$ owing to the fact that $x_i = x_j$. Use the definition of $\hat{\boldsymbol{\theta}}'$, we have

$$L(\hat{\boldsymbol{\theta}}) - L(\hat{\boldsymbol{\theta}}') = n\lambda \left(|\hat{\theta}_j| + \sqrt{(C + |\hat{\theta}_j|)^2 + D + \hat{\theta}_i^2} - \sqrt{C^2 + D + (\hat{\theta}_i + \hat{\theta}_j)^2} \right)$$

Claim $L(\hat{\boldsymbol{\theta}}) - L(\hat{\boldsymbol{\theta}}') \geq 0$ with equality only if $\hat{\theta}_j = 0$

Proof. Since $n\lambda$ is positive, the claim is equivalent to

$$\sqrt{(C + |\hat{\theta}_j|)^2 + D + \hat{\theta}_i^2} \geq \sqrt{C^2 + D + (\hat{\theta}_i + \hat{\theta}_j)^2} - |\hat{\theta}_j|$$

If the right hand side is negative, we are done since the left hand side is non-negative. Otherwise, both sides are non-negative. We square them and rearrange to get the equivalent form

$$\hat{\theta}_j^2 + 2\hat{\theta}_i\hat{\theta}_j \leq 2|\hat{\theta}_j|\sqrt{C^2 + D + (\hat{\theta}_i + \hat{\theta}_j)^2} + 2C|\hat{\theta}_j|$$

which is true following

$$\begin{aligned}
\hat{\theta}_j^2 + 2\hat{\theta}_i\hat{\theta}_j &\leq 2\hat{\theta}_j^2 + 2\hat{\theta}_i\hat{\theta}_j - \hat{\theta}_j^2 && \text{(B.9)} \\
&\leq 2|\hat{\theta}_j||\hat{\theta}_i + \hat{\theta}_j| && \text{(B.10)} \\
&= 2|\hat{\theta}_j|\sqrt{(\hat{\theta}_i + \hat{\theta}_j)^2} && \text{(B.11)} \\
&\leq 2|\hat{\theta}_j|\sqrt{C^2 + D + (\hat{\theta}_i + \hat{\theta}_j)^2} + 2C|\hat{\theta}_j| && \text{(B.12)}
\end{aligned}$$

Again if $\hat{\theta}_j \neq 0$, the inequality is strict from **Equation (B.9)** to **Equation (B.10)**

□

Since we assumed that $\hat{\boldsymbol{\theta}}$ is optimal, the equality in B.6 must hold, thus $\hat{\theta}_j = 0$.

□

- $r_i = 1, r_j = 1, x_i = x_j \implies \hat{\theta}_i = \hat{\theta}_j$ (feature weights are dense in known risk factors).

Proof. Assume $\hat{\boldsymbol{\theta}}$ is optimal, consider $\hat{\boldsymbol{\theta}}'$ that is the same as $\hat{\boldsymbol{\theta}}$ except $\hat{\theta}'_i = \hat{\theta}'_j = \frac{\hat{\theta}_j + \hat{\theta}_j}{2}$. Assume $\hat{\boldsymbol{\theta}} \neq \hat{\boldsymbol{\theta}}'$: $\hat{\theta}_i \neq \hat{\theta}_j$. As in the last proof, the data loss (sum of squared residual) is the same in both solutions because $x_i = x_j$. As a result we have $L(\hat{\boldsymbol{\theta}}) - L(\hat{\boldsymbol{\theta}}') = n\lambda \left(\sqrt{\left(C + |\hat{\theta}_i| + |\hat{\theta}_j|\right)^2 + D + \hat{\theta}_i^2 + \hat{\theta}_j^2} - \sqrt{\left(C + 2\frac{|\hat{\theta}_i + \hat{\theta}_j|}{2}\right)^2 + D + 2\frac{|\hat{\theta}_i + \hat{\theta}_j|^2}{4}} \right)$, which is greater or equal to $n\lambda \left(\sqrt{\left(C + |\hat{\theta}_i| + |\hat{\theta}_j|\right)^2 + D + \hat{\theta}_i^2 + \hat{\theta}_j^2} - \sqrt{\left(C + |\hat{\theta}_i| + |\hat{\theta}_j|\right)^2 + D + \frac{|\hat{\theta}_i + \hat{\theta}_j|^2}{2}} \right)$. Again, C and D are non-negative constant involving entries other than i and j.

Since

$$\hat{\theta}_i^2 + \theta_j^2 - \frac{|\hat{\theta}_i + \hat{\theta}_j|^2}{2} = \frac{(\hat{\theta}_i - \hat{\theta}_j)^2}{2} > 0$$

by assumption that $\hat{\theta}_i \neq \hat{\theta}_j$ for the optimal solution. This shows $L(\hat{\boldsymbol{\theta}}) - L(\hat{\boldsymbol{\theta}}') > 0$, which contradict our assumption. Thus $\hat{\theta}_i = \hat{\theta}_j$ for the optimal solution. □

- $r_i = 0, r_j = 0, x_i = x_j \implies$ back to LASSO continuum

  Note that fixing $\theta_k \; \forall k \notin \{i, j\}$, solving for θ_i and θ_j reduces the problem to LASSO, thus all properties of LASSO carry over for θ_i and θ_j. Thus sparsity is maintained in unknown features.

B.7 General Correlation

Grouping effect in elastic net is still present in eye penalty within groups with similar level of risk.

Theorem B.7.1. *if $\hat{\theta}_i \hat{\theta}_j > 0$ and design matrix is standardized, then*

$$\frac{|r_i^2 \hat{\theta}_i - r_j^2 \hat{\theta}_j|}{Z} \leq \frac{\sqrt{2(1-\rho)}\|\boldsymbol{y}\|_2}{n\lambda} + |r_i - r_j| \left(1 + \frac{\|(\mathbf{1} - \boldsymbol{r}) \odot \hat{\theta}\|_1}{Z}\right)$$

where $Z = \sqrt{\|(\mathbf{1} - \boldsymbol{r}) \odot \hat{\boldsymbol{\theta}}\|_1^2 + \|\boldsymbol{r} \odot \hat{\boldsymbol{\theta}}\|_2^2}$, ρ is the sample covariance between x_i and x_j.

Proof. Denote the objective in **Equation (B.5)** as L. Assume $\hat{\theta}_i\hat{\theta}_j > 0$, $\hat{\boldsymbol{\theta}}$ is the optimal weights, and the design matrix X is standardized to have zero mean and unit variance in its column. Via the optimal condition and (B.6), subgradient $\boldsymbol{g}$ at $\hat{\boldsymbol{\theta}}$ is 0. Hence we have

$$-x_i^\top(\boldsymbol{y} - X\hat{\boldsymbol{\theta}}) + n\lambda((1-r_i)s_i + \frac{\|(\mathbf{1}-\boldsymbol{r})\odot\hat{\boldsymbol{\theta}}\|_1}{Z}((1-r_i)s_i + r_i^2\hat{\theta}_i)) = 0 \tag{B.13}$$

$$-x_j^\top(\boldsymbol{y} - X\hat{\boldsymbol{\theta}}) + n\lambda((1-r_j)s_j + \frac{\|(\mathbf{1}-\boldsymbol{r})\odot\hat{\boldsymbol{\theta}}\|_1}{Z}((1-r_j)s_j + r_j^2\hat{\theta}_j)) = 0 \tag{B.14}$$

The assumption that $\hat{\theta}_i\hat{\theta}_j > 0$ implies $sgn(\hat{\theta}_i) = sgn(\hat{\theta}_j)$ and eliminates the need to discuss the subgradient issue. Subtract B.14 from B.13, we have $(x_j^\top - x_i^\top)(\boldsymbol{y} - X\hat{\boldsymbol{\theta}}) + n\lambda((r_j - r_i)sgn(\hat{\theta}_i) + \frac{\|(\mathbf{1}-\boldsymbol{r})\odot\hat{\boldsymbol{\theta}}\|_1}{Z}((r_j - r_i)sgn(\hat{\theta}_i) + r_i^2\hat{\theta}_i - r_j^2\hat{\theta}_j)) = 0$. We can further rearrange the equation to get

$$\frac{r_i^2\hat{\theta}_i - r_j^2\hat{\theta}_j}{Z} = \frac{(x_i^\top - x_j^\top)(\boldsymbol{y} - X\hat{\theta})}{n\lambda} + (r_i - r_j)sgn(\hat{\theta}_i)\left(1 + \frac{\|(\mathbf{1}-\boldsymbol{r})\odot\hat{\boldsymbol{\theta}}\|_1}{Z}\right) \tag{B.15}$$

Being the optimal weights, $L(\hat{\boldsymbol{\theta}}) \leq L(\mathbf{0})$, which implies $\|y - X\hat{\boldsymbol{\theta}}\|_2^2 \leq \|\boldsymbol{y}\|_2^2$. Moreover, the standardized design matrix gives $\|x_i - x_j\|_2^2 = \langle x_i, x_i\rangle + \langle x_j, x_j\rangle - 2\langle x_i, x_j\rangle = 2(1-\rho)$. Taking the absolute value of **Equation (B.15)** and applying Cauchy Schwarz inequality, we get

$$\frac{|r_i^2\hat{\theta}_i - r_j^2\hat{\theta}_j|}{Z} \leq \frac{\|x_i - x_j\|_2\|\boldsymbol{y} - X\hat{\boldsymbol{\theta}}\|_2}{n\lambda} + |r_i - r_j|\left(1 + \frac{\|(\mathbf{1}-\boldsymbol{r})\odot\hat{\boldsymbol{\theta}}\|_1}{Z}\right) \tag{B.16}$$

which is less or equal to

$$\frac{\sqrt{2(1-\rho)}\|\boldsymbol{y}\|_2}{n\lambda} + |r_i - r_j|\left(1 + \frac{\|(\mathbf{1}-\boldsymbol{r})\odot\hat{\boldsymbol{\theta}}\|_1}{Z}\right) \tag{B.17}$$

□

Corollary B.7.2. *If $\hat{\theta}_i\hat{\theta}_j > 0$, design matrix is standardized, and $r_i = r_j \neq 0$*

$$\frac{|\hat{\theta}_i - \hat{\theta}_j|}{Z} \leq \frac{\sqrt{2(1-\rho)}\|\boldsymbol{y}\|_2}{r_i^2 n\lambda}$$

where $Z = \sqrt{\|(\mathbf{1} - \boldsymbol{r}) \odot \hat{\boldsymbol{\theta}}\|_1^2 + \|\boldsymbol{r} \odot \hat{\boldsymbol{\theta}}\|_2^2}$, ρ is the sample covariance between x_i and x_j.

This verifies the existence of the grouping effect: highly correlated features (with similar risk) are grouped together in the parameter space.

Appendix C

Concept Credible Model Appendix

C.1 Derivation of the least squared solution for Section 5.3.1

Given $X = [C, S, U]$, $Y = C + U$, $corr(C, U) \neq 1$, and $C = S$ in $\mathcal{D}$, a least squares linear regression solution gives a prediction of $\hat{Y} = (1 - t)C + U + tS$.

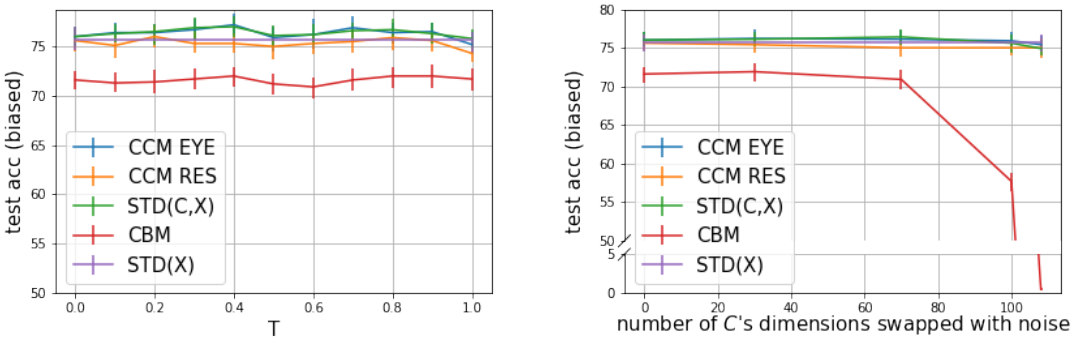
Proof. We know $C + U$ is a solution because they give 0 loss. Since $C = S$ in the dataset, $(1 - t)C + tS + U$ is also a solution for any $t \in \mathbb{R}$. Since U and C are not co-linear, the solution has rank 2. By the rank nullity theorem, we know the null space has dimension 1 (because the dimension of input is 3), thus its least squares solutions also has dimension of 1, which shows that $(1 - t)C + tS + U$ are all the solutions that minimizes the loss. □

The minimum L_2 norm solution of this problem results in $t = 0.5$

Proof. Given the solution is $(1 - t)C + tS + U$, we minimize the coefficient with L_2 loss: $\arg\min_t (1 - t)^2 + t^2 + 1$, which solves to $t = 0.5$. □

If we only use C for prediction (*i.e.*, CBM), the solution will not achieve a loss of 0 since it ignores U.

Proof. Even with infinite training data, fitting Y using C results in $\mathbb{E}(Y|C) = C + \mathbb{E}(U|C)$. The L_2 loss with Y is thus $\mathbb{E}((U - \mathbb{E}(U|C))^2)$, which is non-zero when C and U are not co-linear. □



(A) Test Accuracy (biased) violating **A1**

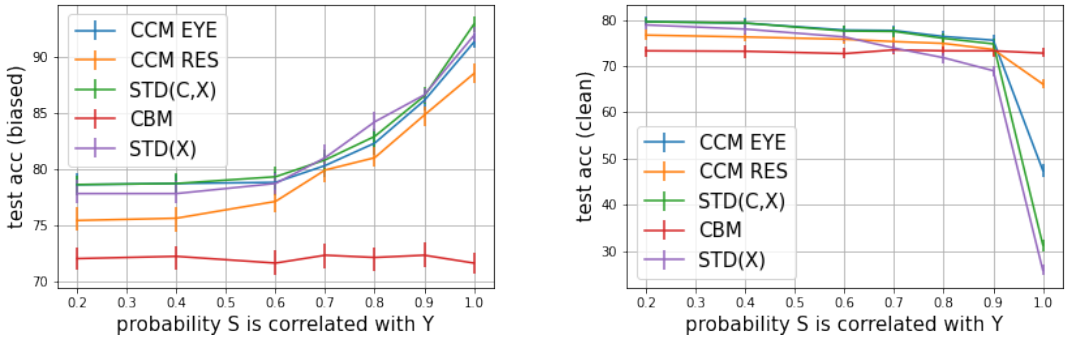
(B) Test Accuracy (biased) violating **A2**

FIGURE C.1: **(a)** When **A1** is broken by adding bias to how C is trained, the biased dataset performances are constant across methods. Note that except for CBM, all methods performed about the same. **(b)** When **A2** is broken by replacing dimensions of C with random noise, the predictive power of CBM decreases, yet other methods have similar performance on the biased dataset because they can learn from X in addition to C.

C.2 Additional CUB results

Test accuracy of CUB experiments on the biased dataset? In the main text, we showed that CCM methods perform well when shortcuts are violated. Here we present the result of methods on the biased dataset when **A1** and **A2** are broken in **Figure C.1a** and **Figure C.1b** respectively. Overall, CBM approaches perform worse on the biased dataset because it lacks the ability to learn U, while all other approaches perform similarly.

What if S carries information outside of $C \cup U$? The second way to break **A2** is to directly correlate S with Y on line 4 of the BIAS function. We sweep T to control the correlation between S and Y. As shown in **Figure C.2**, as T increases, the performance on the biased test set increase as well, but not the clean dataset performance, confirming that S contains information outside of C and U. We also observe that CCM RES performs worse than the standard model on both the biased and portions of the clean dataset. In contrast, CCM EYE is consistently better than $STD(X)$ on the clean dataset and comparable to it on the biased dataset. Moreover, $STD(C, X)$ is comparable to CCM EYE until the shortcut is



(A) Test Acc (biased)

(B) Test Acc (clean)

FIGURE C.2: Results of relaxing **A2** by making S more informative. Here, instead of generating S from CBM, we correlate S with Y directly and sweep the value of T. This experiment demonstrates what happens when S contains information beyond C and U.

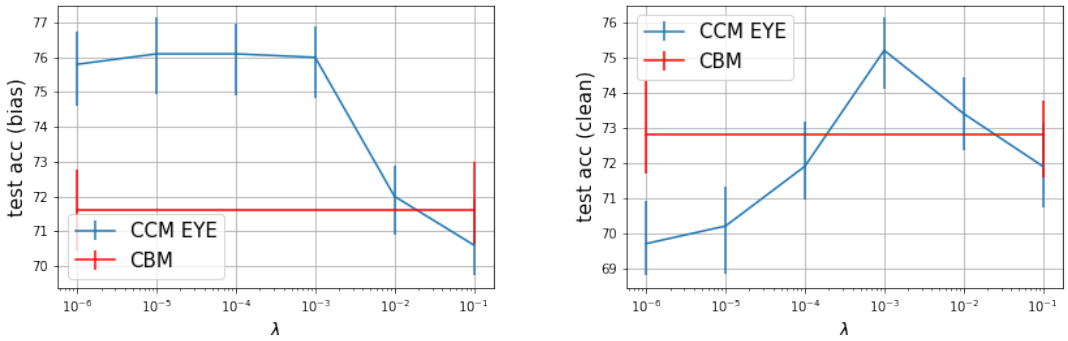
the strongest (*i.e.*, $T = 1$). This makes sense because when shortcuts are weak compared to C, they are not taken by $STD(C, X)$.

How does λ affect model performance? Since CCM EYE has an additional parameter λ, we want to understand its effect on model performance. **Figure C.3** summarizes the results on the CUB dataset. Fixing test accuracy on the biased dataset (*e.g.*, $\lambda \leq 10^{-4}$), increasing the EYE penalty monotonically increases model performance on the clean dataset. This means that credible model's principle (*i.e.*, increasing alignment with expert without sacrificing performance) could help mitigate the use of shortcut when **A1** and **A2** hold.

How does CCM responds to different levels of shortcut? Regardless of n_σ, CCM outperforms baselines (**Figure C.4**).

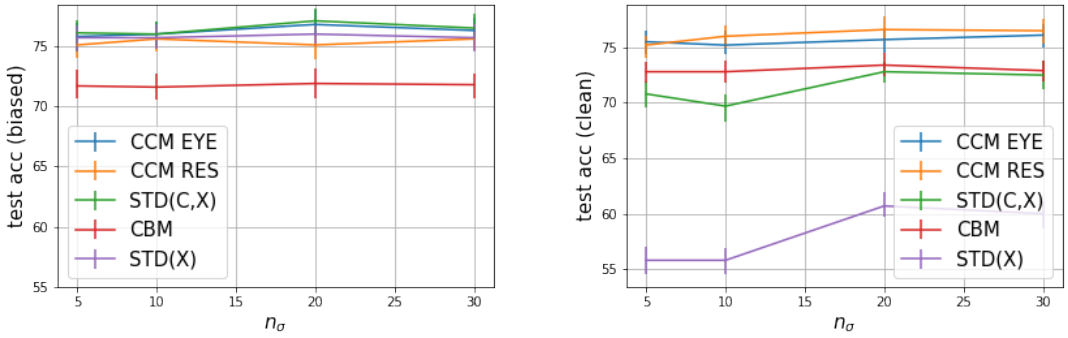
C.3 Additional MIMIC results

We include more results of the MIMIC dataset in **Figure C.3** and **Figure C.3**. Each plot varies the training distribution (noted by the black line) used to train the models and



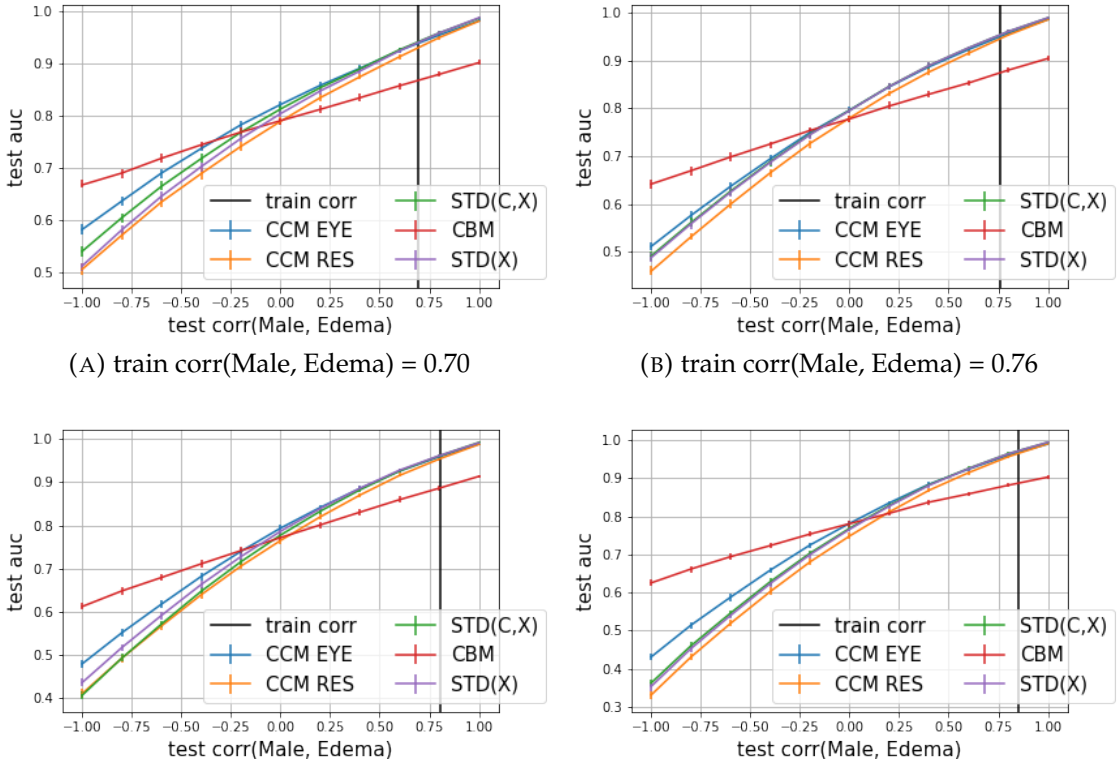
(A) Test Acc (biased) (B) Test Acc (clean)

FIGURE C.3: Results of sweeping λ. Without sacrificing test accuracy on the biased dataset ($\lambda \leq 10^{-4}$ in this case for the CUB dataset), increasing λ boosts performance on the clean test set, justifying our choice of hyperparameter for CCM EYE.



(A) Test Acc (biased) (B) Test Acc (clean)

FIGURE C.4: Results of sweeping number of noises (n_σ). Regardless of n_σ, CCM EYE and CCM RES outperform baselines on the clean dataset, while maintaining similar performance on the biased dataset.



(A) train corr(Male, Edema) = 0.70

(B) train corr(Male, Edema) = 0.76

(C) train corr(Male, Edema) = 0.81

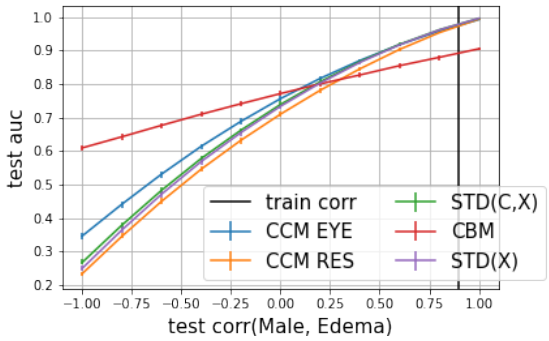
(D) train corr(Male, Edema) = 0.85

FIGURE C.5: Result of the MIMIC-CXR experiment for different training distributions. CCM EYE consistently outperforms baselines models when the training and testing distribution are close. It only performs worse against CBM when the testing distribution is very different from the training.

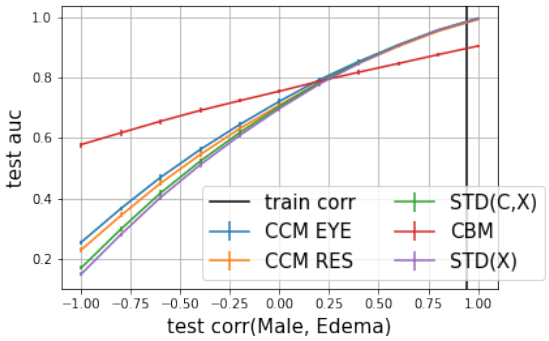
compares their results on different test distributions. Note that CCM EYE consistently outperforms baselines models when the training and testing distribution are close. It only performs worse against CBM when the testing distribution is very different from the training.

C.4 Experiments on the Physionet Challenge dataset

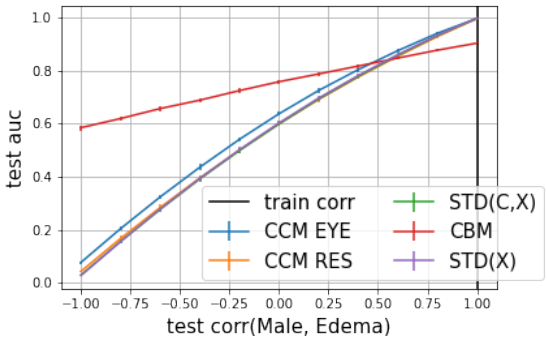
The **Physionet Challenge 2012** dataset [152] is a publicly available benchmark dataset from [151] in which one aims to predict in-hospital mortality using electronic health



(A) train corr(Male, Edema) = 0.90



(B) train corr(Male, Edema) = 0.95



(C) train corr(Male, Edema) = 1.00

FIGURE C.6: Result of the MIMIC-CXR experiment for different training distributions (correlations between male and edema are 0.9, 0.95, and 1 respectively). CCM EYE consistently outperforms baselines models when the training and testing distribution are close. It only performs worse against CBM when the testing distribution is very different from the training.

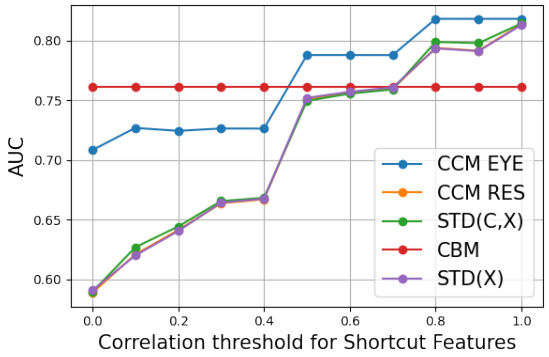


FIGURE C.7: Treating features correlated with C as shortcuts in the Physionet Challenge 2012 dataset, we measure the performance when shortcuts break (set to 0). As expected, when shortcuts are highly correlated with C, CCM EYE outperforms all baselines. Even when shortcuts are not highly correlated with C (violating **A2**), CCM EYE is only second to CBM. In contrast, CCM RES has trouble beating the baselines because U is correlated with S.

record collected in intensive care units for $4,000$ patients. Our preprocessing follows [13], obtaining a feature set of size 130. In addition to the features, we have 15 variables corresponding to the Simplified Acute Physiology Score (SAPS-I) that are developed by physicians to predict ICU mortality in patients greater than the age of 15 [153]. We use those features along with age as C. This mimics setting where the true concepts are learned based on medical knowledge.

Here, we define shortcut variables to be variables correlated with the 15 SAPS-I variables and age. In other words, S is composed of all non C features that have a correlation with features in C above a certain threshold. Other features are regarded as U as their value is not causally related to the shortcuts. This setup mimics the setting in which shortcuts are correlated with known risk factors, motivated in [13].

Model Training. Following [13], we train linear models on this dataset with the Adam optimizer [175]. We randomly reserve 25% of patients as the test set. Of the remaining data, we randomly split 25% for validation and the rest for training. We train baseline models as well as our models using the full set of features and duplicate features in C for $STD(C, X)$ to increase its chance to use C.

Evaluation. We test model performance by setting the value for shortcut variables to 0, making them uninformative at test time. If a model is robust to S, this change shouldn't affect its prediction accuracy. The biased dataset performance is reported when no shortcuts are "zeroed out". This happens with a correlation threshold of 1 as no features other than C have a perfect correlation with features in C.

Results. CCM EYE outperforms all other baselines in **Figure C.7**. CCM EYE does not use features that are highly correlated with C, thus eliminating the use of S. In contrast, CCM RES does not do as well even when C and S are highly correlated (**A2**). This happens because U is correlated with S (yet not causally related). For example, if we fix the correlation threshold at 0.8 for shortcuts, 57% of variables in U have at least 0.1 correlation with a variable in S.

Bibliography

[1] D. Silver, A. Huang, C. J. Maddison, *et al.*, "Mastering the game of go with deep neural networks and tree search," *nature*, vol. 529, no. 7587, pp. 484–489, 2016.

[2] J. Jumper, R. Evans, A. Pritzel, *et al.*, "High accuracy protein structure prediction using deep learning," *Fourteenth Critical Assessment of Techniques for Protein Structure Prediction (Abstract Book)*, vol. 22, p. 24, 2020.

[3] P. He, X. Liu, J. Gao, and W. Chen, "Deberta: Decoding-enhanced bert with disentangled attention," *arXiv preprint arXiv:2006.03654*, 2020.

[4] K. He, X. Zhang, S. Ren, and J. Sun, "Delving deep into rectifiers: Surpassing human-level performance on imagenet classification," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 1026–1034.

[5] A. Esteva, B. Kuprel, R. A. Novoa, *et al.*, "Dermatologist-level classification of skin cancer with deep neural networks," *Nature*, vol. 542, no. 7639, p. 115, 2017.

[6] A. D'Amour, K. Heller, D. Moldovan, *et al.*, "Underspecification presents challenges for credibility in modern machine learning," *arXiv preprint arXiv:2011.03395*, 2020.

[7] M. T. Ribeiro, T. Wu, C. Guestrin, and S. Singh, "Beyond accuracy: Behavioral testing of nlp models with checklist," *arXiv preprint arXiv:2005.04118*, 2020.

[8] A. Torralba and A. A. Efros, "Unbiased look at dataset bias," in *CVPR 2011*, IEEE, 2011, pp. 1521–1528.

[9] V. Veitch, A. D'Amour, S. Yadlowsky, and J. Eisenstein, "Counterfactual invariance to spurious correlations: Why and how to pass stress tests," *arXiv preprint arXiv:2106.00545*, 2021.

[10] F. Doshi-Velez and B. Kim, "Towards a rigorous science of interpretable machine learning," *arXiv preprint arXiv:1702.08608*, 2017.

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

[12] J. Wang, J. Wiens, and S. Lundberg, "Shapley flow: A graph-based approach to interpreting model predictions," in *International Conference on Artificial Intelligence and Statistics*, PMLR, 2021, pp. 721–729.

[13] J. Wang, J. Oh, H. Wang, and J. Wiens, "Learning credible models," in *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2018, pp. 2417–2426.

[14] S. M. Lundberg and S.-I. Lee, "A unified approach to interpreting model predictions," in *Advances in neural information processing systems*, 2017, pp. 4765–4774.

[15] M. Sundararajan, A. Taly, and Q. Yan, "Axiomatic attribution for deep networks," in *International Conference on Machine Learning*, PMLR, 2017, pp. 3319–3328.

[16] C. Frye, D. de Mijolla, L. Cowton, M. Stanley, and I. Feige, "Shapley-based explainability on the data manifold," *arXiv preprint arXiv:2006.01272*, 2020.

[17] K. Aas, M. Jullum, and A. Løland, "Explaining individual predictions when features are dependent: More accurate approximations to shapley values," *arXiv preprint arXiv:1903.10464*, 2019.

[18] L. S. Shapley, "A value for n-person games," *Contributions to the Theory of Games*, vol. 2, no. 28, pp. 307–317, 1953.

[19] P. W. Koh, T. Nguyen, Y. S. Tang, *et al.*, "Concept bottleneck models," in *International Conference on Machine Learning*, PMLR, 2020, pp. 5338–5348.

[20] J. Von Neumann and O. Morgenstern, *Theory of games and economic behavior (commemorative edition)*. Princeton university press, 2007.

[21] E. Štrumbelj and I. Kononenko, "Explaining prediction models and individual predictions with feature contributions," *Knowledge and information systems*, vol. 41, no. 3, pp. 647–665, 2014.

[22] A. Datta, S. Sen, and Y. Zick, "Algorithmic transparency via quantitative input influence: Theory and experiments with learning systems," in *2016 IEEE symposium on security and privacy (SP)*, IEEE, 2016, pp. 598–617.

[23] C. Frye, I. Feige, and C. Rowat, "Asymmetric shapley values: Incorporating causal knowledge into model-agnostic explainability," *arXiv preprint arXiv:1910.06358*, 2019.

[24] D. Janzing, L. Minorics, and P. Blöbaum, "Feature relevance quantification in explainable ai: A causal problem," in *International Conference on Artificial Intelligence and Statistics*, PMLR, 2020, pp. 2907–2916.

[25] K. D. Pandl, F. Feiland, S. Thiebes, and A. Sunyaev, "Trustworthy machine learning for health care: Scalable data valuation with the shapley value," in *Proceedings of the Conference on Health, Inference, and Learning*, 2021, pp. 47–57.

[26] A. Ghorbani and J. Zou, "Data shapley: Equitable valuation of data for machine learning," in *International Conference on Machine Learning*, PMLR, 2019, pp. 2242–2251.

[27] R. Jia, X. Sun, J. Xu, C. Zhang, B. Li, and D. Song, "An empirical and comparative analysis of data valuation with scalable algorithms," *arXiv preprint arXiv:1911.07128*, 2019.

[28] S. E. Page, *The model thinker: What you need to know to make data work for you*. Hachette UK, 2018.

[29] J. Pearl, *Causality*. Cambridge university press, 2009.

[30] J. Peters, D. Janzing, and B. Schölkopf, *Elements of causal inference*. The MIT Press, 2017.

[31] Z. C. Lipton, "The mythos of model interpretability: In machine learning, the concept of interpretability is both important and slippery.," *Queue*, vol. 16, no. 3, pp. 31–57, 2018.

[32] C. Rudin, "Please stop explaining black box models for high stakes decisions," *arXiv preprint arXiv:1811.10154*, vol. 1, 2018.

[33] B. Kim and F. Doshi-Velez, "Machine learning techniques for accountability," *AI Magazine*, vol. 42, no. 1, pp. 47–52, 2021.

[34] T. Wang, C. Rudin, F. Doshi-Velez, Y. Liu, E. Klampfl, and P. MacNeille, "A bayesian framework for learning rule sets for interpretable classification," *The Journal of Machine Learning Research*, vol. 18, no. 1, pp. 2357–2393, 2017.

[35] G. Meyfroidt, F. Güiza, J. Ramon, and M. Bruynooghe, "Machine learning techniques to examine large patient databases," *Best Practice & Research Clinical Anaesthesiology*, vol. 23, no. 1, pp. 127–143, 2009.

[36] I. Kononenko, "Machine learning for medical diagnosis: History, state of the art and perspective," *Artificial Intelligence in medicine*, vol. 23, no. 1, pp. 89–109, 2001.

[37] P. W. Koh and P. Liang, "Understanding black-box predictions via influence functions," in *International Conference on Machine Learning*, PMLR, 2017, pp. 1885–1894.

[38] M. Mitchell, S. Wu, A. Zaldivar, *et al.*, "Model cards for model reporting," in *Proceedings of the conference on fairness, accountability, and transparency*, 2019, pp. 220–229.

[39] G. Bansal, B. Nushi, E. Kamar, E. Horvitz, and D. S. Weld, "Is the most accurate ai the best teammate? optimizing ai for teamwork," 2021.

[40] K. Simonyan, A. Vedaldi, and A. Zisserman, "Deep inside convolutional networks: Visualising image classification models and saliency maps," *arXiv preprint arXiv:1312.6034*, 2013.

[41] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba, "Learning deep features for discriminative localization," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 2921–2929.

[42] A. Shrikumar, P. Greenside, and A. Kundaje, "Learning important features through propagating activation differences," *arXiv preprint arXiv:1704.02685*, 2017.

[43] D. Baehrens, T. Schroeter, S. Harmeling, M. Kawanabe, K. Hansen, and K.-R. Müller, "How to explain individual classification decisions," *The Journal of Machine Learning Research*, vol. 11, pp. 1803–1831, 2010.

[44] A. Binder, G. Montavon, S. Lapuschkin, K.-R. Müller, and W. Samek, "Layer-wise relevance propagation for neural networks with local renormalization layers," in *International Conference on Artificial Neural Networks*, Springer, 2016, pp. 63–71.

[45] J. T. Springenberg, A. Dosovitskiy, T. Brox, and M. Riedmiller, "Striving for simplicity: The all convolutional net," *arXiv preprint arXiv:1412.6806*, 2014.

[46] A. Fisher, C. Rudin, and F. Dominici, "All models are wrong but many are useful: Variable importance for black-box, proprietary, or misspecified prediction models, using model class reliance," *arXiv preprint arXiv:1801.01489*, pp. 237–246, 2018.

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

[48] T. J. Hastie and R. J. Tibshirani, *Generalized additive models*. CRC press, 1990, vol. 43.

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

[50] Q. Zhang, Y. N. Wu, and S.-C. Zhu, "Interpretable convolutional neural networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 8827–8836.

[51] C.-K. Yeh, B. Kim, S. O. Arik, C.-L. Li, T. Pfister, and P. Ravikumar, "On completeness-aware concept-based explanations in deep neural networks," *arXiv preprint arXiv:1910.07969*, 2019.

[52] C. Chen, O. Li, C. Tao, A. J. Barnett, J. Su, and C. Rudin, "This looks like that: Deep learning for interpretable image recognition," *arXiv preprint arXiv:1806.10574*, 2018.

[53] O. Li, H. Liu, C. Chen, and C. Rudin, "Deep learning for case-based reasoning through prototypes: A neural network that explains its predictions," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 32, 2018.

[54] B. Kim, C. Rudin, and J. A. Shah, "The bayesian case model: A generative approach for case-based reasoning and prototype classification," in *Advances in neural information processing systems*, 2014, pp. 1952–1960.

[55] E. Choi, M. T. Bahadori, J. A. Kulas, A. Schuetz, W. F. Stewart, and J. Sun, "Retain: An interpretable predictive model for healthcare using reverse time attention mechanism," *arXiv preprint arXiv:1608.05745*, 2016.

[56] N. Jethani, M. Sudarshan, Y. Aphinyanaphongs, and R. Ranganath, "Have we learned to explain?: How interpretability methods can learn to encode predictions in their interpretations.," in *International Conference on Artificial Intelligence and Statistics*, PMLR, 2021, pp. 1459–1467.

[57] P. Dabkowski and Y. Gal, "Real time image saliency for black box classifiers," *arXiv preprint arXiv:1705.07857*, 2017.

[58] J. Chen, L. Song, M. Wainwright, and M. Jordan, "Learning to explain: An information-theoretic perspective on model interpretation," in *International Conference on Machine Learning*, PMLR, 2018, pp. 883–892.

[59] J. Yoon, J. Jordon, and M. van der Schaar, "Invase: Instance-wise variable selection using neural networks," in *International Conference on Learning Representations*, 2018.

[60] M. D. Zeiler and R. Fergus, "Visualizing and understanding convolutional networks," in *European conference on computer vision*, Springer, 2014, pp. 818–833.

[61] B. Kim, M. Wattenberg, J. Gilmer, C. Cai, J. Wexler, F. Viegas, *et al.*, "Interpretability beyond feature attribution: Quantitative testing with concept activation vectors (tcav)," in *International conference on machine learning*, PMLR, 2018, pp. 2668–2677.

[62] G. Hinton, O. Vinyals, and J. Dean, "Distilling the knowledge in a neural network," *arXiv preprint arXiv:1503.02531*, 2015.

[63] M. Wu, M. Hughes, S. Parbhoo, M. Zazzi, V. Roth, and F. Doshi-Velez, "Beyond sparsity: Tree regularization of deep models for interpretability," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 32, 2018.

[64] J. Adebayo, M. Muelly, I. Liccardi, and B. Kim, "Debugging tests for model explanations," *arXiv preprint arXiv:2011.05429*, 2020.

[65] R. Guidotti, A. Monreale, S. Ruggieri, F. Turini, F. Giannotti, and D. Pedreschi, "A survey of methods for explaining black box models," *ACM computing surveys (CSUR)*, vol. 51, no. 5, pp. 1–42, 2018.

[66] A. Ghorbani, A. Abid, and J. Zou, "Interpretation of neural networks is fragile," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, 2019, pp. 3681–3688.

[67] S. Srinivas and F. Fleuret, "Rethinking the role of gradient-based attri-bution methods for model interpretability," *arXiv preprint arXiv:2006.09128*, 2020.

[68] A. Altmann, L. Toloşi, O. Sander, and T. Lengauer, "Permutation importance: A corrected feature importance measure," *Bioinformatics*, vol. 26, no. 10, pp. 1340–1347, 2010.

[69] A. Chattopadhyay, P. Manupriya, A. Sarkar, and V. N. Balasubramanian, "Neural network attributions: A causal perspective," in *International Conference on Machine Learning*, PMLR, 2019, pp. 981–990.

[70] K. Dhamdhere, M. Sundararajan, and Q. Yan, "How important is a neuron?" *arXiv preprint arXiv:1805.12233*, 2018.

[71] W. Samek, A. Binder, G. Montavon, S. Lapuschkin, and K.-R. Müller, "Evaluating the visualization of what a deep neural network has learned," *IEEE transactions on neural networks and learning systems*, vol. 28, no. 11, pp. 2660–2673, 2016.

[72] C.-K. Yeh, C.-Y. Hsieh, A. S. Suggala, D. I. Inouye, and P. Ravikumar, "On the (in) fidelity and sensitivity for explanations," *arXiv preprint arXiv:1901.09392*, 2019.

[73] A. A. Ismail, M. Gunady, H. C. Bravo, and S. Feizi, "Benchmarking deep learning interpretability in time series predictions," *arXiv preprint arXiv:2010.13924*, 2020.

[74] C. Molnar, *Interpretable machine learning*. Lulu. com, 2020.

[75] S. M. Lundberg, G. G. Erion, and S.-I. Lee, "Consistent individualized feature attribution for tree ensembles," *arXiv preprint arXiv:1802.03888*, 2018.

[76] S. M. Lundberg, G. Erion, H. Chen, *et al.*, "From local explanations to global understanding with explainable ai for trees," *Nature machine intelligence*, vol. 2, no. 1, pp. 2522–5839, 2020.

[77] M. Sundararajan and A. Najmi, "The many shapley values for model explanation," *arXiv preprint arXiv:1908.08474*, 2019.

[78] R. Tibshirani, "Regression shrinkage and selection via the lasso," *Journal of the Royal Statistical Society. Series B (Methodological)*, pp. 267–288, 1996.

[79] A. E. Hoerl and R. W. Kennard, "Ridge regression: Biased estimation for nonorthogonal problems," *Technometrics*, vol. 12, no. 1, pp. 55–67, 1970.

[80] H. Zou and T. Hastie, "Regularization and variable selection via the elastic net," *Journal of the Royal Statistical Society*, vol. 67, no. 2, pp. 301–320, 2005.

[81] M. A. Figueiredo and R. D. Nowak, "Sparse estimation with strongly correlated variables using ordered weighted l1 regularization," *arXiv preprint arXiv:1409.4005*, 2014.

[82] Y. Grandvalet and S. Canu, "Outcomes of the equivalence of adaptive ridge with least absolute shrinkage," *Advances in neural information processing systems*, vol. 11, 1998.

[83] I. Goodfellow, Y. Bengio, A. Courville, and Y. Bengio, *Deep learning*, 2. MIT press Cambridge, 2016, vol. 1.

[84] D. A. Van Dyk and X.-L. Meng, "The art of data augmentation," *Journal of Computational and Graphical Statistics*, vol. 10, no. 1, pp. 1–50, 2001.

[85] Y. Ganin, E. Ustinova, H. Ajakan, *et al.*, "Domain-adversarial training of neural networks," *The journal of machine learning research*, vol. 17, no. 1, pp. 2096–2030, 2016.

[86] Y. Yao, L. Rosasco, and A. Caponnetto, "On early stopping in gradient descent learning," *Constructive Approximation*, vol. 26, no. 2, pp. 289–315, 2007.

[87] H. Zou, "The adaptive lasso and its oracle properties," *Journal of the American statistical association*, vol. 101, no. 476, pp. 1418–1429, 2006.

[88] L. C. Bergersen, I. K. Glad, and H. Lyng, "Weighted lasso with data integration," *Statistical applications in genetics and molecular biology*, vol. 10, no. 1, 2011.

[89] C. Zeng, D. C. Thomas, and J. P. Lewinger, "Incorporating prior knowledge into regularized regression," *bioRxiv*, 2020.

[90] L. Meier, S. Van De Geer, and P. Bühlmann, "The group lasso for logistic regression," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 70, no. 1, pp. 53–71, 2008.

[91] Z. Shen, J. Liu, Y. He, *et al.*, "Towards out-of-distribution generalization: A survey," *arXiv preprint arXiv:2108.13624*, 2021.

[92] V. Vapnik, "Principles of risk minimization for learning theory," *Advances in neural information processing systems*, vol. 4, 1991.

[93] R. Geirhos, J.-H. Jacobsen, C. Michaelis, *et al.*, "Shortcut learning in deep neural networks," *Nature Machine Intelligence*, vol. 2, no. 11, pp. 665–673, 2020.

[94] S. Beery, G. Van Horn, and P. Perona, "Recognition in terra incognita," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 456–473.

[95] R. Geirhos, P. Rubisch, C. Michaelis, M. Bethge, F. A. Wichmann, and W. Brendel, "Imagenet-trained cnns are biased towards texture; increasing shape bias improves accuracy and robustness," *ICLR*, 2019.

[96] J. R. Zech, M. A. Badgeley, M. Liu, A. B. Costa, J. J. Titano, and E. K. Oermann, "Variable generalization performance of a deep learning model to detect pneumonia in chest radiographs: A cross-sectional study," *PLoS medicine*, vol. 15, no. 11, e1002683, 2018.

[97] J. Shane, "Do neural nets dream of electric sheep," *AI Wierdness*, 2018. [Online]. Available: `https://aiweirdness.com/post/171451900302/do-neural-nets-dream-of-electric-sheep`.

[98] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," *arXiv preprint arXiv:1412.6572*, 2014.

[99] Y. S. Abu-Mostafa, M. Magdon-Ismail, and H.-T. Lin, *Learning from data*. AMLBook New York, 2012, vol. 4.

[100] M. Makar, J. Guttag, and J. Wiens, "Learning the probability of activation in the presence of latent spreaders," in *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

[101] L. Torrey and J. Shavlik, "Transfer learning," in *Handbook of research on machine learning applications and trends: algorithms, methods, and techniques*, IGI global, 2010, pp. 242–264.

[102] Y. Ganin and V. Lempitsky, "Unsupervised domain adaptation by backpropagation," in *International conference on machine learning*, PMLR, 2015, pp. 1180–1189.

[103] Z. Shen, P. Cui, K. Kuang, B. Li, and P. Chen, "Causally regularized learning with agnostic data selection bias," in *Proceedings of the 26th ACM international conference on Multimedia*, 2018, pp. 411–419.

[104] Z. Shen, P. Cui, J. Liu, T. Zhang, B. Li, and Z. Chen, "Stable learning via differentiated variable decorrelation," in *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2020, pp. 2185–2193.

[105] E. D. Cubuk, B. Zoph, D. Mane, V. Vasudevan, and Q. V. Le, "Autoaugment: Learning augmentation policies from data," *arXiv preprint arXiv:1805.09501*, 2018.

[106] D. Yin, R. G. Lopes, J. Shlens, E. D. Cubuk, and J. Gilmer, "A fourier perspective on model robustness in computer vision," *arXiv preprint arXiv:1906.08988*, 2019.

[107] R. G. Lopes, D. Yin, B. Poole, J. Gilmer, and E. D. Cubuk, "Improving robustness without sacrificing accuracy with patch gaussian augmentation," *arXiv preprint arXiv:1906.02611*, 2019.

[108] D. Hendrycks, S. Basart, N. Mu, *et al.*, "The many faces of robustness: A critical analysis of out-of-distribution generalization," *arXiv preprint arXiv:2006.16241*, 2020.

[109] M. Nauta, R. Walsh, A. Dubowski, and C. Seifert, "Uncovering and correcting shortcut learning in machine learning models for skin cancer diagnosis," *Diagnostics*, vol. 12, no. 1, p. 40, 2022.

[110] A. S. Ross, M. C. Hughes, and F. Doshi-Velez, "Right for the right reasons: Training differentiable models by constraining their explanations," *arXiv preprint arXiv:1703.03717*, 2017.

[111] M. Makar, B. Packer, D. Moldovan, D. Blalock, Y. Halpern, and A. D'Amour, "Causally-motivated shortcut removal using auxiliary labels," *arXiv preprint arXiv:2105.06422*, 2021.

[112] M. Du, V. Manjunatha, R. Jain, *et al.*, "Towards interpreting and mitigating shortcut learning behavior of nlu models," *arXiv preprint arXiv:2103.06922*, 2021.

[113] M. Arjovsky, L. Bottou, I. Gulrajani, and D. Lopez-Paz, "Invariant risk minimization," *arXiv preprint arXiv:1907.02893*, 2019.

[114] D. Krueger, E. Caballero, J.-H. Jacobsen, *et al.*, "Out-of-distribution generalization via risk extrapolation (rex)," in *International Conference on Machine Learning*, PMLR, 2021, pp. 5815–5826.

[115] E. Delage and Y. Ye, "Distributionally robust optimization under moment uncertainty with application to data-driven problems," *Operations research*, vol. 58, no. 3, pp. 595–612, 2010.

[116] L. Breiman, "Random forests," *Machine learning*, vol. 45, no. 1, pp. 5–32, 2001.

[117] T. Heskes, E. Sijben, I. G. Bucur, and T. Claassen, "Causal shapley values: Exploiting causal knowledge to explain individual predictions of complex models," *Advances in neural information processing systems*, 2020.

[118] S. López and M. Saboya, "On the relationship between shapley and owen values," *Central European Journal of Operations Research*, vol. 17, no. 4, p. 415, 2009.

[119] R. Moraffah, M. Karami, R. Guo, A. Raglin, and H. Liu, "Causal interpretability for machine learning-problems, methods and evaluation," *ACM SIGKDD Explorations Newsletter*, vol. 22, no. 1, pp. 18–33, 2020.

[120] H. Chen, J. D. Janizek, S. Lundberg, and S.-I. Lee, "True to the model or true to the data?" *arXiv preprint arXiv:2006.16234*, 2020.

[121] A. Shrikumar, P. Greenside, A. Shcherbina, and A. Kundaje, "Not just a black box: Learning important features through propagating activation differences," *arXiv preprint arXiv:1605.01713*, 2016.

[122] A. Adadi and M. Berrada, "Peeking inside the black-box: A survey on explainable artificial intelligence (xai)," *IEEE Access*, vol. 6, pp. 52 138–52 160, 2018.

[123] B. Mittelstadt, C. Russell, and S. Wachter, "Explaining explanations in ai," in *Proceedings of the conference on fairness, accountability, and transparency*, 2019, pp. 279–288.

[124] C. Glymour, K. Zhang, and P. Spirtes, "Review of causal discovery methods based on graphical models," *Frontiers in genetics*, vol. 10, p. 524, 2019.

[125] C. S. Cox, *Plan and operation of the NHANES I Epidemiologic Followup Study, 1992*, 35. National Ctr for Health Statistics, 1998.

[126] W. R. Robinson, A. Renson, and A. I. Naimi, "Teaching yourself about structural racism will improve your machine learning," *Biostatistics*, vol. 21, no. 2, pp. 339–344, 2020.

[127] L. Merrick and A. Taly, "The explanation game: Explaining machine learning models using shapley values," in *Machine Learning and Knowledge Extraction*, A. Holzinger, P. Kieseberg, A. M. Tjoa, and E. Weippl, Eds., Cham: Springer International Publishing, 2020, pp. 17–38, ISBN: 978-3-030-57321-8.

[128] S. Hara and T. Maehara, "Finding alternate features in lasso," *arXiv preprint arXiv:1611.05940*, 2016.

[129] H. Lakkaraju and C. Rudin, "Learning cost-effective and interpretable treatment regimes," in *Artificial Intelligence and Statistics*, 2017, pp. 166–175.

[130] Z. C. Lipton, "The mythos of model interpretability," *ICML Workshop on Human Interpretability in Machine Learning*, 2016.

[131] B. Ustun and C. Rudin, "Methods and models for interpretable linear classification," *arXiv preprint arXiv:1405.4047*, 2014.

[132] J. Sun, J. Hu, D. Luo, *et al.*, "Combining knowledge and data driven insights for identifying risk factors using electronic health records.," in *AMIA*, vol. 2012, 2012, pp. 901–10.

[133] V. Vapnik and R. Izmailov, "Learning using privileged information: Similarity control and knowledge transfer.," *Journal of Machine Learning Research*, vol. 16, pp. 2023–2049, 2015.

[134] T. Helleputte and P. Dupont, "Partially supervised feature selection with regularized linear models," in *Proceedings of the 26th Annual International Conference on Machine Learning*, ACM, 2009, pp. 409–416.

[135] E. Choi, M. T. Bahadori, L. Song, W. F. Stewart, and J. Sun, "Gram: Graph-based attention model for healthcare representation learning," in *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ACM, 2017, pp. 787–795.

[136] A. Ben-David, "Monotonicity maintenance in information-theoretic machine learning algorithms," *Machine Learning*, vol. 19, no. 1, pp. 29–43, 1995.

[137] W. Kotłowski and R. Słowiński, "Rule learning with monotonicity constraints," in *Proceedings of the 26th Annual International Conference on Machine Learning*, ACM, 2009, pp. 537–544.

[138] D. Martens, J. Vanthienen, W. Verbeke, and B. Baesens, "Performance of classification models from a user perspective," *Decision Support Systems*, vol. 51, no. 4, pp. 782–793, 2011.

[139] M. J. Pazzani, S. Mani, W. R. Shankle, *et al.*, "Acceptance of rules generated by machine learning among medical experts," *Methods of information in medicine*, vol. 40, no. 5, pp. 380–385, 2001.

[140] W. Verbeke, D. Martens, C. Mues, and B. Baesens, "Building comprehensible customer churn prediction models with advanced rule induction techniques," *Expert Systems with Applications*, vol. 38, no. 3, pp. 2354–2364, 2011.

[141] E. E. Altendorf, A. C. Restificar, and T. G. Dietterich, "Learning from sparse data by exploiting monotonicity constraints," *arXiv preprint arXiv:1207.1364*, 2012.

[142] J. Sill, "Monotonic networks," *Advances in neural information processing systems*, pp. 661–667, 1998.

[143] M. Velikova, H. Daniels, and A. Feelders, "Solving partially monotone problems with neural networks," in *Proceedings of the International Conference on Neural Networks, Vienna, Austria*, 2006.

[144] V. Chandrasekaran, B. Recht, P. A. Parrilo, and A. S. Willsky, "The convex geometry of linear inverse problems," *Foundations of Computational mathematics*, vol. 12, no. 6, pp. 805–849, 2012.

[145] N. Hurley and S. Rickard, "Comparing measures of sparsity," *IEEE Transactions on Information Theory*, vol. 55, no. 10, pp. 4723–4741, 2009.

[146] M. D. Zeiler, "Adadelta: An adaptive learning rate method," *arXiv preprint arXiv:1212.5701*, 2012.

[147] J. Oh, M. Makar, C. Fusco, *et al.*, "A generalizable, data-driven approach to predict daily risk of Clostridium difficile infection at two large academic health centers," *Infection Control and Hospital Epidemiology*, 2018.

[148] K. Garey, T. Dao-Tran, Z. Jiang, M. Price, L. Gentry, and H. Dupont, "A clinical risk index for Clostridium difficile infection in hospitalised patients receiving broad-spectrum antibiotics," *Journal of Hospital Infection*, vol. 70, no. 2, pp. 142–147, 2008.

[149] E. R. Dubberke, Y. Yan, K. A. Reske, *et al.*, "Development and validation of a Clostridium difficile infection risk prediction model," *Infection Control & Hospital Epidemiology*, vol. 32, no. 04, pp. 360–366, 2011.

[150] J. Wiens, W. N. Campbell, E. S. Franklin, J. V. Guttag, and E. Horvitz, "Learning data-driven patient risk str. jpegication models for clostridium difficile," in *Open forum infectious diseases*, Oxford University Press, vol. 1, 2014, ofu045.

[151] A. L. Goldberger, L. A. N. Amaral, L. Glass, *et al.*, "PhysioBank, PhysioToolkit, and PhysioNet: Components of a new research resource for complex physiologic signals," *Circulation*, vol. 101, no. 23, e215–e220, 2000, Circulation Electronic Pages: http://circ.ahajournals.org/content/101/23/e215.full PMID:1085218; doi: 10.1161/01.CIR.101.23.e215.

[152] I. Silva, G. Moody, D. J. Scott, L. A. Celi, and R. G. Mark, "Predicting in-hospital mortality of icu patients: The physionet/computing in cardiology challenge 2012," in *Computing in Cardiology, 2012*, IEEE, 2012, pp. 245–248.

[153] J.-R. Le Gall, P. Loirat, A. Alperovitch, *et al.*, "A simplified acute physiology score for icu patients.," *Critical care medicine*, vol. 12, no. 11, pp. 975–977, 1984.

[154] A. Subbaswamy, P. Schulam, and S. Saria, "Preventing failures due to dataset shift: Learning predictive models that transport," in *The 22nd International Conference on Artificial Intelligence and Statistics*, PMLR, 2019, pp. 3118–3127.

[155] S. Jabbour, D. Fouhey, E. Kazerooni, M. W. Sjoding, and J. Wiens, "Deep learning applied to chest x-rays: Exploiting and preventing shortcuts," in *Machine Learning for Healthcare Conference*, PMLR, 2020, pp. 750–782.

[156] R. Geirhos, C. R. M. Temme, J. Rauber, H. H. Schütt, M. Bethge, and F. A. Wichmann, "Generalisation in humans and deep neural networks," *arXiv preprint arXiv:1808.08750*, 2018.

[157] M. Du, N. Liu, F. Yang, and X. Hu, "Learning credible deep neural networks with rationale regularization," in *2019 IEEE International Conference on Data Mining (ICDM)*, IEEE, 2019, pp. 150–159.

[158] J. M. Wooldridge, *Introductory econometrics: A modern approach*. Cengage learning, 2015.

[159] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 2818–2826.

[160] C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie, "The caltech-ucsd birds-200-2011 dataset," 2011.

[161] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *2009 IEEE conference on computer vision and pattern recognition*, Ieee, 2009, pp. 248–255.

[162] A. E. Johnson, T. J. Pollard, S. J. Berkowitz, *et al.*, "Mimic-cxr, a de-identified publicly available database of chest radiographs with free-text reports," *Scientific data*, vol. 6, no. 1, pp. 1–8, 2019.

[163] A. L. Goldberger, L. A. Amaral, L. Glass, *et al.*, "Physiobank, physiotoolkit, and physionet: Components of a new research resource for complex physiologic signals," *circulation*, vol. 101, no. 23, e215–e220, 2000.

[164] A. Johnson, L. Bulgarelli, T. Pollard, S. Horng, L. A. Celi, and R. Mark, *Mimic-iv*, 2020. DOI: 10.13026/A3WN-HQ05. [Online]. Available: https://physionet.org/content/mimiciv/0.4/.

[165] J. Irvin, P. Rajpurkar, M. Ko, *et al.*, "Chexpert: A large chest radiograph dataset with uncertainty labels and expert comparison," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 33, 2019, pp. 590–597.

[166] C. Lovering, R. Jha, T. Linzen, and E. Pavlick, "Predicting inductive biases of pre-trained models," in *International Conference on Learning Representations*, 2020.

[167] M. T. Bahadori and D. E. Heckerman, "Debiasing concept bottleneck models with instrumental variables," *arXiv preprint arXiv:2007.11500*, 2020.

[168] I. Covert, S. Lundberg, and S.-I. Lee, "Explaining by removing: A unified framework for model explanation," *Journal of Machine Learning Research*, vol. 22, no. 209, pp. 1–90, 2021.

[169] R. Singal, G. Michailidis, and H. Ng, "Flow-based attribution in graphical models: A recursive shapley approach," *Available at SSRN 3845526*, 2021.

[170] I. E. Kumar, C. Scheidegger, S. Venkatasubramanian, and S. Friedler, "Shapley residuals: Quantifying the limits of the shapley value for explanations.," in *ICML Workshop on Workshop on Human Interpretability in Machine Learning (WHI)*, 2020.

[171] J. F. Banzhaf III, "Weighted voting doesn't work: A mathematical analysis," *Rutgers L. Rev.*, vol. 19, p. 317, 1964.

[172] V. Viswanathan and Y. Zick, "Model explanations via the axiomatic causal lens," *arXiv preprint arXiv:2109.03890*, 2021.

[173] L. von Rueden, S. Mayer, K. Beckh, *et al.*, "Informed machine learning-a taxonomy and survey of integrating prior knowledge into learning systems," *IEEE Transactions on Knowledge and Data Engineering*, 2021.

[174] A. Ratner, S. H. Bach, H. Ehrenberg, J. Fries, S. Wu, and C. Ré, "Snorkel: Rapid training data creation with weak supervision," in *Proceedings of the VLDB Endowment. International Conference on Very Large Data Bases*, NIH Public Access, vol. 11, 2017, p. 269.

[175] D. Kingma and J. Ba, "Adam: A method for stochastic optimization (2014)," *International Conference on Learning Representations*, vol. 15, 2015.