ABSTRACT

CHAUDHARY, MANDAR SANJAY. Methods for Learning and Predicting Causal Relationships from Observational Data. (Under the direction of Dr. Nagiza F. Samatova).

Discovering causal relationships from observational data has contributed significantly towards understanding mechanisms of complex systems. Specifically, the domains of biology and climate science have benefited from observational studies where performing external interventions is infeasible or impractical. For example, in biology identifying genes having a causal effect on a phenotype (such as riboflavin production rate) has led to prioritizing gene knock-out experiments. However, the increasing amount of observational data is accompanied by its own set of challenges such as high-dimensionality, smaller sample size and missing ground truth to validate causal relationships. Thus, developing algorithms that effectively discover causal relationships becomes a challenging task.

In this dissertation, we propose two main contributions for extracting causal information from observational data. First, in contrast to existing causal feature selection methods, we identify meaningful predictive features by incorporating the causal structure as well as the strength of causal effects. This leads to discovering relationships that cannot be detected by causal structure learning methods, especially while dealing with high-dimensional data sets with smaller sample size. Second, we mine observational data to discover patterns of association between variables that discriminate causal relationships from non-causal relationships. This results in a significant improvement in the quality of output causal relationships.

Traditional causal discovery methods focus on learning the causal structure of a network. The potential causal relationships in the network are studied by domain scientists to validate their hypotheses. Studying each causal relationship can become infeasible in high-dimensional settings. In the first part of this dissertation, we develop an algorithm to reduce the search space of causal relationships and evaluate the predictive ability of the factors in such relationships with respect to a given phenomenon of interest such as seasonal rainfall in a geographic region. Results show that our algorithm is consistently among the top performing methods compared to state-of-art feature selection methods. Additionally, our algorithm identifies physically interpretable features that can explain the prediction performance of the model.

In the second part, we develop a graph partitioning based approach to complement a constraint-based causal structure learning algorithm. Partitioning a graph into smaller subgraphs guides the process of discovering causal relationships. Consequently, reducing the error of falsely removing a causal relationship. Experiments on networks of different sizes, complexities and data types show the effectiveness of our algorithm.

In the final component, we posit that causal discovery from continuous additive noise models

can be transformed into a supervised learning problem. We develop an algorithm that utilizes kernel mean embedding to generate patterns from a parent-child and a non-parent child relationship. Simulations show the presence of a discriminative pattern in networks generated with non-linear data. Experiments on synthetic data and real-world data illustrate the effectiveness of our algorithm in terms of the quality of causal graphs as it outperforms state-of-art causal discovery methods.

Methods for Learning and Predicting Causal Relationships from Observational Data

by
Mandar Sanjay Chaudhary

A dissertation submitted to the Graduate Faculty of
North Carolina State University
in partial fulfillment of the
requirements for the Degree of
Doctor of Philosophy

Computer Science

Raleigh, North Carolina

2019

APPROVED BY:

Dr. Dennis R. Bahler	Dr. Steffen Heber
Dr. R. Raju Vatsavai	Dr. Nagiza F. Samatova Chair of Advisory Committee

DEDICATION

To my parents for their unwavering love, support and encouragement.

BIOGRAPHY

Mandar Chaudhary is a PhD student in the department of Computer Science at the North Carolina State University advised by Dr. Nagiza F. Samatova. He received his M.S. in Computer Science from North Carolina State University in 2014 and a B.E. in Computer Engineering from Gujarat Technological University, India in 2012. During his doctoral study, Mandar had appointments as a research assistant and a teaching assistant. He also had the opportunity to work as a data science intern at Motorola Mobility. His current research focuses on developing algorithms using machine learning and graph mining for effectively discovering causal relationships from observational data.

ACKNOWLEDGEMENTS

My journey as a PhD student would not have been possible without the constant support and guidance from a number of people. First, I would like to thank my advisor Dr. Nagiza F. Samatova for her continued support and guidance during my doctoral studies. Her wisdom, advice and encouragement have enabled me to become the person and researcher I am today.

I would like to express my gratitude towards the faculty at the Computer Science department of North Carolina State University. Particularly, I'm thankful to Dr. Dennis Bahler, Dr. Steffen Heber and Dr. R. Raju Vatsavai for their participation in my advisory committee and for providing insightful discussions to improve my dissertation. I would also like to thank Dr. George N. Rouskas, Director of Graduate Programs for his support during my graduate studies.

I was also fortunate to have valuable collaborations from different research institutions. I would like to thank my collaborators from the National Science Foundation (NSF) Expeditions in Computing project "Understanding Climate Change: A Data Driven Approach," particularly Dr. Fredrick H. M. Semazzi at North Carolina State University's Department of Marine, Earth, and Atmospheric Sciences (MEAS) and Dr. Vipin Kumar at the University of Minnesota. I would also like to thank Dr. Semazzi's research group member Michael Angus for his valuable insights regarding the application of my work to climate science domain in Chapter 2. I would also like to thank Dr. Murali Doraiswamy and Dr. Elizabeth Cirulli at Duke University for their insightful feedback about the application of my work in biology domain.

Additionally, I am grateful for having the opportunity to work with fellow students and colleagues at Dr. Samatova's research lab. I would like to thank Gonzalo Bello, Steve Harenberg, Stephen Ranshous, Shiou "Claude" Tian Hsu, Dhara Desai, Changsung Moon, Khelan Patel, Ameeta Muralidharan, Doel Gonzalez, and David "Drew" Boyuka.

Finally, I am eternally grateful to my family, particularly my parents, Sanjay and Sunita, my grandparents, Paruben and Raghuveerbhai, my sister Anuradha and my wife Foram for their unconditional love, motivation and support throughout this journey.

This dissertation is based upon work supported in part by the Laboratory for Analytic Sciences (LAS) and the NSF grant 1029711. Any opinions, findings, conclusions, or recommendations expressed in this dissertation are those of the author and do not necessarily reflect the views of any agency or entity of the United States government.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER

1

INTRODUCTION

Causal inference provides meaningful insights into the mechanism of a complex system. Identifying cause-and-effect relationships can facilitate domain knowledge by confirming existing hypothesis as well as developing new ones. Additionally, these relationships can be used for prioritizing experiments to perform external interventions. The gold-standard approach of inferring causal relationships is to perform randomized controlled experiments (RCEs). However, performing RCEs might be infeasible, computationally expensive or even unethical. For example, it is impossible to externally intervene on the sea surface temperature of a geographical region and observe its effect on the seasonal rainfall. An alternative approach is to derive causal information from observational studies. For example, in the domain of climate science, methods for discovering causal relationships have captured interesting patterns from time series data of an atmospheric field such as geopotential height [EUD12; EUD14]. Figure 1.1 shows these patterns as the strongest pathways of interactions which have been identified as storm tracks. It is important to note that discovering such patterns would not have been possible using traditional climate science methods [EUD12].

Discovering causal relationships can illuminate the understanding of complex systems. Another significant contribution of causal inference methods has been in the field of machine learning. In particular, feature selection is defined as the problem of identifying a subset of relevant features with respect to a target variable of interest that can be used for building prediction models. The local causal structure of a target variable can be used to select a subset of features that can predict its values accurately. In other words, discovering causal relationships around a target variable provides

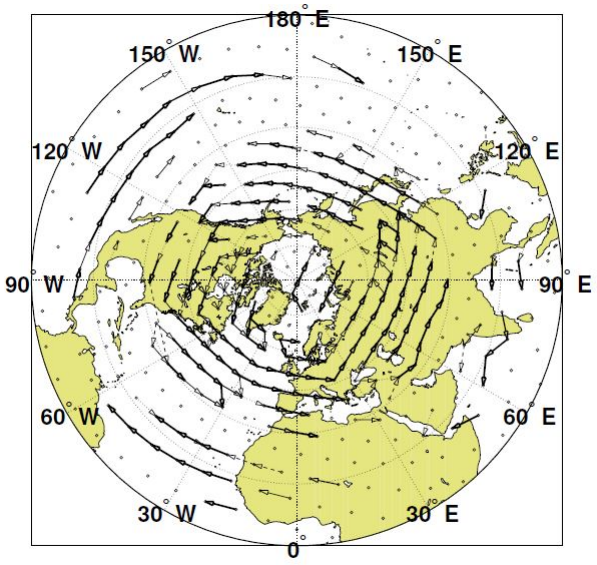


Figure 1.1 An example graph represents information flow for an atmospheric field such as geopotential height around the globe. The nodes (dots) represent spatial points that provide time series data and the directed edges represent the strongest causal relationships between the different values of geopotential height [EUD14].

significant benefits in feature selection.

Figure 1.2 shows the local causal structure of a target variable T which includes its direct causes (A,B), effects (C) and spouses (F). This local structure of target variable T is known as its Markov Blanket (MB). Given its Markov Blanket, T is independent of every other variable in the graph. That is, no other variable provides additional information about T than the variables in its Markov Blanket [Ali10a]. Thus, estimating this local structure around T yields a reduced set of features that can be used for building prediction models. Additionally, these features are likely to be more interpretable than those selected by association-based feature selection methods, and they can be used by domain experts to evaluate the physical interpretability of prediction models.

There are two primary approaches for discovering cause-effect relationships from observational data: causal structure learning and causal effect estimation. The first approach learns a causal graph where each variable in the observational data is represented as a node in the graph, and a directed edge between two nodes, $X \rightarrow Y$ represents a causal relationship with X as the cause and Y as the effect. Several methods have been developed that can learn a causal structure based on the multivariate distribution in the observational data [MN16]. While this approach is widely accepted and applied in several domains, the lack of sufficient observational data can lead to very few directed edges. For example, in biology, the number of samples can be much smaller (few hundreds) than the size of the variable set (tens of thousands). Also, it is not possible to estimate the bounds on the

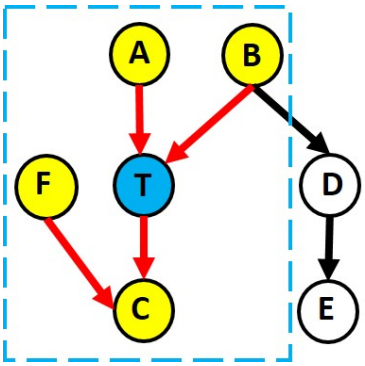


Figure 1.2 The local causal structure of a target variable T consists of its direct causes (A, B), direct effects (C) and spouses (F). This structure is referred to as the Markov Blanket of T. Conditioned on its Markov Blanket, T is independent of the remaining variables (D, E) [Ali10a].

causal effects using only the causal structure.

As a result, the second approach was developed to estimate bounds on causal effects of a factor with respect to the target variable. For example, in biology, the estimated causal effects of genes on a phenotype (such as riboflavin production rate) can be used to prioritize gene knock-out experiments. In this dissertation, we focus on constructing the unknown causal structure from observational data by combining both approaches. We will refer to this problem as causal discovery.

Causal discovery from observational data has become a ubiquitous tool to infer causal relationships, but it is not possible to interpret the output of these methods without making assumptions. In this work, we make three important assumptions about the underlying causal structure,

1. Acyclic causal structure: We assume that the underlying causal structure does not contain directed loops.

2. Causal faithfulness: We assume that the causal relationships discovered in the DAG are a result of the joint distribution of variables in the observational data. This assumption is referred to as causal faithfulness.

3. Causal sufficiency: We also assume that any variable that causes two or more variables is recorded in the observational data.

1.1 Causal Discovery Methods

We present an overview of the causal discovery methods in Figure 1.3. These methods have been mainly categorized based on their assumptions about the joint distribution of the variables in the data. A brief description of these methods is given below,

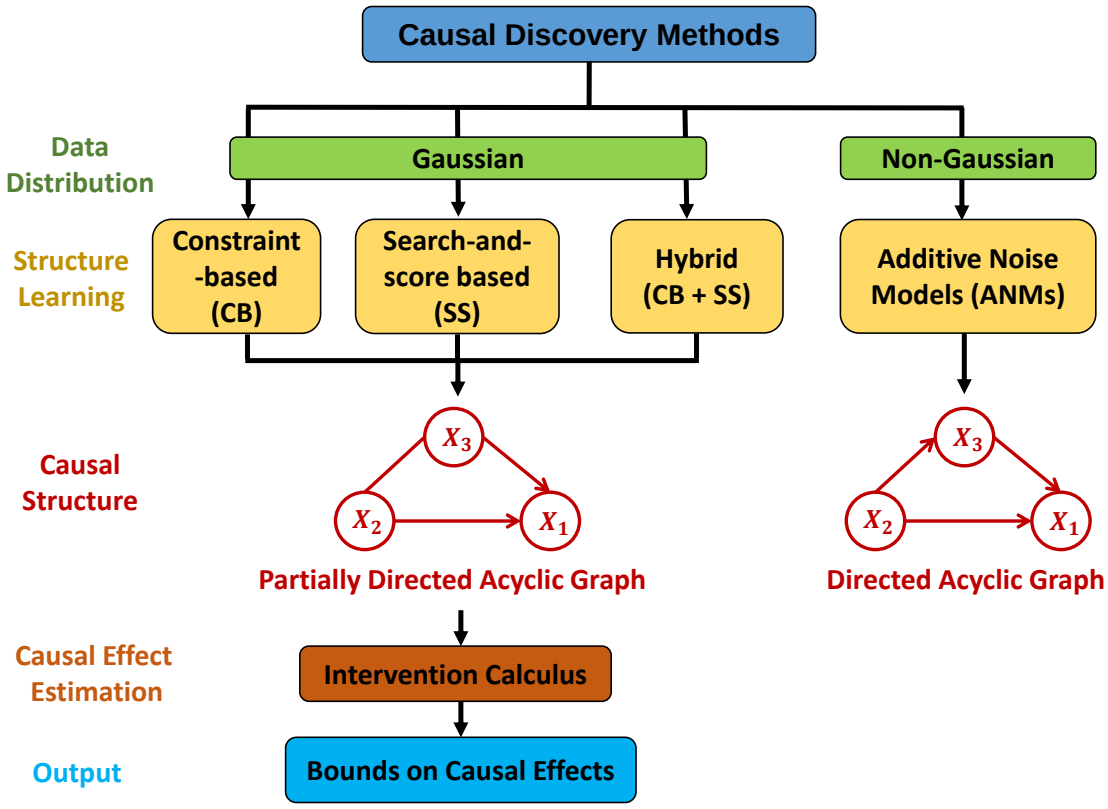


Figure 1.3 An overview of causal discovery methods categorized based on their assumptions about the joint distribution in the observational data. Constraint-based , search-and-score based, hybrid, and causal effect estimation methods assume multivariate Gaussian distribution and estimate a partially directed acyclic graph (PDAG). The PDAG can be used to estimate bounds on the causal effects using the so-called intervention calculus. On the other hand, by relaxing the assumption of Gaussianity in the observational data, additive noise models can estimate the complete directed acyclic graph (DAG).

1.1.1 Causal Structure Learning

The initial works in the development of causal structure learning assumed that the observational variables follow a multivariate Gaussian distribution. While this approach spurred the development of numerous algorithms, their output causal structure was restricted to a partially directed acyclic graph (PDAG). A PDAG contains a mixture of directed, undirected and bidirected edges, and it represents a set of Markov equivalent graphs which encode the same conditional independences. A drawback of algorithms relying on this assumption is that they cannot infer the direction of causal relationship in a bivariate case.

- **Constraint-based (CB)** methods aim to remove spurious associations between variables by performing statistical tests. The goal is to test the dependence between every pair of adjacent

variables X_i and X_j after explaining away the effects from the remaining variables. CB methods have gained immense popularity due to their simplicity and scalability in high-dimensional settings. The most prominent example of a CB method is the PC algorithm[1][Spi00]. Several modifications of the PC algorithm have been developed over the years due to its attractive property of being consistent in sparse high-dimensional settings [KB07].

- **Search-and-score-based (SS)** methods adopt a greedy approach to construct a causal graph. These methods perform a series of operations that include edge additions, deletions and flips with the goal of optimizing a score function. The most popular example of a SS method is the Greedy Equivalence Search (GES). While the GES algorithm does not assume causal faithfulness, they do not scale well in high-dimensional settings.

- **Hybrid (CB+SS)** methods adopt the best of CB and SS approaches. In other words, hybrid methods utilize the scalability of CB methods to eliminate spurious associations and the greedy search technique of SS methods to orient the edges. The Max-Min Hill Climbing (MMHC) method has shown remarkable success in constructing causal graphs even in high-dimensions [Ali10a].

An interesting work in the field of causal discovery showed it was possible to recover the exact directed acyclic graph if the assumption of Gaussian distribution was relaxed. Allowing the presence of non-Gaussianity in the data made it possible to detect exact causal identities of the variables [Shi06; Hoy09].

- **Additive Noise Models (ANMs)** identify the direct causes of a variable by exploiting the asymmetry between the cause and the effect. For example, to infer the direction of causality between two variables X and Y, the idea is to fit a regression model $X \sim Y + \epsilon_X$, calculate the residual ϵ_X and test whether ϵ_X is independent of Y. If so, then Y is the direct cause of X and if not repeat this process by regressing Y against X.

1.1.2 Causal Effect Estimation

The intervention calculus was developed to simulate RCEs and estimate bounds on the causal effects [KK01; Pea95]. The biggest advantage of this approach is a significant reduction in the cost of performing RCEs especially in domains where such experiments are time consuming and expensive.

The Intervention calculus when DAG is Absent (IDA) method was developed and empirically validated to rank the factors based on their causal effects with respect to a target variable of interest [Maa09]. For example, a ranked list of genes can be obtained to prioritize gene knock-out experiments. IDA utilizes the PDAG output by a causal structure learning method such as the PC

[1]The algorithm is named after its inventors Peter Spirtes and Clark Glymour

algorithm and calculates causal effects from the set of Markov equivalent graphs (MEGs) using linear regression.

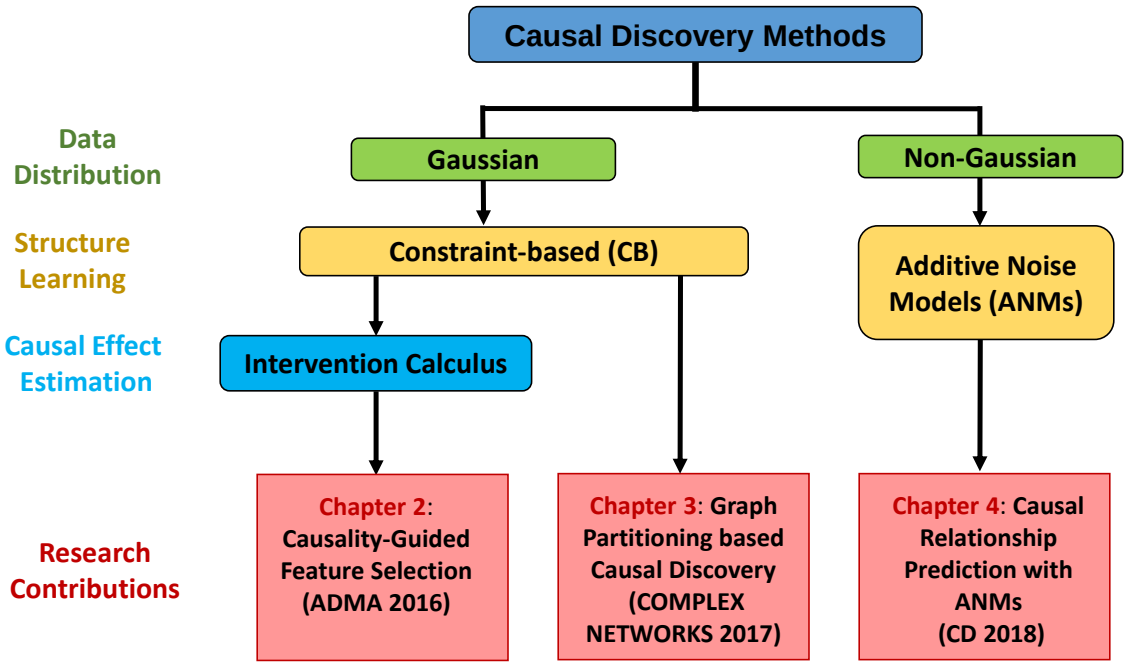


Figure 1.4 An overview of the dissertation chapters highlighting our contributions to the research objectives of effective causal discovery.

1.2 Methods for Learning and Predicting Causal Relationships

Given the rich set of causal discovery methods, identifying the causal structure in general is still a challenging task. The readily available observational data is accompanied by its own set of complexities such as high-dimensionality, missing ground-truth to validate causal relationships and smaller sample size. Therefore, there is a need for developing methods to effectively discover causal relationships. In this dissertation, we focus on developing solutions for the following research objectives,

1. Causal feature selection with constraint-based methods (Chapter 2)
2. Effective causal discovery with constraint-based methods (Chapter 3)

3. Effective causal discovery with additive noise models (Chapter 4)

Figure 1.4 presents an overview of the dissertation chapters. In Chapter 2, we present a novel feature selection method that incorporates the causal structure from a constraint-based method as well as the causal strength of a factor on a target variable. Chapter 3 improves a prominent constraint-based method by effectively eliminating spurious associations using a divide-and-conquer approach. Finally, in Chapter 4, we propose a feature engineering approach that estimates a directed acyclic causal graph by predicting the direct causes of each variable.

1.2.1 Causality-Guided Feature Selection

One of the main issues in building robust predictive models is to find the optimal feature set with maximum predictability. While there exist a plethora of association-based feature selection methods [Jov15], not all associations are causal in nature. Therefore, identifying causal features requires an additional effort of pruning non-causal association-based features. This problem is exacerbated with the additional complexities in the data such as smaller sample size and high-dimensionality. Nevertheless, the benefits of a causal feature selection method are twofold: first, they select a minimal feature set with optimum prediction performance, and second, causal features aid interpretability of the prediction model.

In Chapter 2 we develop a novel feature selection method that identifies causal features with respect to a target variable of interest. We develop an algorithm that prunes the existing feature set by incorporating the causal structure as well as the estimated causal effects of features on the target variable. We evaluate the prediction performance of the selected features from the data sets in the domains of climate science and biology, and compare it with state-of-art feature selection methods. Further, we evaluate the physical interpretability of the frequently selected causal features using domain knowledge. This work was published in [Cha16].

1.2.2 Graph Partitioning based Causal Discovery

Causal structure learning from the constraint-based PC-stable algorithm mainly depends on building conditioning sets which are used to eliminate non-causal associations. While some variations based on variable association have been developed to improve its performance [Spi00; Abe06], recent works based on recursive divide-and-conquer approach have shown promising results [XG08; Cai13; Liu17].

For this reason, in Chapter 3, we propose a recursive graph partitioning based algorithm by incorporating both variable association and divide-and-conquer strategies. The key idea is to partition a large graph into smaller subgraphs and restrict the task of performing conditional independence tests to a smaller subset of variables. In the process, conditional independence tests are performed

by building conditioning sets that leverage the graph topology and variable association. We evaluated our proposed algorithm on several real-world networks of variable sizes and complexities. The results obtained show a significant improvement in the quality of causal graphs without increasing the rate of false positives. These findings were published in [Cha17].

1.2.3 Causal Relationship Prediction with Additive Noise Models

As mentioned in section 1.1, causal discovery methods that assume multivariate Gaussian distribution in the observational data can recover the unknown causal structure in the form of a partially directed acyclic graph. On the other hand, by relaxing this assumption, additive noise models (ANMs) have been able to estimate the exact causal structure in the form of a directed acyclic graph.

Recent works have presented causal discovery from ANMs as a supervised learning problem [Fon16; LP15a]. The main idea is to learn patterns from causal relationships and non-causal relationships by engineering features. A binary classifier is trained using these features to predict causal relationships. However, these methods have been developed only for a bivariate setting. Motivated by the recent works, in Chapter 4, we propose a methodology for constructing a directed acyclic graph in a multivariate setting. To do this, we present a data partition approach that groups variables based on their causal influence with respect to each variable in the observational data. In the next step, we create new features from these groups using kernel embeddings and train a binary classifier to learn the patterns of causal and non-causal relationships. Finally, we evaluate the predictive power of these features by comparing the quality of causal structures with the state-of-art methods. The results obtained show an improvement in the quality of causal graphs, especially in observational data generated from non-linear structural equation models.

CHAPTER

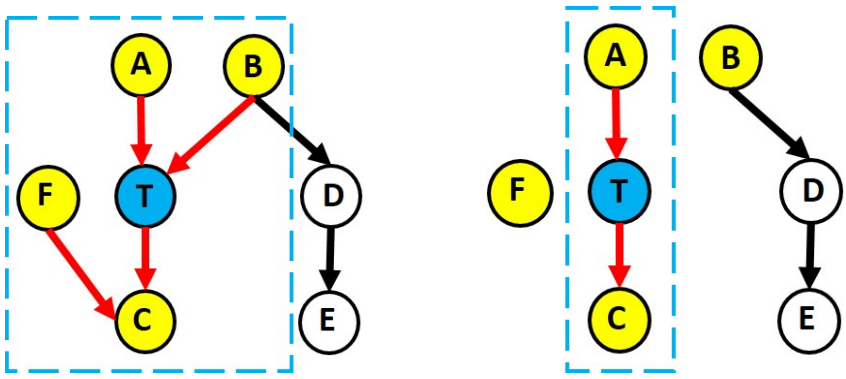
2

CAUSALITY-GUIDED FEATURE SELECTION

2.1 Introduction

Complex systems, such as the climate and biological systems, are characterized by an intricate interconnected network of interacting factors. These interactions often represent causal relationships among the factors (predictors), as well as between the factors and a phenomenon of interest (response). For example, in the climate science domain, factors may represent large scale ocean and atmospheric patterns, summarized as time series called *climate indices*, whose interactions are known to influence extreme weather phenomena, such as droughts and floods [And04; BL11]. Similarly, in biology, factors may represent genes, the expression levels of which have been found to have an effect on a phenotype of interest, such as disease status [Maa10]. Identifying meaningful factors in these complex systems that can be used to predict the response is a challenging task.

Traditionally, causality-driven methods have been applied to investigate causal relationships between variables. In the climate science domain, they have been used to construct causal graphs (Definition 2.2) that capture interactions among climate indices [EUD12; EUD14]. The relationships found in these causal graphs are further studied to confirm existing hypotheses and, if possible, generate new ones. On the other hand, in biology, cause-effect relationships are established by performing randomized gene knock-out experiments. Estimating bounds on the potential causal



(a) True Markov Blanket **(b)** Estimated Markov Blanket

Figure 2.1 2.1a The Markov Blanket of a target variable T consists of its direct causes (A, B), direct effects (C) and spouses (F). T is independent of other variable given its Markov Blanket [Ali10a]. 2.1b An estimated Markov Blanket might be incomplete due to errors made by statistical tests, thereby providing suboptimal prediction performance. In this case, the edges B → T and F → C were erroneously removed thereby excluding them from the Markov Blanket

effects of genes on a phenotype has been helpful for prioritizing such experiments [Maa10; Maa09]. However, these approaches require further domain expertise to validate the results and do not focus on identifying predictive factors for any specific response.

Recently the concept of constructing the local causal structure of a response variable for predictive modeling has gained lot of attention [Tsa03; Peñ07; Ali10a]. In other words, estimating the Markov Blanket (MB) of a response variable has shown to include all the relevant features that contain maximum predictive information about the response variable. Figure 2.1a shows an example of the MB of a target variable T, which includes its direct causes (A, B), effects (C) and spouses (F). T is independent of every other variable in the graph given its MB. Therefore, variables D and E do not provide additional information about T and hence they can be excluded while predicting T.

An extensive empirical study was conducted to compare the prediction models built using the features in the MB of a response variable with the features selected by state-of-art feature selection methods [Ali10b]. It was shown that MB-based methods provide a significant reduction in the features as well as improved prediction performance [Ali10a; Ali10b]. However, MB-based feature selection methods rely only on the estimated structure of the causal graph which can be incomplete due to erroneously removed edges. Figure 2.1 shows an example of this. Figure 2.1a shows the true MB of T and figure 2.1b shows the estimated MB. Notice that the edges $B \rightarrow T$ and $F \rightarrow C$ are missing compared to the true MB. Such estimation errors might occur due to noise in observational data which leads to errors in statistical tests.

To the best of our knowledge, there has been no work that tries to recover such missing edges. In this work we have made an attempt to identify features that might not appear in the MB of the target

variable but can be considered as potential candidates for feature selection. Therefore, we introduce the problem of *causality-guided feature selection*, that incorporates the causal structure learning and causal effect estimation to select features causally associated with a target variable. To do this, we construct causal graphs using a well-known constraint-based structure learning algorithm, such as PC-stable [CM14] and leverage causal relationships in this graph to estimate the causal effect of a predictor on the response. We assess the stability of each predictor by performing a random permutation test, to obtain a set of predictors having statistically significant causal effect on the response. In the end, we cluster the predictors and select features from each cluster with the most significant causal effect to form the new feature space.

Finally, we validate our proposed methodology on two motivational use cases in the domains of climate science and biology. Specifically, in climate science, we apply this methodology to select features for predicting seasonal rainfall in the regions of African Sahel and East Africa, and in biology, for predicting riboflavin production rate in bacterium *B. Subtilis* and the cognitive score measured using the Mini-Mental State Exam in male and female patients respectively. In climate science, the African Sahel region has been studied extensively following a series of severe droughts in the 1970s and 1980s [BL11]. Repeated droughts throughout the 2000s have led to a humanitarian crisis in the region, with approximately 10.3 million food-insecure people in 2013 [1]. East Africa is a similarly vulnerable region, including within it Lake Victoria, a mostly precipitation-fed resource for millions of people. In biology, identifying genes having significant causal effect on a phenotype of interest such as the riboflavin (vitamin B_2) production rate is a challenging task [Büh14; Maa09]. Another important task is to identify biomarkers that can be used to detect the phase of mild cognitive impairment (MCI) which is preceded by Alzheimer's disease (AD) among individuals. Currently, no biomarkers have been validated for predicting the risk of AD, and hence there is a greater need to discover key biomarkers [Che12].

In addition to evaluating the prediction performance of the selected features, we also validate the physical interpretability of the frequently selected features from climate data sets. To do this, we selected the climate indices that were used as features in more than 50% of the cross-validation folds and compared them with the features chosen by existing MB-based feature selection methods.

2.2 Problem Statement

Let $X = \{X_1, X_2, .., X_p, Y\}$ be a set of variables consisting of p predictors, $\{X_1, X_2, .., X_p\}$, and a response, Y. For example, for our use case in the climate science domain, X may be a set of p climate indices and Y may be seasonal rainfall at a target region (e.g., the African Sahel or East Africa). Informally, we define *causality-guided feature selection* as the task of selecting features based on the

[1] `http://www.fao.org/fileadmin/user_upload/emergencies/docs/SITUATION%20UPDATE%20Sahel%201%2007%202013.pdf`

potential causal relationships among the variables in X, with the goal of improving the prediction of Y. To do this, we first introduce the concepts of *causal graph* and *causal effect*.

Definition (CAUSAL GRAPH) Given a set of variables X, a *causal graph* $\mathbb{G} = (V, E)$ is defined as a graph where $V = X$ is the set of nodes and E is the set of edges, such that each directed edge, $X_i \rightarrow X_j$, represents a *potential causal relationship* where X_i is a *potential cause* of X_j, and each undirected edge, $X_i - X_j$, or bidirected edge, $X_i \longleftrightarrow X_j$, represents an *ambiguous relationship* between X_i and X_j.

Definition (CAUSAL EFFECT) Given a predictor X_i and a response Y in a causal graph $\mathbb{G}$, the *causal effect* of X_i on Y is defined as the change in Y for a unit change in X_i. Then, the estimated causal effect of X_i on Y is given by the regression coefficient of X_i, θ_i, when Y is regressed on X_i and its parents S_i; that is,

$$Y = \theta_i X_i + \theta_{S_i}^{\top} S_i + \epsilon_i \tag{2.1}$$

where ϵ_i is the residual of Y. The parent set of X_i are the variables that have a directed edge towards X_i in the causal graph $\mathbb{G}$.

Finally, we formally define the problem of *causality-driven feature selection*: Given a set of variables X consisting of p predictors and a response Y, a causal graph $\mathbb{G}$, and a set of causal effects of the predictors in X on Y, cluster the predictors in X based on their causal effect, and select predictors (i.e., features) from each cluster with the most statistically significant causal effect on the response.

2.3 Method

In this section, we describe our causality-guided feature selection methodology, as outlined in Algorithm 1. First, we use a constraint-based learning algorithm to construct a causal graph and select the potential causal relationships in the graph (Section 2.3.1). Second, for each predictor that takes part in a potential causal relationship, we compute its causal effect on the response using a methodology that addresses multicollinearity and then estimate the significance of this causal effect by performing a random permutation test (Section 2.3.2). Finally, we cluster the predictors based on their causal effect and select features from each cluster with the most statistically significant causal effect on the response (Section 2.3.3).

2.3.1 Constructing Causal Graphs and Selecting Potential Causal Relationships

We construct a graph of potential causal relationships using a constraint-based structure learning algorithm. Specifically, we use the PC-stable algorithm because of its ability to construct graphs with order-independent adjacency structure and to mitigate the effect of false positives edges [CM14]. In the next two paragraphs we present a summary of the constraint-based structure learning algorithm.

Algorithm 1 Causality-Guided Feature Selection

Require: A set of variables $X = \{X_1, X_2, .., X_p, X_{p+1}\}$ consisting of p predictors, $\{X_1, X_2, ..., X_p\}$, and a response, $Y = X_{p+1}$

Ensure: A set of causal features f_{new}

1: Let $f_{new} = \emptyset$ be the new feature space
2: Let $\mathbb{G}$ be a CPDAG constructed using the PC-stable algorithm (see Section 2.3.1)
3: Let C be the set of potential causal relations in $\mathcal{G}$
4: Let $\Theta = \emptyset$ be a set of statistically significant causal effects
5: Let Φ be a set of p-values of the statistically significant causal effects
6: **for each** $c = X_i \rightarrow X_j$ **in** C **do**
7: Let $\Theta_i = \emptyset$
8: Let G be the set of Markov equivalent graphs generated for X_i
9: **for each** $g \in G$ **do**
10: Let $\theta_{X_{i,1}}$ be the causal effect of X_i on Y computed with PCR using X_i and its set of parents S_i in g (see Section 2.3.2.1)
11: $[\theta_{X_{i,1}}, p\text{-}value_{\theta_{i,1}}]$=ASSESS_STABILITY($\theta_{X_{i,1}}$, X_i, S_i, Y)
12: **if** $p-value_{\theta_{i,1}} < 0.05$ **then**
13: $\Theta_i = \Theta_i \cup \theta_{X_{i,1}}$
14: **end if**
15: **end for**
16: **if** $\Theta_i \neq \emptyset$ **then**
17: $\theta_i = \arg\min_{\theta \in \Theta_i} |\theta|$
18: $\Theta = \Theta \cup \theta_i$
19: $\Phi = \Phi \cup p\text{-}value_{\theta_i}$
20: **end if**
21: Repeat steps 7-18 for X_j and Θ_j if $X_j \neq Y$
22: **end for**
23: f_{new} =FEATURE_SELECTION(Θ, Φ, X)
24: **return** f_{new}

Algorithm 2 ASSESS_STABILITY

Require: A causal effect, $\theta_{X_{i,1}}$, a predictor, X_i, its parents S_i, and a response Y

Ensure: Estimated causal effect and its p-value

1: Let N=100
2: Let Θ_{rand} be a set of randomized causal effects computed from N random permutations of the response (see Section 2.3.2.1)
3: $p\text{-}value = \frac{\left|\{\theta'_m \in \Theta_{rand} \text{ s.t. } |\theta'_m| \geq |\theta_{X_{i,1}}|\}\right| + 1}{N+1}$
4: **return** $[\theta_{X_{i,1}}, p\text{-}value]$

In the first step, the PC-stable algorithm constructs a completed undirected graph G over a set of variables X and initializes the size of the conditioning set, m, to zero. Next, for each variable

Algorithm 3 FEATURE_SELECTION

Require: A set of statistically significant causal effects, Θ, its corresponding set of p-values Φ, and set of predictors in X.

Ensure: A reduced set of features f_{new}

1: Perform K-Means clustering on Θ by identifying the optimal number of clusters, k, using the elbow method
2: **for each** cluster c_i **do**
3: Let $\phi \subset \Phi$ be the p-values of statistically significant causal effects in c_i
4: Select all predictors, $p_i \subset X$, in c_i whose causal effect has p-value equal to $min(\phi)$
5: $f_{new} = f_{new} \cup p_i$
6: **end for**
7: **return** f_{new}

$X_i \in X$, it stores the nodes adjacent to X_i in its adjacency set $a(X_i)$. For every pair of adjacent variables X_i and X_j in G, it checks whether the two variables are independent conditioned on S (i.e., $X_i \perp X_j | S$), such that $S \subseteq a(X_i)$ or $S \subseteq a(X_j)$ and $|S| = m$. If the variables are conditionally independent, the edge between them is removed from G and the conditioning variable(s), S, is stored in their separating set, $sepset(X_i, X_j)$ and $sepset(X_j, X_i)$. Similarly, the remaining pairs of adjacent variables are checked for conditional independence. This completes the first iteration of the conditional independence test. The adjacency set is updated for every variable, and the value of the conditioning set, m, is incremented by 1 in the next iteration. At the end of this step, the algorithm yields a *skeleton* of the causal graph, which contains undirected edges between variables that were not found to be conditionally independent. In our experiments, we use the Fisher's Z test to determine if two variables are conditionally independent at a significance threshold $\alpha = 0.05$.

In the next step, for every unshielded triple $(X_i - X_j - X_k)$ such that X_i and X_k are not adjacent, the algorithm orients $X_i - X_j - X_k$ into a v-structure, $X_i \rightarrow X_j \leftarrow X_k$, if and only if $X_j \notin sepset(X_i, X_k)$. Then the algorithm tries to orient as many remaining edges as possible using the following set of rules:

- Rule 1: Given $X_i \rightarrow X_j$ and $X_j - X_k$, orient $X_j - X_k$ to $X_j \rightarrow X_k$ such that X_i and X_k are not adjacent.
- Rule 2: Given $X_i - X_k$ and a chain $X_i \rightarrow X_j \rightarrow X_k$, orient $X_i - X_k$ into $X_i \rightarrow X_k$.
- Rule 3: Given two chains, $X_i - X_j \rightarrow X_k$ and $X_i - X_l \rightarrow X_k$, orient $X_i - X_k$ into $X_i \rightarrow X_k$.

The output of the PC-stable algorithm is a completed partially directed acyclic graph (CPDAG), which represents an approximation of the Markov equivalence class of the data. An estimated CPDAG can contain directed, undirected and bidirected edges, as shown in Figure 2.2. A directed edge $X_1 \rightarrow Y$ represents a cause-effect relationship where X_1 is a potential cause of Y, an undirected edge

$X_1 - X_2$ implies some association between the variables, and a bidirected edge $X_2 \longleftrightarrow X_3$ represents a sampling error or a hidden common cause of X_2 and X_3 that is not present in the data.

2.3.1.1 Assumptions

In order to interpret a CPDAG causally, we consider the following three assumptions:

- The underlying causal structure is sparse and acyclic.
- The graph built by the PC-stable algorithm is a well approximated representation of the underlying probability distribution in the data (i.e., causal faithfulness).
- There is absence of confounding variables in the data (i.e., causal sufficiency); i.e., given any two variables X and Y in the data having a common cause Z, then Z is also present in the data.

In real-world scenarios, it is difficult to assume that a system is causally sufficient, since there might be factors that are difficult to observe and measure. As a result, we cannot prove the existence of the potential causal relationships in the CPDAG. Nonetheless, these can provide an approximation of the underlying interactions between the variables. There are causal inference algorithms such as the Fast Causal Inference (FCI) algorithm and the Really Fast Causal Inference (RFCI) that detect the presence of hidden causes [Spi00].For future work, we plan to explore and determine the applicability of such causal inference algorithms to our proposed methodology.

Once the CPDAG is constructed using the PC-stable algorithm, we select all the directed edges in the graph for further processing because they represent potential causal relationships between pairs of variables. Moreover, these directed edges are present across all the Markov equivalent DAGs that can be generated from the estimated CPDAG. Figure 2.2 shows a CPDAG and a set of Markov equivalent DAGs (Figures 2.3b to 2.3d) generated from the CPDAG by orienting all the undirected and bidirected edges into directed edges such that no additional v-structure or cycle is created.

While in this work we used the PC-stable algorithm to construct the causal graph, any algorithm can be used in its place. We believe using more sophisticated causal structure learning methods using greedy or hybrid approaches can potentially improve the quality of estimated causal graphs. Consequently, with increase in the number of directed edges in the graph, our method can benefit from improved search space of predictors.

2.3.2 Estimating Causal Effects and Assessing its Statistical Significance

We estimate the causal effect on the response of every predictor that takes part in a potential causal relationship. For example, consider a potential causal relationship $X_i \rightarrow X_j$, where an external intervention on predictor X_i leads to a unit change in X_i. To estimate the causal effect of X_i on Y,

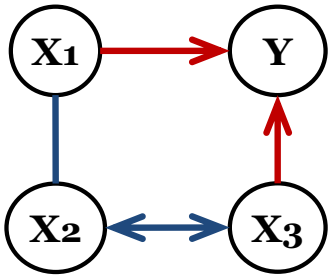
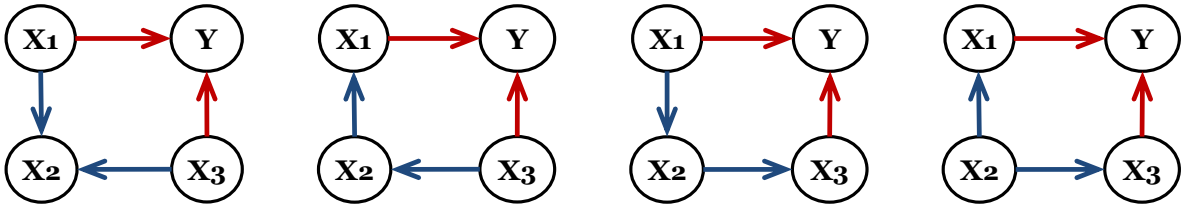


Figure 2.2 A completed partially directed acyclic graph (CPDAG) over a set of variables $X = \{X_1, X_2, X_3, Y\}$ comprising of two directed edges, $X_1 \rightarrow Y$ and $X_3 \rightarrow Y$, one undirected edge $X_1 - X_2$ and one bi-directed edge $X_2 \longleftrightarrow X_3$.



(a) D1 **(b)** D2 **(c)** D3 **(d)** D4

Figure 2.3 Figures (a)-(d) represent the set of possible DAGs D1-D4. D1 is an invalid DAG as it contains an additional v-structure, $X_1 \rightarrow X_2 \leftarrow X_3$, that is not present in the estimated CPDAG. D2-D4 are valid DAGs and they belong to the same Markov equivalence class.

the Intervention calculus when DAG is Absent (IDA) method has been proposed, which explores the local neighborhood of X_i by changing every undirected and bidirected edge incident on X_i to a directed edge [Maa09]. Thus, if there is a total of m undirected and bidirected edges incident on X_i, then up to 2^m Markov equivalent graphs can be generated. For each graph, the IDA method identifies the parents of X_i (i.e., predictors having a directed edge towards X_i). The response Y is then regressed on X_i and its parents, $S_i \subseteq \{X_1, X_2, ..., X_p, Y\} \setminus X_i$; that is,

$$Y = \theta_i X_i + \theta_{S_i}^{\top} S_i + \epsilon_i \tag{2.2}$$

where the regression coefficient of X_i, θ_i, is the estimated causal effect of X_i on Y for the corresponding Markov equivalent graph and ϵ_i is the residual of Y.

Thus, the IDA method yields a set of causal effects for X_i, $\Theta_i = \{\theta_i^1, \theta_i^2, ..., \theta_i^k\}$, across $k \leq 2^m$ Markov equivalent DAGs. The causal effect θ_i is 0 if $Y \in S_i$, since no change in X_i can have an effect on its parents; otherwise, θ_i is estimated as shown above.

For several real-world applications, such as our use cases in the domains of climate science and biology, the data exhibits a high degree of multicollinearity; that is, near-linear dependencies among the predictors. We determine the presence of multicollinearity by computing the variance

inflation factor (VIF); for example, for our climate data set, a high VIF (i.e., greater than 10) indicates a high degree of multicollinearity in the data [Net96]. The presence of multicollinearity leads to a poor estimation of the coefficients in the linear regression models in Equation 2.2 [Mon15]. To address this issue, we perform Principal Component Regression (PCR), instead of linear regression, to estimate the causal effects.

2.3.2.1 Principal Component Regression

Given a Markov equivalent graph, let $\mathbb{X}_i$ be a matrix whose columns consist of predictor X_i and its parents (i.e., the predictors in S_i). Then, Principal Component Regression (PCR) computes n principal components of $\mathbb{X}_i$, where n is less than or equal to the number of columns of $\mathbb{X}_i$, using singular value decomposition (SVD) as follows:

$$\mathbb{X}_i = T_i {P_i}^\top + \epsilon_{\mathscr{X}_i} \tag{2.3}$$

where T_i is the score matrix, P_i is the loading matrix and $\epsilon_{\mathbb{X}_i}$ is the unexplained variance of $\mathbb{X}_i$. Next, PCR builds a regression model with Y as the response and the score matrix T_i as the predictors; that is,

$$Y = \theta_{T_i} T_i + \epsilon_{T_i} \tag{2.4}$$

We solve for the regression coefficients θ_{T_i} of the score matrix T_i using the least squares method as follows:

$$\theta_{T_i} = (T_i^\top T_i)^{-1} T_i^\top Y \tag{2.5}$$

The regression coefficients of the score matrix T_i, θ_{T_i}, are then used to compute the regression coefficients of the original predictor matrix $\mathbb{X}_i$ as follows:

$$\theta_{\mathbb{X}_i} = P_i \theta_{T_i} \tag{2.6}$$

where $\theta_{\mathbb{X}_i}$ contains the regression coefficients of X_i and of the predictors in S_i across n principal components. We then select, as the causal effect of X_i on Y, the regression coefficient of X_i that corresponds to the first principal component, $\theta_{X_{i,1}}$, since it captures the maximum variance of $\mathscr{X}_i$.

2.3.2.2 Permutation Test

We assess the significance of a causal effect $\theta_{X_{i,1}}$ by performing the statistical test described in Algorithm 2. We randomize the response, Y, $N = 100$ times and compute the corresponding set of randomized causal effects $\Theta_{rand} = \{\theta_1^{'}, \theta_2^{'}, ..., \theta_N^{'}\}$ using Equation 2.6. Permuting the response randomly breaks any causal association between the response and the predictor. A p-value is calculated to measure the probability that the magnitude of a randomized causal effect is greater than

or equal to the magnitude of $\theta_{X_{i,1}}$. The causal effect $\theta_{X_{i,1}}$ is considered to be statistically significant if this p-value is less than 0.05.

This test is performed for each causal effect of X_i on Y computed across all the Markov equivalent graphs. The result is a set of statistically significant causal effects of X_i on Y, $\Theta_i = \{\theta^1_{X_{i,1}}, \theta^2_{X_{i,1}}, ..., \theta^l_{X_{i,1}}\}$, where $l \leq k$. From a set of statistically significant causal effects, Θ_i, we select θ_i, such that it represents the minimum change in response Y for a unit change in X_i. This is illustrated by the equation below,

$$\theta_i = \arg\min_{\theta \in \Theta_i} |\theta| \tag{2.7}$$

Similarly, we compute the set of statistically significant causal effects of X_j on Y, Θ_j, for predictor X_j taking part in the potential causal relationship $X_i \rightarrow X_j$.

The causal effect θ_i represents the minimum change in the response Y for a unit change in X_i. We compute the causal effects θ_i for every predictor X_i that takes part in a potential causal relationship in the CPDAG. This results in a set of predictors having a statistically significant causal effect on the response.

2.3.3 Feature Selection via Clustering

Finally, we prune additional features, by clustering the predictors based on their causal effects using the k-means algorithm as described in Algorithm 3. The optimum number of clusters k was selected using the elbow method. From each cluster, we select the predictors with the most statistically significant causal effect on the response i.e., the ones with the lowest p-value from the statistical test (see Section 2.3.2.2). By doing so, we minimize the number of redundant predictors having similar causal effect on the response.

2.4 Empirical Evaluation

In this section, we describe the application of our proposed causality-guided feature selection methodology. Specifically, our goal is to select features that improve the prediction of 1) seasonal rainfall at the African Sahel and East Africa regions, 2) riboflavin production rate and 3) cognitive score of male and female patients.

2.4.1 Data Description

We used two real-world data sets from the climate science domain and three real-world data sets from the biology domain to demonstrate the performance of our methodology.

Climate Science: Climate scientists are often faced with having to use datasets that are predom-

inantly large in scale. For example, the University of Delaware Air Temperature & Precipitation [2] contains gridded data over a 0.5°×0.5° grid, resulting in 259,200 data points to analyze. As such, climate scientists leverage climate indices as a way to quantify the characteristic of a geophysical system occurring in nature, which is measured using a single variable (e.g. precipitation, temperature, pressure etc.) or a combination of variables (e.g. temperature and wind). Climate indices across different geographical locations, like the climate factors that they are abstractions of, have been observed to interact with one another, these communications drive the climatic conditions of a region. For example, rainfall anomalies in the Sahel and East Africa have been linked to different climate indices, locally and globally. This gives us an opportunity to explore the causal connections affecting rainfall in the region, and leverage them to improve the forecast of seasonal rainfall. We present a description of the datasets and the preprocessing steps involved to prepare the experimental data.

Our predictor set consists of 30 climate indices, most of which are made available by NOAA ESRL[3]. Five additional indices were constructed to incorporate regional local atmospheric conditions, following established conventions for index construction. Details of the techniques employed to construct additional indices as well as data origin are described further in Table 2.1. A total of 35 climate indices are collected over a period of N=57 years from 1951 to 2007. The rainfall data for both the regions is the Gridded Precipitation Climatology Centre (GPCC) V6, gridded at 0.5×0.5 degree resolution. A box average over the geographical coordinates of the Sahel 10°N-20°N, 20°W-35°E and East Africa 13°N-13°S, 24°E-55°E was constructed to build the rainfall data across the regions. The rainfall season in the Sahel has been observed during the months of Jul-Aug-Sep (JAS) and during the months of Oct-Nov-Dec (OND) in East Africa. For the Sahel region, the monthly values of 34 climate indices are taken from Jan-Jun in a given year and for East Africa region, the monthly values of 33 climate indices are taken from Jan-Sep. We consider two principal components of the LLW climate index for the Sahel region based on our domain knowledge. The resulting datasets have dimensions 57ÃŮ(210 + 3) and 57ÃŮ(297 + 3) for the Sahel and East Africa regions respectively, where the additional 3 variables represent rainfall variables for the three months of the rainfall season.

Biology: For our first data set, we used publicly available data[4] containing 71 observations of 101 variables measuring the logarithm of the expression level of 100 genes and the riboflavin production rate (RPR) in the bacterium *B. subtilis* [Büh14]. Our second and third data sets were collected from the Alzheimer's Disease Neuroimaging Initiative (ADNI). The data sets contained microarray gene expression data collected from 266 male patients and 219 female patients. The cognitive score of the subjects in each gender was used as the response variable, i.e., CS_M for male patients and CS_F for female patients.

[2] http://www.esrl.noaa.gov/psd/data/gridded/data.UDel_AirT_Precip.html

[3] http://www.esrl.noaa.gov/psd/data/climateindices/list

[4] http://www.annualreviews.org/doi/suppl/10.1146/annurev-statistics-022513-115545/suppl_file/riboflavinv100.csv

Table 2.1 Climate Indices constructed from Reanalysis data to include local variability.

Feature represented	Region	Abbr.	Location	Variable	Methodology
Mediterranean Sea	Sahel/ East Africa	MSEA	30°N-46°N, 6°W-36°E	SST anomaly (subtract 1979-2000 average)	Box average [Row03].
Indian Ocean	Sahel/ East Africa	IOD	10°S-10°N, 50°E-70°E and 10°S - 0°S, 90°E - 110°E	SST anomaly (subtract 1979-2000 average)	Box average, difference between the two locations [Saj99].
Local variation in atmospheric pressure	Sahel	GHT	0°N-40°N, 40°W-30°E	Geopotential Height at 850mb	First mode of Principal Component Analysis.
Local variation in atmospheric winds	Sahel	LLW	0°N-20°N, 60°W-25°E	Zonal (u) wind at 850mb	First and Second mode of Principal Component Analysis [NG01].
Somali Jet	East Africa	VWSI	7.5°S-7.5°N, 37.5°E-45°E	Meridional (v) wind at 850mb	Box average [CP02].

2.4.2 Data Preprocessing

The following pre-processing steps were performed after dividing the data into training and test sets using leave-one-out cross-validation (LOOCV),

1. **De-trending**: Given the temporal nature of the climate data sets, the predictors and the response variable were detrended to remove seasonal trends. This step was performed only on climate data sets.

2. **Normalization**: The predictors and the response variable were standardized using their z-scores. Note that the z-scores were computed using the average and the standard deviation from the training sets.

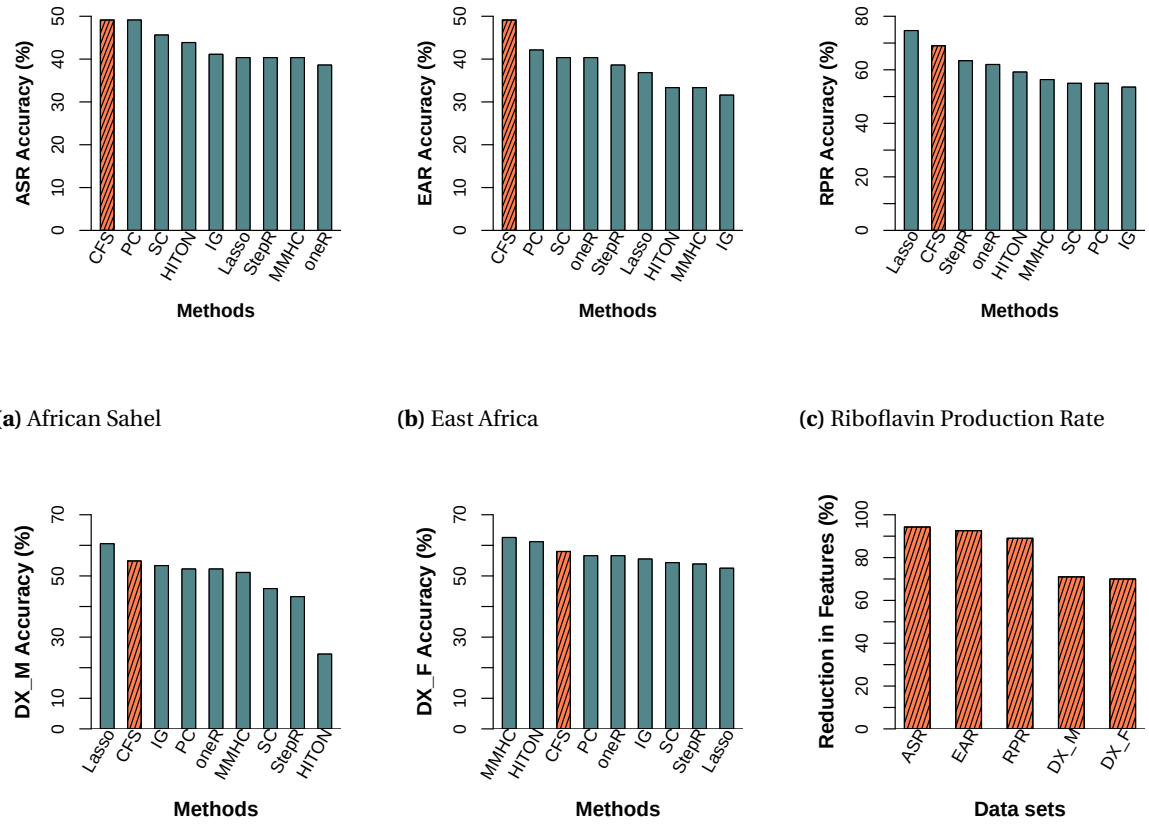
Due to the indefinite amount of time taken by the PC-stable algorithm for constructing causal graphs from the microarray gene expression data, we used the training set to select top-100 genes correlated with the response as predictors.

2.4.3 Temporal Constraints

While constructing causal graphs from climate data sets, we explicitly specify certain constraints to obtain temporally coherent causal relationships. Since each predictor contains the monthly value of a climate index, a causal relationship between monthly values of the same climate index can only exist if cause precedes the effect. For example, we would observe the causal relationship

$AMM\_1 \rightarrow AMM\_2$ where AMM_1 is the value of the climate index in the month of January and AMM_2 is the value in the month of February. On the other hand, we can observe causal relationships between two different climate indices in the same month. For example the climate index AMO can effect AMM in the same month or in different months.

2.4.4 Performance Comparison



(a) African Sahel **(b)** East Africa **(c)** Riboflavin Production Rate

(d) Cognitive Score in Male **(e)** Cognitive Score in Female **(f)** Percentage Feature Reduction

Figure 2.4 Mean classification accuracy over leave-one-out cross validation (Accuracy) for the African Sahel (2.4a), East Africa (2.4b), *B. subtilis* (2.4c), diagnostic information for Male (2.4d) and Female (2.4e) obtained from Causality-Guided Feature Selection (CFS), Max-Min Hill Climbing (MMHC), HITON Markov Blanket (HITON) Spearman Correlation (SCorr), Pearson Correlation (PCorr), oneR (oneR), Lasso Regression (Lasso), Stepwise Regression (StepR), and Information Gain (IG). 2.4f shows the percentage reduction in the features using CFS across five data sets.

Table 2.2 RMSE scores for prediction of seasonal rainfall at African Sahel (ASR) and East Africa (EAR), riboflavin production rate (RPR) and cognitive score for male (CS_M) and female (CS_F) patients obtained from all the feature selection methods. RMSE scores within 6% of the best performing method are highlighted in bold. (*) indicates that a feature selection method could not find any feature from some of the training sets during cross-validation

Data set	Feature Selection Methods								
	CFS	PCorr	SCorr	oneR	StepR	IG	Lasso	MMHC	HITON
ASR	**0.9787**	1.1499	1.0942	**0.9985**	1.7362	**0.9978**	1.2304	*	*
EAR	0.9860	1.0720	1.0448	1.3769	1.3885	1.3751	3.5515	**0.7599**	**0.7655**
RPR	**0.5483**	**0.6112**	**0.5335**	0.8288	1.0956	0.6077	0.6432	*	0.7114
CS_M	**1.0576**	1.1976	1.1805	1.1939	1.134	1.2086	1.2039	**1.0477**	**1.0148**
CS_F	**1.1079**	**1.0646**	1.1285	1.1337	1.1859	1.1321	1.2397	**1.1033**	**1.0996**

We evaluate the performance of our causality-guided feature selection methodology by training classification models using C5.0 decision trees and regression models using the linear regression method. The response variables for ASR, EAR and RPR data sets were discretized into three categories: high, normal or low (i.e., values in the higher 66.7^{th} percentile, between the lower 33.3^{rd} and the higher 66.7^{th} percentile, and in the lower 33.3^{rd} percentile, respectively). For the two ADNI data sets, CS_M and CS_F, the cognitive score collected from the Mini-Mental State Exam (MMSE) was used as the continuous response variable. Additionally, there were two groups of patients within each gender based on the diagnostic information made available by ADNI. The first group comprised of controls (i.e., patients who did not suffer from Alzheimer's), and the second group comprised of patients suffering from late Mild Cognitive Impairment (LMCI). This diagnostic information for males and females was used as the discretized response variable in both the data sets. The decision trees were trained on these five data sets using the corresponding discretized response variables. The regression models were built using the z-scores of the response variables.

2.4.4.1 Prediction Performance

The performance of our proposed methodology was compared against eight feature selection methods, including four univariate methods namely, Spearman correlation (SCorr), Pearson correlation (PCorr), one rule classifier (oneR) and information gain (IG), two regression methods namely, Lasso Regression (Lasso) and Stepwise Regression (StepR), and two local causal discovery-based methods namely, HITON and Max-Min Hill Climbing (MMHC). The univariate methods require an input K which specifies the top-K predictive features to be chosen for building the prediction model. We specified the input K to be the average number of features chosen by our method across all cross-validation folds. All the above mentioned methods were used to select features which were

then used to train classifiers and regression models.

The classification accuracies of the feature selection methods on five real-world data sets are shown in Figure 2.4a-2.4e. We observe that our proposed methodology is the best performing method for ASR and EAR, and its accuracy is within 6% of the best performing method for RPR, CS_-M and CS_F. While Lasso outperforms all the methods on RPR and CS_M, it is the worst performing method for CS_F and our methodology has a minimum improvement of 8% over Lasso for ASR and EAR. Similarly, MMHC and HITON have higher accuracy values while predicting CS_F, but their performance is as good as random guessing for EAR, and HITON is the worst performing method for CS_M.

We now compare the root mean squared errors (RMSE) for each method to evaluate the performance of the selected features on the regression models. Tables 2.2 presents our findings that four out of five times the RMSE score of our proposed methodology is within 6% of the best performing method. The remaining methods, however, don't show the same level of consistency in terms of the results obtained. For example, Pearson Correlation was among the top performing methods in terms of classification accuracy for ASR and EAR, but its performance is relatively poor when evaluating regression models. Furthermore, although Lasso has higher accuracy values for RPR and CS_M (see Figure 2.4c-2.4d), it is the worst performing method for EAR and CS_F. While MMHC and HITON have high accuracy values for CS_F, they had the worst RMSE scores for ASR as they could not find any feature from some of the training sets during cross-validation.

A summary of the results is shown in Table 2.3. We observe that the classification accuracy and RMSE score of our methodology is consistently within 6% of the best performing method. Note that while methods such as Pearson Correlation, Lasso, MMHC and HITON perform well on few data sets they are also the worst performing methods or within 6% of the worst performing method on other data sets. This indicates that their prediction performance varies significantly with data sets from different domains, whereas our methodology shows predictive skill across all of data sets in terms of classification accuracy and RMSE score. Moreover, we also observe a percentage reduction in the selected features by at least 70% as compared to the original feature space across all the data sets (see Figure 2.4f).

2.4.4.2 Confidence Intervals

We also calculated 95% confidence intervals of the classification accuracy and RMSE values for all the methods. Specifically, we used bootstrapping to generate sample means of classification accuracy and RMSE scores. One of the advantages of choosing bootstrap to generate confidence intervals is that we do not have to assume a normal distribution of the sample statistic.

We start by illustrating an example of computing confidence interval for the mean classification accuracy, μ. In our experiments, we calculated μ from leave-one-out cross-validation. For each data

set comprising of n observations, we obtain a boolean vector of size n containing the number of correct predictions where a 1 in this vector indicates our model correctly predicted the output class label. Next, we generate bootstrap samples of size n and compute the sample mean classification accuracy, $\overline{\mu}$. This process is repeated 1000 times to generate a sufficiently large distribution of sample means. From each bootstrap sample, we compute the difference $\delta = \overline{\mu} - \mu$, and we use these differences to calculate the bootstrap confidence interval for μ.

Confidence intervals for the RMSE scores of all the methods across different data sets were generated in a similar manner. Instead of the classification accuracy, RMSE score was calculated from the bootstrap samples. Tables 2.4 and 2.5 show the mean classification accuracy, RMSE values and the 95% confidence interval for each feature selection method across all the data sets.

$$\delta = \overline{\mu} - \mu \tag{2.8}$$

$$\mu - \delta_{0.05} < \mu < \mu + \delta_{0.95} \tag{2.9}$$

Table 2.3 A summary of the performance of all the feature selection methods in terms of classification accuracy and RMSE scores across all five data sets. A solid upper triangle indicates the best performing method, whereas a solid lower triangle indicates the worst performing method on a given data set. An upper triangle with pattern indicates that a method's performance was within 6% of the accuracy and RMSE score of the best performing method, and a lower triangle with pattern indicates a method's performance was within 6% of the accuracy and RMSE score of the worst performing method.

Performance Metrics	Data set	Feature Selection Methods								
		CFS	PCorr	SCorr	oneR	IG	StepR	Lasso	MMHC	HITON
Accuracy	ASR									
	EAR									
	RPR									
	CS_M									
	CS_F									
RMSE	ASR									
	EAR									
	RPR									
	CS_M									
	CS_F									

Table 2.4 Confidence interval of classification accuracy for prediction of seasonal rainfall at African Sahel (ASR) and East Africa (EAR), riboflavin production rate (RPR) and cognitive score for male (CS_M) and female (CS_F) patients obtained from all the feature selection methods. A threshold of 95% was used to measure the confidence intervals. A value highlighted in bold indicates the best performance in terms of classification accuracy on a given dataset and a value underlined indicates the smallest confidence interval width.

Feature Selection	Data set				
	Sahel	East Africa	Riboflavin	CS_M	CS_F
CFS	**49 [39,60]**	**49 [37,61]**	69 [61,79]	58 [53,63]	55 [50,60]
PCorr	**49 [39,60]**	42 [32,53]	55 [45,65]	57 [51,62]	52 [47,57]
SCorr	46 [35,56]	40 [30,51]	55 [45,65]	54 [49,60]	46 [41, 51]
oneR	42 [30,53]	32 [21,42]	55 [45,65]	56 [50,62]	53 [48,58]
IG	42 [32,53]	32 [21,40]	54 [44,65]	56 [51,60]	53 [48,59]
Lasso	40 [28,51]	37 [26,46]	**75 [66,83]**	53 [48,59]	**61 [56,66]**
StepR	40 [28,51]	39 [26,47]	63 [54,73]	54 [48,59]	43 [38,48]
MMHC	28 [18,37]	35 [25,46]	51 [41,61]	60 [55,66]	49 [45,55]
HITON	44 [32,54]	33 [23,44]	59 [51,69]	**61 [56,67]**	24 [20,29]

Table 2.5 Confidence interval of root mean squared errors (RMSE) for prediction of seasonal rainfall at African Sahel (ASR) and East Africa (EAR), riboflavin production rate (RPR) and cognitive score for male (CS_M) and female (CS_F) patients obtained from all the feature selection methods. A threshold of 95% was used to measure the confidence intervals. A (∗) indicates that RMSE value for a given method was not computed as it failed to select any features in at least one of the cross-validation folds. A value highlighted in bold indicates the best performance in terms of RMSE on a given dataset and a value underlined indicates the smallest confidence interval width.

Feature Selection	Data set				
	Sahel	East Africa	Riboflavin	CS_M	CS_F
CFS	**0.98 [0.8,1.1]**	0.99 [0.9,1.1]	0.55 [0.4,0.7]	1.11 [1,1.2]	1.06 [1, 1.1]
PCorr	1.15 [1,1.3]	1.1 [0.9,1.2]	0.61 [0.5,0.7]	1.13 [1.1,1.2]	1.2 [1.1,1.3]
SCorr	1.1 [0.9,1.3]	1.05 [0.9,1.2]	**0.53 [0.4,0.6]**	1.06 [1,1.2]	1.18 [1.1,1.3]
oneR	0.99 [0.8,1.2]	1.38 [1.1,1.6]	0.65 [0.5,0.8]	1.13 [1,1.2]	1.2 [1.1,1.3]
IG	0.99 [0.8,1.2]	1.38 [1.1, 1.6]	0.6 [0.5,0.7]	1.13 [1,1.2]	1.2 [1.1,1.3]
Lasso	1.2 [1,1.4]	3.6 [2.1,5.1]	0.64 [0.6,0.7]	1.24 [1.1,1.4]	1.2 [1.1,1.3]
StepR	1.7 [1,1.4]	1.4 [1.2,1.7]	1.1 [0.9,1.3]	1.19 [1.1,1.3]	1.13 [1.1,1.2]
MMHC	∗	**0.72 [0.6,0.8]**	∗	**1.04 [1,1.1]**	1.1 [1,1.2]
HITON	∗	0.76 [0.6,0.9]	0.71 [0.6,0.9]	1.1 [1,1.2]	**1.01 [1,1.1]**

where $\delta_{0.05}$ is the value at the 5^{th} percentile in δ. Similarly, $\delta_{0.95}$ is the value at the 95^{th} percentile.

Table 2.6 Mean classification accuracy and RMSE scores over leave-one-out cross-validation for the African Sahel (ASR), East African (EAR), riboflavin production rate (RPR), and cognitive score for male (CS_M) and female (CS_F) patients obtained using MMHC and MMHC+CFS as feature selection methods. A $*$ indicates that MMHC did not select any feature in at least one of the cross-validation folds.

Dataset	Accuracy		RMSE	
	MMHC	MMHC + CFS	MMHC	MMHC + CFS
ASR	28.1 [18,37]	56.1 [46,67]	$*$	0.95 [0.78,1.14]
EAR	35.1 [25,46]	40.4 [30,51]	0.72 [0.6,0.8]	1.1 [0.98,1.3]
RPR	50.7 [41,61]	66.2 [58,76]	$*$	0.64 [0.56,0.73]
CS_F	60.3 [55,66]	59.3 [54,65]	1.1 [1,1.2]	1.3 [1.2,1.4]
CS_M	49.2 [45,55]	61.6 [57,67]	1.04 [1,1.1]	1.2 [1.1,1.3]

2.4.5 Additional Experiments

We performed another set of experiments by replacing the PC-stable algorithm for constructing causal graphs (see Section 2.3.3) with a hybrid method called Max-Min Hill Climbing (MMHC) algorithm [Tsa06]. The rest of the steps in the method remain unchanged. We used the causal graph estimated by the MMHC algorithm and identified all the predictors that participated in at least one causal relationship. All the remaining steps mentioned in Sections 2.3.2-2.3.3 were followed to identify a reduced feature set. These features were then used for training classifiers and regression models.

MMHC combines constraint-based and search-and-score based approaches to estimate a causal graph. To summarize, MMHC estimates the skeleton of a causal graph using conditional independence tests, and orients the undirected edges in the skeleton such that the estimated causal structure optimizes a score function. While any causal structure learning algorithm can be used instead of PC-stable, we chose MMHC given its effectiveness in constructing causal graphs [Ali10a; Ali10b]. This demonstrates that our method can be easily adapted to fit any causal structure learning method.

Table 2.6 shows the prediction performance of MMHC and combination of MMHC and CFS (MMHC+CFS) in terms of classification accuracy and root mean squared errors. We observe an improvement of 5%-28% in terms of classification accuracy in four out of five datasets. In case of RMSE scores, MMHC could not identify any feature in the Markov Blanket of the target variable across ASR and RPR datasets. As a result, we could not compute its mean RMSE score. However, the combination MMHC+CFS can identify features across all the folds in the two datasets thereby providing an improvement over using only MMHC. On the other hand, for the remaining three datasets, the RMSE score of MMHC is better than MMHC+CFS. This suggests that the improvement in classification accuracy does not translate to an improvement in RMSE scores. We believe this could be an interesting avenue of future research work.

2.4.6 Physical Interpretability

While improving the prediction performance of the response is critical for any application, understanding the features that lead to this improved performance is also of importance to the domain scientists. In this work, we collaborated with climate scientists and they provided us with the domain knowledge to understand the selection of features from the climate data sets. We evaluated the physical interpretability of the causal relationships that were selected to identify features for predicting the seasonal rainfall in the African Sahel and the East African regions, respectively. Specifically, we selected frequent causal relationships (i.e., causal relationships that are observed in at least 50% of the cross-validation folds) such that both the climate indices in a relationship have a statistically significant causal effect (i.e. p-value < 0.05) on the seasonal rainfall.

We compared the frequently selected features identified by our method and two state-of-art local causal feature selection methods, HITON and MMHC. To do this, we identified features that were chosen in at least 50% of the cross-validation folds for building prediction models.

Figures 2.5 and 2.6 show a summary graphs that represent the frequently selected causal relationships for the African Sahel and East African regions, respectively. In Figures 2.5a and 2.6a, directed edges between different climate indices represent causal relationships in the same or different months. For example as shown in Figure 2.5, a directed edge from AMO to CAR represents a causal relationship where both AMO and CAR are in the same month or AMO is recorded in a month preceding the monthly value of CAR. A self-loop indicates a causal relationship between the monthly values of a climate index. However, a self-loop cannot occur between the same monthly values of the climate index.

The climate indices in solid line shapes were found during their predictive months (i.e., months when the climate indices are known to influence the response according to climate literature), while the climate indices in dotted line shapes were not found during their predictive months. As shown in Figure 2.5a, for the African Sahel region our method finds most of the indices during their predictive months except the causal influence from climate indices MSEA_SST and PACWARM. However, HITON and MMHC can identify only one climate index, AMO, to be causally associated with the seasonal rainfall in the region. This means that AMO was the only climate index present in the Markov Blanket of the response variable. While our method also captures AMO's effect on the seasonal rainfall, it identifies several other climate indices that are known to influence the seasonal rainfall.

The climate indices shown in dotted shapes require further investigation to validate their selection by our method. While at this point, we do not have enough domain knowledge to validate them, it is possible that they are false positives output by our method.

Similarly, in figure 2.6 we observe that our method captures more diverse climate indices known to affect the seasonal rainfall in the East African region compared to the features selected by HITON

and MMHC methods. These observations provide some promising evidence that our method is able to identify missing features that were eliminated by errors in statistical tests.

We studied the causal graphs output by the PC-stable, HITON and MMHC methods from the training sets in two different regions. We found that the climate index AMO had a directed edge towards the African Sahel seasonal rainfall in almost all the causal graphs and similarly, the climate index IOD had a directed edge towards the East African seasonal rainfall in majority of the causal graphs. However, several climate indices are known to affect the seasonal rainfall and therefore selecting AMO or IOD would not output the best prediction model or physically interpretable results. While HITON and MMHC select features from the causal structure, they were able to identify only AMO and IOD indices. On the other hand, our method benefits by finding relevant features using causal structure as well as causal effect estimation.

In addition to validating the selection of individual climate indices with respect to the response variable, our method offers another dimension that can be useful to the domain scientists. Recent studies in climate science have benefited from studying causal relationships in the CPDAG output by the PC algorithm [EUD12; EUD14]. Studying every causal relationship might become infeasible in high-dimensional settings. To address this issue, our method provides a list of prioritized causal relationships that can be studied by the domain scientists. As shown in figures 2.5a and 2.6a, the directed edges represent causal relationships between frequently selected climate indices. We posit that since almost all the frequent climate indices have been validated using domain knowledge, these relationships can be prioritized over the rest in the CPDAG.

We now present tables 2.7 and 2.8 that list the climate indices known to influence the seasonal rainfall in the two African regions along with the prominent predictive seasons found in the climate literature and the prominent season found by our method. In Table 2.7 we observe that our method identifies seven out of ten climate indices known to influence the seasonal rainfall in the African Sahel region. While the monthly values of climate indices such as ENSO, AMO, LLW1 and GHT align with the expected prominent seasons, our method also finds MSEA, PACWARM and LLW1 earlier than expected. Similarly, Table 2.8 presents the same information for the East African region. We observe that our method finds two of the three climate indices known to influence the seasonal rainfall in the region and a high overlap with the prominent predictive season.

Lastly, we also present a list of causal relationships with frequency of at least 50% in Tables 2.7 and 2.8. Each causal relationship is represented with a directed edge from the cause to the effect. Additionally, we annotate the climate indices with the months they were found in a cause and effect relationship. The months were labeled from 1 to 12 where 1 indicates the month of January and 12 indicates the month of December. For example, the directed edge LLW1_1→LLW1_4 indicates that the climate index LLW1 during the month of January had a causal influence on its value recorded in the month of April. These relationships represent a reduced search space of all the relationships that can exist between climate indices. Therefore, unlike existing studies which require each causal

association to be studied by the domain scientists, we present a list of prioritized causal relationships that can be studied to validate hypotheses.

For the African Sahel, we identified a potential causal relationship between the Atlantic Meridional Oscillation (AMO) index and the response. The influence of the AMO on the Sahel rainfall is well documented in climate literature, and has been attributed to the displacement of the Inter-tropical Convergence Zone [ZD06] and the African Easterly Jet [MT14]. Additionally, the AMO was also found to drive the Atlantic Tripole SST (ATLANTIC SST) and the tropical Caribbean SST (CAR) indices, which also have an impact on rainfall at the region. Positive (negative) Atlantic SST anomalies before the season are predictive of high (low) rainfall anomalies in the Sahel due to an increase in moisture from the Atlantic. The selection of features linking the northern and tropical Atlantic is therefore physically coherent. More locally, the atmospheric pressure's (GHT) influence on the local wind field (LLW) is observed just before the rainfall season. The selection of these features agrees well with the expectation that the position of the African Easterly Jet in uences Sahel rainfall [NG01]. In the Indo-Pacific, the only frequently selected feature observed is the PACWARM index. This is unexpected, given the previous work connecting the Indian Ocean variability to Sahel rainfall [BL03].

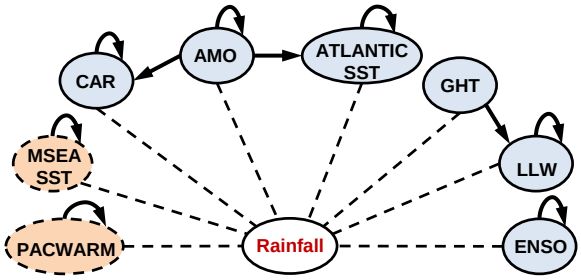
Similarly, for the East African region, we observe a potential causal relationship between the Indian Ocean Dipole (IOD) and the response just before the rainfall season. This is to be expected due to the strong easterly wind anomalies that weaken the flow of moisture out of East Africa [Bla03]. ENSO is also a major source of East African variability [Omo13], with strong ENSO events associated with higher than normal rainfall. This tropical Pacific variability is captured by our proposed methodology, with the ENSO indices observed to influence each other. Although the PACWARM index is not chosen, this may be due to the strong overlap with the IOD, causing the IOD to be selected preferentially.

Overall, the physical interpretability of the results obtained using our methodology is supported by its agreement with climate literature for both regions.

Analyzing the frequently discovered causal relationships has a twofold advantage. First, in contrast to previous studies it presents a reduced search space of causal relationships to be studied by the domain scientists. These relationships can be used to confirm existing hypothesis and develop new ones. Second, it allows visibility into our method which is based on selecting features having a causal influence on the response instead of mere association, thereby justifying the prediction performance of our method.

2.4.7 Time Complexity

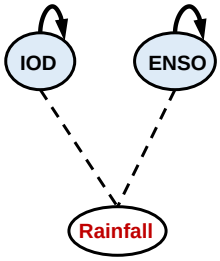
The majority of the time taken by the proposed methodology is due to constructing a causal graph, computing causal effects and their significance. The asymptotic time complexity of the PC-stable algorithm is polynomial time on sparse graphs, i.e., $O(p^q)$ where p is the number of variables



(a) Causality-Guided Feature Selection

(b) HITON and MMHC

Figure 2.5 A summary graph representing frequently selected causal relationships with both climate indices in a relationship having a statistically significant causal effect on the seasonal rainfall in the African Sahel region. A directed edge represents a causal relationship between climate indices in the same month or different months. A self-loop indicates a causal relationship between the same climate index in different months. A dotted line between a climate index and seasonal rainfall indicates the climate index has a statistically significant causal effect on the seasonal rainfall. Figure 2.5a shows the frequently selected features and the causal relationships between them. Figure 2.5b shows the frequently selected features by existing state-of-art local causal feature selection methods namely HITON and MMHC.



(a) Causality-Guided Feature Selection

(b) HITON and MMHC

Figure 2.6 A summary graph representing frequently selected causal relationships with both climate indices in a relationship having a statistically significant causal effect on the seasonal rainfall in the East African region. A directed edge represents a causal relationship between climate indices in the same month or different months. A self-loop indicates a causal relationship between the same climate index in different months.Figure 2.6a shows the frequently selected features and the causal relationships between them. Figure 2.6b shows the frequently selected features by existing state-of-art local causal feature selection methods namely HITON and MMHC.

in the graph and q is the maximum number of vertices adjacent to any vertex in the graph such that $q = O(n^{1-b})$ where n is the sample size and $0 < b \leq 1$. To estimate a causal effect, the time complexity of linear regression is dominated by matrix multiplication i.e., $O(p^2 n)$ if $n > p$ else

Table 2.7 A list of ground truth containing the prominent predictive season of the climate indices and their prominent season predicted by our method for the African Sahel region. A - indicates that our method did not find the corresponding climate index as a feature. The months are represented as, 1=Jan, 2=Feb, ..., 6=Jun.

Region	Ground truth	Prominent predictive season	Prominent season found
Sahel	El Niño Southern Oscillation (ENSO)	6	6
	North Atlantic Oscillation (NAO)	3-5	-
	Atlantic Multidecadal Oscillation (AMO)	1-6	1-3, 5
	Atlantic Tripole SST	1-3	1-2, 5
	Local variation in atmospheric winds EOF1 (LLW1)	6	1,3,4,6
	Mediterranean Sea Index (MSEA)	6	1-4
	Local variation in atmospheric pressure (GHT)	6	6
	Tropical Southern Atlantic Index (TSA)	6	-
	Indian Ocean Dipole (IOD)	6	-
	Pacific Warm Pool (PACWARM)	6	2-3

Table 2.8 A list of ground truth containing prominent predictive seasons of the climate indices and their prominent season predicted by our method for the East African region. A - indicates that our method did not find the corresponding climate index as a feature. The months are represented as, 1=Jan, 2=Feb, ..., 9=Sep.

Region	Ground truth	Prominent predictive season	Prominent season found
East Africa	Indian Ocean Dipole (IOD)	7-9	7-9
	El Niño Southern Oscillation (ENSO)	7-9	6-9
	Pacific Warm Pool (PACWARM)	7-9	-

it is $O(p^3)$. In the worst case, we may end up computing causal effect of every predictor on the response variable. As a result, when $n > p$ the time taken to compute the causal effect of each predictor on the response variable and its significance across all the Markov equivalent graphs is $O(2p \cdot 2^q \cdot (p^2 n + 100p^2 n)) \approx O(2^{q+1} p \cdot c\, p^2 n)$ where 2^q is the number of Markov equivalent graphs and c is a constant. Thus, the total time complexity is $O(p^q + 2^{q+1} p \cdot c\, p^2 n)$. Similarly, for $n < p$ it is $O(p^q + 2^{q+1} p \cdot c\, p^3)$.

2.5 Related Work

The underlying behavior of a complex system can be attributed to the intricate network of potential cause-effect relationships between factors. Identifying meaningful predictors that can further the understanding of such complex systems is a challenging task. In the climate science domain, constraint-based structure learning methods for causal discovery have been applied to generate

Table 2.9 A list of frequently selected causal relationships identified by our methodology along with its frequency count for African Sahel region.

Causal Relationships	Frequency (%)
LLW1_1→LLW1_4	98.2
MseaSSTindex_1→ MseaSSTindex_2	98.2
AMO_2→AtlanticTripoleSST_2	94.7
AtlanticTripoleSST_1→ AtlanticTripoleSST_2	94.7
AMO_1→AMO_2	91.2
EOFPAC_6→Nino3_6	87.7
LLW1_1→LLW1_3	87.7
AMO_5 → AtlanticTripoleSST_5	85.9
AMO_3→CAR_5	78.9
AMO_2→AMO_3	75.4
PACWARM_2→PACWARM_3	75.4
BEST_6→SOI_6	71.9
CAR_4→CAR_5	68.4
LLW1_3→LLW1_4	68.4
MseaSSTindex_2→MseaSSTindex_3	66.7
GHT1_6→LLW1_6	59.6
AtlanticTripoleSST_1→AMM_1	54.4
MseaSSTindex_2→MseaSSTindex_4	54.4
MseaSSTindex_3→MseaSSTindex_4	52.6

graphs of information flow (i.e., causal graphs) describing the interactions between four climate indices [EUD12; EUD14]. Similarly in the domain of bioinformatics, causal inference algorithms have been applied to estimate the brain network structure from fMRI data and to explain the variations observed in high-throughput gene expression data [Chi12; Iye13]. Note that these methods focus on studying potential causal relationship using domain knowledge, which may not be feasible for high-dimensional systems.

Feature selection methods based on local causal structure learning identify variables within the Markov Blanket of a target variable [Ali10a; Peñ07; Tsa03; Guy07]. HITON and Max-Min Hill Climbing (MMHC) are prominent examples of such feature selection methods, which employ the divide and conquer approach to find the parents, children and the spouses of the target variable[Ali10a; Ali03]. However, these methods are based on constraint-based learning and do not incorporate the causal information between the variables to select the features. In this work, we have proposed a novel method that integrates the causal strength of the predictors on the target variable to select meaningful predictors. The results show that our method produces better predictive performance over these methods.

Table 2.10 A list of frequently selected causal relationships identified by our methodology along with its frequency count for East African region.

Causal Relationships	Frequency (%)
BEST_7→BEST_8	100
BEST_7→SOI_7	100
IOD_7→IOD_8	100
IOD_8→IOD_9	100
SOI_9→BEST_9	100
BEST_6→BEST_7	98.2
BEST_8→BEST_9	98.2
MEI_9→EOFPAC_9	98.2
EOFPAC_9→Nino3_9	96.5
Nina4_8→Nina4_9	96.5
Nino3_7→Nino3_8	96.5
BEST_6→MEI_7	94.7
EOFPAC_7→MEI_7	91.2
NAO_3→IOD_7	89.5
Nino34_8→ONI_8	89.5
ONI_7→Nino34_8	89.5
Nina4_7→Nina4_8	85.9
Nina12_8→Nina12_9	84.2
Nino34_7→ONI_7	78.9
BEST_6→SOI_6	77.2
BEST_2→MEI_2	64.9
BEST_7→SOI_8	64.9
Nina4_7→Nino34_7	64.9
MseaSSTindex_1→MseaSSTindex_2	56.1
Nino3_7→EOFPAC_7	52.6

2.6 Conclusion

Causality-guided methods have been used in multiple domains to facilitate the understanding of complex systems. Traditionally, the application of these methods has been limited to descriptive and inferential purposes. In this work, we propose a causality-guided feature selection methodology that identifies predictors taking part in potential causal relationships and selects features based on their causal strength with respect to the response via clustering. We achieve this through a number of technical contributions, such as estimating the statistical significance of causal effects on the response, while addressing multicollinearity in the data. Our proposed methodology was found to perform consistently in terms of classification accuracy and RMSE score across real-world data sets from the domains of climate science and biology, suggesting that the newly selected features

have predictive skill for the response variable of interest. Additionally, in contrast to existing feature selection methods, our method also identifies a list of prioritized causal relationships that can be studied by domain scientists.

In addition to selecting features with greater predictive power, we also evaluated the physical interpretability of the frequently selected features and their causal relationships in climate data sets. Most of the features were found to be in agreement with the studies in climate science.

CHAPTER

3

GRAPH PARTITIONING BASED CAUSAL DISCOVERY

3.1 Introduction

In the previous chapter, we utilized the order independent version of the well-known PC-stable algorithm to learn a PDAG. The algorithm initializes a complete undirected graph and performs conditional independence (CI) tests between pairs of variables to remove non-causal relationships. Further, it is also commonly used as the first step of the FCI-, RFCI- and CCD-algorithms. The FCI- and RFCI-algorithms can learn causal structure in the presence of hidden variables, whereas the CCD-algorithm can be used to learn cyclic causal structures [Spi95; Col12; Ric96]. Among the rich set of causal discovery methods available, the constraint-based PC algorithm has gained much attention, due to its uniform consistency and computational feasibility in sparse high-dimensional settings [KB07].

However, the causal graph output by the PC-algorithm is vulnerable to the order of variables used in CI tests. This issue was found to be especially pronounced in high-dimensional data sets, which led to the advent of the PC-stable algorithm [CM14]. Due to the exhaustive number of CI tests performed by PC-stable, it tends to have the lowest rate of false positives, but often fails to find true causal relationships due to erroneously removed edges as a result of incorrect high order CI tests decisions [Liu17; XG08; ZM17]. Algorithms such as stable FCI, RFCI, and CCD share the

PC-stable algorithm as their first step towards building a causal graph. Thus, an improvement in the performance of PC-stable is desirable and can benefit all its extensions.

Inspired by the recent success of the divide-and-conquer strategy to improve the quality of causal graphs [Cai13; Liu17; XG08; Xie06], we propose the first algorithm that: (i) uses divide-and-conquer strategy to improve the performance of the PC-stable algorithm and (ii) presents a general strategy to determine the conditioning set size for performing CI tests. At its core, our effort is centered around an intelligent ordering of the conditioning sets for finding d-separators. To do this, we recursively partition an undirected graph (skeleton) into communities, considering both the graph topology and strength of association between variables, and perform well designed CI tests on every variable in a community that shares an edge with a variable in a different community. We then proceed with a bottom-up approach, merging the smaller subgraphs to form the final causal structure. Simulations on real-world discrete and Gaussian bayesian networks show a significant improvement in the quality of output causal graphs, without increasing the rate of false positives.

3.2 Related Work

The exponential time complexity of CB methods has encouraged the development of scalable techniques for causal discovery [Sil00]. These approaches include: ordering the CI tests based on variable association [Spi00], developing a variable ordering agnostic PC [CM14], reducing the number of CI tests by identifying specific structures in the undirected graph [Abe06], or leveraging parallelism to reduce the runtime [Le16]. However, all of these methods focus on building the entire skeleton at once, which makes them computationally expensive and susceptible to errors.

An alternative approach is to use a divide-and-conquer strategy. An earlier work in this direction aimed at decomposing a moral graph[1] into smaller subgraphs by searching for a complete separator set [Gen05]. The condition of finding a complete separator set was relaxed by introducing d-separation trees [Xie06]. Both these approaches decomposed only the original undirected graph, but none of the subgraphs. A recursive approach for CB learning was later proposed, which addressed this and further decomposed the subgraphs [XG08]. More recently this idea has been explored to develop hybrid methods with promising results. The SADA framework divides the graph by initializing a cut set that removes conditional dependency between two sets of variables [Cai13]. Each set of variables is recursively divided until the size of the variable set is below a threshold and a linear non-gaussian additive model (LiNGAM) is applied on smaller subgraphs. The Large-Scale Causal Discovery (LCSD) framework extends SADA by building skeleton on smaller subgraphs using CB methods and orienting the edges using additive noise models (ANM) [Hon17]. The recently proposed Separation and Union (SAR) method partitions the graph into two separate variable sets using RatioCut and performs CI tests up to a fixed order between the variables to optimize the

[1] A moral graph is the same as a skeleton except it has edges between pairs of variables having a common child

cut set [Liu17]. The variable sets are further partitioned until their size is below a threshold and a score-based approach is used to build DAGs on smaller subgraphs.

Our work has been motivated by the encouraging results of the divide-and-conquer strategy. We propose a novel approach to improve the order independent version of the PC algorithm, using community detection as a graph partitioning approach. Based on community detection, our method is the first that is capable of both variable association driven partitioning, and using more than two partitions to find separator sets. Although our method improves a CB method, it can be easily extended to score-based or hybrid methods.

3.3 Preliminaries

In this section we define terms and notations that will be used in the rest of the paper. A *graph* G is defined over a set of nodes (variables)[2], $\mathbf{X}$, a set of edges $\mathbf{E}$ with no self-loops, and a set of edge weights $\mathbf{W}$ in the graph. The edge weights $\mathbf{W}$ represent the association between a pair of variables, and will only be used for community detection. For the sake of consistency, we use the same notation of graph when referring to community detection method and the PC-stable algorithm. In the context of our work, graphs can contain two types of edges between variables, directed ($\rightarrow$) and undirected ($-$). Given a graph G, any pair of variables $X_i, X_j \in \mathbf{X}$ are *adjacent* to each other if they are connected by a directed or an undirected edge. A *path* between two variables X_i and X_j is defined by a sequence of distinct variables. A directed path from variable X_i to X_j is a sequence of directed edges beginning at X_i terminating at X_j, and an undirected path is a sequence of undirected edges from X_i to X_j. X_i is an *ancestor* of X_j and X_j is a *descendant* of X_i if there is a directed path from X_i to X_j. A graph is known as a directed acyclic graph (DAG) if it consists only of directed edges, and contains no cycles. In other words, no (directed) path exists in the graph that both starts and ends at any variable X_i. The *skeleton* of a graph is the same graph but with directed edges replaced by undirected edges.

A triple (X_i, X_j, X_k) in graph G is known as a *v-structure* if X_i and X_k are not adjacent to each other, but are both a *parent* of X_j, i.e., $X_i \rightarrow X_j \leftarrow X_k$. Variables such as X_j are referred as colliders.

A path between variables X_i and X_j is said to be blocked by a variable set $\mathbf{Z} \subseteq \mathbf{X} \setminus \{X_i, X_j\}$ if no variable in $\mathbf{Z}$ is a collider or a descendant of a collider. X_i and X_j are said to be conditionally independent given $\mathbf{Z}$, denoted as $X_i \perp\!\!\!\perp X_j \mid \mathbf{Z}$, if all the paths between them are blocked by the variables in $\mathbf{Z}$. In order to recognize such conditional independence relationships in DAGs, a graphical criterion known as *d-separation* was developed [Pea03]. Two variables X_i and X_j are d-separated by $\mathbf{Z}$ if $X_i \perp\!\!\!\perp X_j \mid \mathbf{Z}$. We refer to the variables in $\mathbf{Z}$ as *d-separators* in the rest of the paper.

It is possible to have two or more DAGs that encode the same conditional independence relationships. Such DAGs are said to be *Markov equivalent*, as they have the same skeleton and the same

[2] In this work, we will refer to nodes and variables interchangeably.

set of *v-structures*. DAGs in a Markov equivalence class can be uniquely represented by a completed partially directed acyclic graph (CPDAG). We refer to a CPDAG as a causal graph in this paper.

In this paper, we make the following assumptions about the variables in the data set, and interpretation of the output graph. Given a variable set $\mathbf{X}$, with a probability distribution P, the assumptions are as follows:

- *Causal sufficiency:* There are no missing common causes of variables in the data set.
- *Causal faithfulness:* The conditional independence relations in the output causal graph are entailed by the probability distribution P.
- *Causal Markov condition:* Given its parents, a variable is independent of every variable in the graph except its descendants.

A community is defined as a group of variables that are densely connected within a community than outside their community. We denote a set of disjoint communities as $\mathbf{C} = \{C_1, C_2, ..., C_k\}$, and $X_{C_1}, X_{C_2}, ..., X_{C_k}$ as the variable set for the corresponding community, such that $\bigcup_{i=1}^{k} X_{C_i} = \mathbf{X}$ and for every pair of communities $C_i, C_j \in \mathbf{C}, X_{C_i} \cap X_{C_j} = \emptyset$ if $i \neq j$. That is, every variable is part of exactly one community. For a variable $X_i \in X_{C_k}$, the set of variables adjacent to X_i in its community, C_k, are denoted by adj(X_i, G_{C_k}) and the set of adjacent variables not in its community, $C_{\neq k}$, are denoted by adj$(X_i, G_{C_{\neq k}})$.

3.4 Method

The goal of our algorithm, as with PC-stable and all other CB methods, is to d-separate as many pairs of variables as possible. To do this, PC-stable selects pairs of adjacent variables in lexicographical order and performs CI tests with increasing conditioning set sizes. Instead of using lexicographical ordering, as PC-stable does, we posit that the graph structure and the strength of the association between variables can be used to guide the order in which CI tests are performed. Existing methods have successfully used graph-based divide-and-conquer approaches to yield promising results [Liu17; XG08]. However, none of them have leveraged variable association and the ability to deal with more than two graph partitions. Our method builds on the idea of divide-and-conquer, extending it for CB methods by incorporating variable association and generalizing from using two-way graph partitioning to being able to handle an arbitrary number of partitions at each recursive split step. Moreover, we also develop a general strategy to determine the conditioning set size for performing CI tests.

3.4.1 Constructing Undirected Weighted Independence Graphs

In this subsection, we describe the approach of constructing an undirected weighted independence graph. This approach is divided into two steps: first, we remove spurious associations between variables using conditional independence tests, and second, we augment the remaining edges with edge weights.

The goal is to remove spurious associations between two variables by conditioning on a variable set such that the size of this variable set can be at most one. In the first step, we start with a complete undirected graph, and then check each edge between a pair of variables X_i and X_j. An edge between X_i and X_j is tested using conditional independence test of zero and first order [WB06]. If the edge is not removed it indicates that the dependence between the two variables cannot be explained by any other variable, X_k. Let ρ_{ij} denote the correlation coefficient of variables X_i and X_j, and $\rho_{ij|k}$ denote the partial correlation coefficient of X_i and X_j given X_k. There exists an edge between X_i and X_j if the following condition holds,

$$\rho_{ij} \neq 0 \quad \text{and} \quad \rho_{ij|k} \neq 0 \text{ for all } k \in \{1,2,...,p\} \setminus \{i,j\} \tag{3.1}$$

We present the details of estimating the undirected independence graph using the zero- and first-order conditional independence tests. For every pair of adjacent variables, (X_i, X_j), we test the null hypothesis $H_0 : \rho_{ij|k} = 0$ versus the alternative hypothesis, $H_1 : \rho_{ij|k} \neq 0$ by performing statistical tests, where $k \in \{\emptyset, 1, 2, ..., p\} \setminus \{i, j\}$. For continuous data we test the null hypothesis using the Fisher's Z test, and for discrete data we use the G-squared test.

First, we build a complete undirected graph where each variable is connected to every other variable in the graph. For a pair of adjacent variables, (X_i, X_j), we construct a conditioning set, X_k such that $k \in \{\emptyset, 1, 2, ..., p\} \setminus \{i, j\}$ and compute the corresponding p-value, $P(i, j|k)$ to test the null hypothesis $H_0 : \rho_{ij|k} = 0$ against the alternative $H_1 : \rho_{ij|k} \neq 0$. Note that due to symmetry, $\rho_{ij|k} = \rho_{ji|k}$, we perform the statistical tests only once for a pair of adjacent variables. At the end of this step, for every pair of adjacent variables we have recorded all the p-values.

Second, for every pair (X_i, X_j) we select the maximum p-value, $P_{max}(i, j|k)$ from the set of all the p-values.

$$P_{max}(i, j) = \max_{k \in \{\emptyset, 1, 2, ..., p\} \setminus \{i, j\}} P(i, j|k) \tag{3.2}$$

Third, we correct the p-values computed over $p(p-1)/2$ edges using the Benjamini-Hochberg correction method for controlling false discovery rate [BH95]. We denote the corrected maximum p-values as, $P_{corr,max}(i, j)$. In the final step, an edge between X_i and X_j remains if $P_{corr,max}(i, j)$ is less than some specified significant level α. In our experiments, we initialize $\alpha = 0.05$.

At the end of this step, we expect to get rid of spurious associations of zero- and first-order. The corrected maximum p-value represents a dependence measure between the two variables.

Figure 3.1 shows an example of an undirected independence graph constructed from the Insurance data set. For the remaining edges in the graph, we generate their edge weights by computing the association between a pair of variables. In case of continuous data, we calculate the absolute value of correlation, abs($\rho(ij)$) between the variables as their measure of association, and in case of discrete data we calculate the Cramer's V coefficient, $cv(ij)$. The equations for computing the two different association measures are as follows,

$$\rho(ij) = \frac{\sum_{k=1}^{n}(x_{ik} - \bar{x_i})((x_{jk} - \bar{x_j})}{(n-1)s_i s_j} \tag{3.3}$$

where n is the number of observations in the data, $\bar{x_i}$ and $\bar{x_j}$ are the sample means, and s_i and s_j are the corrected sample standard deviations.

$$cv(ij) = \sqrt{\frac{\chi^2/n}{\min(k-1, r-1)}} \tag{3.4}$$

where χ^2 is the Pearson's chi-squared statistic, n is the number of observations in the data, k and r equals the number of columns and rows in the data respectively.

3.4.2 Community-driven Causal Discovery (CDCD)

In this subsection we present our divide-and-conquer approach (see Algorithms 4 and 5) of constructing the skeleton of a causal graph. Specifically, we refer to the split phase for dividing the undirected weighted independence graph into smaller subgraphs until they cannot be divided further, and we refer to the merge phase for merging the smaller subgraphs to construct the full skeleton.

Having weighted the edges, we partition the graph using a community detection algorithm, which identifies groups of similar nodes (variables), taking into account both the graph structure as well as the edge weights (variable association). Inter-community variables, those with neighbors in other communities, serve as a good starting point for finding the conditioning set that d-separates the variables in their community from the rest of the graph. Figure 3.1 illustrates this idea.

Our method consists of two phases, namely, the *split* phase and the *merge* phase.

Split. First, we apply community detection (explained in Section 3.4.2) on the undirected weighted independence graph and decompose it into a set of disjoint communities, $\mathbf{C} = \{C_1, C_2, \ldots, C_k\}$, such that for every pair of communities $C_i, C_j \in \mathbf{C}$, $X_{C_i} \cap X_{C_j} = \emptyset$ if $i \neq j$. The variable set X_{C_i} can consist of two types of variables: intra-community variables, $X_{i,intra}$ and inter-community variables, $X_{i,inter}$, where $X_{i,intra}$ are variables whose neighbors all belong to C_i i.e., $\forall X_i \in X_{i,intra}$, $\left|\text{adj}(X_i, G_{C_{\neq i}})\right| = 0$, and $X_{i,inter}$ are variables whose neighbors don't necessarily belong to C_i i.e., $\forall X_i \in X_{i,inter}$, $\left|\text{adj}(X_i, G_{C_{\neq i}})\right| \geq 1$. In Figure 3.1, for $X_{C_i} = \{6, 8, 15, 20, 21\}$, variables 6, 20 and 21 are intra-community variables, and variables 8 and 15 are inter-community variables. The inter-

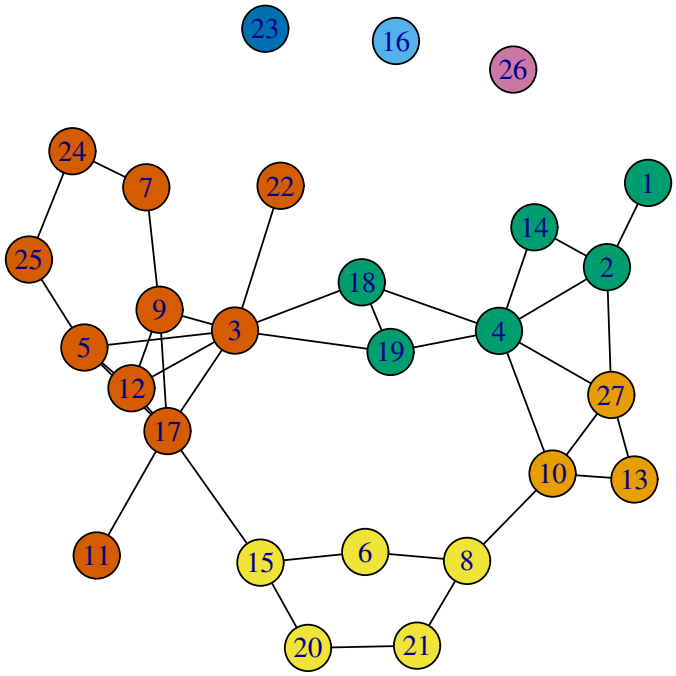


Figure 3.1 Sample output of community detection on an undirected weighted independence graph, after removing edges using zero- and first-order CI tests [WB06]. Variables in the same community share the same color. The inter-community variables, $\mathbf{Z}$={3, 4, 8, 10, 15, 17, 18, 19}, serve as the potential d-separators that separate variables in their communities from the rest of the network.

community variables form a set of potential d-separators, $\mathbf{Z}$, as they block the flow of information between the variables in their community and the rest of the network. Unlike PC-stable, which performs CI tests on every pair of adjacent variables, we contain these tests to the edges incident on the variables in $\mathbf{Z}$. These edges are tested with second-order and higher conditioning sets, since all the first-order CI tests were performed while building the undirected independence graph. While performing these tests with increasing conditioning set size, we update the adjacency set for each $Z_i \in \mathbf{Z}$ in the same manner as PC-stable (see lines 5-7 Algorithm 4.1 in [CM14]).

Inter-community variables can have a high degree, e.g. they may be sparsely connected outside their community but densely connected within the community. One such example is the connectivity of variable 3 in the undirected weighted independence graph shown in Figure 3.1. Because of this, removing edges incident on variable 3 can lead to several high-order tests being performed, which increases the risk of false positives. Thus, there is a need to restrict the size of the conditioning set without increasing false negatives. Existing methods treat the conditioning set size q as a parameter, either initializing it to a fixed value or trying every possible combination of adjacent variables as the conditioning set [Cai13; Liu17].

We propose a general strategy to initialize the conditioning set size using the local structural

Algorithm 4 Community-driven Causal Discovery (CDCD)

Require: Data set D, an undirected graph $G = (\mathbf{X}, \mathbf{E}, \mathbf{W})$ with variable set $\mathbf{X}$, edges $\mathbf{E}$ and edge weights $\mathbf{W}$

Ensure: A completed directed acyclic graph G

if $|\mathbf{X}| \leq 4$ **then**
  Perform CI tests for every pair of adjacent variables in G
  return G
end if
Perform community detection on an undirected weighted graph G and obtain a set of disjoint communities $\mathbf{C} = \{C_1, C_2, \ldots, C_K\}$.
Let $\mathbf{Z}$ be the set of intercommunity variables in G
$q = \emptyset$
for each $X_i \in \mathbf{Z}$ **do**
  Build a Markov Blanket of X_i using the MMHC algorithm [Ali10a]
  Let PC_i be the set of parents and children of X_i in its Markov Blanket
  $q = q \cup |PC_i|$
end for
$q = \min(q)$
for each $X_i \in \mathbf{Z}$ **do**
  G=CI_TESTS(D, G, X_i, $\mathbf{C}$, q)
end for
$G_C = \emptyset$
for all communities $C_k \in \mathbf{C}$ **do**
  X_{C_k}= variables in community C_k
  Z_{C_k}=variables that are not in C_k and are adjacent to inter-community variables in C_k
  $D_{V_{C_k}}$ = data set containing variables $V_{C_k} = X_{C_k} \cup Z_{C_k}$
  G_{C_k}= graph over variables V_{C_k}
  G_{C_k}= CDCD($D_{V_{C_k}}$, V_{C_k}, G_{C_k})
  $G_C = G_C \cup G_{C_k}$
end for
Update G with edges in G_C given by equation 3.5
return G

information of the inter-community variables. To do this, we build a Markov Blanket (MB) for each variable $Z_i \in \boldsymbol{Z}$ and record the number of parents and children in it, $nPC(Z_i)$. While building the MB for Z_i, we include only those variables that are adjacent to it, not the full variable set $\mathbf{X}$, thereby efficiently building each MB. For example, in Figure 3.1, the MB for variable 4 will be built using the variables $\{2, 10, 14, 18, 19, 27\}$. Several algorithms exist that can build the Markov Blanket for a given variable. In our case, we use the Max-Min Hill Climbing (MMHC) algorithm [Tsa06]. Once the number of parents and children for every $Z_i \in \mathbf{Z}$ have been recorded, we initialize $q = \min_{Z_i \in \boldsymbol{Z}}(nPC(Z_i))$. Every edge incident on Z_i is then conditionally tested, where the size of

Algorithm 5 Perform tests of Conditional Independence: CI_TESTS

Require: A data set D, an undirected graph G, variable X_i, a set of communities **C**, and a conditioning set size q

Ensure: An updated undirected graph G

1: **for each** $X_j \in adj(X_i, G_{C_{\neq k}})$ **do**
2: $S_i = \text{adj}(X_i, G_{C_k}) \cup \text{adj}(X_i, G_{C_{\neq k}}) \setminus X_j$
3: $\forall S \subseteq S_i$ s.t. $|S| \leq q$, test if $X_i \perp\!\!\!\perp X_j \mid S$ using data set D and update G.
4: **end for**
5: **for each** $X_l \in \text{adj}(X_i, G_{C_k})$ **do**
6: $S_i = \text{adj}(X_i, G_{C_k}) \setminus X_l \cup \text{adj}(X_i, G_{C_{\neq k}})$
7: $\forall S \subseteq S_i$ s.t. $|S| \leq q$, test if $X_i \perp\!\!\!\perp X_j \mid S$ using data set D and update G
8: **end for**
9: **return** G

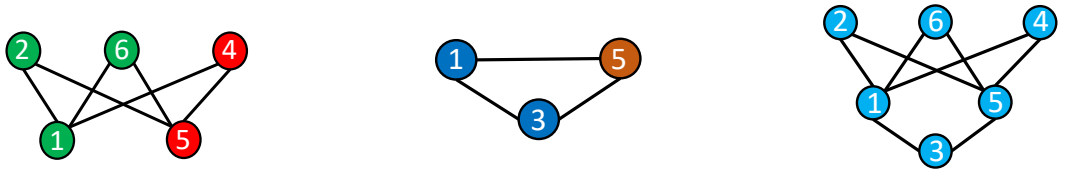
the conditioning set can grow up to q. It might happen that $q = 1$, in which case we increment its value to 2 since all CI tests of first-order have been performed while building the undirected independence graph. The split phase is recursively performed on each community with the variable set $X'_{C_i} = X_{C_i} \cup Z_{C_i}$ where $Z_{C_i} = \bigcup_{X_i \in X_{i,inter}} adj(X_i, G_{C_{\neq i}})$, until there is no change in the community structure, or the community is comprised of four or fewer variables. Four is chosen as the threshold to exit the split phase as it is the minimum number required to perform second-order pairwise CI tests. Once the graph cannot be decomposed further, we test every pair of adjacent variables by performing CI tests up to order q. We now present two theorems that validate the completeness of the split phase.

Theorem [XG08]: Given three disjoint sets, X, Y and Z, such that $X \perp\!\!\!\perp Y \mid Z$. For $X_i \in X$ and $Z_j \in X \cup Z$, then X_i and Z_j are d-separated by a subset of $X \cup Y \cup Z$ if and only if they are d-separated by a subset of $X \cup Z$.

Theorem [XG08]: Given three disjoint sets, X, Y and Z, such that $X \perp\!\!\!\perp Y \mid Z$. Every pair of adjacent variables $Z_i, Z_j \in Z$ is d-separated by a subset of $X \cup Y \cup Z$ if and only if they are d-separated by a subset of $X \cup Z$ or $Y \cup Z$.

Merge. After all the recursive split calls have terminated, we proceed with bottom-up merging to recover the full skeleton. Let $G_{C_U} = (U, E_U)$ and $G_{C_V} = (V, E_V)$ be two subgraphs over two communities C_U and C_V, with variable sets U and V, and edge sets E_U and E_V. While merging these subgraphs, we keep the edges that are found only in G_U or G_V with the exception of edges between variables common to both the subgraphs. For such variables, we keep an edge in the merged graph only if it is present in G_U and G_V. The following equation demonstrates the edges that will be present in the merged subgraph $G_{U \cup V}$

$$E_{U \cup V} = (E_U \cup E_V) \setminus \{(U_i, V_j) : U_i, V_j \in U \cap V \text{ s.t. } (U_i, V_j) \notin (E_U \cap E_V)\} \tag{3.5}$$



(a) Subgraph G_1 **(b)** Subgraph G_2 **(c)** Merged graph G

Figure 3.2 Subgraphs G_1 and G_2 are output from the split phase and they are to be merged to build the full skeleton. Variables in the same community are represented in the same color. Note that in subgraph G_1 the edge $1-5$ is removed whereas in G_2 the edge is still present. Therefore, in the final merged graph, the edge $1-5$ is removed since it is likely that the two variables are d-separated by the variables in G_1.

Figure 3.2 illustrates the process of merging two subgraphs G_1 and G_2 that have been output by the split phase. Variables that belong to the same community are represented by the same color. Note that variables 1 and 5 are common to both subgraphs. Thus, while merging G_1 and G_2 we keep the edge between the two variables in the merged graph only if it is present in both the subgraphs.

We observe that the edge $1-5$ is removed in G_1 but not in G_2, which would mean that the variables that d-separate 1 and 5 are present in G_1. Thus, while merging the two subgraphs, the edge between 1 and 5 is not present as it is likely that the two variables are also d-separated in the true causal graph. Finally, the merged graph G is shown in Figure 3.2c

Once the full skeleton has been recovered, we orient every unshielded triple into a v-structure [CM14]. The remaining undirected edges are oriented using Meek's rules [Mee95] to output a completed partially directed acyclic graph (CPDAG). The following theorem validates the completeness of the merge phase.

Theorem [XG08]: Suppose that $X \perp\!\!\!\perp Y \mid Z$, the skeleton $G_V = (V, E_V)$ can be constructed by merging local skeletons $G_{X\cup Z} = (X \cup Z, E_{X\cup Z})$ and $G_{Y\cup Z} = (Y \cup Z, E_{Y\cup Z})$ as follows:

1. $V = X \cup Y \cup Z$
2. $E_V = (E_{X\cup Z} \cup E_{Y\cup Z}) \backslash \{(Z_i, Z_j) : Z_i, Z_j \in Z \text{ s.t. } (Z_i, Z_j) \notin (E_{X\cup Z} \cap E_{Y\cup Z})\}$

Graph Partitioning using Community Detection. Community detection is defined as the task of identifying densely connected groups of nodes with sparse inter-group connections. A group of densely connected nodes is known as a community. One approach for community detection iteratively partitions the graph, at each split trying to optimize an objective function. Several community detection methods have been developed, for both undirected and directed graphs [Har14].

In this work, we apply the Louvain community detection algorithm [Blo08], which is based on maximizing modularity, to find communities in a weighted undirected graph. For a given partitioning of the graph, modularity (Q) is measured by comparing the density of edges within communities

to that of between communities. Since optimizing exact modularity is an NP-complete problem [Bra07], the Louvain algorithm aims to maximize the gain in modularity (ΔQ) in each iteration. The algorithm iterates over two phases that can be summarized as follows.

Phase I: Initially, each variable is treated as a community, and the gain in modularity is calculated by moving a variable i from its current community, to the community of it's neighbor variable, j. The two communities are merged if there is a positive gain in the modularity, calculated as follows,

$$\Delta Q = \left[\frac{\sum_{in}+k_{i,in}}{2m} - \left(\frac{\sum_{tot}+k_i}{2m}\right)^2\right] \quad - \left[\frac{\sum_{in}}{2m} - \left(\frac{\sum_{tot}}{2m}\right)^2 - \left(\frac{k_i}{2m}\right)^2\right] \tag{3.6}$$

where $\sum_{in}$ is the summation of the edge weights inside a community C, $\sum_{tot}$ is the summation of edges incident to variables in C, k_i is the sum of edge weights incident on variable i, $k_{i,in}$ is the sum of edge weights between variable i and the variables in community C, and m is the sum of all of the edge weights in the network. If no positive gain in modularity is achieved, variable i stays in its own community. This process is sequentially applied to all the variables in the graph and repeated until there is no further improvement in the modularity.

Phase II: A new network is built where the nodes correspond to the communities formed in *Phase I*. The weight of an edge in the new graph is the summation of the inter-community edges in the old graph. Edges between nodes in the same community are represented as self-loops in the new network. The algorithm iterates over these two phases until there are no more changes in the graph and a maximum modularity is attained.

We would like to point out that during the split phase, it may happen that the Louvain algorithm outputs only one community. In such scenarios, we switch to another community detection algorithm known as FastGreedy for mining communities in a given subgraph. Similar to the Louvain algorithm, FastGreedy aims to optimize modularity using a hierarchical agglomeration technique [Cla04]. The community structure is depicted by a dendrogram which is cut into two communities since it is the minimum number of communities our method requires to proceed with the split phase.

Community detection methods harness the structural property of the graph as well as the association between variables to find groups of closely associated variables. This is an interesting feature that we utilize to guide the skeleton phase of the PC-stable algorithm. Intuitively, variables inside a community tend to be more similar and closely connected than those outside a community. We build our method on this notion of *similarity* by first removing the edges between communities and then within each community.

3.4.3 PC algorithm

One of the most prominent examples of constraint-based causal discovery methods is the PC algorithm (named after its inventors Peter Spirtes and Clark Glymour) [Spi00]. The algorithm starts with a complete undirected graph and applies conditional independence (CI) tests on every pair of adjacent variables. The size of the conditioning set in these CI tests is initialized to 0 and incremented by 1 at each level. The output of this step is an undirected graph which contains edges between variables that are not conditionally independent, given any combination of the remaining variables. This undirected graph is known as the skeleton. In the next step, the algorithm orients every triple $X - Y - Z$ into $X \rightarrow Y \leftarrow Z$ if and only if Y does not d-separate X and Z, X is adjacent to Y, Z is adjacent to Y, but X and Z are not adjacent. The triple $X \rightarrow Y \leftarrow Z$ is known as a *v-structure*. The remaining undirected edges are oriented using a set of orientation rules [Mee95], and the output is a completed partially directed acyclic graph (CPDAG). A CPDAG represents a set of Markov equivalent directed acyclic graphs (DAGs), where each DAG has the same skeleton and the same *v-structures*. The PC algorithm has been shown to be sound and complete [Spi00; Mee95; And97].

Building a skeleton by performing pairwise CI tests contributes significantly to the computational cost of the algorithm. In high-dimensional settings, a large number of CI tests can lead to incorrect edges being added or removed from the skeleton. Additionally, the estimated skeleton has been shown to be sensitive to the variable ordering in high-dimensional data sets [CM14]. While this problem has been addressed by developing an order independent version known as the PC-stable algorithm, it comes at the cost of performing more CI tests than the PC algorithm [CM14]. In the worst-case, the time complexity of the PC-stable algorithm is exponential. As a result, we focus on improving the skeleton phase of the PC-stable algorithm. The details of building a skeleton are shown in Algorithm 1.

Succinctly, the goal of constraint-based methods such as the PC-stable algorithm is to d-separate as many pairs of variables as possible. While CI tests are pivotal to the PC-stable algorithm, the association between variables and the graph structure at each level of the skeleton phase can be used to find d-separators efficiently. Graph mining methods such as community detection have been widely used to find groups of densely connected variables. Therefore, we utilize the structural information provided by the community formation to drive the skeleton phase. We now present a description of the well-known Louvain community detection algorithm.

3.5 Empirical Evaluation

We evaluate the performance of our method on several commonly used Bayesian Networks (BNs) of varying complexity and sizes. Table 3.1 presents a list of the different data types and networks used

Algorithm 6 PC-stable Algorithm: Skeleton phase

Require: Data set D with variables X, a complete undirected graph G, and a significance threshold α

Ensure: A completed directed acyclic graph G

$l = 0$
repeat
  for all $X_i \in V$ **do**
    $a(X_i) = \text{adj}(X_i, G)$
  end for
  repeat
    Select a new pair of adjacent variables X_i and X_j in G
    $\forall S \in a(X_i) \setminus X_j$ such that $|S| = l$
    if $X_i \perp\!\!\!\perp X_j | S$ at threshold α **then**
      Remove edge $X_i - X_j$ from G
      Store S in the separating set, sepset(X_i, X_j)
      Continue with the next pair of adjacent variables
    end if
    $l = l + 1$
  until X_i and X_j are no longer adjacent or all conditioning sets of size l have been tested
until $l >$max$(|\text{a}(X_i, G) \setminus X_j|)$
return G

Table 3.1 A list of different networks and data types used in our experiments. Discrete data sets were generated from five discrete bayesian networks and continuous data sets were generated from two gaussian bayesian networks.

Data type	Network	Nodes	Edges
Discrete	Insurance	27	52
	Alarm	37	46
	Hailfinder	56	66
	Hepar2	70	123
	Winpts95	76	112
Continuous	Magic-Niab	44	66
	Magic-Irri	64	102

in our experiments. These datasets are publicly available from the online BN repository[3].

Simulations were performed by randomly generating 10 data sets for each network over three different sample sizes, $N = \{1000, 2000, 5000\}$. For discrete data we used Cramer's V coefficient to generate edge weights, and G^2 statistical test to measure conditional independence. For continuous data we used the absolute value of correlation as edge weights, and Fisher's Z test to measure conditional independence. A significance threshold of $\alpha = 0.05$ was used for CI tests. All the experiments

[3] http://www.bnlearn.com/bnrepository/

were conducted using R as the programming language.

3.5.1 Results

We compare the quality of the causal graphs output by our method, PC, and PC-stable, based on five performance metrics. In particular, we compute: (1) Structural Hamming Distance (SHD), which indicates the number of edge additions, deletions, and flips required to convert the estimated graph into the true graph; (2) True Discovery Rate (TDR), which is the number of correctly found edges divided by the number of edges in the estimated graph; (3) True Positive Rate (TPR), which records the number of correctly found edges divided by the number of edges in the true graph; (4) False Positive Rate (FPR), which is the number of edges found in the estimated graph that are not present in the true graph; (5) and the standard accuracy metric, d, which measures the distance between the true causal graph and the estimated causal graph, $d = \sqrt{(1-TDR)^2 + (1-TPR)^2}$.

Figures 3.3 and 3.4 show a comparison of the three methods on discrete and Gaussian BNs, respectively. In contrast to the PC algorithm, PC-stable is robust against cascading incorrect CI test decisions, and agnostic to variable ordering [CM14]. As a result, PC-stable achieves a higher TDR and lower FPR compared to PC. However, this comes at the cost of missing true causal edges resulting in lower TPR and higher SHD. On the other hand, PC may have lower SHD and higher recall than PC-stable but it fails to remove causal relationships that are not present in the true causal graph. We observe that with our recursive community-driven graph partitioning approach we are able to overcome the drawbacks of both the algorithms while improving the quality of causal graphs, as described below.

As shown in Figure 3.3, the SHD of our method is the lowest in all cases except the Alarm network and the Win95pts network with $N = 1000$, where PC has a marginally lower SHD with a difference of 1 and 1.2 respectively. We observe that the TDR of our method is comparable to PC-stable and higher than PC. The Win95pts network is the only case where PC-stable maintains a higher TDR than CDCD in two out of three sample sizes, with a maximum difference of 3.64%. Nonetheless, CDCD achieves a significant improvement, at least 10% in all sample sizes, over PC-stable in the TDR of the Hepar2 network. Comparing TPR, we observe that CDCD substantially outperforms PC-stable with an improvement of up to 26%. Though PC has a slightly higher TPR in some cases with $N = 1000$, the difference is compensated by CDCD performing better with increasing sample size. Additionally, PC has a much higher rate of false causal relationships, while CDCD is able to achieve FPR that is comparable or lower than PC-stable. This suggests that our method is able to recover true causal relationships without increasing the rate of false positives. Lastly, we also compare the standard accuracy metric, d, and observe that our method consistently outputs a lower d value. This suggests that the estimated causal graphs are closer to the true causal graph than PC and PC-stable.

Figure 3.4 shows a similar comparison of the three methods on Gaussian BNs. We observe that

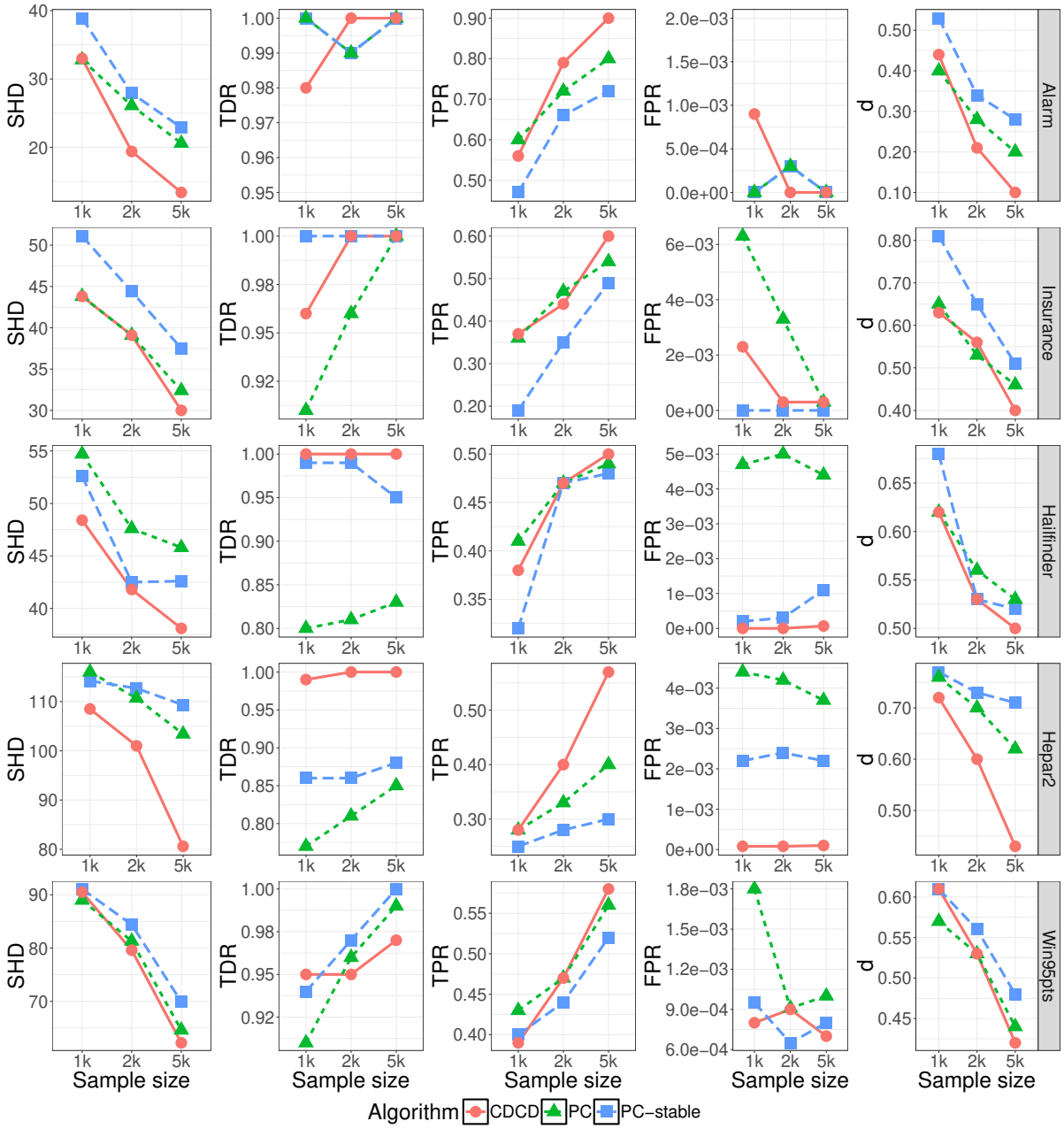


Figure 3.3 Comparison of the CDCD, PC, and PC-stable algorithms in terms of Structural Hamming Distance (SHD), True Discovery Rate (TDR), True Positive Rate (TPR), False Positive Rate (FPR) and the standard accuracy metric, d, across five discrete Bayesian Networks and three sample sizes. The reported metric for each network is its mean value from 10 randomly generated data sets.

CDCD significantly outperforms PC and PC-stable in terms of SHD, TDR, FPR and d value across all sample sizes. PC and PC-stable have a higher TPR for $N = 1000$, but with increasing sample size,

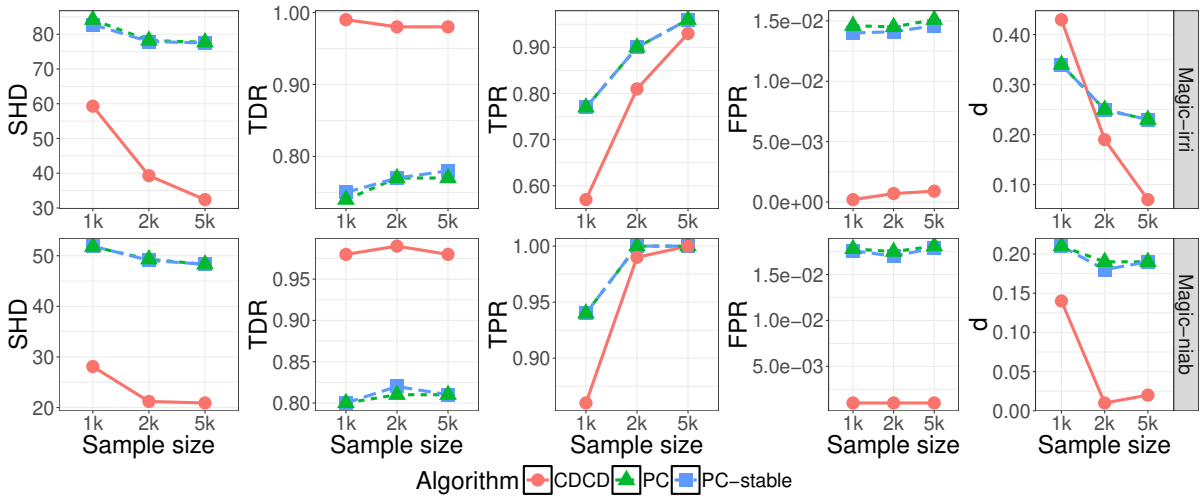


Figure 3.4 Comparison of the CDCD, PC, and PC-stable algorithms in terms of Structural Hamming Distance (SHD), True Discovery Rate (TDR), True Positive Rate (TPR), False Positive Rate (FPR) and the standard accuracy metric, d, across two Gaussian Bayesian Networks and three sample sizes. The reported metric for each network is its mean value from 10 randomly generated data sets.

CDCD achieves comparable TPR while maintaining better quality of output causal graphs.

Overall, the results indicate that CDCD achieves higher TDR and TPR in most cases while eliminating false causal relationships. Consequently, the quality of output causal graphs improves when compared to the PC and PC-stable algorithms.

3.5.2 Time Complexity Analysis

The runtime of our method is dominated by the CI tests performed for building the undirected independence graph. In a network with p variables, a total of $\frac{p(p-1)}{2}$ zero- and first-order CI tests are conducted. As with PC and PC-stable algorithms, the worst case time complexity of CDCD is exponential, $O(p^q)$, where p is the number of variables in the graph and q is the maximum number of variables adjacent to any variable. The time complexity of the Louvain community detection algorithm has been estimated to be $O(n \log n)$ [Blo08]. Therefore, including an additional step to find communities does not incur any overhead to the time complexity.

3.6 Conclusion

Efficiently searching for d-separators is a primary concern for the majority of constraint-based methods. In this chapter, we proposed an algorithm for improving the performance of the PC-stable algorithm when searching for d-separators. To the best of our knowledge, this is the first divide-and-conquer based approach that takes into account both variable association and graph structure to

find d-separators. Our key insight is to leverage community detection for partitioning the graph. Communities serve as an important graph structure to identify d-separators, thereby enabling our method to efficiently perform conditional independence tests.

We performed experiments on several real-world networks networks of different sizes, complexities, and data types to evaluate our proposed algorithm. The results obtained suggest that the graph partitioning approach identifies effective d-separators than PC and PC-stable, thereby improving the quality of output causal graphs.

CHAPTER

4

CAUSAL RELATIONSHIP PREDICTION WITH ADDITIVE NOISE MODELS

4.1 Introduction

As explained in Chapter 1, causal discovery methods such as constraint-based, search-and-score and hybrid methods output a partially directed graph that represents a set of Markov equivalent causal graphs. These structures encode the same set of conditional independence relationships but not all causal relationships can be determined. On the other hand, causal discovery using additive noise models (ANM) has gained much attention given their ability of identifying exact cause-effect relationships [Shi06; Moo09; Hoy09; Jan12; Pet14]. The presence of non-linearity in the data generating process coupled with additive noise structure has made it possible to fully discover directed acyclic causal graphs. The main idea is to detect asymmetry between the cause and effect using regression and statistical tests to establish causality [Shi06; Hoy09]. This idea has been extended from a two-variable case to construct causal graphs in a multivariate setting [Shi06; Jan12; Pet14]. Overall, these methods output better quality of causal graphs which were previously restricted to a set of Markov equivalent graphs.

More recently, the problem of causal discovery in a bivariate case has been presented as a learning problem [Guy13; Guy14]. Given two variables X and Y the goal is to predict the causal relationship between them. These competitions have brought forward another opportunity of

solving the problem of causal discovery from a machine learning perspective. Encouraging results have been obtained in predicting the true causal relationship between two variables [LP15a; LP15b; Fon16]. These methods extract features from the input random variables and train classifiers. Their prediction performance has been shown to outperform the state-of-the-art methods. However, there has been no work on constructing a complete multivariate causal graph purely by learning patterns that can predict causal relationships.

In this work, we propose a Causal Relationship Prediction in Additive Noise Models (CRPAM) framework that learns patterns from the true parents and the non-parents of a given variable, X, and predicts causal relationships. Informally, we define a non-parent as any variable that is not a parent or an ancestor of X. To this end, we use a characteristic kernel to featurize: 1) the residuals from regressing X on its parents, and 2) residuals from regressing X on its non-parents. Our major contributions are as follows,

1. We present the first method to construct a multivariate causal graph purely by finding discriminative patterns between a variable and its parents, and non-parents.
2. We develop an approach with kernel mean embedding for creating a training set that is used to train a nonlinear binary classifier.
3. We perform simulations to compare the effectiveness of our method with state-of-the-art causal discovery methods.
4. We present an analysis of the patterns from kernel mean embeddings and illustrate their discriminative ability.

4.2 Preliminaries

In this section, we define the concepts and notations that will be used throughout the paper.

Definition 1 A causal graph G over a variable set $\mathbf{X}$ is defined as a graph containing directed edges and no directed cycles. A directed edge $X_j \to X_i$ indicates a causal relationship between X_j and X_i, where X_j is the *direct cause* or the *parent* and X_i is the *effect* or the *child*. We denote pa_i as the parent set of X_i.

Definition 2 A *path* in a causal graph G is defined as a sequence of at least two distinct adjacent nodes (variables). A *directed path* between X_i and X_j is a path with all edges oriented in the same direction.

Definition 3 An *ancestor* of variable X_i is a variable that has a directed from towards X_i. We denote an_i as the variable set containing all the ancestors of X_i.

Next, we define two kinds of additive noise models (ANM) namely linear structural equation models (Linear SEMs) and nonlinear structural equation models (Nonlinear SEMs).

Definition 4 [Shi06]: A linear SEM with a causal graph G represents each variable as a linear function of its direct causes and an additive non-Gaussian noise structure,

$$X_i = \sum_{j \in pa_i} \beta_{ij} X_j + \epsilon_i \tag{4.1}$$

where β_{ij} are non-zero causation coefficients for all $i \in \{1,2,...,p\}$ and $j \in pa_i$, and all ϵ_i are mutually independent.

Definition 5 [Hoy09]: A nonlinear SEM with a causal graph G represents each variable as a nonlinear function of its direct causes and normally distributed noise structure ϵ_i,

$$X_i = \sum_{j \in pa_i} f_{ij}(X_j) + \epsilon_i \tag{4.2}$$

where the function $f_{ij}()$ is nonlinear and three times differentiable.

Definition 6: In a causal graph G defined over a variable set $\mathbf{X}$, we define three sets of variables for a variable $X_i \in \mathbf{X}$ with at least one parent,

1. **True parent set** of X_i contains all the variables that are direct causes of X_i. In other words, any variable is adjacent to X_i and has a directed edge towards X_i is part of its true parent set. We denote it as $\mathbf{X_{pa_i}}$.

2. **Non-parent set** of X_i is the set of variables that do not have any causal influence on X_i. Any variable that does not have a direct edge or a directed path towards X_i is a non-parent of X_i. We denote the non-parent set as $\mathbf{X_{non\_pa_i}}$.

3. **Mix parent set** contains all the variables in the true parent set, $\mathbf{X_{pa_i}}$, and some randomly chosen variables from the ancestors of X_i. We denote the mix parent set as $\mathbf{X_{mix\_pa_i}}$.

We assume there are no hidden common causes of variables in the observational data. In other words, we assume causal sufficiency. We also assume that the underlying causal graph in the observational data is acyclic, i.e., there exist no directed cycles in the graph.

4.3 Related work

The field of causal discovery has seen several interesting developments over the past few years. As mentioned in Chapter 1, identifying exact causal identities in the output graph has become possible by introducing assumptions about the data generating process. One of the first works that presented promising results was the Linear Non-Gaussian Additive Model (LINGAM) [Shi06]. LINGAM is able to recover the entire causal graph assuming that the data generating process is additive, linear and the noise structure follows a non-Gaussian distribution. Inspired by LINGAM's success, several methods were developed that could recover the exact causal graph. The Post-Nonlinear Model (PNL) was developed to find true causal graphs when the data is generated with nonlinear effect of the causes and the noise variables [ZH08]. Another set of methods use nonlinear regression and p-values from kernel-based statistical tests to identify the true parents of a variable [Hoy09; Moo09]. The idea behind these methods is to regress a variable on different sets of the remaining variables, and test for independence between the residuals and the variable set. The smallest variable set that leads to independence is considered as the true parent set. This process is repeated for every variable in the data. An improvement over these methods was proposed by using the least dependent residuals instead of relying on the hypothesis test [Pet14].

Recently, causal discovery has been presented as a learning problem of classifying the causal relationships between two variables [Guy13; Guy14]. The training set consists of a large number of cause-effect samples $\{(S_i, l_i)\}_{i=1}^{n}$ where each sample consists of data collected over two random variables X and Y. A sample is a tuple, $\{(x_{ij}, y_{ij})\}_{j=1}^{n_i}$ attested with a binary label l_i which indicates different types of relationships. Two state-of-the-art methods have emerged from these competitions by developing new features that are used to learn the patterns between different types of causal relationships [LP15b; Fon16]. However, these methods are not developed to build an entire multivariate causal graph.

In this work, we develop the first causal relationship prediction framework that builds a causal graph by predicting the true parent set of each variable. Our framework is developed for causal graphs generated from additive noise models.

4.4 Method

In this section we present the details of the causal relationship prediction in additive noise models (CRPAM) framework. There are three main steps involved, first simulate causal graphs and for each variable having at least one parent, build regression models by regressing the variable on its parents and non-parents respectively (Section 4.4.1). Second, embed the distributions of the residuals and the regressors from the regression models with kernel mean embedding (Section 4.4.2) and third, train a binary classifier to learn the patterns of the parents and the non-parents to predict the causal

relationships in the observational data (Section 4.4.2).

4.4.1 Data partition for building regression models

Causal discovery between two variables X and Y generated from an additive noise model, $Y = \beta X + \epsilon_Y$, can be performed by fitting two regression models: 1) a forward model $Y \sim X + \epsilon_Y$ and 2) a reverse model $X \sim Y + \epsilon_X$. Assuming the data follows a non-Gaussian distribution, the residuals from the forward model ϵ_Y should be independent of X, and the causality $X \rightarrow Y$ would be inferred. The Linear Non-Gaussian Additive Model (LiNGAM) was the first method to develop this idea. Several methods were later developed to exploit this property between the residuals and a parent set to identify the true parent set of each variable.

Our framework borrows the inspiration of the unique association between the residuals and the parent set to learn different patterns. In contrast to existing methods we do not perform statistical tests to identify the parent set. Instead, our aim is to find patterns between the residuals obtained by regressing a variable, X_i, on its parents, and the residuals from non-parents. We define a non-parent set as a set of variables that does not contain the parents and ancestors of X_i (see Definition 6). Regressing a variable, X_i, on its parent set yields a unique set of residuals that cannot be obtained from any other variable set. We posit that this pattern of residuals from the parent set is distinct from the pattern of residuals from any other variable set not containing the parent set. This idea forms the intuition of our method.

Assuming the underlying causal graph is sparse, we simulate data sets from $n_{cg} = 10$ different causal graphs with p variables and sparsity level $p_{con} = 2/(p-1)$, where p is the number of variables in the observational data. We explain the process of creating features from a causal graph, C_g, where $1 \leq g \leq n_{cg}$ and repeat this process n_{cg} times to build the training set. First, we generate $n_{train} = 100$ random data sets of sample size, n, from a simulated causal graph, C_g. Next, we identify the variables having at least one parent in the causal graph. These variables are referred to as the child variables, $\mathbf{X_{child}}$. For each $X_i \in \mathbf{X_{child}}$, we create three groups of variables. The first group contains variables that are parents of X_i denoted by $\mathbf{X_{pa_i}}$, the second group contains variables that are neither parents nor ancestors of X_i denoted by $\mathbf{X_{non\_pa_i}}$, and the third group consists of all the parents and ancestors of X_i, and it is denoted by $\mathbf{X_{mix\_pa_i}}$ (see Definition 6). Note that there is one set of variables in $\mathbf{X_{pa_i}}$ since there can be only one parent set of X_i. As a result, we obtain one set of residuals from the parent set. On the other hand, any variable that does not have a directed path towards X_i is its non-parent. As a result, the number of variable combinations from $\mathbf{X_{non\_pa_i}}$ can be exponential. This creates a significant imbalance in the distribution of residual patterns that can be learned from the parent set and the non-parent set. To prevent this imbalance, we created a third group of variables called the mix parent set, $\mathbf{X_{mix\_pa_i}}$, which contains all the variables in $\mathbf{X_{pa_i}}$ and any variable that has a directed path towards X_i. We believe that by including the parent set in $\mathbf{X_{mix\_pa_i}}$, the residuals

obtained by regressing X_i against the variables in this set would be similar to the residuals obtained from regressing against $\mathbf{X_{pa_i}}$.

In the next step, we build regression models to record the residuals of X_i where the regressors are chosen from: $\mathbf{X_{pa_i}}$, $\mathbf{X_{non\_pa_i}}$, and $\mathbf{X_{mix\_pa_i}}$ respectively. The regression model for the variable set, $\mathbf{X_{pa_i}}$, consists of the parent set as the regressor. On the other hand, there can be an exponential number of combinations to build a regressor set from $\mathbf{X_{non\_pa_i}}$ and $\mathbf{X_{mix\_pa_i}}$ respectively. To prevent an unnecessary computational load of building a large number of regression models, we randomly select few variable sets of different sizes. We create $r_{non\_pa_i}$ sets of randomly chosen variables from $\mathbf{X_{non\_pa_i}}$ where the size of these variable sets can grow from 1 to $s_{non\_pa_i}$. As mentioned earlier, $\mathbf{X_{mix\_pa_i}}$ contains the parents and ancestors of X_i. We build $r_{mix\_pa_i}$ regressor sets of a given size from $\mathbf{X_{mix\_pa_i}}$, where each regressor set is formed by taking the union of all the parents and randomly chosen ancestors. The size of the variable sets chosen from $\mathbf{X_{mix\_pa_i}}$ can grow from $n_{pa_i}+1$ to $n_{pa_i}+s_{mix\_pa_i}$, where n_{pa_i} is the number of variables in $\mathbf{X_{pa_i}}$. Thus, a total of $n_{reg} = 1 + r_{non\_pa_i} \cdot s_{non\_pa_i} + r_{mix\_pa_i} \cdot s_{mix\_pa_i}$ regressor sets are generated and an equal number of regression models are built. The residuals and the corresponding regressor sets from n_{reg} regression models are recorded. We refer to a function $\upsilon()$ that takes as input a variable, X_i and a variable set to produce a set of residuals and the corresponding regressor sets. This is illustrated as follows,

$$\upsilon(X_i, \mathbf{X_{pa_i}}) = \{(\epsilon_{ij}, X_{pa_{ij}})\}_{j=1}^{n_{train}} \tag{4.3}$$

$$\upsilon(X_i, \mathbf{X_{non\_pa_i}}) = \{\{(\epsilon_{ijk}, X_{non\_pa_{ijk}})\}_{j=1}^{n_{train}}\}_{k=1}^{r_{non\_pa_i} \cdot s_{non\_pa_i}} \tag{4.4}$$

$$\upsilon(X_i, \mathbf{X_{mix\_pa_i}}) = \{\{(\epsilon_{ijk}, X_{mix\_pa_{ijk}})\}_{j=1}^{n_{train}}\}_{k=1}^{r_{mix\_pa_i} \cdot s_{mix\_pa_i}} \tag{4.5}$$

We denote $\mathbf{S_{pa_i}} = \{(\epsilon_{ij}, X_{pa_{ij}})\}_{j=1}^{n_{train}}$ to contain the samples generated from equation 4.3. Similarly, $\mathbf{S_{non\_pa_i}}$ and $\mathbf{S_{mix\_pa_i}}$ contain the samples from equations 4.4 and 4.5 respectively.

4.4.2 Feature creation with kernel mean embeddings

For any prediction problem, one of the main concerns is to find the feature space that gives the best prediction performance. In order to find the discriminative patterns, we use kernel mean embedding to project the residuals and the corresponding regressor set into a new feature space. Recently, kernel mean embeddings have been used to create features by projecting the probability distribution P over a set of variables in d-dimensional space, $\mathbb{R}^d$, using a kernel function k. Specifically, the randomized causation coefficient (RCC) was developed to featurize the distributions to predict causal relationships in a bivariate case [LP15a; LP15b]. The mathematical notation of a kernel mean

embedding of a probability distribution P over a variable set in $\mathbb{R}^d$ is shown below,

$$\mu_k(P) = \int_{\mathbb{R}^d} k(x,.)dP(x) \in \mathscr{H}_k \tag{4.6}$$

where $\mathscr{H}_k$ is the reproducible kernel Hilbert space (RKHS) associated with the kernel function k. The prediction performance of RCC outperformed state-of-the-art methods when evaluated on an independent test set. However, their method has not been developed to construct the full multivariate causal graph. Nonetheless, motivated by their results, we use kernel mean embeddings to create features from the distributions of the residuals and the corresponding set of regressors to build the training set.

In this work, we calculate the embeddings by computing the kernel matrix $\mathbf{K}$ from the empirical distribution. Specifically, we consider the gaussian kernel function, k, for embedding the distributions due to its attractive property of uniquely projecting every distribution into a new feature space i.e., $||\mu_k(P) = \mu_k(Q)||$ iff $P = Q$. The mapping of two samples x and x' with the Gaussian kernel is defined by the following equation,

$$k(x, x') = \exp\big(-\gamma||x - x'||_2^2\big) \tag{4.7}$$

where $\gamma > 0$ and is known as the inverse kernel width.

In the previous subsection we explained how three different groups of variables: $\mathbf{X_{pa_i}}$, $\mathbf{X_{non\_pa_i}}$ and $\mathbf{X_{mix\_pa_i}}$ are used to generate the sample sets, $\mathbf{S_{pa_i}}$, $\mathbf{S_{non\_pa_i}}$ and $\mathbf{S_{mix\_pa_i}}$ (see equations 4.3-4.5). Each sample set contains the empirical distribution of the residuals and the regressor sets. For the parent set of a variable X_i, we refer to these distributions as P_{ϵ_i, pa_i} and P_{X, pa_i} respectively. These distributions are featurized by embedding them into kernel matrices using the gaussian kernel function. Thus, for a given pair of residuals and regressor set, we compute three kernel matrices: $\mathbf{K}_{\epsilon_i}$, $\mathbf{K}_{pa_i}$ and $\mathbf{K}_{\epsilon_i, pa_i}$. To summarize the information contained in a kernel matrix, we take the column-wise mean over the matrix and featurize the distributions as illustrated in the following equation,

$$m_k(P_{\epsilon_i, pa_i}, P_{X, pa_i}) = \left\{\left(\overline{\mathbf{K}}_{pa_{il}}, \overline{\mathbf{K}}_{\epsilon_{il}}, \overline{\mathbf{K}}_{\epsilon_{il}, pa_{il}}\right)\right\}_{l=1}^{|\mathbf{S_{pa_i}}|} \tag{4.8}$$

where $\overline{\mathbf{K}}_{pa_{il}}$, $\overline{\mathbf{K}}_{\epsilon_{il}}$, and $\overline{\mathbf{K}}_{\epsilon_{il}, pa_{il}}$ represent the column-wise mean vectors of the corresponding kernel matrices $\mathbf{K}_{pa_{il}}$, $\mathbf{K}_{\epsilon_{il}}$ and $\mathbf{K}_{\epsilon_{il}, pa_{il}}$.

The output of this equation is a matrix with the number of rows equal to the number of pairs of residuals and regressor sets in $\mathbf{S_{pa_i}}$, and the number of columns equal to $3 \cdot n$ where n is the number of observations in the data. Likewise, the empirical distributions from the sample sets $\mathbf{S_{non\_pa_i}}$ and $\mathbf{S_{mix\_pa_i}}$ are featurized using the above equation.

In the last step we attest a class label for every pair of featurized residual and regressor set that will

be used as the ground truth for training a classifier. Recall from section 4.4.1 that every regressor set from $\mathbf{X_{mix\_pa_i}}$ will be a superset of the parent set, $\mathbf{X_{pa_i}}$, therefore we expect the residual embeddings obtained from these two variable sets to be similar to each other than to $\mathbf{X_{non\_pa_i}}$. Based on this assumption, every pair of featurized residuals and regressor set from $\mathbf{X_{pa_i}}$ and $\mathbf{X_{mix\_pa_i}}$ is labeled as "+1", and the featurized residuals and the regressor sets from $\mathbf{X_{non\_pa_i}}$ are labeled as "0". The embedded distributions of the residuals and the regressor sets: $m_k(P_{\epsilon_i,pa_i}, P_{X,pa_i})$, $m_k(P_{\epsilon_i,non\_pa_i}, P_{X,non\_pa_i})$, and $m_k(P_{\epsilon_i,mix\_pa_i}, P_{X,mix\_pa_i})$ generated from a causal graph C_g are combined row-wise to form a part of the training set.

$$\begin{bmatrix} (m_k(P_{\epsilon_i,pa_i}, P_{X,pa_i}), +1) \\ (m_k(P_{\epsilon_i,non\_pa_i}, P_{X,non\_pa_i}), 0) \\ (m_k(P_{\epsilon_i,mix\_pa_i}, P_{X,mix\_pa_i}), +1) \end{bmatrix} \tag{4.9}$$

The above mentioned process is repeated for each variable in causal graph C_g, and then performed for n_{cg} simulated causal graphs. The training set is built by combining all the embedded distributions from n_{cg} graphs.

A random forest classifier is used to learn the patterns of parents and non-parents from the training set. The classifier is then used to predict causal relationships in the observational data. Prior to making predictions, we need to convert the observational data in the same feature space as the training set. The projection of the observational data into this new feature space is simpler than the training set since the parents and non-parents are not known. Thus we do not have to create separate sample sets for creating features. Instead, for each variable X_j in the observational data, we create a combination of variables that will be considered as its parent set. These variable sets are featurized as per the steps mentioned in Sections 4.4.1 and 4.4.2. From hereon, we will refer to the transformed observational data as the test set. While generating combinations of variables, the size of the combinatorial variable set can grow from 1 to s_{test}. Note that we do not set the value of s_{test} higher than the maximum size of the variable set used to build the training set. In other words, s_{test} is always less than or equal to $max(s_{non\_pa_i}, n_{pa_i} + s_{mix\_pa_i})$. Deciding the value of s_{test} is a trade-off between time consumption and accuracy. Initializing s_{test} to a high value creates exponentially more combinations of variables for building the test set. As a result, this increases the runtime to featurize the distributions.

Once the classifier is trained, for each variable in the test set we predict whether a variable combination is closer to the pattern of a parent set or a non-parent set. Ideally, if there exists a parent set of a variable, the classifier should predict only the variable combination containing the parent set as the true parent set. Unfortunately, this does not happen in practice and the classifier often predicts more than one combination of variables as the true parent set. We develop a heuristic to aggregate these predictions. To do this, we keep a track of the frequency of each variable being predicted as the true parent set. In the end, the predicted parent set contains the variables having

more than 50% frequency.

4.5 Simulations

We perform a series of simulations to evaluate the effectiveness of the proposed causal relationship prediction framework.

4.5.1 Synthetic Data

We simulate $n_{cg} = 10$ synthetic sparse causal graphs of p variables to build the training set. The value of p is equal to the number of variables in the observational data. To enforce sparsity, we set the probability of an edge being present between two variables to be $p_{con} = 2/(p-1)$. The resulting causal graph is expected to have p edges which is considered as a sparse setting [Pet14]. We generate $n_{train} = 100$ synthetic datasets for each causal graph and leverage the true causal relationships to build the training set (see Section 4.4). While building the regression models, for any child node, $X_i \in \mathbf{X_{child}}$, we initialize $m_{non\_pa} = 3$, $m_{mix\_pa} = 2$, $1 \leq k_{non\_pa_i} \leq 4$, and $1 \leq k_{mix\_pa_i} \leq 3$. These values are chosen after conducting several experiments and observing their impact on runtime. We test the framework to predict causal relationships in test set with variables $p = \{10, 15\}$ across two different sample sizes, $n = \{100, 200\}$. While embedding the test data into the new feature space, we initialize $k_{test} = 5$ for $p = 10$, and $k_{test} = 3$ for $p = 15$. We initialize the inverse kernel width, $\gamma = 1$ to generate the kernel embeddings as shown in equation 4.7

We generate 20 test sets from a linear as well as a non-linear setting for a given set of variables and sample size. For the linear setting, we use linear regression and for the nonlinear setting we use generalized additive model regression to obtain the residuals. The output causal graph is evaluated in terms of the structural hamming distance (SHD) which calculates the number of edges to be added, removed or flipped in the estimated causal graph to match the true causal graph; the accuracy value $d = \sqrt{(1-precision)^2 + (1-recall)^2}$ where precision is the fraction of true edges found in the estimated causal graph, and recall is the fraction of true edges in the estimated graph that are also present in the true causal graph.

We compare the performance of CRPAM against regression with subsequent independence test (RESIT), greedy DAG search (GDS), linear non-Gaussian additive models (LINGAM), the PC algorithm with Fisher's Z test and significance level of 0.05, the greedy equivalence search (GES), and max-min hill climbing algorithm (MMHC).

4.5.1.1 Models for Data Generation

We simulate training set and test set from two kinds of models: linear structural equation models and non-linear structural equation models (see Definition 4 and 5). We follow the procedure mentioned

Table 4.1 Performance metrics for Linear Structural Equation Models on sparse causal graphs, $p_{con} = 2/(p-1)$. The reported metric is its mean value over 20 simulations. The best performance is highlighted.

	CRPAM	RESIT	GDS	LINGAM	PC	GES	MMHC
$p = 10, n = 100$							
SHD	7.3 ± 3.3	14.7 ± 3.8	8.3 ± 4.5	7.5 ± 2.8	8.4 ± 3.4	7.3 ± 2	6.6 ± 3.9
d	0.4 ± 0.2	0.7 ± 0.1	0.6 ± 0.3	0.7 ± 0.1	0.4 ± 0.2	0.3 ± 0.1	0.4 ± 0.2
$p = 15, n = 100$							
SHD	8.7 ± 3.5	25.9 ± 11.8	14.3 ± 5.2	10.2 ± 2.4	11.6 ± 3.6	10.7 ± 3.1	9.8 ± 3.1
d	0.5 ± 0.2	0.8 ± 0.1	0.7 ± 0.2	0.7 ± 0.1	0.4 ± 0.1	0.4 ± 0.1	0.4 ± 0.1
$p = 10, n = 200$							
SHD	3.8 ± 2.8	15.8 ± 7.7	4.9 ± 3.2	4.4 ± 3	7 ± 3.6	6.7 ± 4	6 ± 3.5
d	0.3 ± 0.2	0.6 ± 0.2	0.3 ± 0.2	0.4 ± 0.2	0.3 ± 0.2	0.3 ± 0.2	0.4 ± 0.2
$p = 15, n = 200$							
SHD	7.3 ± 3.4	36.6 ± 10.7	11.1 ± 5.5	6.8 ± 2.8	11 ± 3.7	10.3 ± 3.8	8.4 ± 3.4
d	0.4 ± 0.1	0.8 ± 0.1	0.5 ± 0.2	0.4 ± 0.1	0.3 ± 0.1	0.3 ± 0.1	0.4 ± 0.2

in [Pet14] and use their publicly available code[1] to simulate causal graphs and synthetic data sets. In the linear setting, each variable is a linear combination of its parents and an additive non-Gaussian noise variable (see Definition 4). The coefficients β_{ij} are uniformly chosen from [-2, -0.1] $\cup$ [0.1, 2] and the noise variables are independent and distributed according to $K_i \cdot \text{sign}(M_i) \cdot |M_i|^{\alpha_i}$ where M_i is normally distributed with mean 0 and standard deviation 1, K_i is uniformly distributed between [0.1, 0.5] and α_i is also uniformly distributed between [2, 4]. We also generate test data sets from nonlinear SEMs (see Definition 5). The nonlinear functions f_{ij} are sampled from a Gaussian process with bandwidth one.

Table 4.1 summarizes the performance of the CRPAM framework against state-of-the-art causal discovery methods with Linear SEMs. We observe that CRPAM has the lowest SHD in two out of four cases and the next best SHD in the remaining two cases. The performance of CRPAM improves with increasing sample size. LINGAM and MMHC have the lowest SHD values in one case each. However, LINGAM performs better than MMHC with increasing sample size. The PC and GES algorithms have the lowest accuracy value d in most cases with GES having slightly better mean SHD values than PC. However, their SHD values are not the lowest which indicates that the output graph contains false positives or the output edges have incorrect orientations or it could be both. While the accuracy value d of CRPAM is not the best in all cases, it is consistently better than RESIT, GDS, LINGAM and MMHC.

There is no method that performs the best in all cases across both performance metrics. Nonethe-

[1] `https://github.com/bquast/ANM/tree/master/codeANM`

Table 4.2 Performance metrics for Nonlinear Structural Equation Models on sparse causal graphs, $p_{con} = 2/(p-1)$. The reported metric is its mean value over 20 simulations. The best performance is highlighted.

	CRPAM	RESIT	GDS	LINGAM	PC	GES	MMHC
$p = 10, n = 100$							
SHD	4.6 ± 3.3	8.9 ± 3.6	9 ± 4.9	10.9 ± 3	10.7 ± 3.3	14 ± 4.6	9.8 ± 3.2
d	0.2 ± 0.2	0.6 ± 0.2	0.5 ± 0.2	0.8 ± 0.2	0.5 ± 0.1	0.6 ± 0.2	0.6 ± 0.1
$p = 15, n = 100$							
SHD	4.6 ± 2.9	14 ± 5.3	16.6 ± 6	14 ± 3.4	13.6 ± 3.5	22.7 ± 5.1	12 ± 4.2
d	0.3 ± 0.2	0.7 ± 0.2	0.6 ± 0.2	0.9 ± 0.2	0.5 ± 0.2	0.8 ± 0.2	0.5 ± 0.2
$p = 10, n = 200$							
SHD	2.7 ± 2.5	5.8 ± 4	8 ± 5.1	10 ± 3	9.3 ± 3.2	13.3 ± 5	8.2 ± 3.3
d	0.1 ± 0.1	0.4 ± 0.2	0.5 ± 0.1	0.5 ± 0.2	0.5 ± 0.2	0.7 ± 0.2	0.5 ± 0.2
$p = 15, n = 200$							
SHD	2.7 ± 3	9 ± 3.7	13.9 ± 8	14.4 ± 3.8	12.3 ± 3.5	23.3 ± 7.4	11.1 ± 4.4
d	0.2 ± 0.2	0.4 ± 0.1	0.5 ± 0.2	0.6 ± 0.1	0.5 ± 0.1	0.7 ± 0.1	0.4 ± 0.2

less, one can conclude that CRPAM and GES seem to perform better for linear SEMs compared to the remaining methods.

Table 4.2 presents the results with non-linear SEMs. We observe that CRPAM significantly outperforms all the other methods. The mean values of SHD and accuracy value d output by CRPAM are the lowest in all cases. There is a minimum difference of at least 3 and 0.2 in the mean values of SHD and d respectively when compared with the next best performing method. Note that both CRPAM and RESIT make use of the nonlinearity of the functions in the SEMs to identify causal relationships. However, the features extracted by CRPAM seem to have greater discriminative power in identifying the true parent sets than RESIT. These results suggest that there is a distinct pattern in the featurized distributions of the residuals and the true parent sets that differentiates them from the non-parent sets.

Motivated by the performance of CRPAM with nonlinear SEMs we also perform another set of experiments. We increase the complexity of the simulated causal graphs by setting $p_{con} = 2*2/(p-1)$ so that the probability of an edge between two variables is twice that in sparse causal graphs. We refer to this setting as dense causal graphs. Next, we generate data with $p = \{10, 15\}$, $n = 100$ and compare the performance with the other methods. All the parameters used in the previous experiments remain the same as mentioned in Section 4.5.1. The results are presented in Table 4.3. We observe that CRPAM continues to outperform all the methods in terms of both SHD and accuracy d. The

Table 4.3 Performance metrics for Nonlinear Structural Equation Models on dense causal graphs, $p_{con} = 2*2/(p-1)$. The reported metric is its mean value over 20 simulations. The best performance is highlighted.

	CRPAM	RESIT	GDS	LINGAM	PC	GES	MMHC
$p = 10, n = 100$							
SHD	11.2 ± 4.6	17.3 ± 5.6	20.4 ± 5.4	18.8 ± 3.9	18.2 ± 3.6	21.6 ± 4.5	16.6 ± 3.4
d	0.3 ± 0.1	0.6 ± 0.1	0.6 ± 0.1	0.9 ± 0.1	0.7 ± 0.1	0.6 ± 0.1	0.7 ± 0.1
$p = 15, n = 100$							
SHD	16.7 ± 8.3	32 ± 8.8	36.1 ± 9	29.5 ± 6.4	28.2 ± 6.9	39.2 ± 7.8	26.4 ± 7.5
d	0.5 ± 0.2	0.8 ± 0.1	0.7 ± 0.1	0.9 ± 0.1	0.7 ± 0.1	0.7 ± 0.1	0.7 ± 0.1

Table 4.4 Performance metrics for real-world protein network. The reported metric is its mean value over 10 simulations. The best performance is highlighted.

	CRPAM	RESIT	GDS	LINGAM	PC	GES	MMHC
$p = 11, n = 100$							
SHD	19 ± 2	19.7 ± 2.4	18.4± 2	18.8 ± 1.03	19.2 ± 0.9	19.4 ± 1.7	16.9 ± 1
d	0.7 ± 0.1	0.83 ± 0.1	0.72± 0.1	1.04 ± 0.1	0.76 ± 0.04	0.74 ± 0.1	0.75 ± 0.04
$p = 11, n = 200$							
SHD	17.3 ± 0.9	20.4 ± 1.8	18.8± 1	17.3 ± 0.7	18.7 ± 0.9	19.6 ± 0.7	16.4 ± 1.2
d	0.67 ± 0.1	0.73 ± 0.1	0.68 ± 0.1	0.81 ± 0.1	0.69 ± 0.1	0.7 ± 0.1	0.71 ± 0.1

next best performing method has 1.5 times the mean SHD value than CRPAM and at least 1.4 times the mean accuracy d.

4.5.1.2 Real-world Data

We now illustrate the performance of CRPAM framework on a real-world problem of reconstructing a protein network [Sac05]. We used Gaussian Process regression to model the relationships between each variable and its set of parents. Experiments were performed over two different sample sizes, 100, 200 and the reported values are averaged over 10 runs.

Table 4.4 shows the results of our framework along with the remaining causal discovery methods. We observe that CRPAM has the lowest accuracy value, d. This indicates that CRPAM recovers more edges in the output causal structure compared to the other methods. However, it does not have the lowest SHD value when compared to MMHC and LiNGAM. This indicates that some edge orientations in the output causal graph are incorrect. While MMHC has lowest SHD value, its high accuracy value, d, suggests that it does not recover as many true edges as CRPAM.

The selection of the best causal discovery method on real-world problems is not easy and one must assess the trade-offs involved before making a decision. If domain knowledge is available

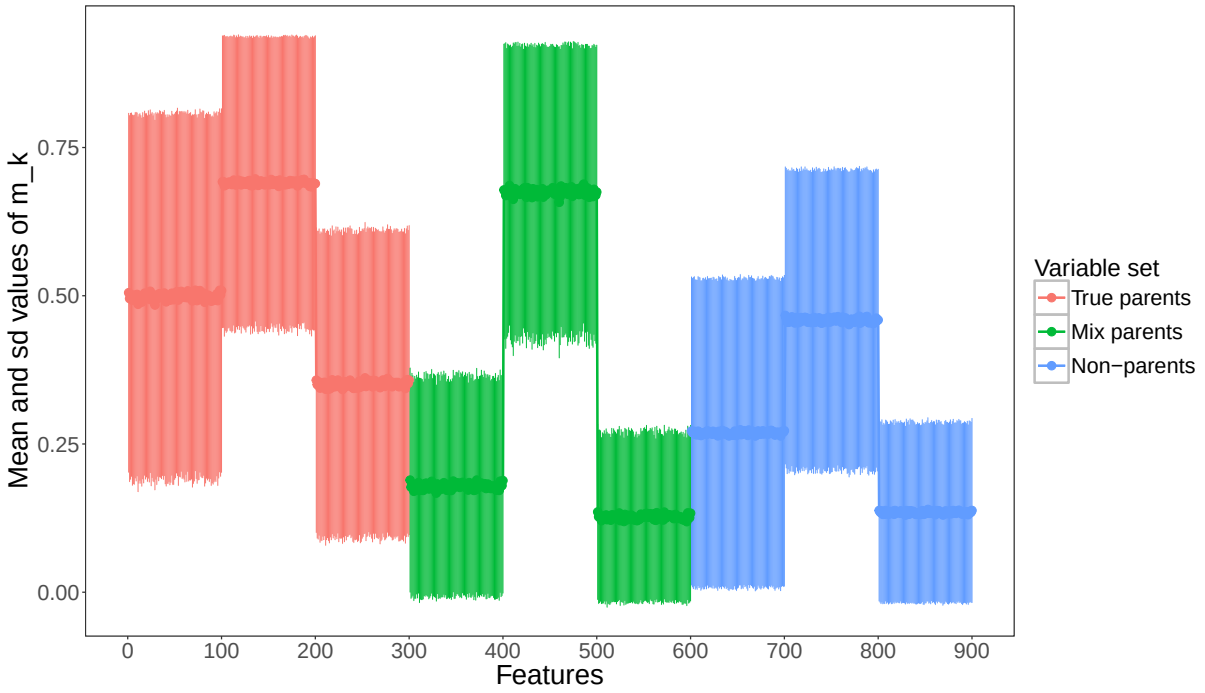


Figure 4.1 The mean and standard deviation values of features generated from the true parents (see Equation 4.8) (1:300), mix-parents (301:600), and non-parents (601:900) for all the child variables in the training set with $p = 10$ and $n = 100$ for linear SEMs. The first hundred features in a variable set represent the embedding of the regressor set, the next hundred represent the embedding of residuals and the last hundred represent the embedding of both. The true-parent set (pink) and the mix parent set (green) are assigned the same class label and the non-parent set (blue) is assigned a different class label.

then the output from CRPAM might be more useful as it recovers more edges in the output graph. The incorrect edge orientations can be examined using domain knowledge. However, if no such expertise is available, using a causal discovery method that has lowest false positive rate might be a better option.

4.5.1.3 Feature Analysis

The prediction performance of CRPAM on linear SEMs and nonlinear SEMs is driven by the features created using kernel mean embedding (see Section 4.4.2). We provide some understanding behind the behavior of our framework by presenting a visual analysis of the distribution of the features in the training sets and their importance in training the classifier. Recall from equation 4.9 that the distributions of the residuals and the corresponding regressor sets across three sample sets were embedded into a new feature space. The features from the true parent set are stored in $m_k(P_{res,pa}, P_{X,pa})$, the features from the non-parent set are stored in $m_k(P_{res,non\_pa}, P_{X,non\_pa})$, and the features from the third group with the mix parent set are stored in $m_k(P_{res,mix\_pa}, P_{X,mix\_pa})$.

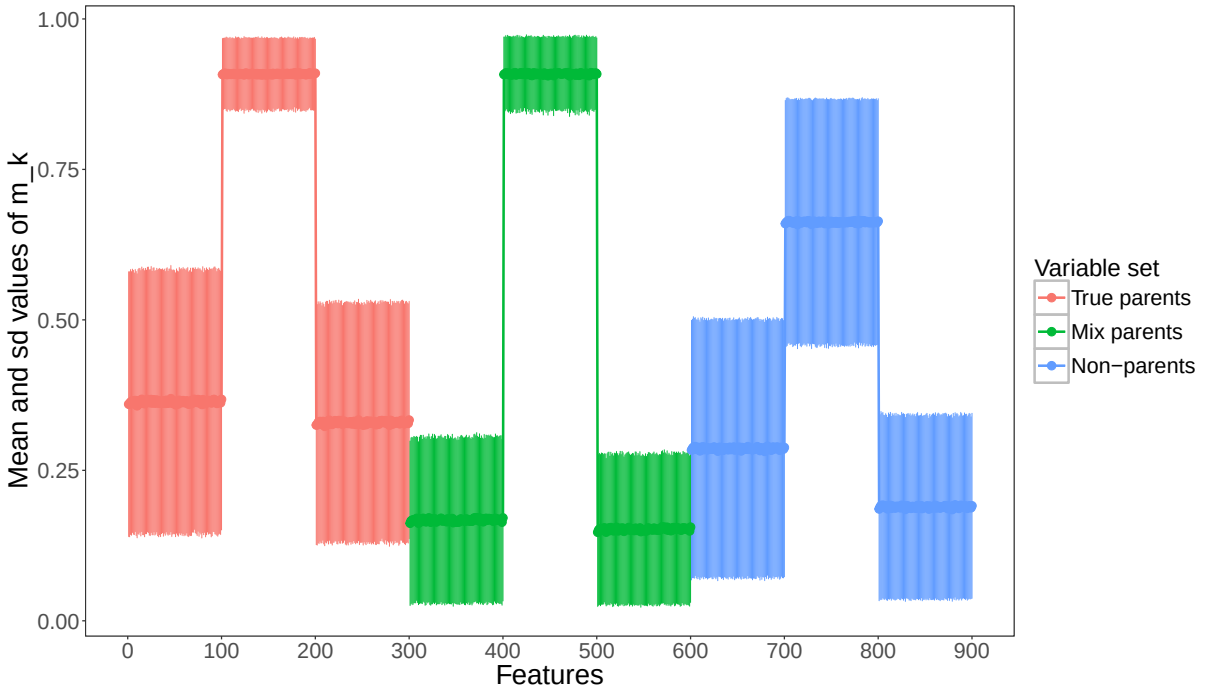


Figure 4.2 The mean and standard deviation values of features generated from the true parents (see Equation 4.8) (1:300), mix-parents (301:600), and non-parents for all the child variables in the training set with $p = 10$ and $n = 100$ for nonlinear SEMs. The first hundred features in a variable set represent the embedding of the regressor set, the next hundred represent the embedding of residuals and the last hundred represent embedding of both. The true-parent set (pink) and the mix parent set (green) are assigned the same class label and the non-parent set (blue) is assigned a different class label.

Figures 4.1 and 4.2 summarize the information contained in the training set with $p = 10$ and $n = 100$ for linear SEMs and nonlinear SEMs respectively. In particular, we present the mean and standard deviation values of the features in the training set. For sample size $n = 100$ the training set has $3n$ features where the first hundred features are the embeddings of the regressor set, the next hundred features are the embeddings of the residuals and the last hundred are the embeddings of both the regressor set and the residuals (see equation 4.8). In figure 4.1, we observe that the mean and standard deviation values of the residual embeddings of the true parent set and the mix parent set are similar to each other than the non-parent set. Although there is some discriminative pattern between the two classes, there is a high overlap between their values. This can potentially explain the prediction performance of CRPAM on linear SEMs (see Table 4.1).

Figure 4.2 presents the same information for nonlinear SEMs. Although there is some overlap between the features, we observe a distinct pattern in the residual embeddings with very little overlap. These features are very similar for the true parent set and the mix parent set but different for the non-parent set. The mean values of the residual embeddings from the true parent set and

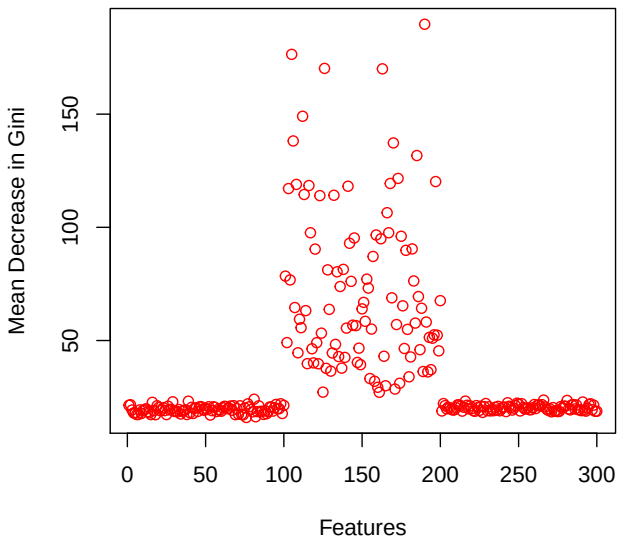


Figure 4.3 The importance of features in training set with $p = 10$ and $n = 100$ in linear SEMs. The features are represented on the X-axis and the variable importance is represented on the Y-axis. The importance of a variable is measured by the mean decrease in gini if the variable were included in training the classifier. The first hundred features represent the embeddings of the parent set, the next hundred features represent the embedding of the residuals and the last hundred features represent embedding of both.

the mix parent set are centered around 0.9 with a very small standard deviation. On the other hand, the same values for non-parent sets are centered around 0.7 with a much higher standard deviation. We believe this discriminative pattern would be leveraged by the classifier to accurately predict the true causal relationships. This pattern also supports our assumption made in Section 4.4.2 about the residual embeddings being similar for the true parent set and the mix parent set.

Lastly, we analyze the features based on their ability to discriminate the patterns of the true parent sets and the non-true parent sets. Figure 4.3 shows the feature importance from the random forest classifier for linear SEMs and nonlinear SEMs. The importance of a feature is measured by the mean decrease in Gini if that variable was included in training the classifier. In figure 4.4 we observe that the features created by embedding the residuals are the most important to the classifier as they lead to highest reduction in mean Gini values. This confirms our observations from figures 4.1 and 4.2. Additionally, this also provides insight into reducing the feature space by excluding the other two sets of features containing the regressor sets. It would be interesting to experiment with only the features containing the residual embedding given this evidence. The benefits would be twofold: one, this would lead to a 66% reduction in the number of features and two, it would significantly

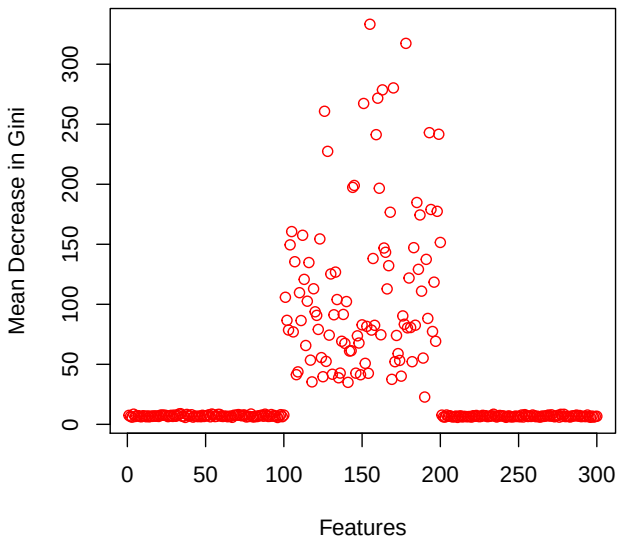


Figure 4.4 The importance of features in training set with $p = 10$ and $n = 100$ in nonlinear SEMs. The features are represented on the X-axis and the variable importance is represented on the Y-axis. The importance of a variable is measured by the mean decrease in gini if the variable were included in training the classifier. The first hundred features represent the embeddings of the parent set, the next hundred features represent the embedding of the residuals and the last hundred features represent embedding of both.

reduce the runtime of the framework. We leave these experiments as part of our future work.

4.6 Conclusions and future work

In this work, we have approached the problem of causal discovery as a learning problem. The proposed framework builds on the ideas of additive models to create discriminative patterns for true parent sets and non-parent sets from kernel mean embeddings. A nonlinear binary classifier is trained to learn these patterns and predict the parent set of each variable in the test data. The framework is evaluated on linear SEMs and nonlinear SEMs to demonstrate its prediction performance. Finally, we present evidence of a strong discriminative pattern of the features in nonlinear SEMs which gives insight into the performance of the framework.

We have planned the research progress of the framework in three directions. First, perform an analysis of the different parameters used in the framework for differing sizes of the data and network complexity. Second, develop a parallel version of the framework since all the steps can be easily parallelized. A parallel CRPAM framework would greatly reduce the runtime and provide better scal-

ability. Third, develop ensemble methods for data sets with large sample size that extracts patterns from different parts of the data and performs majority voting to predict the causal relationships.

CHAPTER

5

CONCLUSION

Discovering causal relationships from observational data has been one of the most fundamental questions in scientific domains. Studying every causal relationship can provide meaningful insight about the behavior of factors in the underlying complex system. Another useful application of causal discovery involves estimating the Markov Blanket of a response variable of interest for feature selection and predictive modeling. Validating every causal relationship requires significant amount of resources and the estimated local causal structure of a target variable is vulnerable to statistical errors and noise in the data. Therefore, developing methods that can discover causal structure effectively has multiple benefits.

In this dissertation, we focused on the problem of causal discovery in the context of feature selection and causal structure learning. Estimating the Markov Blanket of a target variable provides a reduced set of features for predictive modeling. However, errors made by statistical tests can provide an incomplete Markov Blanket that leads to suboptimal prediction performance. In Chapter 2, we developed a method to incorporate causal structure learning and causal effect estimation to identify meaningful predictive features. In Chapter 3, we identified areas of improvement for a well-known causal discovery method and introduced a graph partitioning based approach to improve its quality of causal graphs. Finally, in Chapter 4 we transitioned to additive noise models for accurately predicting causal relationships using feature engineering.

A summary of our main contributions across the dissertation are as follows,

First, we presented a feature selection method that incorporates causal effect estimation along

with causal structure learning, allows us to identify meaningful predictive features with respect a response variable of interest. In Chapter 2 we showed that the features selected by our method contains greater predictive power compared to state-of-art feature selection methods. We also evaluated the physical interpretability of the frequently selected climate indices as features and most of them were found to be in agreement with the climate literature. Our method also presents a list of prioritized causal relationships which can be studied by domain scientists to confirm existing hypothesis.

Second, we developed a graph partitioning based approach to estimate a causal structure using a divide-and-conquer strategy. The key idea is to divide a large graph into smaller subgraphs and remove spurious associations using a clever ordering of pairwise edges based on the structural properties of the graph and pairwise association between variables. The results from smaller subgraphs are merged together in a bottom-up fashion to obtain the full causal structure. Results on real-word networks of differing data types and complexities demonstrate the effectiveness of our method.

In the final component, we transition to additive noise models (ANMs) for discovering causal relationships from non-Gaussian data. ANMs have become popular due to their ability of extracting the exact causal identities from observational data assuming that the variables follow non-Gaussian distribution. We present a framework for predicting causal relationships in a multivariate setting by creating features using kernel embeddings. Simulations show the effectiveness of our framework on non-linear data. Results on real-world data show that our method shows higher predictive power compared to existing ANM based methods. However, approximation based approaches are needed to engineer features in large data sets.

5.1 Future Work

We conclude this dissertation by identifying potential areas of future directions that can benefit from our work.

First, we postulated that incorporating causal effect estimation to learn the local causal structure of a response variable of interest can identify meaningful predictive features. In this work, we used a well-known constraint-based structure learning algorithm to identify causal relationships. A simple extension of this work would be to learn causal structures using more sophisticated approaches such as greedy or hybrid methods as mentioned Chapter 1.

Second, predicting causal relationships from observational data relies on the underlying data generation mechanism. In Chapter 4 we showed that additive noise models are effective when the data follows non-Gaussian distribution. ANMs can benefit immensely if the underlying mechanism of data generation were known. However, identifying the true mechanism of data generation is often difficult or impossible. To overcome this issue, causal graphs can be simulated from a broader class of non-Gaussian distributions. Patterns of causal relationships can be learned by performing

feature engineering and training classifiers. While this increases the complexity of training sets, recent advancements in deep learning can be leveraged to learn discriminative patterns effectively. Recent works have already explored the application of deep learning to reconstruct the causal graphs [Gou17; Kal18]. However, their application in learning to predict causal relationship remains to be explored.

BIBLIOGRAPHY

[Abe06] Abellán, J. et al. "Some Variations on the PC Algorithm." *Probabilistic Graphical Models*. 2006, pp. 1–8.

[Ali03] Aliferis, C. F. et al. "HITON: a novel Markov Blanket algorithm for optimal variable selection". *AMIA Annu. Symp. Proc.* Vol. 2003. 2003, p. 21.

[Ali10a] Aliferis, C. F. et al. "Local causal and markov blanket induction for causal discovery and feature selection for classification part i: Algorithms and empirical evaluation". *J. Mach. Learn. Res.* **11**.Jan (2010), pp. 171–234.

[Ali10b] Aliferis, C. F. et al. "Local causal and markov blanket induction for causal discovery and feature selection for classification part ii: Analysis and extensions". *Journal of Machine Learning Research* **11**.Jan (2010), pp. 235–284.

[And97] Andersson, S. A. et al. "A characterization of Markov equivalence classes for acyclic digraphs". *Ann. Stat.* **25**.2 (1997), pp. 505–541.

[And04] Andrews, E. et al. "Influence of ENSO on flood frequency along the California coast". *J. Climate* **17**.2 (2004), pp. 337–348.

[BL03] Bader, J. & Latif, M. "The impact of decadal-scale Indian Ocean sea surface temperature anomalies on Sahelian rainfall and the North Atlantic Oscillation". *Geophysical Research Letters* **30**.22 (2003).

[BL11] Bader, J. & Latif, M. "The 1983 drought in the West Sahel: a case study". *Climate Dynam.* **36**.3 (2011), pp. 463–472.

[BH95] Benjamini, Y. & Hochberg, Y. "Controlling the false discovery rate: a practical and powerful approach to multiple testing". *Journal of the royal statistical society. Series B (Methodological)* (1995), pp. 289–300.

[Bla03] Black, E. et al. "An observational study of the relationship between excessively strong short rains in coastal East Africa and Indian Ocean SST". *Monthly Weather Review* **131**.1 (2003), pp. 74–94.

[Blo08] Blondel, V. D. et al. "Fast unfolding of communities in large networks". *J. Stat. Mech. Theor. Exp.* **2008**.10 (2008), P10008.

[Bra07] Brandes, U. et al. "On finding graph clusterings with maximum modularity". *International Workshop on Graph-Theoretic Concepts in Computer Science*. Springer. 2007, pp. 121–132.

[Büh14] Bühlmann, P. et al. "High-dimensional statistics with a view toward applications in biology". *Annu. Rev. of Stat. Appl.* **1** (2014), pp. 255–278.

[Cai13] Cai, R. et al. “Sada: A general framework to support robust causation discovery”. *International Conference on Machine Learning*. 2013, pp. 208–216.

[CP02] Camberlin, P & Philippon, N. “The East African March-May rainy season: Associated atmospheric dynamics and predictability over the 1968-97 period”. *J. Climate* **15**.9 (2002), pp. 1002–1019.

[Cha16] Chaudhary, M. S. et al. “Causality-Guided Feature Selection”. *Advanced Data Mining and Applications*. Springer. 2016, pp. 391–405.

[Cha17] Chaudhary, M. S. et al. “A Community-Driven Graph Partitioning Method for Constraint-Based Causal Discovery”. *International Workshop on Complex Networks and their Applications*. Springer. 2017, pp. 253–264.

[Che12] Chen, Z. et al. “Spice: discovery of phenotype-determining component interplays”. *BMC Syst. Biol.* **6**.1 (2012), pp. 1–19.

[Chi12] Chindelevitch, L. et al. “Causal reasoning on biological networks: interpreting transcriptional changes”. *Bioinformatics* **28**.8 (2012), pp. 1114–1121.

[Cla04] Clauset, A. et al. “Finding community structure in very large networks”. *Physical review E* **70**.6 (2004), p. 066111.

[CM14] Colombo, D. & Maathuis, M. H. “Order-independent Constraint-based Causal Structure Learning”. *J. Mach. Learn. Res.* **15**.1 (2014), pp. 3741–3782.

[Col12] Colombo, D. et al. “Learning high-dimensional directed acyclic graphs with latent and selection variables”. *Ann. Stat.* (2012), pp. 294–321.

[EUD12] Ebert-Uphoff, I. & Deng, Y. “Causal Discovery for Climate Research Using Graphical Models”. *J. Climate* **25**.17 (2012), pp. 5648–5665.

[EUD14] Ebert-Uphoff, I. & Deng, Y. “Causal Discovery from Spatio-Temporal Data with Applications to Climate Science”. *Proc. of the 2014 13th Int Conf. on Machine Learning and Applications*. IEEE, 2014, pp. 606–613.

[Fon16] Fonollosa, J. A. “Conditional distribution variability measures for causality detection”. *arXiv preprint arXiv:1601.06680* (2016).

[Gen05] Geng, Z. et al. “Decomposition of search for v-structures in DAGs”. *J. Multivariate Anal.* **96**.2 (2005), pp. 282–294.

[Gou17] Goudet, O. et al. “Causal Generative Neural Networks”. *arXiv preprint arXiv:1711.08936* (2017).

[Guy13] Guyon, I. *Cause-effect pairs kaggle competition*. 2013. URL: `https://www.kaggle.com/c/cause-effect-pairs/`.

[Guy14] Guyon, I. *Chalearn fast causation coefficient challenge*. 2014. URL: https://www.codalab.org/competitions/1381.

[Guy07] Guyon, I. et al. "Causal feature selection". *Computational methods of feature selection* (2007), pp. 63–86.

[Har14] Harenberg, S. et al. "Community detection in large-scale networks: a survey and empirical evaluation". *Wiley Interdisciplinary Reviews: Computational Statistics* **6**.6 (2014), pp. 426–439.

[Hon17] Hong, Y. et al. "An efficient algorithm for large-scale causal discovery". *Soft Computing* **21**.24 (2017), pp. 7381–7391.

[Hoy09] Hoyer, P. O. et al. "Nonlinear causal discovery with additive noise models". *Advances in neural information processing systems*. 2009, pp. 689–696.

[Iye13] Iyer, S. P. et al. "Inferring functional connectivity in MRI using Bayesian network structure learning with a modified PC algorithm". *Neuroimage* **75** (2013), pp. 165–175.

[Jan12] Janzing, D. et al. "Information-geometric approach to inferring causal directions". *Artificial Intelligence* **182** (2012), pp. 1–31.

[Jov15] Jović, A. et al. "A review of feature selection methods with applications". *Information and Communication Technology, Electronics and Microelectronics (MIPRO), 2015 38th International Convention on*. IEEE. 2015, pp. 1200–1205.

[Kal18] Kalainathan, D. et al. "SAM: Structural Agnostic Model, Causal Discovery and Penalized Adversarial Learning". *arXiv preprint arXiv:1803.04929* (2018).

[KB07] Kalisch, M. & Bühlmann, P. "Estimating high-dimensional directed acyclic graphs with the PC-algorithm". *J. Mach. Learn. Res.* **8** (2007), pp. 613–636.

[KK01] Kaufman, J. S. & Kaufman, S. "Assessment of structured socioeconomic effects on health". *Epidemiology* (2001), pp. 157–167.

[Le16] Le, T. et al. "A fast PC algorithm for high dimensional causal discovery with multi-core PCs". *IEEE/ACM Trans. Comput. Biol. Bioinf.* (2016).

[Liu17] Liu, H. et al. "A new hybrid method for learning bayesian networks: Separation and reunion". *Knowl.-Based Syst.* **121** (2017), pp. 185 –197.

[LP15a] Lopez-Paz, D. et al. "The randomized causation coefficient". *The Journal of Machine Learning Research* **16**.1 (2015), pp. 2901–2907.

[LP15b] Lopez-Paz, D. et al. "Towards a learning theory of cause-effect inference". *International Conference on Machine Learning*. 2015, pp. 1452–1461.

[MN16] Maathuis, M. H. & Nandy, P. *A Review of Some Recent Advances in Causal Inference.* 2016.

[Maa09] Maathuis, M. H. et al. "Estimating high-dimensional intervention effects from observational data". *Ann. of Stat.* **37**.6A (2009), pp. 3133–3164.

[Maa10] Maathuis, M. H. et al. "Predicting causal effects in large-scale systems from observational data". *Nature Methods* **7**.4 (2010), pp. 247–248.

[MT14] Martin, E. R. & Thorncroft, C. D. "The impact of the AMO on the West African monsoon annual cycle". *Quarterly Journal of the Royal Meteorological Society* **140**.678 (2014), pp. 31–46.

[Mee95] Meek, C. "Causal inference and causal explanation with background knowledge". *Proc. of the Eleventh conference on Uncertainty in Artificial Intelligence*. Morgan Kaufmann Publishers Inc. 1995, pp. 403–410.

[Mon15] Montgomery, D. C. et al. *Introduction to linear regression analysis*. John Wiley & Sons, 2015.

[Moo09] Mooij, J. et al. "Regression by dependence minimization and its application to causal inference in additive noise models". *Proceedings of the 26th annual international conference on machine learning*. ACM. 2009, pp. 745–752.

[Net96] Neter, J. et al. *Applied linear statistical models*. Vol. 4. Irwin Chicago, 1996.

[NG01] Nicholson, S. E. & Grist, J. "A conceptual model for understanding rainfall variability in the West African Sahel on interannual and interdecadal timescales". *International Journal of Climatology: A Journal of the Royal Meteorological Society* **21**.14 (2001), pp. 1733–1757.

[Omo13] Omondi, P. et al. "Linkages between global sea surface temperatures and decadal rainfall variability over Eastern Africa region". *International Journal of Climatology* **33**.8 (2013), pp. 2082–2104.

[Pea95] Pearl, J. "Causal diagrams for empirical research". *Biometrika* **82**.4 (1995), pp. 669–688.

[Pea03] Pearl, J. "Causality: models, reasoning and inference". *Econometric Theory* **19**.675-685 (2003), p. 46.

[Peñ07] Peña, J. M. et al. "Towards scalable and data efficient learning of Markov boundaries". *Int. J. Approx. Reason.* **45**.2 (2007), pp. 211–232.

[Pet14] Peters, J. et al. "Causal discovery with continuous additive noise models". *The Journal of Machine Learning Research* **15**.1 (2014), pp. 2009–2053.

[Ric96] Richardson, T. “A discovery algorithm for directed cyclic graphs”. *Proc. of the Twelfth international conference on Uncertainty in Artificial Intelligence*. Morgan Kaufmann Publishers Inc. 1996, pp. 454–461.

[Row03] Rowell, D. P. “The impact of Mediterranean SSTs on the Sahelian rainfall season”. *J. Climate* **16**.5 (2003), pp. 849–862.

[Sac05] Sachs, K. et al. “Causal protein-signaling networks derived from multiparameter single-cell data”. *Science* **308**.5721 (2005), pp. 523–529.

[Saj99] Saji, N. et al. “A dipole mode in the tropical Indian Ocean”. *Nature* **401**.6751 (1999), pp. 360–363.

[Shi06] Shimizu, S. et al. “A linear non-Gaussian acyclic model for causal discovery”. *Journal of Machine Learning Research* **7**.Oct (2006), pp. 2003–2030.

[Sil00] Silverstein, C. et al. “Scalable techniques for mining causal structures”. *Data Mining and Knowledge Discovery* **4**.2-3 (2000), pp. 163–192.

[Spi95] Spirtes, P. et al. “Causal inference in the presence of latent variables and selection bias”. *Proc. of the Eleventh conference on Uncertainty in Artificial Intelligence*. Morgan Kaufmann Publishers Inc. 1995, pp. 499–506.

[Spi00] Spirtes, P. et al. *Causation, prediction, and search*. Vol. 81. MIT press, 2000.

[Tsa03] Tsamardinos, I. et al. “Time and Sample Efficient Discovery of Markov Blankets and Direct Causal Relations”. *Proc. of the Ninth ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining*. Washington, D.C.: ACM, 2003, pp. 673–678.

[Tsa06] Tsamardinos, I. et al. “The max-min hill-climbing Bayesian network structure learning algorithm”. *Mach. Learn.* **65**.1 (2006), pp. 31–78.

[WB06] Wille, A. & Bühlmann, P. “Low-order conditional independence graphs for inferring genetic networks”. *Stat. Appl. Genet. Molec. Biol.* **5**.1 (2006).

[XG08] Xie, X. & Geng, Z. “A recursive method for structural learning of directed acyclic graphs”. *J. Mach. Learn. Res.* **9**.Mar (2008), pp. 459–483.

[Xie06] Xie, X. et al. “Decomposition of structural learning about directed acyclic graphs”. *Artificial Intelligence* **170**.4-5 (2006), pp. 422–439.

[ZM17] Zhang, J., Mayer, W., et al. “Weakening faithfulness: some heuristic causal discovery algorithms”. *Int. J. Data Sci. Anal.* **3**.2 (2017), pp. 93–104.

[ZH08] Zhang, K. & Hyvärinen, A. “Distinguishing causes from effects using nonlinear acyclic causal models”. *Proceedings of the 2008th International Conference on Causality: Objectives and Assessment-Volume 6*. JMLR. org. 2008, pp. 157–164.

[ZD06] Zhang, R. & Delworth, T. L. “Impact of Atlantic multidecadal oscillations on India/Sahel rainfall and Atlantic hurricanes”. *Geophysical Research Letters* **33**.17 (2006).