July 2017

Method for Enabling Causal Inference in Relational Domains

David Arbour

METHODS FOR ENABLING CAUSAL INFERENCE IN RELATIONAL DOMAINS

A Dissertation Presented

by

DAVID ARBOUR

Submitted to the Graduate School of the
University of Massachusetts Amherst in partial fulfillment
of the requirements for the degree of

DOCTOR OF PHILOSOPHY

May 2017

College of Information and Computer Sciences

METHODS FOR ENABLING CAUSAL INFERENCE IN RELATIONAL DOMAINS

A Dissertation Presented

by

DAVID ARBOUR

Approved as to style and content by:

David Jensen, Chair

Ben Marlin, Member

Nick Reich, Member

Dan Sheldon, Member

James Allan, Chair
College of Information and Computer Sciences

DEDICATION

To Rachel for her unwavering support, love and understanding, and to my daughters who brighten every morning with their zest for life.

Well, how did I get here?

David Byrne

ACKNOWLEDGMENTS

I would like to thank David Jensen for providing me the opportunity to pursue research under his guidance and his thoughtful input over these years. His sense of pragmatism and his relentless empiricism have been well appreciated, and has benefited my work tremendously. I would also like to thank my committee members for their careful comment and consideration throughout the thesis process.

It has been a pleasure to work with the past and present members of the Knowledge Discovery Lab at UMass. Their time and discussions helped me gain deeper understanding and clarity. I would like to especially thank my co-authors. Marc Maier for working with me in my earliest days as a researcher and providing a seamlessly endless amount of useful advice about research and life. Katerina Marazopoulou for her willingness to relentlessly challenge assumptions and spend many long hours at the whiteboard working through problems. Dan Garant whose conversations over coffee and long walks through campus provided both grounding and enthusiasm for new ideas.

I would also like to thank Rachel and the girls for offering their unending support and love, welcome distractions from work, and providing perspective. My family for their good humor, love, and great memories throughout the years. Finally, thank you to Allen and Pam for their incredible hospitality, bottomless well of babysitting, and quiet understanding.

ABSTRACT

METHODS FOR ENABLING CAUSAL INFERENCE IN RELATIONAL DOMAINS

MAY 2017

DAVID ARBOUR

B.Sc., UNIVERSITY OF MASSACHUSETTS AMHERST

M.Sc., UNIVERSITY OF MASSACHUSETTS AMHERST

Ph.D., UNIVERSITY OF MASSACHUSETTS AMHERST

Directed by: Professor David Jensen

The analysis of data from complex systems is quickly becoming a fundamental aspect of modern business, government, and science. The field of causal inference is concerned with developing a set of statistical methods that allow practitioners make inferences about unseen interventions. This field has seen significant advances in recent years. However, the vast majority of this work assumes that data instances are independent, whereas many systems are best described in terms of interconnected instances, i.e. relational systems. This discrepancy prevents causal inference techniques from being reliably applied in many real-world settings.

In this thesis, I will present three contributions to the field of causal inference that seek to enable the analysis of relational systems. First, I will present theory for consistently testing statistical dependence in relational domains. I then show how the significance of this test can be measured in practice using a novel bootstrap

method for structured domains. Second, I show that statistical dependence in relational domains is inherently asymmetric, implying a simple test of causal direction from observational data. This test requires no assumptions on either the marginal distributions of variables or the functional form of dependence. Third, I describe relational causal adjustment, a procedure to identify the effects of arbitrary interventions from observational relational data via an extension of Pearls backdoor criterion. A series of evaluations on synthetic domains shows the estimates obtained by relational causal adjustment are close to those obtained from explicit experimentation.

TABLE OF CONTENTS

LIST OF TABLES

Table **Page**

LIST OF FIGURES

Figure **Page**

CHAPTER 1

INTRODUCTION

Advances within the field of machine learning have introduced a number of powerful techniques for analyzing and modeling relationships in observational data sets. These algorithms are often characterized by their ability perform prediction or provide descriptive information of data, e.g. determining the presence and location of an object in an image; finding clusterings in a given data set. This has lead to a number of successful applications in fields ranging from human level performance in computer vision [47], to expert level playing of the game Jeopardy! [27], and automatic discovery of topics in large text corpora [8].

The vast majority of machine learning methods operate by exploiting *statistical dependence* between variables, i.e., the focus is to provide very accurate predictions about unseen observational data, rather than learning underlying mechanisms that gave rise to that data. However, in many situations the goal is to predict the effect of an *action* or *intervention*, rather than simple prediction. To illustrate the difference between prediction and intervention, we will consider a simple example. Suppose that we are interested in understanding the weather and that we have collected historical data with observations of whether or not it rained on a given day and the number of people carrying umbrellas during their morning commute. If we are interested in building a machine learning classifier that will predict whether or not it will rain on a given day, the number of commuters with umbrellas is a very informative measure, and would likely result in high predictive accuracy. However, the learned model is

useless for intervention. If we were somehow able to make everyone carry an umbrella to work, there would be no effect on whether or not it rains that day.

The field of causal inference seeks to explicitly model the effect of intervention on an observed system. The most widely known causal inference method is randomized experimentation. In this setting, a small representative group is assigned to either treatment or control, an outcome is measured, and the results are extrapolated to infer the effect of the treatment on a larger population [1]. When experimentation is infeasible or undesirable, methods for observational causal inference can be used to infer the effect of intervention from purely observational data[2]. There are a number of approaches for observational causal inference that have been proposed in the literature (c.f. Pearl [71], Rubin [77], Dawid [21]). While these approaches differ in philosophy and specific algorithmic detail, they all consist of specifying the *structure* of a domain, and then leveraging the specified structure to reason over the set of variables necessary to condition on in order to infer the direct causal effect.

When causal structure cannot be reliably specified using *a priori* knowledge, causal discovery can be used to infer the structure of a domain from observational data [83, 70]. The most commonly employed methods for causal discovery (e.g., Spirtes, et al. [83]) rely on a series of dependence tests to infer the causal structure of a domain. There has been a number of advances in testing statistical dependence in the machine learning community in recent years (c.f., Gretton, et al. [34], Lopez-Paz, et al. [54], Margaritis and Thrun [63]). The most successful of these is the Hilbert-Schmidt independence criterion (HSIC) [34]. HSIC provides a consistent, non-parametric measure of dependence between variables which has been shown to

[1]This extrapolation need not be complex. In fact, for binary treatments a simple difference between the means of treatment and control is employed.

[2]We note that observational causal inference methods can often be employed for experimental analysis as well.

provide state of the art performance in terms of type I and type II errors, and causal structure learning [95].

The majority of the existing literature concerning causal inference and discovery rely heavily on the assumption of independent and identically distributed (i.i.d) instances. However, many real-world systems arise from systems that are relational, i.e. structured as networks [3]. Examples of these systems include social, technological and biological networks. Instances in these systems can be represented as interconnected nodes in a graph in which the attributes of each node are often correlated. Such *relational* data contain dependent instances, and thus violate the i.i.d. assumption. As a result, much of the existing algorithmic mechanisms for inferring causal structure and effects cannot be applied to relational data. Further, there are classes of interventions, such as peer effects, that cannot be easily expressed in frameworks designed to model i.i.d. data.

Recent work has developed methods for causal inference and discovery in relational domains. Maier, et al. [57] introduce, relational d-separation, a theory for graphically reasoning about conditional independence in relational domains. Subsequent work leveraged relational d-separation to learning causal structure of relational domains [55, 61, 48]. While this work represents a significant advance, the results (both theoretical and experimental) assume an idealized setting with the presence of an independence oracle. This prevents these methods from being reliably used by practitioners.

This thesis presents a set of techniques that aim to enable causal discovery in relational domains, as well as causal inference from the learned structure. Toward this end, we provide the following contributions:

- Consistent non-parametric testing of statistical dependence in relational domains. We show that the Hilbert-Schmidt independence criterion can provide

[3]Throughout this thesis we use the terms relational and structured interchangeably to denote domains that can be represented via network structure.

consistent measures of dependence in the presence of non-i.i.d. data having arbitrary dependence structure between instances. This is an improvement of prior results for testing dependence in non-i.i.d. domains (i.e., Zhang et al. [98, 97]) weakening the required assumptions on both the structure of dependence, as well as the strength of dependence between instances. In order to assess statistical significance from finite samples, we provide an extension of the wild bootstrap to relational domains that is consistent, and provides state of the art results in terms of type I error. To our knowledge this is the first provably consistent bootstrap procedure for arbitrarily structured domains.

- An examination of the relationship between association and causality in relational domains. Specifically, we show that, in contrast to the i.i.d. setting, dependence in relational domains can exhibit inherent asymmetry, regardless of the form of functional dependence or marginal distributions. We show how these results imply a simple test for causal direction from purely observational data which is simple to implement and effective in practice. We also discuss the implications of this finding for causal discovery in relational domains.

- A novel method for inferring the effect of unseen interventions from observational relational data. This consists of presenting an adaptation of Pearl's adjustment criterion [71] to relational domains. Comparing to the state of the art in experimental design for relational domains, we show that we can recover causal estimates which are close to those obtained via randomization. We also show how the method derived for observational causal inference can be used to improve the estimates of experiments performed in relational domains.

The remainder of this thesis is structured as follows. Chapter 2 provides the necessary concepts required for understanding the contributions of the thesis. This consists of three distinct sections: Bayesian networks and relational models, repro-

ducing kernel Hilbert spaces, and weak dependence. Chapter 3 examines the problem of testing for dependence in relational data. Specifically, we will define tests of autodependence, marginal dependence and conditional dependence for relational data and introduce the structured wild bootstrap to simulate from the null distribution of these tests where necessary. Chapter 4 examines the problem of identifying causal direction from observational relational data. Chapter 5 introduces relational causal adjustment (RCA), an algorithm for determining causal effects in observational relational data. We show the efficacy of RCA with respect to explicit experimentation, and then show how RCA can be used as a post-hoc adjustment to improve the estimates that have been obtained via randomization. Finally, Chapter 6 summarizes the contributions of this thesis, and suggests directions for future work.

CHAPTER 2

BACKGROUND

We will now introduce concepts necessary for the reader for understanding the contributions and context of our work. We will first introduce the formalizations of causal graphical models and relational causal graphical models. These concepts will be principally be used in chapter 5 where they are directly utilized to define novel methods of causal effect estimation in graphical models. They also provide valuable context for the entire thesis, in particular chapter 4 where our results on the inference of causal direction in relational models have implications for learning the structure of relational causal models. These two representations provide a basis for reasoning over causal dependencies and conditional dependence in both i.i.d. and relational domains. We then provide a brief introduction to reproducing kernel Hilbert spaces (RKHS). This will be used in chapters 3 and 4 where we utilize and extend measures of dependence constructed using the RKHS framework. Finally, we introduce the notion of weak dependence and associated measures, which are a necessary component of the proofs contained in chapter 4.

Finally, before proceeding, we will define some notation used throughout the remainder of the thesis.

$P(X)$	Probability distribution of the random variable X
$P(X\|Z)$	Probability distribution of the random variable X after conditioning on Z
$p(x)$	Probability density of x
$X \perp\!\!\!\perp Y \| Z$	Random variables X and Y are independent after conditioning on Z

2.1 Bayesian Networks

Bayesian networks are widely used graphical model for i.i.d. data that are able to compactly represent a joint probability distribution, while admitting a set of efficient algorithmic tools for reasoning about various properties of its corresponding joint distribution. They have been successfully used to model domains in a number of fields, including epidemiology [80], cognitive science [32], and ecology [9].

The structure of a Bayesian network is given as a directed graph $\mathcal{G} =< V, E >$. Each vertex, $v \in V$ represents a random variable. Each edge, $e \in E$ represents a probabilistic dependency between variables. For any two nodes, X and Y, in the network, if $X \rightarrow Y$ then X is called a *parent* of Y and Y the *child* of X. A node Z is an *ancestor* of Y if there is a directed path beginning at Z that reaches X in the network. A node W is a *descendant* of Y if there is a directed path beginning at Y that reaches W in the network.

Compact factorization of the joint distribution represented by a Bayesian network is possible because of the Markov condition. The Markov condition states that a variable X is rendered conditionally independent of all its non-descendants given its parents, i.e. $P(X|V\backslash X) = P(X|parents(X))$, where '\' is the set difference operator. For joint probability distributions satisfying the Markov condition the distribution can be factorized as

$$P(V) = \prod_{v \in V} P(v|parents(v))$$

A simple example of a Bayesian network can be seen in Figure 2.1. This network represents a joint distribution of four random variables A, B, C, D. For the sake of simplicity, assume that each is a binary random variable. Assuming the Markov condition, the factorization implied by the network is given by $P(A, B, C, D) = P(A)P(B|A)P(C|A)P(D|B, C)$. Using this representation requires $2^0+2(2^1)+2^2 = 9$ entries. Contrast this to a naive representation using which would require 15 param-

eters representing $2^4 = 16$ states. The magnitude of this relative advantage increases as a function of the number of variables, since the number of states required to naively represent a joint probability distribution grows exponentially as a function of the size of the joint distribution, while the number of parameters required for a Bayesian network is a function of the number of the incoming degrees of each node in the network.

The Markov condition provides a binding between between the network structure and the set of independencies that are present in all probability distributions that are compatible with the network structure. The rules of d-separation [70] leverage this to provide a set of algorithmic rules to answer arbitrary conditional independence queries in Bayesian networks, given in the following definition.

Definition 1. *Given three disjoint sets of random variables* $\mathbf{X}, \mathbf{Y}, \mathbf{Z}$*,* $\mathbf{X}$ *and* $\mathbf{Y}$ *are said to be d-separated given* $\mathbf{Z}$ *if:*

- *No member of* $\mathbf{Z}$ *is a descendant of both* X *and* Y*.*
- *All paths from* X *to* Y *are blocked after conditioning on* $\mathbf{Z}$*, i.e., there is at least one member of* Z *that sits along any undirected path between* X *and* Y*.*

The rules of d-separation allow for reasoning about conditional (in)dependencies that hold in a distribution. For example, the network shown in Figure 2.1 entails the following dependencies:

- $B \not\perp\!\!\!\perp C | \emptyset$
- $B \perp\!\!\!\perp C | A$
- $B \not\perp\!\!\!\perp C | A, D$

We refer to A as a *confounder* of B and C, i.e. a variable that is a common cause of both B and C. Similarly, we refer to D as a *collider* of B and C, i.e. a variable

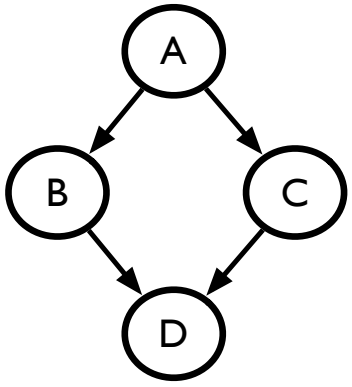


Figure 2.1: An example Bayesian network representing the joint distribution $P(A, B, C, D) = P(B|A)P(C|A)P(D|B, C)$.

that is a common effect of B and C. The introduction of dependence between two variables after conditioning on a collider is a well known statistical phenomenon that is referred to as Berkson's fallacy[1] [92] and explaining away [71] in the statistics and machine learning communities, respectively.

2.1.1 Causal Bayesian networks

In addition to defining a joint probability distribution, Bayesian networks can also be endowed wtih *causal* semantics, provided a few additional assumptions [83, 71].

A1. *(Faithfulness) All independencies entailed by the structure of G are present in D.*

A2. *(Causal Sufficiency) For all variables $X, Y, Z \in \mathcal{D}$, if Z is a common cause of X and Y, then Z has been measured and is present in $\mathcal{X}$.*

A3. *(Invariance under experimentation). The conditional distribution of outcomes, $P(Y|pa(Y))$, under intervention is identical to the conditional distribution observationally.*

[1]Berkson's fallacy is also commonly referred to as Berkson's paradox and Berkson's bias.

The *do* operator [71] explicitly models the effect of an intervention on a system modeled by a Bayesian network. The operation itself is simple to perform, and can be explained in a single step. The structure G' representing the joint distribution $\mathcal{D}'$ that results after intervening on a variable X, is given by taking the original structure G, and removing all incoming edges to X.

The *do*-operator and approximations of interventional effects from observational data is covered in much greater detail in chapter 5, where we also provide an extension to relational domains.

2.1.2 Learning Causal Structure from Observational Data

In general, there are two approaches for learning the causal structure of a Bayesian network. Search and score methods, explicitly optimize the likelihood of a network given some observed data, penalizing for the number of parameters (a function of the number of edges in the network) [11]. The second category are known as "constraint based" methods. Broadly[2], these methods begin with a fully connected graph and undirected graph, and then perform the following steps:

1. For each size conditioning set (beginning with the empty set) test for (conditional) dependence of each pair of variables. If two variables can be rendered independent then remove the edge between them.

2. Once the set of possible conditional independencies has been exhausted, orient as many edges as possible by recursively applying rules leveraging either testable statistical properties or assertions based on modeling assumptions.

The accuracy and ability of constraint based methods is due almost exclusively to the accuracy of marginal and conditional independence tests. It has been shown that

[2]We refer the reader to Spirtes, et al.[83] for a complete treatment of constraint based learning.

Type II errors, i.e. false conclusions of independence, can lead to arbitrarily poor performance [17].

2.2 Modeling Relational Domains

Bayesian networks assume independence between instances, however for many systems observed in the real world, such as social systems, instances are dependent. This dependence has been described in a number of fields and is referred to as spillover effects, SUTVA violations, interference, and relational dependence. In keeping with existing work within the machine learning community [66, 65, 57, 31], we will use the phrase relational dependence. Modeling relational dependence requires additional representational semantics to allow the explicit modeling of dependence between instances, which we will now describe. This extension from traditional Bayesian networks to relational domains can often lead to numerous points of confusion. In order to avoid this, we will build up from a simple i.i.d. example to a fully specified relational model using an example social network. Consider a group of N people who are members of a social networking site. For each person, we will assign an (arbitrary) index, $i = 1, \ldots, n$. Assume that we have measured three attributes on each individual: privacy ($Priv$), social disposition ($Disp$), and average time on site per day (ToS). In addition to the our random variables, we are also given an undirected network, $G =< V, E >$, where each individual is represented by a vertex and an edge between vertexes denotes the presence of "friendship" between them on the site. It follows that each vertex, $v_i \in V$ is associated with random variables with the same index, i.e. x_i, y_i are associated with v_i. Under the assumption that the value of a person's attributes are independent from those of their friends G is superfluous for probabilistic modeling. In this setting, we are assuming that:

1. For each attribute the corresponding random variables have *identical* marginal and conditional distributions for all $i = 1, \ldots, n$.

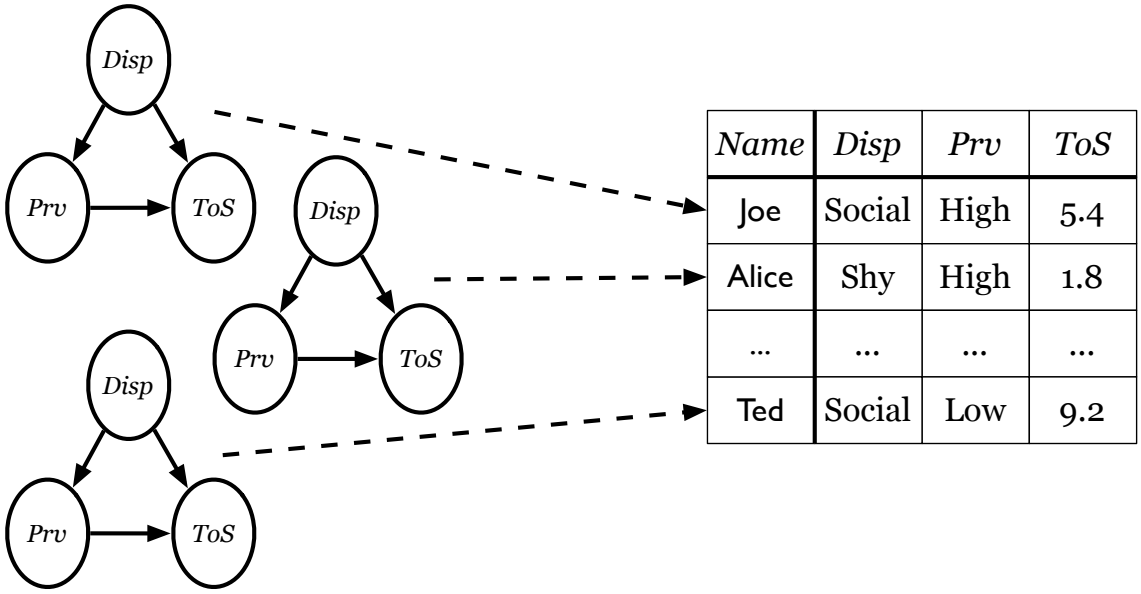


Figure 2.2: An example Bayesian network representing the joint distribution $P(A, B, C, D) = P(B|A)P(C|A)P(D|B, C)$.

2. Can display dependence if and only if they have identical indices, i.e. $Pr(Priv_k, Disp_{k'}) \neq Pr(Priv_k)P(Disp_{k'}) \rightarrow k = k'$.

These conditions are commonly referred to as the independent and identically distributed (i.i.d.) assumption. However, if a person's friends affects their behavior, then the data necessarily violate Assumption 2, i.e. that of independent instances. A consequence of this is we are no longer able to treat each person's values as a sample from an i.i.d. process. Instead, we must consider the interactions between instances directly. In order to reason probabilistically we introduce the following definition and impose some accompanying alternative assumptions. These assumptions are far from novel to this thesis, indeed they are central assumptions in much of the statistical relational learning [43, 65, 72, 58] where they are referred to collectively as the *templating assumption.*

Definition 2. *A relational dependence is defined as any dependence between nodes who do not have an identical index.*

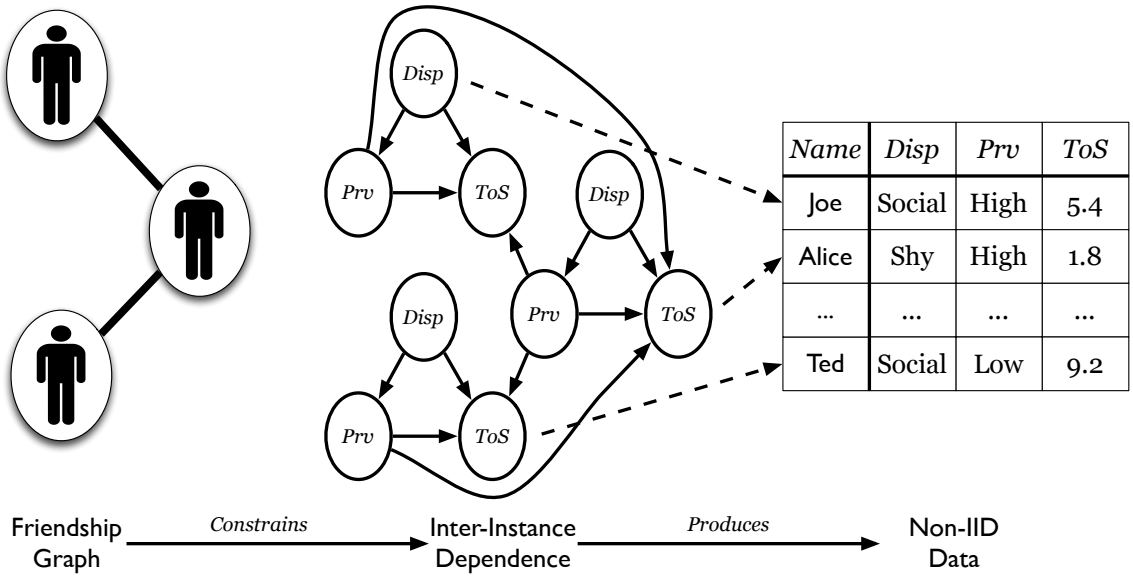


Figure 2.3: An example Bayesian network representing the joint distribution $P(A, B, C, D) = P(B|A)P(C|A)P(D|B, C)$.

A4. *G is given a priori and fixed, i.e. the structure of G is not affected by the values of associated random variables.*

A5. *All relational dependencies are defined with respect to a path predicate, π, obeying first order logic. For instance i and predicate π, the set of instances for an attribute, X, obtained by traversing the G via π from i is denoted as $\pi_i(X)$.*

A6. *For any relational dependence, $f(y_i|\pi_i(x))$ where Y and X are arbitrary random variables, the functional relationship is defined with respect to the sufficient statistics, θ of $\pi_i(x)$. In other words, all relational dependencies are defined with respect to aspects of the distributions defined by instances reached via the path predicate.*

Note that Assumption A6 implies that instances are exchangeable up to a path constraint, i.e. the position of a node in the network is irrelevant after determining that it is reachable via the defined path.

2.3 Relational Causal Models

So far we have considered fairly simple relational models that have a single entity and relationship type. In practice, many relational systems are significantly more complex and require reasoning over multiple entity and relationship types. For example, we could have a data set that consists of people, their social and work relationships among each other, as well as information about the city or town they reside in. To reason over such systems in a clear and concise manner, we will use the framework of relational causal model (RCM) [57]. RCMs allow for reasoning at multiple levels of abstraction, while maintaining the causal semantics of Bayesian networks. In this section, we introduce the key concepts of RCMs, following the notation and terminology of Maier, Marazopoulou, and Jensen [57] that will be used throughout the dependence testing, and structure learning sections. We will begin with describing the class dependency graph, which describes a template of a relational system. We will then introduce abstract ground graphs, an intermediate representation between the individual level model and the class dependency graph that admits sound and complete d-separation semantics [57].

2.3.1 Class Dependency Graphs

A *relational schema* $\mathcal{S} = (\mathcal{E}, \mathcal{R}, \mathcal{A}, card)$ specifies the set of entity, relationship, and attribute classes of a domain. It includes a cardinality function that imposes constraints on the number of times an entity instance can participate in a relationship. A relational schema can be graphically represented with an Entity-Relationship (ER) diagram. Figure 2.5 shows the ER diagram for the Foursquare domain. Foursquare is an online social network where users "check-in" to locations using their mobile phones. In this example, there are three entity classes (*User*, *Place*, *Hometown*), and three relationship classes, (*Friends*, *ChecksIn*, *From*). The entity class *User* has three attributes: *Smokes*, *Weight*, and *Drinks*. The cardinality constraints are depicted

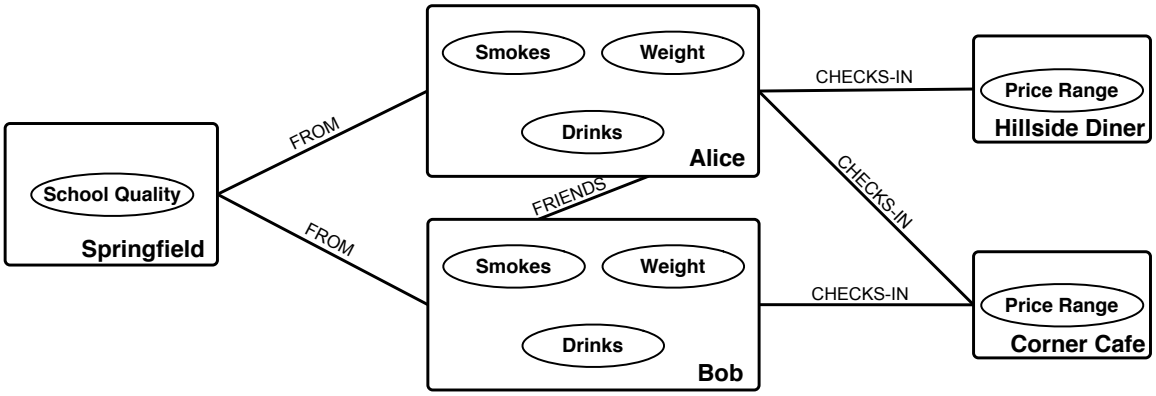


Figure 2.4: Example relational skeleton for the Foursquare domain. This could be a small fragment of a (potentially) larger skeleton.

using crow's feet notation. For example, the cardinality of the *From* relationship is one-to-many, indicating that one user has one hometown, but many users can be from the same hometown.

A *relational skeleton* is a partial instantiation of a relational schema that specifies the set of entity and relationship instances that exist in the domain. Figure 2.4 depicts an example relational skeleton for the Foursquare domain. The network consists of two *User* instances, Alice and Bob, who are friends with each other and come from the same hometown. There are two *Place* instances, Hillside Diner and Corner Cafe.

Given a relational schema, one can specify *relational paths*, which intuitively correspond to possible ways of traversing the schema (see Maier, et al.[57] for a formal definition). For the schema shown in Figure 2.5, possible paths include $[User, Friends, User]$ (a person's friends), and $[User, Friends, User, From, Hometown]$ (the hometowns of a person's friends). *Relational variables* consist of a relational path and an attribute that can be reached through that path. For example, the relational variable $[User, Friends, User].Drinks$ corresponds to the alcohol consumption of a person's friends. We briefly note that the logical predicate used to construction of this set can be defined in a number of ways. For example, we can define $[User, Friends, User]$ to be the set of friends for an individual either exclusive or in-

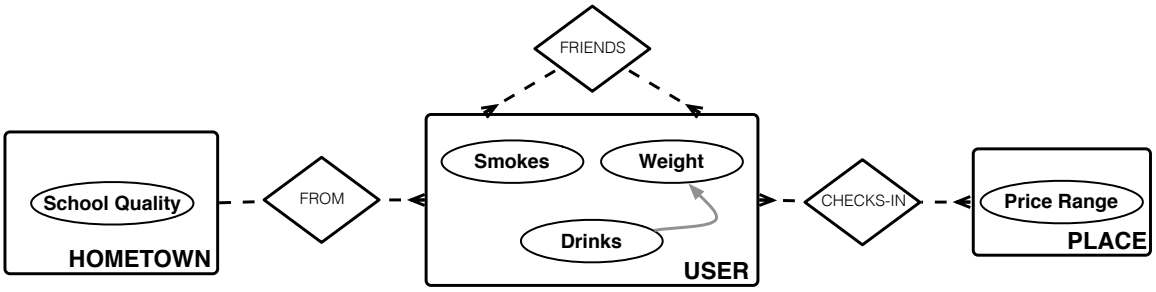


Figure 2.5: Relational model for the Foursquare domain. The underlying relational schema (ER diagram) is shown in black. The attributes on the entities are fictional. A relational dependency is shown in gray. The model shown represents the joint distribution of the domain.

clusive of that individual's friends. Marazopoulou, Arbour, and Jensen [60] further details and show the impact of the choice of path predicates on effect estimation. Probabilistic dependencies can be defined between these relational variables. In this work, we consider dependencies where the path of the outcome relational variable is a single item. In this case, the path of the treatment relational variable describes how dependence is induced. For example, the *relational dependency*

$$[User, Friends, User].Drinks \rightarrow [User].Weight$$

states that the alcohol consumption of a user's friends affects that user's weight.

A *relational model* $\mathcal{M} = (\mathcal{S}, \mathcal{D}, \Theta)$ is a collection of relational dependencies $\mathcal{D}$ defined over a relational schema along with their parameterizations Θ (a conditional probability distribution for each attribute given its parents). The structure of a relational model can be depicted by superimposing the dependencies on the ER diagram of the relational schema, as shown in Figure 2.5, and labeling each arrow with the dependency it corresponds to. If labels are omitted, the resulting graphical representation is known as a *class-dependency graph*.

2.3.2 Abstract Ground Graphs

Recent work by Maier, et. al [57] provides a framework that enables reasoning about d-separation in relational models. Toward that end, they introduce *abstract ground graphs* (AGGs), a graphical structure that captures relational dependencies and can be used to answer relational d-separation queries. Abstract ground graphs are defined from a given perspective, the base item of the analysis, and include nodes that correspond to relational variables. Returning to our example, suppose we are interested in examining the causal structure of our system from the perspective of a user, i.e. we wish to understand the effects of interventions in the system on individual users. The construction of these graphs consist of all singleton variables, i.e. relational variables whose path consists of a single item, and all variables whose path *begins* with the same base item. The key innovation of the abstract ground graph is that, in contrast to the class dependency graph, d-separation semantics can be applied directly to the graph as they are in the case of Bayesian networks. This has enabled a number of novel algorithms for causal discovery in relational domains, (e.g. Marazopoulou, et al. [61], Maier, et al. [55]), as well as a novel method for estimating causal effects in relational domains which we present as the final contribution of this thesis.

2.4 Kernel Embeddings

In this section we will review reproducing kernel Hilbert spaces (RKHS), kernel mean embeddings, and the use of kernel mean embeddings as representations of probability distributions.

2.4.1 Reproducing Kernel Hilbert Space

There are a wealth of provably correct and efficient algorithms for analyzing data under the assumption of linearity. However, in practice we mostly live in a non-linear

world. For roughly the past thirty years the field of machine learning has proposed methods for dealing with non-linear data. For most of the most effective estimators, such as random forests and neural networks, theoretical analysis proves to be very difficult. The kernel embedding framework provides a rich set of non-linear estimators that are considerably more amenable to theoretical analysis, while still providing state of the art or near state of the art performance. Kernel embeddings work by implicitly projecting the data into a potentially infinite space, where a linear method can then be applied.

We will first begin with some definitions.

Definition 3. *Given a vector space, $\mathcal{H}$ the inner product is a function, $\langle \cdot, \cdot \rangle_{\mathcal{H}}, \mathcal{H} \times \mathcal{H} \mapsto \mathbb{R}$ with the following properties:*

1. *Symmetry, i.e. $\langle f, f' \rangle_{\mathcal{H}} = \langle f, f' \rangle_{\mathcal{H}}$*
2. *Linearity, i.e. $\langle \alpha f_1 + \alpha' f_2, f' \rangle_{\mathcal{H}} = \alpha \langle f_1, f' \rangle_{\mathcal{H}} + \alpha' \langle f_2, f' \rangle_{\mathcal{H}}$*
3. *$\langle f, f \rangle_{\mathcal{H}} \geq 0$, with $\langle f, f' \rangle_{\mathcal{H}} = 0 \iff f = 0$*

The norm given by an inner product is defined as $\|f\|_{\mathcal{H}} = \sqrt{\langle f, f \rangle_{\mathcal{H}}}$.

We call $\mathcal{H}$ a *Hilbert space* if it possesses an inner product and also contains Cauchy sequence limits[3].

Definition 4. *Given a non-empty set $\mathcal{X}$, a kernel, $k(\cdot, \cdot)$, is a function from $\mathcal{X} \times \mathcal{X} \mapsto \mathbb{R}$ that has a corresponding $\mathbb{R}$-Hilbert space, $\mathcal{H}$, and a mapping function $\phi : \mathcal{X} \mapsto \mathcal{H}$ with the property that for all $x, x' \in \mathcal{X}$, $k(x, x') := \langle \phi(x), \phi(x') \rangle_{\mathcal{H}}$.*

Note that we have not needed to put any restrictions on $\mathcal{X}$ itself, other than it being non-empty.

---

[3]Given a metric space, (X, d), with distance function $d(\cdot, \cdot)$, a cauchy sequence is a sequence $x_1, \ldots, x_n$ if for any positive real number $\epsilon > 0$, there exists some integer n such that for all i, j, where $j < n$, $d(x_i, x_j) < \epsilon$

Definition 5. *We say that a kernel has the reproducing property if for all* $x \in \mathcal{X}$ *and* $f \in \mathcal{H}$, $\langle \phi(x), f \rangle_{\mathcal{H}} = f(x)$.

Within our particular context we see that this implies for any $x, x' \in \mathcal{X}$, $\langle k(x, \cdot), k(x', \cdot) \rangle_{\mathcal{H}} = k(x, x')$. A *reproducing kernel Hilbert space* (RKHS) is a Hilbert space containing a reproducing kernel.

Thus far we have described kernel embeddings with respect to an explicit mapping function. However, in practice these mappings can be extremely difficult to define and infeasible to apply in practice to data. The key insight is that if an algorithm can be defined with respect to inner products only, it is never necessary to explicitly create the feature embeddings, only the inner product. There are a large number of functions that allow for this implicit definition. The most commonly used is the radial basis function (RBF) kernel which is given by $k(x, x') = \exp -\frac{\|x-x'\|^2}{2\sigma^2}$, where $\|\cdot\|^2$ is the squared euclidean norm and σ^2 is a user defined parameter.

2.4.2 Kernel Mean Embeddings

Definition 6. *Let* $\mathcal{X}$ *be a non-empty set,* $(\mathcal{X}, \mathcal{A})$ *be a measurable space where* $\mathcal{A}$ *is a* σ*-algebra on* $\mathcal{X}$*, and let* $\mathscr{P}$ *be the set of all probability measures,* P*, on* $\mathcal{X}$*.* $\mathcal{H}$ *is the RKHS of the functions* $f : \mathcal{X} \to \mathbb{R}$ *with the reproducing kernel* $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$*. The* mean map *is a function* $\mu : \mathscr{P} \to \mathcal{H}$ *that defines a kernel embedding of a distribution into* $\mathcal{H}$*:*

$$\mu_P = \mu(P) = \int_{\mathcal{X}} k(x, \cdot) dP(x)$$

A *characteristic* kernel is one that defines an injective mapping between a distribution and the kernel mean, i.e. the kernel mean uniquely identifies the underlying distribution that that data was drawn from. The conditions required for a kernel to be characteristic are well studied, but beyond the scope of this thesis. We refer the

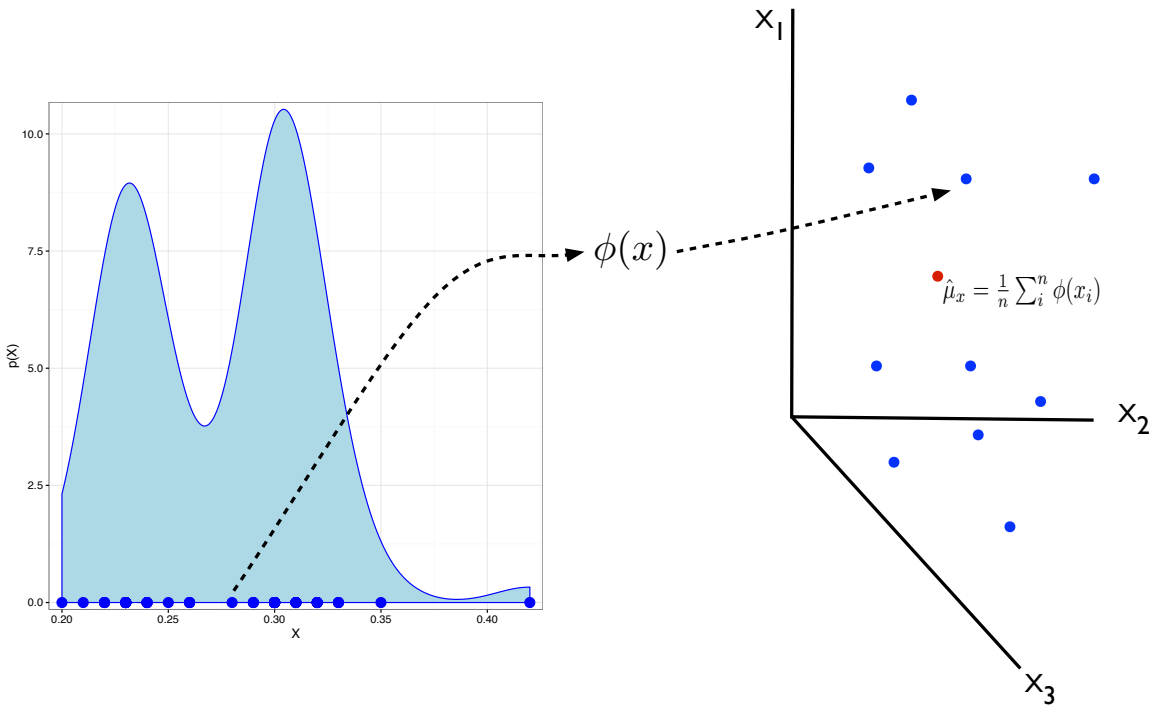


Figure 2.6: Illustrative example of kernel embedding. In this case $\phi(x)$ defines a mapping from $x \in \mathbb{R}$ to $\mathbb{R}^3$. In practice this mapping can be to an arbitrary number of dimensions. The empirical kernel mean, $\hat{\mu}_x$, is the mean in the projected, i.e. feature, space. If the kernel is characteristic the kernel mean, μ_x, uniquely identifies the underlying distribution of x, $P(X)$.

reader to Sripumbudur, et al. [85], and Sripumbudur, et al. [84] for a more comprehensive treatment of what constitutes a characteristic kernel. For the purposes of this thesis it suffices to note that many common kernels such as the RBF and Laplacian kernel are characteristic.

The kernel mean embedding framework enables a number of non-parametric procedures that reason over the space of distributions. For instance, the maximum mean discrepancy (MMD) [36] leverages the injective property of kernel mean embeddings to provide a robust and flexible two sample test of equality between two distributions. The MMD is the squared norm of the difference of two mean embeddings which can

be estimated using only inner products, i.e. kernel estimates[4]:

$$\text{MMD}(X,Y) = \|\mu_x - \mu_y\|^2_{\mathcal{H}} = \|\frac{1}{n}\sum_i^n \phi(x_i) - \frac{1}{n}\sum_j^n \phi(y_j)\|^2_{\mathcal{H}} \tag{2.1}$$

$$= \frac{1}{n^2}\sum_{i,j}^n k(x_i,x_j) + k(y_i,y_j) + 2k(x_i,y_j) \tag{2.2}$$

2.4.3 Testing Marginal Dependence via Kernel Embeddings

The RKHS framework can also be used to reason about the distributions of random variables [82]. Given two random variables, X and Y, the joint distribution can be embedded by considering the kernel mean embedding of the tensor product of the two embeddings:

$$P(X,Y) \approx \frac{1}{n}\sum_i^n \phi(x) \otimes \phi(y)$$

Similarly, the product distribution, $P(X)P(Y)$, is approximated by considering the tensor product of the respective kernel means:

$$P(X)P(Y) \approx \mu_x \otimes \mu_y = \frac{1}{n}\sum_i^n \phi(x) \otimes \frac{1}{n}\sum_i^n \phi(y)$$

This embedding of distributions can be used for robust measures of marginal and conditional dependence between variables, by considering the difference between two distributions: the joint distribution and the product of the marginals. This intuition leads to the Hilbert-Schmidt independence criterion [34]. The Hilbert-Schmidt independence criteria is defined as

$$\text{HSIC}(X,Y) = \|\frac{1}{N}\sum_i^N \phi(x) \otimes \psi(y) - \mu_x \otimes \mu_y\|^2_{\mathcal{H}} \tag{2.3}$$

[4]For notational convenience, and without loss of generality, we assume that the samples for X and Y are of equal length

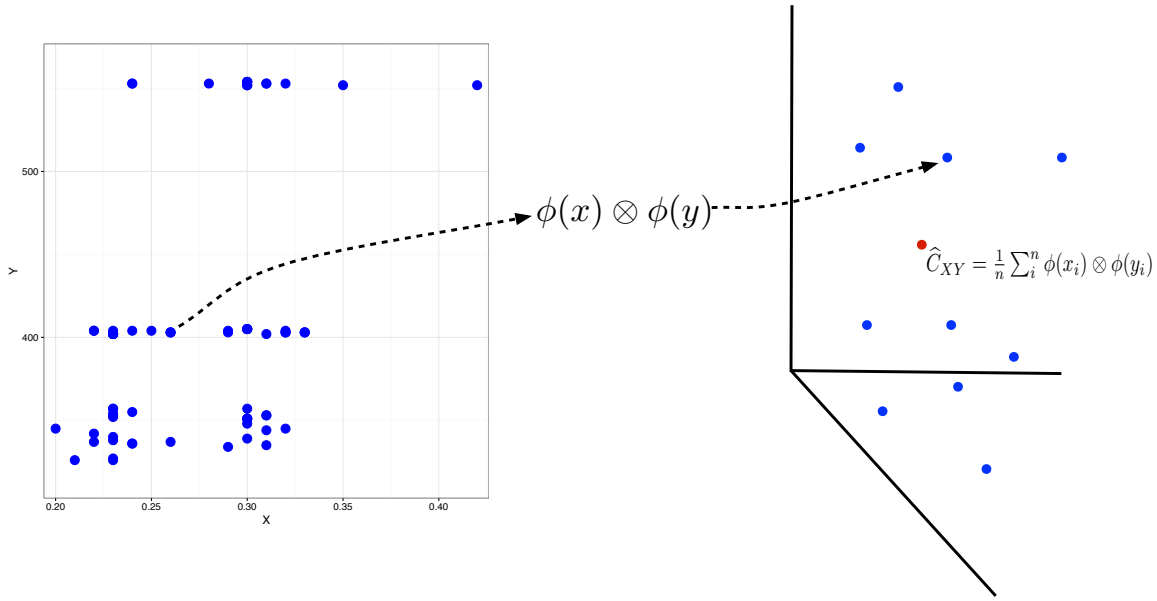


Figure 2.7: Illustrative example of a joint kernel embedding. Rather than considering a single projection the operation now considers the tensor product of the embeddings of instances of X and Y. If characteristic kernels are used for both X and Y, the empirical mean of this mapping, $\widehat{C}_{xy} = \frac{1}{n}\sum_i^n \phi(x_i) \otimes \phi(y_i)$ defines an injective mapping of the joint distribution of X and Y, i.e. $P(X, Y)$ is uniquely identified by C_{xy}.

A biased estimate of HSIC can be achieved with

$$\text{HSIC}(X, Y) = K_{\mathbf{x}} H K_{\mathbf{y}} H \tag{2.4}$$

Where $H = I - \frac{1}{n}\mathbf{1}\mathbf{1}^T$ is a centering matrix, and $K_{\mathbf{x}} = \phi(\mathbf{x})\phi(\mathbf{x})^T, K_{\mathbf{y}} = \psi(\mathbf{y})\psi(\mathbf{y})^T$ are the Gram matrices for $\mathbf{X}$ and $\mathbf{y}$, respectively.

A closely related measure is the centered kernel target alignment, which is a normalized measure of dependence introduced by Cortes, et al. [18] within the context of multiple kernel learning. The measure is defined as:

$$\text{KTA}(\mathbf{x}, \mathbf{y}) = \frac{\|\frac{1}{N}\sum_i^N \phi(x) \otimes \psi(y) - \mu_x \otimes \mu_y\|_{\mathcal{H}}^2}{\|\frac{1}{N}\sum_i^N \phi(x) - \mu_x\|_{\mathcal{H}} \|\frac{1}{N}\sum_i^N \phi(y) - \mu_y\|_{\mathcal{H}}} \tag{2.5}$$

$$= \frac{\langle K_{\mathbf{x}}^c, K_{\mathbf{y}}^c \rangle_{\mathcal{H}}}{\|K_{\mathbf{x}}^c\|_{\mathcal{H}} \|K_{\mathbf{y}}^c\|_{\mathcal{H}}} \tag{2.6}$$

Zhang, et al. [95] extend this idea to provide a definition of partial dependence which can be applied in the kernel setting. To do so, they make use the following characterization, due to Daudin [20]:

Lemma 1. *[20] Let $\mathcal{E}_{YZ}, \mathcal{E}_{XZ}$ be the space of all functions of X, Z and Y, Z respectively. The following conditions are equivalent:*

1. $X \perp\!\!\!\perp Y | Z$
2. $\mathbb{E}(\tilde{f}\tilde{g}) = 0, \forall \tilde{f} \in \mathcal{E}_{XZ}$ *and* $\tilde{g} \in \mathcal{E}_{YZ}$
3. $\mathbb{E}(\tilde{(f)}g) = 0, \forall f \in \mathcal{E}_{XZ}$ *and* $g \in L^2_{XZ}$
4. $\mathbb{E}(\tilde{f}\tilde{g}') = 0, \forall \tilde{f} \in \mathcal{E}_{XZ}$ *and* $\tilde{g}' \in \mathcal{E}'_{YZ}$
5. $(\tilde{f}g') = 0, \forall \tilde{f} \in \mathcal{E}_{XZ}$ *and* $g' \in L^2_Y$

Intuitively, the second condition of Lemma 1 says that independence can be asserted if any function of the residuals of (X, Z) given Z is uncorrelated with (Y, Z) given Z. While this provides a more general condition for partial correlation, it is infeasible in practice, since it requires the ability to reason over *all* functions of X, Z and Y, Z, in L^2. However, if this requirement is relaxed and the space of functions are restricted to be those residing in Hilbert spaces, $\mathcal{H}_{\ddot{\mathcal{X}}}$, $\mathcal{H}_{\mathcal{Y}}$, where $\ddot{X} = (X, Z)$, then the following characterization, due to Fukumizu, et al. [30] provides a definition which can be practically realized:

Lemma 2. *[30] Let $k_{\ddot{\mathcal{X}}} \triangleq k_{\mathcal{X}\mathcal{Z}}$. Assuming $k_{\ddot{\mathcal{X}}} k_{\mathcal{Y}}$ is characteristic w.r.t. $(\mathcal{X} \times \mathcal{Y}) \times \mathcal{Z}$, $\mathcal{H}_{\mathcal{X}}, \mathcal{H}_{\mathcal{Y}}$, and $\mathcal{H}_{\mathcal{Z}}$ are contained in L^2 and $\mathcal{H}_{\mathcal{Z}} + \mathbb{R}$ is dense in $L^2(P_Z)$, then*

$$\Sigma_{\ddot{X}Y|Z} = 0 \iff X \perp\!\!\!\perp Y | Z \tag{2.7}$$

This implies that conditional dependence can be determined by constructing the spaces resulting from the residuals of a nonparametric regression and testing dependence between the kernel matrices the same way that one determines marginal dependence, i.e. $HSIC_{X,Y|Z} = \frac{1}{n}Tr(\tilde{K}_{\ddot{X}|Z}\tilde{K}_{Y|Z})$. The key contribution of Zhang, et al. [95] was to show that after taking the residual of variables in feature space, i.e. RKHS, the problem of testing conditional dependence reduces to testing the marginal dependence between the residualized variables.

2.5 Sources of Relational Bias

A large portion of this thesis involves reasoning over the bias that arises from assuming independence amongst instances in relational domains. Historically, the entirety of this bias has been attributed to *autodependence*, i.e., dependence that exists between individual instances. There are two forms of bias that must be considered in the context of relational learning. The first is the previously mentioned autodependence, i.e. dependence amongst instances arising from a relational dependence. The second source of bias is what we will refer to as *aggregation-bias*. In contrast to autodependence, aggregation bias is not an inherent property of the underlying generative model, but rather an artifact of analytic procedure.

As an example of aggregation bias, consider the following simple relational model:

$$
\begin{aligned}
X &\sim \mathcal{N}(0,1) \\
\tau.X &\triangleq D^{-1}Ax \\
\epsilon &\sim \mathcal{N}(0,1) \\
Y &\sim \beta\tau.X + \epsilon
\end{aligned}
$$

where A is the adjacency matrix of a single entity/single relationship relational model, D is a diagonal matrix where $D_{i,i}$ is the degree of node i, and $\beta \in \mathbb{R}$ is a coefficient

of linear dependence. Assume we wish to determine linear dependence between $\tau.X$ and Y.

1. For each node i, collect the set of X instances whose associated node lies in the terminal set of the path predicate.

2. Create an appropriate set of aggregations for each set of instances.

Bias occurs whenever the terminal sets produced by predicate ξ are non-distinct, i.e. whenever there exists two nodes i and j such that $\xi(i) \cap \xi(j) \neq \emptyset$.

The population variance, σ_X^2 of a random variable X with chain structured instance dependence can be written as [22]

$$\sigma_X^2 = Var(X_1) + 2\sum_{k=1}^{\infty} Cov(X_1, X_k)$$

Similarly, it follows that a random variable X with instance dependence structure given by a d-regular graph, the variance, σ_X^2 is given as

$$\sigma_X^2 = Var(X_{\pi_1(1)}) + 2\sum_{k \in \pi_1(2,\ldots,\infty)} Cov(X_{\pi_1(1)}, X_k) \tag{2.8}$$

Now consider the relational variable defined by considering the mean of neighboring instances, i.e., $\mathbf{X} = \frac{1}{d}AX$. Assume that the instances of X are independent, i.e. $Var(X) = Var(X_1)$. The population variance of the relational variable, $\sigma_{\mathbf{X}}^2$, is

$$\sigma_{\mathbf{X}}^2 = Var(\mathbf{X}_{\pi_1(1)}) + 2\sum_{k \in \pi_1(2,\ldots,\infty)} Cov(\mathbf{X}_{\pi_1(1)}, \mathbf{X}_k) \tag{2.9}$$

$$= \frac{1}{d^2}\sum_{i=1}^{d} Var(X_{\pi_1(1)}) + 2\sum_{k \in \pi_1(2,\ldots,\infty)} Cov(\mathbf{X}_{\pi_1(1)}, \mathbf{X}_k) \tag{2.10}$$

$$= \frac{1}{d} Var(X_{\pi_1(1)}) + \frac{2}{d^2}\sum_{k \in \pi_1(2,\ldots,\infty)} |\mathbf{X}_{\pi_1(1)} \cap \mathbf{X}_k| \, Var(X_{\pi_1(1)}) \tag{2.11}$$

The important take away here is that even in the case where instances are truly independent, bias is induced by simply by casting the problem as relational and considering relational aggregates.

CHAPTER 3

DETECTING DEPENDENCE IN RELATIONAL DOMAINS

Hypothesis tests based on kernel mean embeddings have been successfully used for a number of applications such as two sample testing [36], marginal and conditional independence testing [35, 95], and detecting three variable interactions [78]. A central assumption to much of the theoretical results supporting these methods is that of independent and identically distributed (i.i.d.) instances. In practice, however, many modern phenomena are relational, i.e., occur in networks. For example, testing peer dependence in social networks and deciding whether metrics of a computer network are drawn from the same distribution. A common trait of relational data is auto-dependence, i.e., values of instances of a random variable are correlated with the values of neighboring instances in the network, which violates the i.i.d. assumption.

Previously, [98] studied the problem of dependence testing in graph-structured domains requiring the graph-structure to be decomposed into cliques, which is not possible for many structures such as lattices. More recent work developing methods for kernel hypothesis testing for non-i.i.d. domains have largely focused on time series. [14] presented a permutation procedure for dependence testing between time-series consisting of shifting one time series relative to another. Noting that many kernel-based hypothesis tests, such as the Hilbert-Schmidt independence criterion (HSIC) [35], and maximum mean discrepancy (MMD) [36] are degenerate V-statistics, [15] showed the dependent wild bootstrap of [50] can be used to provide a consistent estimate of the null distribution under weak dependence.

In this work, we study the problem of dependent testing in relational domains. Toward this end, we present three contributions. First, we provide a proof of consistency for kernel based hypothesis tests in relational domains. This is achieved by extending previous proofs of consistency in the presence of auto-dependent time series [50, 15] for degenerate V-statistics to the relational setting. [1]

The second contribution of this work is a modification of dependent Wild bootstrap for degenerate kernel statistics [50, 14] to relational domains to assess dependence in finite samples. The wild bootstrap is a method of *external randomization*, which allows for creation of pseudosamples that explicitly take into account the dependence structure of a given domain. In addition to showing this is a consistent procedure, we show empirically that it leads to a much smaller number of type I errors, i.e., conclusions of false dependence. This represents the first provably consistent bootstrapping procedure for relational domains.

The third contribution of this work addresses a practical concern. As described later, the wild bootstrap requires the generation of an auxiliary variable that possesses the same auto-dependence structure as the original data. Within the literature of dependent wild-bootstraps, it is assumed that this can be correctly specified manually. This assumption is unlikely to hold in the relational setting, due to the complexity of relational domains. To remedy this, we provide an efficient non-parametric optimization procedure for inferring the covariance matrix from observed data.

The remainder of this work is structured as follows. In section 2, we provide the problem setting and necessary background. Section 3, presents a proof of consistency of degenerate V-statistics for weakly dependent data in relational domains. In section 4, we show that the dependent wild bootstrap provides a consistent etimate of the null distribution for such statistics. A procedure for inferring the correct covariance

[1] This result improves on the previous result from [98], which assumes a stronger condition of ϑ-mixing and restricts the form of dependence to clique structures.

structure of the wild bootstrap is presented in section 5. The final sections provide related work, synthetic experiments, and conclusions and directions for future work.

3.1 Problem Setup and Background

The contributions of this work rely on the notions of relational data, kernels, V-statistics, and weak dependence which we will now introduce.

3.1.1 Relational Structure

The *relational structure* refers to a graph, where each node is associated with instances of a random variable. For example, in a social network, the ndoes are individuals and edges between them denote the presence of a "friendship". More formally, we assume an undirected graph, $G = \langle V, E \rangle$ For every random variable X, each vertex v_i is associated with unique instances of X, i.e. instance x_i is associated with vertex v_i. We will also assume the following throughout the work regarding the structure of the graph and its relationship to the random variables.:

A7. *Each node $v \in V$ has degree of at least 1.*

A8. *The adjacency matrix of G is jointly exchangeable, i.e., binary and symmetric.*

A9. *As the number of nodes in the network grows to infinity, the maximum degree of any node is bounded by a real constant.*

A10. *Dependence between two instances i and j implies the existence of a path in the graph between v_i and v_j.*

Assumptions A8 and A9 represent sufficient, but not strictly necessary conditions. We conjecture that the assumption of finite degree may be exchanged for a condition on the growth rate of the maximum degree of the network with respect to the number of nodes, and that the requirement of uniform edge weights may be replaced with an assumption of finite edge weights. The requirement of symmetric networks provide

convenience, allowing for the specification of the covariance matrix for the wild bootstrap in terms of the combinatorial Laplacian, but is not necessary to assume in order for the proofs of consistency to hold.

3.1.2 V-Statistics

Let $X = \{X_1, \dots X_n\}$ be the set of given observations. Define h to be a symmetric function, taking m arguments. A V-statistic is a function defined with respect to h taking the form

$$V(h, X)_n = \frac{1}{n^m} \sum_{i \in i_1 \dots i_m \in N^m} h(X_{i_1}, \dots Z_{X_m})$$

where N^m is defined as the Cartesian product of the set $1, \dots, n$ and n is the total number of observations. In the sequel, we will write $V(h, X)$ as $V(X)$ to reduce notational clutter. Following [15], we will refer to h as the *core*[2].

We say that a core h is j-*degenerate* if for every $x_1, \dots, z_j$,

$$E[h(X_1, \dots, X_j, X^*_{j+1}, \dots, X^*_m)] = 0$$

where $X^*_{j+1}, \dots, X^*_m$ are independent samples drawn from the same distribution as X_1. A core is called canonical if for all $j \leq m - 1$ it is j-degenerate. Finally, we call a V-statistic with a 1-degenerate core a *degenerate V-statistic*.

In this work, our empirical evaluation will focus on the Hilbert-Schmidt independence criterion [3]. As described in the background, the Hilbert-Schmidt independence criterion (HSIC) is a test of dependence, i.e. a hypothesis test of paired samples where the null hypothesis is that the two samples are generated independently, $\mathbb{P}_{x,y} = \mathbb{P}_x \mathbb{P}_y$.

---

[2]In order to prevent confusion, we do not follow the canonical convention of calling h the kernel.

[3]The resulst presented are applicable to a larger set of degenerate V-statistics as well

In this work, we focus on the empirical estimator of HSIC, which can be written as degree-four V-statistic with a core defined by:

$$h(x_1, x_2, x_3, x_4) = \frac{1}{4!} \sum_{\pi \in S_4} k(x_{\pi(1)}, x_{\pi(2)}) k(y_{\pi(1)}, y_{\pi(2)}) + \quad k(y_{\pi(3)}, y_{\pi(4)}) - 2k(y_{\pi(2)}, y_{\pi(3)})$$

where S_n is the set of permutations over a set of n elements.

3.1.3 Weak Dependence

In order to reason about the behavior of test statistics under non-independent samples we necessarily need to reason about the behavior of dependence amongst instances. To understand asymptotic behavior, we need to be able to characterize this behavior as a function of some notion of distance between instances. There are a number of formalisms for reasoning about dependent data (c.f. [22]). In this work we will focus on weak dependence [22], which we now describe. Within this work we will make use of the notion of weak dependence, i.e. τ-dependence. As we shall shortly, weak dependence provides a flexible notion of dependence that requires only the definition of distance between instances and the presence of a measurable probability space.

Definition 7. *[22] Let $(\Omega, \mathcal{A}, \mathbb{P})$ be a probability space, and $\mathscr{M}$ a σ-algebra of $\mathcal{A}$, and δ a distance metric. For any $\mathbb{L}^p$-integrable $\mathcal{X}$-valued random variable X, the coefficient τ_p is defined as:*

$$\tau_p(\mathcal{M}) = \left\| \sup_{g \in \Lambda^{(1)}(\delta)} \left\{ \int g(x) \mathbb{P}_{X|\mathcal{M}}(dx) - \int g(x) \mathbb{P}_X(dx) \right\} \right\|_p$$

letting $\mathbb{P}_X$ be the distribution of X and $\mathbb{P}_{X|\mathcal{M}}$ be the conditional distribution of X given $\mathcal{M}$. defining the sequence of coefficients $\tau_{p,r}(k)$ as

$$\tau_{p,r}(k) = \max_{\ell \leq r} \frac{1}{\ell} \sup_{(i,j) \in \Gamma(1,\ell,k)} \tau_p(\mathcal{M}_i(X_{j_1}, \ldots, X_{j_1}))$$

Perhaps the most important aspect of $\tau(\mathcal{M})$ is that it provides the minimum L_1 distance between a random vector X and another random vector Y drawn from the same process. We call a process *weakly dependent* if $\tau(r) \underset{r \to 0}{\longrightarrow} 0$, i.e. dependence tends to zero as a as the distance grows to infinity.

Within this setting we will define our distance of interest to be the shortest path distance between two nodes in a graph. Thus, the role of τ as a measure of the decay of dependence between instances as a function of graph distance. A process is said to be *weakly dependent* if $\tau(k) \to_{n \to \infty} 0$, where n is the number of nodes in the network. More formally, we will employ the following assumption:

A11. *$(X_t)_{t \in \pi}$ is a strictly stationary τ-dependent process with $\sum_{r=1}^{\infty} r^2 \sqrt{\tau_r} \leq \infty$ for some filtration π, where r is shortest-path graph distance.*

Where a filtration is an ordering of a set such that for any two subsets, $S_{1,\ldots,j}, S_{1,\ldots,k}$, $i \leq k \to S_{1,\ldots,j} \subseteq S_{1,\ldots,k}$. Less formally this assumption states that as the distance between any two nodes in the network tends to infinity, the dependence between them converges to zero.

The notion of weak dependence within the network setting is not novel to this work, [94] make use of the τ-coefficient in the context of deriving asymptotic consistency for transductive learning with an assumption of linear dependence amongst instances. However, to our knowledge, this is the first to consider weak dependence with arbitrary dependence for hypothesis testing of relational data.

3.2 Consistency of Dependence Testing Under Weak Dependence

We now provide a proof of consistency of degenerate V-statistics for relational data under weak dependence. The strategy of this proof is to first approximate the

V statistic with weighted sums of squares, and then apply the central limit theorem to this approximation. The approximation used is the spectral decomposition of the core

$$h(x, y) = \sum_k \lambda_k \Phi(x)\Phi(y)$$

where λ_k are the nonzero eigenvalues of $E[h(x, X_0)\Phi(X_0)] = \lambda\Phi(x)$, and $\Phi(x)$ are the associated eigenvectors. This strategy largely mirrors what is found in both Leucht and Neumann [50]. However, in that case, the approximations are constructed as a function of distance in time. Our contribution is a generalization of the approximations to network domains that follow the aformentioned assumptions. This is done by considering *sets* of instances separated by shortest path distance of k, rather than assuming that there is always at single instance at distance k, and adapting results accordingly.

Theorem 1. *Let $(Z_k)_k$ be centered, jointly normal random variables with $Cov(Z_j, Z_k) = \sum_{r=-\infty}^{\infty} Cov(\Phi_j(X_0), \Phi_k(X_r))$, and $(\lambda_k)_k, (\Phi_k)_k$ be the sequence of non-zero eigenvalues and corresponding eigenfunctions of $E\left[h(x, X_0)\Phi(X_0)\right] = \lambda\Phi(x)$. Under the aforementioned assumptions, $V_n \xrightarrow{d} Z := \sum_k \lambda_k Z_k^2$, as $n \to \infty$, and $EZ = \sum_{r\in\mathbb{Z}} Eh(X_0, X_r) < \infty$ i.e., the infinite series that defines Z converges in L_1.*

Proof. Let $(\lambda_k)_k$ be an enumeration of the positive eigenvalues of $Eh(x, X_0)\Phi(X_0) = \lambda\Phi(x)$ sorted in decreasing order, and $(\Phi_k)_k$ be the corresponding eigenfunctions. Following Leucht and Neumann [50], we set $\lambda_k := 0, \Phi_k \equiv 0, \forall k > L$, when the number L of non-zeros eigenvalues is finite. We are given from a version of Mercer's theorem (given by Theorem 2 of Sun [87]) that

$$h^{(K)}(x, y) = \sum_{k=1}^{K} \lambda_k \Phi_k(x)\Phi_k(y) \underset{K\to\infty}{\longrightarrow} h(x, y), \forall x, h \in \text{supp}(P^{X_0}) \tag{3.1}$$

Leucht and Neumann [50] provide the prerequisites necessary for equation 3.1 converges absolutely and uniform on compact subsets of $\text{supp}(P^{X_0})$, which apply directly in our setting as well. We will consider an approximation of V_n by a V-statistic with a kernel with finite spectral decomposition given by $V_n^{(K)} = \frac{1}{n}\sum_{s,t}^{n} h^{(K)}(X_s, X_t)$. Because h is positive semi-definite by definition, all eigenvalues are non-negative, implying $V_n - V_n^{(K)} \geq 0$. This implies

$$
\begin{aligned}
&E\left|V_n - V_n^{(K)}\right| = E\left[V_n - V_n^{(K)}\right] \\
&= E\left[h(X_0, X_0) - h^{(K)}(X_0, X_0)\right] + \sum_{r=1}^{n-1} 2(1 - r/n) E\left[h(X_0, X_r) - h^{(K)}(X_0, X_r)\right]
\end{aligned}
$$

By majorized convergence the first term converges to zero as $K \to \infty$. For the second term, repeated application of Cauchy-Schwarz gives

$$
\begin{aligned}
&\sum_{r=1}^{n-1} 2(1 - r/n) E\left[h(X_0, X_r) - h^{(K)}(X_0, X_r)\right] \\
&\leq 2\sum_{r=1}^{\infty} \left| \sum_{j \in \Delta_r} E\left[\sum_{k=K+1}^{\infty} \lambda_k \Phi_k(X_0)\Phi_k(X_j) \right] \right| \\
&= 2\sum_{r=1}^{\infty} \left| E\left[\sum_{j \in \Delta_r} \sum_{k=K+1}^{\infty} \lambda_k \Phi_k(X_0)(\Phi_k(X_j) - \Phi_k(\widetilde{X}_j)) \right] \right| \\
&\leq 2\sum_{r=1}^{\infty} \sqrt{E\left[\sum_{j \in \Delta_r} \sum_{k=K+1}^{\infty} \lambda_k \Phi_k^2(X_0) \right]} \sqrt{E\left[\sum_{j \in \Delta_r} \sum_{k=K+1}^{\infty} \lambda_k \left(\Phi_k(X_r) - \Phi_k(\widetilde{X}_j)\right)^2 \right]} \\
&\leq 2\sqrt{\sum_{r=1}^{\infty} \lambda_k} \sum_{r=1}^{\infty} \sqrt{E\left[\sum_{j \in \Delta_r} \sum_{k=1}^{\infty} \lambda_k \left(\Phi_k(X_j) - \Phi_k(\widetilde{X}_j)\right)^2 \right]} \\
&\leq 2\sqrt{\sum_{k=K+1}^{\infty} \lambda_k} \sum_{r=1}^{\infty} \sqrt{\sum_{j \in \Delta_r} E\left[h(X_j, X_j) - h(X_j, \widetilde{X}_j) - h(\widetilde{X}_j, X_j) + h(\widetilde{X}_j, \widetilde{X}_j) \right]} \\
&\leq 2\sqrt{\sum_{k=K+1}^{\infty} \lambda_k} \sum_{r=1}^{\infty} \sqrt{2\max(\deg)^r \text{Lip(h)}} \sqrt{\tau(r)}
\end{aligned}
$$

Where Δ_r is the set of nodes whose shortest path distance from X_0 is r, max(deg) is the largest degree in the network, and $\widetilde{X}_r$ denotes a copy of X_r that is independent of X_0 and satisfies $E\|X_r - \widetilde{X}_r\|_1 \leq \tau(r)$. Because $\sum_{k=1}^{\infty} \lambda_K = Eh(X_0, X_0) < \infty)$, thus $\sum_{k=K+1}^{\infty} \lambda_k \to 0$ as $K \to \infty$ we arrive at $\sup_n E\left|V_n - V_n^{(K)}\right| \underset{K\to\infty}{\longrightarrow} 0$.

The proof of the central limit theorem for for partial sums, i.e., for $K \leq L$

$$V_n^{(K)} = \sum_{k=1}^{K} \lambda_k \left(n^{-1/2} \sum_{t=1}^{n} \Phi_k(X_t) \right)^2 \xrightarrow{d} \sum_{k=1}^{K} \lambda_k Z_k^2 \tag{3.2}$$

follows a direct application of Leucht and Neumann [50] Theorem 2.1 proof part (ii). Combining these two results, to satisfy the requirements of Theorem 2 of Dehling, et al. [23] we arrive at $V_n \xrightarrow{d} Z := \sum_k \lambda_k Z_k^2$. The only item remaining to be shown is $EX < \infty$, which follows from a direct application of part (iv) of the proof of Theorem 2 provided by Leucht and Neumann [50]. □

3.2.1 The Dependent Wild Bootstrap

Bootstrap methods are a collection of techniques that create pseudo-samples from an initial data set by performing a randomization that preserves the statistical properties of the initial sample. The most well known bootstrap method was first put forth by [26]. In Efron's version of the bootstrap, each pseudo-sample is created by sampling with replacement from the data set until the pseudo-sample has the same number of data points as the original sample. While this method has shown considerable utility and robustness throughout statistics, its correctness relies on the exchangeability, i.e., independence, of instances. One alternative to Efron's bootstrap is the the wild bootstrap [93]. Rather than rely on resampling from the original data set with replacement, the wild bootstrap performs external randomization by multiplying by an external bootstrap process. The dependent wild bootstrap [79, 50] provides an extension of the wild bootstrap to dependent time series samples by replacing $\mathcal{N}(0, 1)$ with a sample from a process that mimics the inter-instance dependence of

the original sample. This strategy has been successfully used previously by [15] for applying HSIC in time series. We now present an extension of the dependent wild bootstrap [50, 79, 15] to relational domains.

We impose the following assumptions on the bootstrap process:

A12. *The row-wise strictly stationary triangular array* $(W_t^*)_{t=1}^n = (W_{t,n}^*)_{t=1}^n$ *with ordering following a given filtration* π *is independent of* $X_1, \ldots, X_n$.

A13. $E^* W_1^* = 0$ *and* $Cov(W_s^*, W_t^*) = \rho(d(s,t)/l_n)$, *where* $d(s,t)$ *is the shortest path graph distance between* s *and* t, $\rho(u) \to_0 1$, $\sum_i^{n-1} \rho(|r|)/l_n = (O)(l_n)$ *with* $l_n \to_{n\to\infty} \infty$ *and* $l_n = o(n)$.

A14. *The variables* $(W_{t,n}^*)_{t=1}^n$ *are* τ *weakly dependent with coefficients* $\tau^*(r) \leq K\zeta^{r/l_n}$ *for* $r = 1, \ldots, n$ *some* $\zeta \in (0,1)$ *and* $K < \infty$.

We note that these are a generalization of the assumption imposed by prior work [50, 15] to allow for processes operating over general graph structures.

With the proper assumptions in place, we now present a proof of consistency for the wild bootstrap on network-structured domains. As with our prior proof, the core contribution is a generalization of the strategy of Leucht and Neumann [50] from the time-series setting to networks where it is possible to reach a set of instances at distance k.

Theorem 2. *Under the aforementioned assumptions, for* $i = 1, \ldots, 4$, $V_{n,1}^* \xrightarrow{d} Z$ *in probability. Further, if the limiting distribution function is continuous then* $\sup_{x\in\mathbb{R}} |P^*(V_{n,i}^* \leq x) - P(V_n \leq x)| \xrightarrow{P} 0$.

Proof. Leucht and Neumann [50], and Chwialkowski, et al. [15] have shown that $V_{1,n}^*, \ldots, V_{1,n}^*$ are asymptotically equivalent. Thus, without loss of generality we will focus on the case of $V_{1,n}$. There are two intermediate results that are needed in order to prove our final result, the correctness of the approximation of the V-statistic by

V^*, and asymptotic normality. We will address these two in order.

Approximation of the V-statistic:

Following Leucht and Neumann [50], let

$$V_{n,1}^{(K)*} := \frac{1}{n}\sum_{s,t=1}^{n} h^{(K)}(X_s, X_t)W_s^* W_t^* = \sum_{k=1}^{K}\lambda_k\left(\frac{1}{\sqrt{n}}\sum_{s=1}^{n}\Phi_k(X_s)W_s^*\right)^2$$

We are given that $V_{n,1}^* \geq V_{n,1}^{(K)*}$ for all K due to the fact that $h(\cdot,\cdot) - h^{(K)}(\cdot,\cdot)$ is a positive semi-definite kernel. We will now show

$$\limsup_{n\to\infty} P\left(P^*(|V_{n,1}^* - V_{n,1}^{(K)*}| > \epsilon) > \delta\right) \underset{K\to\infty}{\longrightarrow} 0 \tag{3.3}$$

$$\forall \delta, \epsilon > 0 \tag{3.4}$$

after applying Markov's inequality we apply the following approximation, which holds under the assumption of bounded degree of all nodes in the graph:

$$\begin{aligned}
&EE^*(V_{n,1}^* - V_{n,1}^{(K)*}) \\
&= \frac{1}{n}\sum_{k=K+1}^{\infty}\lambda_k\sum_{s,t=1}^{n} E\left[\Phi_k(X_s)\Phi_k(X_t)\right]\rho(d(s,t)/l_n) \\
&\leq \sum_{k=K+1}^{\infty}\lambda_k\left\{1 + \sum_{r=1}^{\max(d)}\frac{2(n-|d=r|)}{n}\left|E\left[\Phi_k(X_0)\left(\Phi_k(X_r) - \Phi_k(\widetilde{X}_r)\right)\right]\right||\rho(r/l_n)|\right\} \\
&\leq \sum_{k=K+1}^{\infty}\lambda_k + 2\sum_{r=1}^{\max(d)}|d=r|\sqrt{\sum_{k=K+1}^{\infty}\lambda_k}\sqrt{E\sum_{k=K+1}^{\infty}\lambda_k\left[\Phi_k(X_r) - \Phi_k(\widetilde{X}_r)\right]^2} \\
&\leq \sum_{k=K+1}^{\infty}\lambda_k + 2\sqrt{\sum_{k=K+1}^{\infty}\lambda_k}\sqrt{2\mathrm{Lip}(h)}\sum_{r=1}^{\infty}|d=r|\sqrt{\tau(r)}
\end{aligned}$$

Where $d(s,t)$ is the shortest path graph distance between nodes s and $t, \max(d)$ the maximum distance in the graph, $|d=r|$ is the number of nodes with shortest path distance of r from node 0, and $\widetilde{X}$ is a copy of X_r that is independent of X_0

satisfying $E\|X_r - \widetilde{X}_r\|_1 \leq \tau(r)$. Because $\sum_{k=1}^{\infty} \lambda_k = Eh(X_0, X_0) < \infty$ implying $\lambda_{k=K+1}\infty\lambda_k \to 0$ as $K \to \infty$, we arrive at $\sup_n E\left|V_n - V_n^{(K)}\right| \underset{K\to\infty}{\longrightarrow} 0$.

Central limit theorem for partial sums:

It remains to show that $\frac{1}{\sqrt{n}}\sum_{t=1}^{n} Y_t^* \xrightarrow{d} (Z_1, \ldots, Z_K)' \sim \mathcal{N}(0_K, \Sigma_K)$ in probability, where $(\Sigma_K)_{j,k}) = \Sigma_{r=-\infty}^{\infty} Cov(\Phi_j(X_0), \Phi_k(X_r))$. Assuming there exists a filtration, π, such that for indices in the sequence $i, j = 1, \ldots, n \in \pi$, $\rho(|i-j|/l_n) \leq c\rho(d(i,j)/l_n)$, with $c < \infty$[4], the proof provided by Leucht and Neumann [50] may be applied directly.

We can now apply Corollary 6.1 of Leucht and Neumann [50], which implies $V_{n,1}^{(K^*)} \xrightarrow{d} \sum_{k=1}^{K} \lambda_k Z_k^2$, combining this with equation 3.3 we arrive at $V_{n,1}^* \xrightarrow{d} Z$ in probability by Theorem 2 of Dehling, et al. [23]. □

3.3 Specifying the Covariance Matrix

A core assumption of the wild bootstrap procedure is a faithful model of the covariance structure between instances. So far it has been assumed throughout that we are given access to the true covariance matrix for the bootstrap process. However, in practice this assumption is rarely realistic. Mis-specification of the covariance matrix may lead to increased levels of either type I (if instance dependence is underestimated), or type II (if instance dependence is overestimated) errors. Since we do not assume to have access to the true covariance function amongst instances, we must find a suitable approximation. We will now describe two heuristics. Both methods center around constructing the bootstrap process $(W_{t,n}^*)_{t=1}^n$ by sampling from a Gaussian process with mean zero, unit variance, and l-dependence, i.e., all nodes separated by shortest path graph distance of at least l are independent.

[4]Under our assumption of bounded degree, one such filtration is an ordering defined by a breadth first search beginning at X_0.

3.3.1 Construction of Covariance via Graph Kernels

We first consider the covariance defined by normalized version of the random walk kernel [81] or a diffusion kernel [44]. These are widely used kernels for graph structured data defined respectively as:

$$C_{diff} = \exp\left(\frac{\sigma^2}{2}\widetilde{L}\right)$$

$$C_{rw} = (\alpha I - \widetilde{L})^p, \alpha \geq 2, p \geq 0$$

with $\widetilde{L}$ being the normalized Laplacian, $\widetilde{L} = D^{-1/2}(D-A)D^{-1/2}$, D being a diagonal degree matrix and A an adjacency matrix. Note that exp in this context is the matrix exponential, not element-wise exponentiation of the matrix. The random walk kernel is an approximation of the diffusion kernel with $\frac{p}{\alpha} = \sigma^2$ [81]. To see why these obey l-dependence, consider the random walk kernel where, by definition, the covariance between two nodes with shortest path distance length greater than p is zero. It follows directly that the resulting process displays l-dependence with $l = \frac{p}{\alpha}$. Finally, in order to ensure unit variance local normalization [90] can be employed, i.e., $C_{i,j} = \frac{C_{i,j}}{\sqrt{C_{i,i}}\sqrt{Cj,j}}$.

The main drawback of this approach is the specification of hyper-parameters. Throughout the experiments we will use the following heuristic for the random walk Laplacian. We fix α to a constant, and then iterate over $p = 1, \ldots, k$, choosing the p that maximizes dependence between the graph kernel and the observed values of (X, Y).

3.3.2 Inferring via Eigenvalue Optimization

We now present a non-parametric procedure for inferring the covariance matrix. This formulation allows us to learn the covariance matrix without needing to specify the hyperparmeters of the underlying diffusion process *a priori*. As an objective, we seek to maximize the dependence between the observed values of the random

variables, and the relative position of their corresponding graph-nodes in latent space as given by the inferred graph kernel.

More concretely, Let U, λ be the eigenvectors and values of the normalized graph Laplacian, $\widetilde{L} = I - D^{-\frac{1}{2}} A D^{-\frac{1}{2}}$. Further define $\mathbf{U_i} = U_i U_i^T$, and let the subscript c denote centering, i.e. $\mathbf{K}_{x,y_c} = H\mathbf{K}_{x,y}H$, where $H = I - \frac{1}{n}\mathbf{1}\mathbf{1}^T$ is a centering matrix.

$$\max_{\lambda} \frac{\langle \sum_i \lambda_i \mathbf{U}_c, K_{x,y_c} \rangle_{\mathcal{F}}}{\| \sum_i \lambda_i \mathbf{U}_c \|_{\mathcal{F}} \| K_{x,y_c} \|_{\mathcal{F}}} + \gamma \|\lambda\|_1$$

, $\mathbf{M}$ be a matrix where $\mathbf{M}_{ij} = \langle U_{i_c} U_{i_c}^T, U_{j_c} U_{j_c}^T \rangle_{\mathcal{F}}$, and

$$\mathbf{a} = \left(\langle U_{1_c} U_{1_c}^T, K_{x,y_c} \rangle_{\mathcal{F}}, \ldots, U_{n_c} U_{n_c}^T, K_{x,y_c} \rangle_{\mathcal{F}} \right)$$

Allowing $\mathbf{v} = \frac{\lambda}{\|\lambda\|}$, we recover λ as the solution to the following optimization problem

$$\min_{\mathbf{v}_i > \mathbf{v}_{i-1} \geq 0} \mathbf{v}^T \mathbf{M} \mathbf{v}^T - 2\mathbf{v}^T \mathbf{a} + \gamma \|\mathbf{v}\|_1 \tag{3.5}$$

This problem is equivalent to solving an L_1 regularized hard SVM, and can easily be solved with off-the-shelf optimization software. The inclusion of centering, order constraints, and sparsity via the L_1 norm are important aspects of a complete solution:

- Centering provides a correspondence to dependence maximization and has been shown to be superior to the uncentered kernel target alignment empirically [18].
- The ordering constraint ensures that the learned dependence function is smooth [99]. This is required to satisfy the assumptions made earlier when defining the wild bootstrap procedure.
- The L_1 penalty imposes sparsity on the eigenvalues. By observing that these eigenvalues correspond to edge weights in the original graph [10], we see that this sparsity corresponds to preferring quickly decaying dependence functions.

We note that this problem shows more than a passing resemblance to prior work in (multiple) kernel learning. [18] showed a variant of this problem without the ordering constraint and L_1 penalty is equivalent to a hard margin SVM maximizing the centered kernel target alignment for multiple kernel learning. [99] proposed a semi-definite program for learning the eigenvalues of a graph kernel without either centering or the L_1 penalty. However, the combination of these aspects into an optimization problem for learning a bootstrap process is novel to this work.

3.4 Related Work

Chwialkowski, et al. [14] provide a permutation based procedure for simulating the null distribution of HSIC for weakly-dependent temporal data by considering random shifts of each time series. While effective for dependence testing, permutation via random shifts is not applicable to two-sample tests, or relational domains where there is no corresponding "shifting procedure". Chwialkowski, et al. [15] provide a wild bootstrap for simulating the null distribution of kernel based hypothesis tests in temporal domains. The proofs of consistency in this work are an extension of those results to structured domains under weak dependence. Zhang, et al. [98] provided extensions to the Hilbert-Schmidt independence criterion to structured domains. This work differs from [98] in two important aspects. First, this work only imposes an assumption of sparsity, i.e. finite degree distributions, while [98] require structures that can be decomposed into cliques [5]. The second difference are the mixing assumptions used. Zhang, et al. [98] imposes an assumption of *strong dependence*, which states that instances sufficiently distant from each other are statistically independent. Flaxman, et al. [28] assume an additive structure and remove effects from non-i.i.d. by considering the residuals after a Gaussian process regression. Finally,

[5]For intuition of this difference consider a lattice, this fails to decompose as required by Zhang, e t al. [98], but can easily be seen to be a sparse network.

Rattigan [75] provides a permutation test for assessing independence, and relies on empirical evaluation for evidence of correctness. In contrast, this work assumes *weak dependence* which, as described earlier, only assumes that the dependence tends to zero as distance increases.

3.5 Evaluation

To test the utility of the structured wild bootstrap we performed a series of experiments. Network structure was generated using the Barabasi-Albert, i.e., scale-free network model, and Watts-Strogatz, i.e., small-world network model, algorithms. Two independent draws were then made from a multivariate normal distribution, with the means drawn from $\mathcal{N}(0, 1)$, and covariance set using a random-walk kernel with α set to 2 which was then normalized using the procedure of Urry, et al. [90], outlined previously in this work. We then compared the following methods for testing dependence, using the proportion of type I errors at a 0.01 significance level by each of the following methods:

- HSIC with the wild bootstrap, as described in this work, using a random walk kernel as covariance
- HSIC with the wild bootstrap, as described in this work, using the eigenvalue optimization approach to infer the covariance
- HSIC with significance determined via permutation testing
- Pearson's correlation with asymptotic approximation of significance

Figure 3.2 show the type I errors with the number of nodes fixed to 300 as the autodependence varies. For both small-world and scale-free networsk, we can see that both Pearson and HSIC incur a large number of type I errors as the autodependence becomes more severe. In contrast the wild bootstrap based approach is fairly well

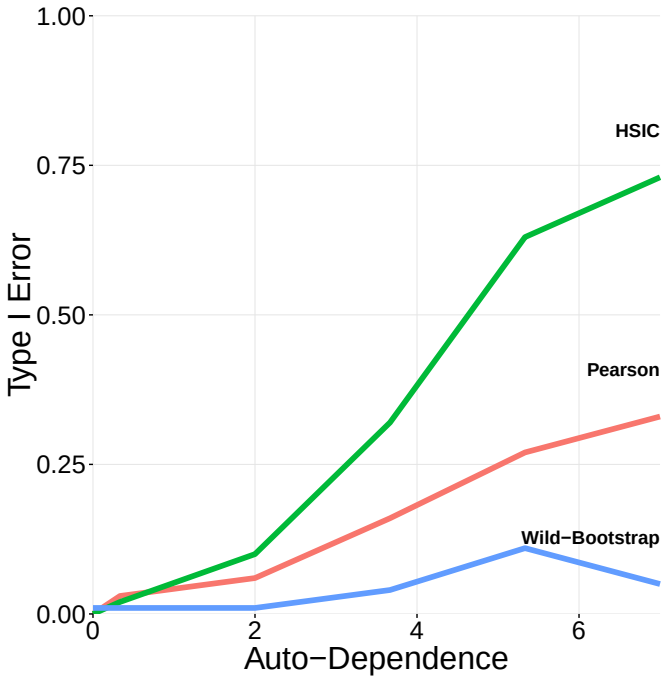


Figure 3.1: Scale-free network

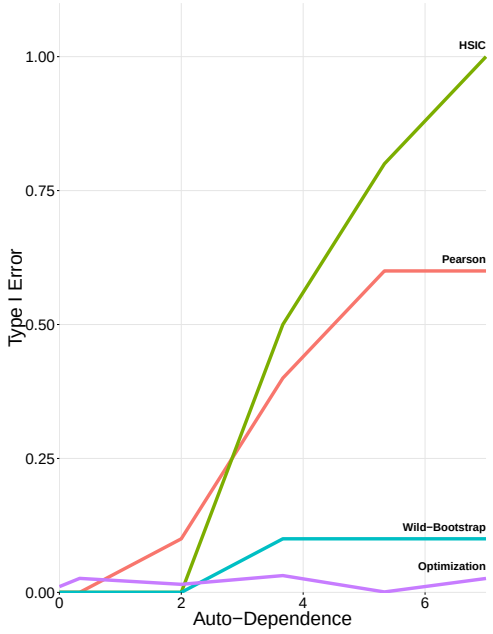


Figure 3.2: Small-world network

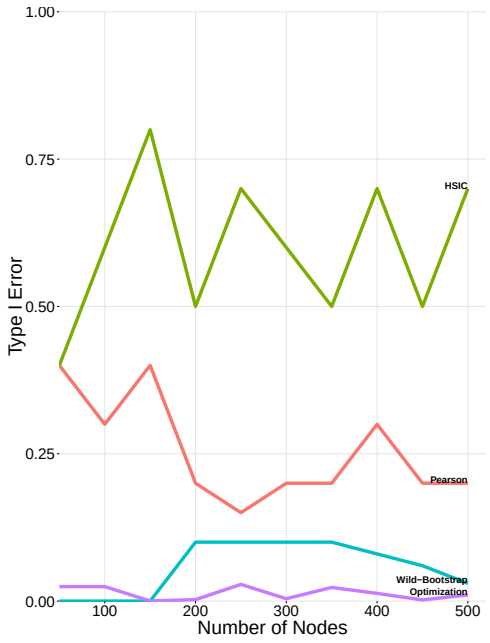


Figure 3.3: Type I errors for graphs with varying number of nodes and fixed auto dependence using the Barabasi-Albert model.

calibrated. At its most extreme the wild bootstrap approach shows a type I error of 10%, which is roughly 2.5 times smaller than Pearson's at the same level of auto-dependence. The optimization based method is more robust, with a type I error of 3%. When interpreting these results it is important to remember that while the total number of nodes stays fixed, the *effective* sample size is much smaller since the autodependence increases the inherent variability and creates uncertainty that are closer to an i.i.d. problem with a smaller number of samples [41].

Figure 3.3 shows the performance of each method with auto-dependence fixed at the largest value used in the prior experimental setting, with results pooled across small-world and scale-free networks. Once again, we see that the error rate for the wild bootstrap approach is much lower than the others. We can also see that the wild bootstrap approach shows less variability across sample sizes.

3.6 Conclusion

In this work we studied the problem of measuring dependence between variables in relational domains. We showed that the Hilbert-Schmidt independence criterion is consistent when relational autodependence is present in the data. We also showed that the null distribution can be efficiently simulated using a relational extension of the wild bootstrap for degenerate kernels. This is the first provably consistent bootstrap method for relational domains. We showed via a set of synthetic experiments that this procedure can yield a substantially lower type I error rate than the non-bootstrap counterparts. While we have focused on HSIC, our results are more general, covering all V-statistics with degree less than or equal to 4. Future work will focus on the extension to U-statistics, a broad class of commonly used statistics that includes absolute differences in means and rank-correlation measures.

CHAPTER 4

INFERRING CAUSAL DIRECTION OF RELATIONAL DEPENDENCE

Inferring[1] the direction of causal dependence between two random variables from observational data is a fundamental problem in statistical reasoning. There have been many advances in this area for data sets that are independent and identically distributed (i.i.d.) [39, 86, 53]. For relational data, recent work has studied the problem of inferring the effects of peers via *experimentation* [64, 7, 88]. However, the problem of identifying causal direction from *observational* relational data has yet to receive the same focus. In this work, we study the problem of inferring the causal direction of peer dependence from observational relational data. We provide theoretical and experimental results to show that the causal direction of peer dependence can be robustly inferred from observational data by comparing the magnitude of two similarity measures (one for each candidate direction).

For example, consider a study on the causes of personal debt. Data consist of the net worth and the average monthly discretionary spending of a large set of individuals, along with the position of each individual within a social network. One reasonable question is whether a person's friends influence his or her spending habits. If a person's spending and wealth are correlated with the wealth and spending of their friends, what can be inferred about the *causal* dependence among these quantities? A person's spending could be caused by their friends' wealth or vice versa

[1]Portions of this chapter previously appeared in UAI 2016 as Arbour, et al., "Inferring Causal Direction from Relational Data."[4].

(direct dependence), or both quantities could be caused by an unobserved variable (confounding).

This chapter examines when and how it is possible to differentiate among these scenarios. Specifically, we:

1. Identify a set of conditions under which the causal direction of relational dependence can be consistently inferred.

2. Investigate the effect of unobserved confounding on this approach to causal inference, and provide a simple test of relational confounding.

3. Provide an extension of our method to the case of non-linear dependence via kernel embeddings.

4. Show that the proposed measures are robust to both the magnitude of the noise and the functional form of the true dependence, through a set of simulations under a variety of graph structures and functional forms.

The rest of the chapter is structured as follows. Section 4.1 describes the problem setting. Section 4.2 presents a test of causal direction under deterministic linear dependence. Section 4.3 considers a relaxation of the assumptions by allowing for latent confounding and discusses the conditions under which latent confounding can be identified. Section 4.4 generalizes these results to the case where the similarity is measured by embedding the data in a reproducing kernel Hilbert space (RKHS). Section 4.5 presents experimental evaluation of these results using synthetic data and a variety of marginal and conditional distributions, as well as networks generated from the Erdős-Rényi, Watts-Strogatz, and Barabási-Albert models. Section 4.6 presents a demonstration of our method on Stack Overflow, a large online community where users ask and answer computer science related questions.

4.1 Problem Setting

Relational domains consist of multiple types of entities that interact with each other through multiple types of relationships. Consider, for example, the domain of academic publishing: authors write chapters, chapters cite other chapters and so on. In this work, for clarity of exposition and without loss of generality, we focus on *networks*, a specific type of relational domains with a single type of entity (e.g., people) and a single type of relationship (e.g., friendship)[2]. An instantiation of a network consists of a set of people and a set of friendships among these people. This can be represented with an undirected graph $G = \langle V, E \rangle$ with n vertices. Nodes correspond to people and an edge denotes friendship between the nodes it connects. Every node of the graph $v_i \in V$ is associated with a pair of random variables, X_i and Y_i. These correspond to attributes of a person, for example wealth and spending habits. For every node, we can define a new random variable as a function of the random variables of its neighboring nodes. Specifically, in this section, we define a new random variable $X_i{}'$ as the sum of X_j over v_i's neighbors:

$$X_i{}' = \sum_{\{v_j | \langle v_i, v_j \rangle \in E\}} X_j$$

Similarly,

$$Y_i{}' = \sum_{\{v_j | \langle v_i, v_j \rangle \in E\}} Y_j.$$

For the remainder of the chapter, we refer to functions of random variables of neighboring nodes, such as $X_i{}'$ and $Y_i{}'$, as *relational variables* and to random variables of the node, such as X_i and Y_i, as propositional variables. To avoid ambiguity, we refer to dependence between a relational variable and a propositional variable as *relational dependence*.

[2]The extension to the more general multi-entity/multi-relationship case is straightforward. We provide the necessary details for this extension in the supplement.

A very common assumption in relational domains is that of *templating*, i.e., random variables in different nodes follow the same distribution [43]. In our case, this would imply that the distribution of X_i is the same for all i (and the same for Y_i, X_i', and Y_i'). This allows us to reason about four random variables on a model level: X, Y, X', and Y'.

Since we are reasoning over random variables across all nodes of the network, it is convenient to represent them as vectors. Let $\mathbf{x} = \langle X_1, \ldots, X_n \rangle$ be a vector with the random variables X_i for every node and, similarly, $\mathbf{x}' = \langle X_1', \ldots, X_n' \rangle$. Let A denote the adjacency matrix of the graph defined as:

$$A_{ij} = \begin{cases} 1, & \text{if } (v_i, v_j) \in E. \\ 0, & \text{otherwise.} \end{cases}$$

We note that A is a symmetric matrix since G is an undirected graph. We can write the vector of the sum of the friends (i.e., the vector $\mathbf{x}'$) as $\mathbf{x}' = A\mathbf{x}$. Similarly, $\mathbf{y}' = A\mathbf{y}$.

We use D to denote the degree matrix of the graph:

$$D_{ij} = \begin{cases} d_i, & \text{if } i = j. \\ 0, & \text{otherwise.} \end{cases}$$

4.1.1 Assumptions

Throughout this chapter, we make the following assumptions:

A15. *G is an undirected graph.*

A16. *Each node $v \in V$ has degree of at least 1.*

A17. *The distribution of X_i and Y_i is the same for all $v_i \in V$ (templating).*

A18. *There are no feedback cycles, i.e. $Y \to X \Rightarrow X \not\to Y$ for any two (relational or propositional) variables.*

Further, we initially assume (and later relax that assumption) that:

A19. *There are no confounding variables, i.e., unobserved variables that are common causes of the observed attributes.*

Section 4.3 is devoted to examining under which conditions this assumption can be loosened, while maintaining the ability to identify causal direction. Moreover, assumptions A18 and A19 mirror those found in the literature on determining causal direction between two propositional variables [86, 39, 53].

4.2 Direction Under Linear Dependence

In this section we show that, under the assumptions of linearity and a small amount of noise, peer dependence is asymmetric and the true causal direction can be consistently inferred. This is an inherent property of relational domains. The extension to non-linear dependencies is provided in Section 4.4.

To measure dependence between variables, we consider the square of Pearson's correlation, a common and widely employed measure of linear correlation between variables. Pearson's correlation between two variables X and Y can be computed from a sample $\mathbf{x}$, $\mathbf{y}$ as follows:

$$\rho(\mathbf{x}, \mathbf{y}) = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2}},$$

where $\bar{x}$ and $\bar{y}$ are the means of $\mathbf{x}$ and $\mathbf{y}$ respectively. We consider the square of the correlation to restrict the range of the metric to [0,1], rather than [-1,1].

Given a measure of dependence, a reasonable question is whether the measure is symmetric for relational data. Surprisingly, it is not. Given this, another reasonable question is what can be inferred by examining the dependence values in both directions. Surprisingly, the causal direction of dependence can be inferred from the resulting asymmetry.

We begin by handling a simplified case: Y is a deterministic function of the X values of related nodes. Specifically, we assume that Y_i is the scaled mean of the X_j variables of the related instances:

$$Y_i = \frac{\beta}{d_i} \sum_{i=1}^{d_i} X_j$$

Or, in matrix notation: $\mathbf{y} = \beta D^{-1} A\mathbf{x}$.

Under certain assumptions about the structure of the graph and the form of the dependence, the squared correlation in the causal direction will be greater that the squared correlation in the opposite direction.

Proposition 1. *Assume that G is a d-regular graph[3], the true generative process is $\mathbf{y} = \beta D^{-1} A\mathbf{x}$ for some constant β, and assumptions A1-A5 hold. Then, $\rho^2(\mathbf{x}', \mathbf{y}) > \rho^2(\mathbf{y}', \mathbf{x})$.*

Proof. The left-hand-side of the inequality, given that by definition $\mathbf{x}' = A\mathbf{x}$, can be written as:

$$\begin{aligned} \rho^2(\mathbf{x}', \mathbf{y}) &= \rho^2(A\mathbf{x}, \beta D^{-1} A\mathbf{x}) \\ &= \rho^2(A\mathbf{x}, \frac{\beta}{d} A\mathbf{x}) = 1 \end{aligned}$$

It remains to show that $1 > \rho^2(\mathbf{y}', \mathbf{x})$ which holds, unless $\rho^2(\mathbf{y}', \mathbf{x}) = 1$. Equality holds only when $\mathbf{y}' = \beta A D^{-1} A\mathbf{x}$ is a linear combination of $\mathbf{x}$, or in words, when the values of a node's friends of friends are a linear combination of that node's value. For random values of X, that happens for a degenerate network structure where every node has one friend of a friend and is the exact same starting node. This would happen, for example, in the case of a regular graph with degree 1 (pairs of nodes). □

[3]A graph is d-regular if every vertex has degree d.

In the case where Y is a noisy function of X, a similar inequality holds.

Proposition 2. *Assume that the true generative process is* $\mathbf{y} = \beta D^{-1}A\mathbf{x} + \epsilon$ *for some constant* β*, where* ϵ *is a vector with the noise terms. Moreover, assume that assumptions A1-A5 hold and* X *and* Y *are scaled to mean 0. Then the following holds:*

$$\rho^2(\mathbf{x}', \mathbf{y}) > \rho^2(\mathbf{y}', \mathbf{x}) \Leftrightarrow$$
$$\frac{Var(AD^{-1}A\mathbf{x}) + Var(A\epsilon)}{Var(D^{-1}A\mathbf{x}) + Var(\epsilon)} > \frac{Var(A\mathbf{x})}{Var(\mathbf{x})}.$$

Proof.

$$\begin{aligned}
\rho(\mathbf{x}', \mathbf{y}) =& \rho(A\mathbf{x}, D^{-1}A\mathbf{x} + \epsilon) && (4.1)\\
=& \frac{Cov(A\mathbf{x}, D^{-1}A\mathbf{x}) + Cov(A\mathbf{x}, \epsilon)}{Var(A\mathbf{x})\big(Var(D^{-1}A\mathbf{x}) + Var(\epsilon)\big)} \\
=& \frac{Cov(A\mathbf{x}, D^{-1}A\mathbf{x})}{Var(A\mathbf{x})\big(Var(D^{-1}A\mathbf{x}) + Var(\epsilon)\big)} && (4.2)
\end{aligned}$$

$$\begin{aligned}
\rho(\mathbf{y}', \mathbf{x}) =& \rho(AD^{-1}A\mathbf{x} + D^{-1}A\epsilon, \mathbf{x}) && (4.3)\\
=& \frac{Cov(AD^{-1}A\mathbf{x}, \mathbf{x}) + Cov(\mathbf{x}, D^{-1}A\epsilon)}{Var(\mathbf{x})\big(Var(AD^{-1}A\mathbf{x}) + Var(D^{-1}A\epsilon)\big)} \\
=& \frac{Cov(AD^{-1}A\mathbf{x}, \mathbf{x})}{Var(\mathbf{x})\big(Var(AD^{-1}A\mathbf{x}) + Var(D^{-1}A\epsilon)\big)} && (4.4)
\end{aligned}$$

The covariance, given that the mean of X and Y is 0, is equal to the inner product of the variables.

$$Cov(A\mathbf{x}, D^{-1}A\mathbf{x}) = \langle A\mathbf{x}, D^{-1}A\mathbf{x}\rangle \tag{4.5}$$

$$= \mathbf{x}^\top A^\top D^{-1} A\mathbf{x} \tag{4.6}$$

$$= \mathbf{x}^\top A D^{-1} A\mathbf{x} \tag{4.7}$$

$$Cov(AD^{-1}A\mathbf{x}, \mathbf{x}) = \langle AD^{-1}A\mathbf{x}, \mathbf{x}\rangle \tag{4.8}$$

$$= \mathbf{x}^\top A D^{-1} A\mathbf{x} \tag{4.9}$$

Therefore, for the square of the correlations we can write:

$$\rho(\mathbf{x}', \mathbf{y}) > \rho(\mathbf{y}', \mathbf{x}) \Leftrightarrow$$

$$\frac{1}{Var(A\mathbf{x})\big(Var(D^{-1}A\mathbf{x}) + Var(\epsilon)\big)} >$$

$$\frac{1}{Var(\mathbf{x})\big(Var(AD^{-1}A\mathbf{x}) + Var(A\epsilon)\big)} \Leftrightarrow$$

$$\frac{Var(AD^{-1}A\mathbf{x}) + Var(A\epsilon}{Var(D^{-1}A\mathbf{x}) + Var(\epsilon)} > \frac{Var(A\mathbf{x})}{Var(\mathbf{x})}$$

□

The implication of proposition 2 is that the causal direction can be accurately inferred, as long as the relative influence of the noise distribution is small in comparison to the relationship between $AD^{-1}\mathbf{x}$ and $\mathbf{y}$. As we show during our experimental evaluation in Section 4.5, the method is quite robust to the effect of noise in practice.

4.3 Reasoning About Confounding

Throughout Section 4.1 we assumed the absence of confounding influences (assumption A19). However, in many real-world settings, this proves to be an unrealistic assumption. Within the relational setting, there are two distinct ways in which the relationship between variables can be confounded:

1. $\mathbf{x}$ and $\mathbf{y}$ may share a common relational cause, $A\mathbf{z}$, i.e., $A\mathbf{z} \rightarrow \mathbf{x}$ and $A\mathbf{z} \rightarrow \mathbf{y}$.

2. There is a variable $\mathbf{z}$ that is a non-relational cause of $\mathbf{x}$ and a relational cause of $\mathbf{y}$, i.e., $\mathbf{z} \to \mathbf{x}$ and $A\mathbf{z} \to \mathbf{y}$.

In what follows, we show that the first scenario is identifiable from data, while the second one is not.

Proposition 3. *If* $Cov(A\mathbf{x}, A\mathbf{y}) \geq Cov(A\mathbf{x}, \mathbf{y})$ *and* $Cov(A\mathbf{x}, A\mathbf{y}) \geq Cov(A\mathbf{x}, \mathbf{y})$*, then there exists a relational variable which is a common cause of* x *and* y*.*

Proof. Assume that the true generative structure is:

$$\mathbf{y} \sim D^{-1}A\mathbf{z} + \epsilon_{\mathbf{y}}$$
$$\mathbf{x} \sim D^{-1}A\mathbf{z} + \epsilon_{\mathbf{x}}$$

The covariance between $A\mathbf{x}$ and $A\mathbf{y}$ is then given by

$$\begin{aligned}
&Cov(Ax, Ay) \\
&\quad = Cov(AD^{-1}A\mathbf{z} + A\epsilon_{\mathbf{y}}, AD^{-1}A\mathbf{z} + A\epsilon_{\mathbf{x}}) \\
&\quad = Cov(AD^{-1}A\mathbf{z} + A\epsilon_{\mathbf{y}}, AD^{-1}A\mathbf{z}) + \\
&\qquad Cov(AD^{-1}A\mathbf{z} + A\epsilon_{\mathbf{y}}, A\epsilon_{\mathbf{x}}) \\
&\quad = Cov(AD^{-1}A\mathbf{z}, AD^{-1}A\mathbf{z}) + Cov(AD^{-1}A\mathbf{z}, A\epsilon_{\mathbf{x}}) \\
&\quad = Cov(AD^{-1}A\mathbf{z}, AD^{-1}A\mathbf{z})
\end{aligned}$$

The covariance between $A\mathbf{x}$ and $\mathbf{y}$, is given by:

$$
\begin{aligned}
&Cov(Ax, y) \\
&\quad = Cov(AD^{-1}A\mathbf{z} + A\epsilon_{\mathbf{x}}, D^{-1}A\mathbf{z} + \epsilon_{\mathbf{y}}) \\
&\quad = Cov(AD^{-1}A\mathbf{z}, D^{-1}A\mathbf{z} + \epsilon_{\mathbf{y}}) + \\
&\qquad Cov(A\epsilon_{\mathbf{x}}, D^{-1}A\mathbf{z} + \epsilon_{\mathbf{y}}) \\
&\quad = Cov(AD^{-1}A\mathbf{z}, D^{-1}A\mathbf{z}) + Cov(D^{-1}A\mathbf{z}, \epsilon_{\mathbf{y}}) \\
&\quad = Cov(AD^{-1}A\mathbf{z}, D^{-1}A\mathbf{z}) \\
&\quad \leq Cov(AD^{-1}A\mathbf{z}, AD^{-1}A\mathbf{z})
\end{aligned}
$$

$AD^{-1}A\mathbf{z}$ and $D^{-1}A\mathbf{z}$ are bounded by the size of the intersection between the set of a node's immediate neighbors and the set of its two-hop neighbors, since we have assumed $\mathbf{z}$ are marginally independent by construction. Each pair of one hop and two hop neighborhoods will diverge for at least the degree of the node for each node, since the two hop walk beginning from node i will return to that node an equal number of its degree, which implies the final inequality. □

Proposition 3 implies a very simple procedure for ruling out the presence of mutual relational confounding between two variables. First, the relative dependence is measured between $A\mathbf{x}, \mathbf{y}$ and $A\mathbf{y}, \mathbf{x}$ respectively. Then, these two values are compared against the measured dependence between $A\mathbf{y}, A\mathbf{x}$. If neither are larger than the between-relational variable dependence no determination of direction is made, since observed dependence is likely due to confounding.

We now turn to scenario two, which yields the following negative result:

Corollary 1. *Under confounding scenario 2, in the absence of noise, a false conclusion of dependence* $A\mathbf{x} \rightarrow \mathbf{y}$ *will be made.*

Proof. Assume the generative structure is given by:

$$\mathbf{x} \sim \mathbf{z}$$

$$\mathbf{y} \sim D^{-1}A\mathbf{z}$$

It can be immediately seen that the form of this dependence is identical to the form of proposition 1, where we substituted $\mathbf{z}$ for the $\mathbf{x}$. It follows that, in the no-noise setting, an incorrect determination of direct causation will be made. □

Note that this also applies in the case of a small amount of noise, as implied by proposition 2. This result shows that without the assumption of no-confounding a determination of non-causation can be reliably implied, but the converse is not necessarily true.

4.4 An Extension To Non-Linear Dependence

In the previous section, we showcased the applicability of our method for detecting linear dependence in relational data using correlation. An extension to more complex variables and non-linear dependence functions can be achieved by applying the kernel trick.

The centered *kernel target alignment* (KTA) is a normalized measure of dependence introduced by Cortes, et al. [18] within the context of multiple kernel learning. The measure is defined as:

$$\mathrm{KTA}(\mathbf{x}, \mathbf{y}) = \frac{\langle K^c_{\mathbf{x}}, K^c_{\mathbf{y}} \rangle_{\mathcal{H}}}{\|K^c_{\mathbf{x}}\|_{\mathcal{H}} \|K^c_{\mathbf{y}}\|_{\mathcal{H}}} \tag{4.10}$$

Where $\|\cdot\|_{\mathcal{H}}$ is the Frobenius norm, $\langle K^c_{\mathbf{x}}, K^c_{\mathbf{y}} \rangle_{\mathcal{H}}$ is the Frobenius norm of the inner product between $K^c_{\mathbf{x}}$ and $K^c_{\mathbf{y}}$ which is calculated by taking the trace of the inner product. $K^c_{\mathbf{x}}$ is a centered kernel matrix, defined as:

$$K^c_{\mathbf{x}} = \left[\mathbf{I} - \frac{1}{m}\mathbf{1}\mathbf{1}^T\right] K_{\mathbf{x}} \left[\mathbf{I} - \frac{1}{m}\mathbf{1}\mathbf{1}^T\right]$$

where $\mathbf{I}$ is the identity matrix and $\mathbf{1}$ is a column vector of ones with length m. If a linear kernel is used, KTA reduces to squared Pearson's correlation, which has been our measure of focus thus far. Using this connection, the following corollary provides for consistent estimation of causal direction under the deterministic case with arbitrary functional dependence.

Corollary 2. *Under assumptions A15, A16, A17, A18, A19, and further assuming that the generative structure is given by* $\mathbf{y} = D^{-1}A\phi(\mathbf{x})\beta$*, then* $KTA(A\mathbf{x}, \mathbf{y}) \geq KTA(A\mathbf{y}, \mathbf{x})$.

This follows as a straightforward extension of proposition 1. Because we are given by assumption that $KTA(A\mathbf{x}, \mathbf{y}) = 1$ and KTA is bounded from above by one, the inequality holds. Equality occurs only when the values of each node's friends of friends can be expressed as a sum of (feature-space embedded) values. For random values of X, this is reduced to the degenerate case of a graph of degree 1, as in proposition 1.

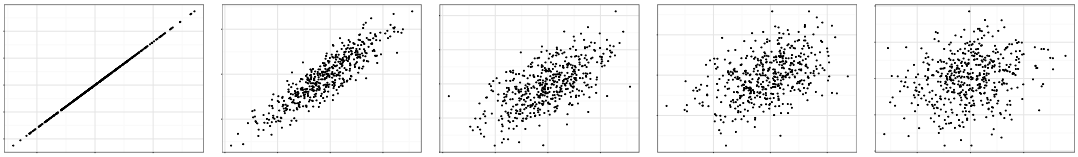
In practice, we note that the KTA based comparison relies on a number of hyper-parameters. The difficulty in choosing these parameters can result in poorer empirical performance. This problem has also been observed for other kernel-based approaches for causal inference [96]. We leave the investigation of hyper-parameter selection as future work.

4.5 Experiments

Our theoretical results focus on regular graphs, linear dependence, and absence of noise. In this section, we examine the effect that the network structure, the functional form of the dependence, and the presence of noise have on the efficacy of the linear and kernel based methods. [4]

[4]Code is available at https://github.com/darbour/RelationalCausalDirection.git.

4.5.1 Regular Networks



(a) $c_\epsilon = 0$ (b) $c_\epsilon = 0.25$ (c) $c_\epsilon = 0.5$ (d) $c_\epsilon = 1$ (e) $c_\epsilon = 2$

Figure 4.1: Scatterplots for the sum of X values of related nodes (x-axis) vs. the sum of X values of related nodes with additive Gaussian noise (y-axis). The noise coefficient (c_ϵ) varies from 0 to 2. The underlying network structure is a regular network of degree 10 with 500 nodes.

We first considered regular graphs with linear dependence—a setting that matches our theoretical analysis—and we examined the effect of noise. We considered networks with the total number of nodes ranging from 100 to 500 and varied the degree between 2 and 22 by increments of 5. For every graph structure, we generated data as follows:

$$\mathbf{x} \sim \mathcal{N}(0, 1)$$

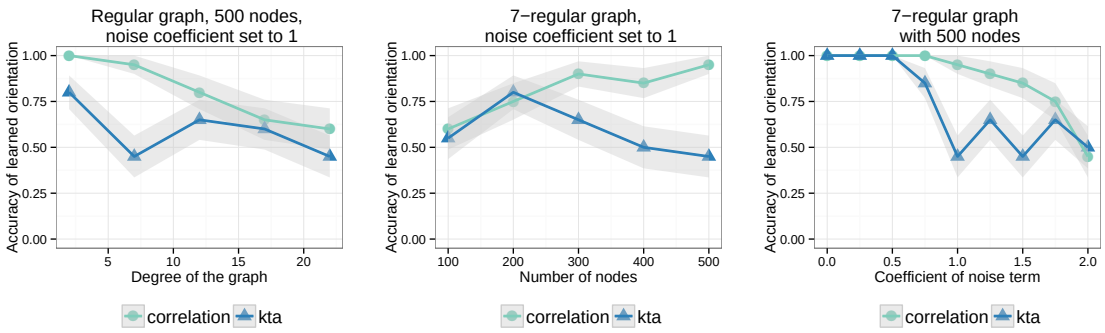
$$\epsilon \sim \mathcal{N}(0, 1)$$

$$\mathbf{y} \sim D^{-1}A\mathbf{x} + \beta\epsilon$$

where β is the coefficient of the noise and was varied between 0 and 2.

Figure 4.1 shows the relationship between $D^{-1}A\mathbf{x}$ and $\mathbf{y}$ for varying values of β. In the noiseless case (Figure 4.1a), $D^{-1}A\mathbf{x}$ and $\mathbf{y}$ are perfectly linearly correlated, as expected from the generating process. However, as the noise increases, the correlation between $D^{-1}A\mathbf{x}$ and $\mathbf{y}$ decays very quickly, approaching an adversarial case by the time the noise coefficient is $\beta = 1.0$.

We then measured dependence in each direction ($\mathbf{x}$ and $A\mathbf{y}$, $\mathbf{y}$ and $A\mathbf{x}$). The direction that produced the higher value for dependence was recorded as the inferred causal direction. To measure dependence, we used



(a) (b) (c)

Figure 4.2: Orientation accuracy for regular graphs for varying degree (4.6a), size of network (4.6b), and noise coefficient (4.6c).

1. the square of Pearson's correlation, and

2. KTA using RBF kernels with a fixed bandwidth of 1.0 for all kernel calculations.

Figure 4.6c shows the accuracy of both methods for a graph with 500 nodes and degree 7, while varying β. As expected from the our earlier theoretical results, both methods perform perfectly in the noise-less case, and continue to do so through $\beta = 0.5$. The linear method is significantly more robust to noise, remaining nearly perfect until $\beta = 1.0$.

We also examined the interplay between the graph structure (degree and number of nodes) and and the performance of each method. Figure 4.6a shows the performance for the case of a 500-node graph with noise coefficient of 1.0 with the degree varied between 2 and 22. Both methods become systematically worse as the degree (and thus the density of the network) increases. This is expected behaviour since an increase in the degree results in a lower *effective sample size* [40], which will reduce the expected efficacy of both methods. The converse of this effect can be seen in Figure 4.6b, where the accuracy of the linear based approach improves significantly as the size of

the network increases while the degree is kept constant (and thus the density of the network decreases).

4.5.2 Non-Regular Networks

We next compared the performance of both methods to a departure from the assumption of network regularity. We considered the three most common generative models of graphs. The Erdős-Rényi model creates networks where two nodes are connected with a given probability. Throughout the experiments, we considered a fixed connection probability equal to 0.2. The Watts-Strogatz model generates "small-world networks". It begins with a lattice with a given neighborhood size and randomly rewires edges according to a fixed probability. For our experiments, we used neighborhood size 5 and rewiring probability equal to 0.2. The final generative model we considered was the Barabási-Albert model. This model generates graphs that display preferential attachment. For our experiments the power of preferential attachment was set to 1.0. For each network we considered sizes between 100 and 1000, by increments of 100, with 20 graphs being drawn for each size.

We then considered the following data generation scenarios for all graph types:

$$\mathbf{x} \sim \mathcal{N}(0, 1)$$
$$\epsilon \sim \mathcal{N}(0, 1)$$
$$\mathbf{y} \sim f(D^{-1}A\mathbf{x}) + \beta\epsilon$$

where $f(\cdot)$ is a function of $D^{-1}A\mathbf{x}$. We considered three functional forms:

- $f(\cdot)$ is a simple linear function (linear)
- $f(D^{-1}A\mathbf{x}) = \tan(D^{-1}A\mathbf{x})$ (nonlinear)
- $f(D^{-1}A\mathbf{x}) = \left(D^{-1}A\mathbf{x}\right)^4$ (quad)

For each setting, β was varied between 0 and 2 by increments of 0.25.

The performance of both the linear and KTA method for fixed network size of 1000 nodes with the magnitude of noise varied is shown in Figure 4.4. For the Barabási model under linear dependence, both the linear and kernel methods appear to be very robust up until a noise coefficient of 2.0. The KTA based method generally outperforms the linear dependence method for non-linear dependencies. This is to be expected, as Pearson's correlation is a measure of linear dependence.

The performance in the case where β is held to 0.5 and the size of the network is varied from 100 to 1000 can be seen in Figure 4.3. Here we can see that in both the Barabási-Albert and Watts-Strogatz graph models, Pearson's correlation and KTA achieve better performance under linear dependence as the size of the network increases. However, for in the case of the Erdős-Rényi models both methods perform poorly consistently as the size of the network increases. This is due to the nature of the graph-generation process. Both the Barabási-Albert and Watts-Strogatz models become increasingly sparse as the size of the network is increased. However, in the case of Erdős-Rényi, the probability connection is constant. As a result, the effective sample size remains low when the number of nodes increases. This likely accounts for the poor performance of the linear estimator. The opposite effect is seen in the case of the Barabási-Albert model. In nearly all cases the performance of the estimators is highest in the case of the Barabási-Albert networks.

4.5.3 A Comparison to Relational Bivariate Edge Orientation

We also compared our results to the relational bivariate edge orientation (RBO) [56], the only other known method for testing causal direction in relational data. Maier et al. [56] introduced the relational bivariate edge orientation (RBO) as an edge-orientation procedure within the context of learning causal models of relational domains. RBO is defined with respect to conditional independence properties of rela-

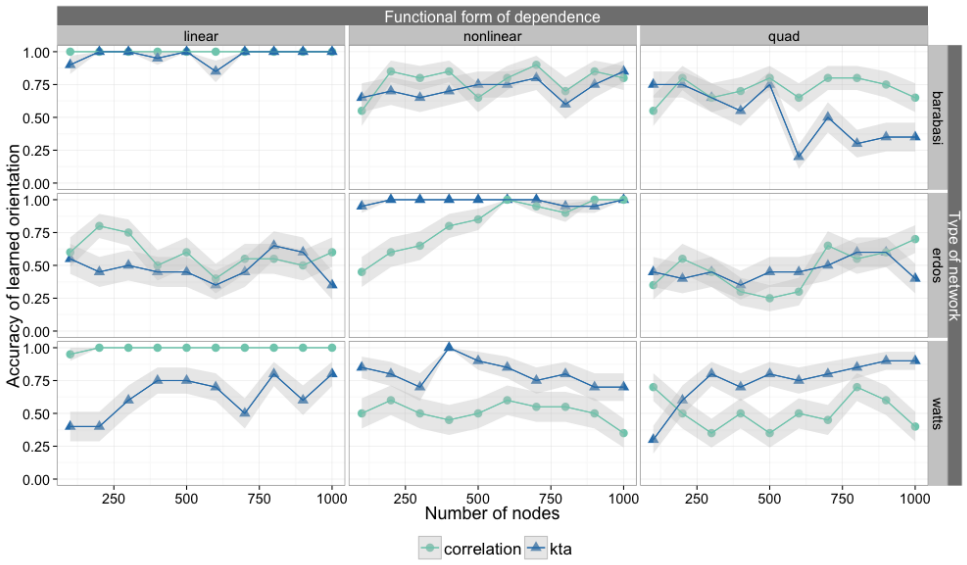


Figure 4.3: Orientation accuracy for various network types and functional forms, as the size of the graph increases. The noise coefficient is set to 0.5.

tional models. Specifically, rephrasing the definition of Maier et al. [56] for single-entity/single-relationship networks, for a relational dependence between Y' and X, RBO checks if Y' is in the separating set of X and X'. If not, then Y' is effectively a "relational" collider and is oriented as such: $Y' \leftarrow X$. Otherwise, the only alternative model is $Y' \rightarrow X$, given that dependencies that induce feedback cycles (such as $X \rightarrow X'$) are excluded by assumption. The correctness of RBO is defined with respect to a conditional dependence oracle. In practice, Maier et al.[56] follow the following procedure to infer causal direction between two relational variables:

1. Learn a linear model $\mathbf{x} \sim D^{-1}A\mathbf{x} + D^{-1}A\mathbf{y}$ to determine if $\mathbf{x} \perp\!\!\!\perp D^{-1}A\mathbf{x} \mid D^{-1}A\mathbf{y}$
2. If $\mathbf{x} \not\perp\!\!\!\perp D^{-1}A\mathbf{x} \mid D^{-1}A\mathbf{y}$, then return $D^{-1}A\mathbf{x} \rightarrow \mathbf{y}$, otherwise return $D^{-1}A\mathbf{y} \rightarrow \mathbf{x}$

We applied this procedure to the linear data-generating scenarios used in the previous two subsections, with one modification. Rather than testing a single perspective, we explicitly tested the conditional independence facts from the perspective of both $\mathbf{x}$

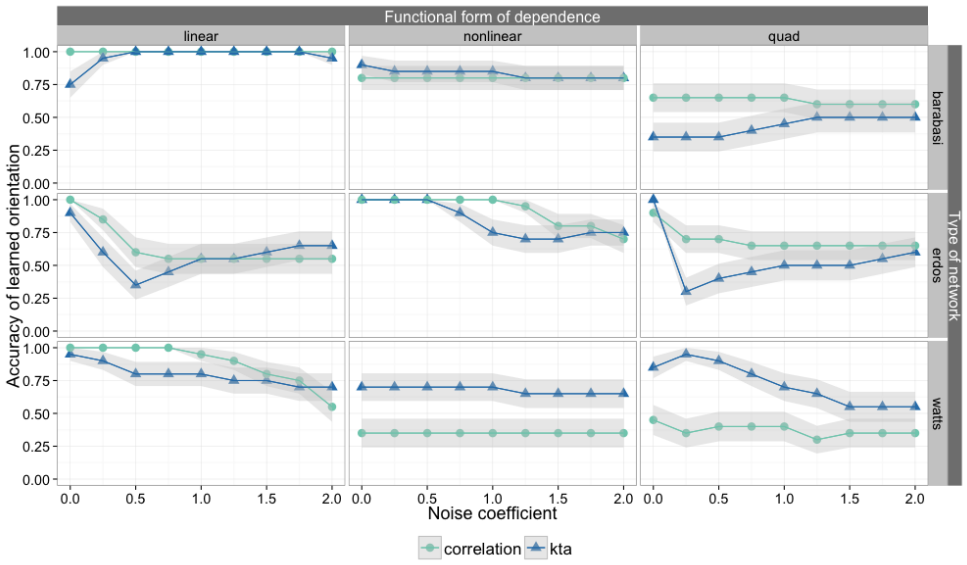
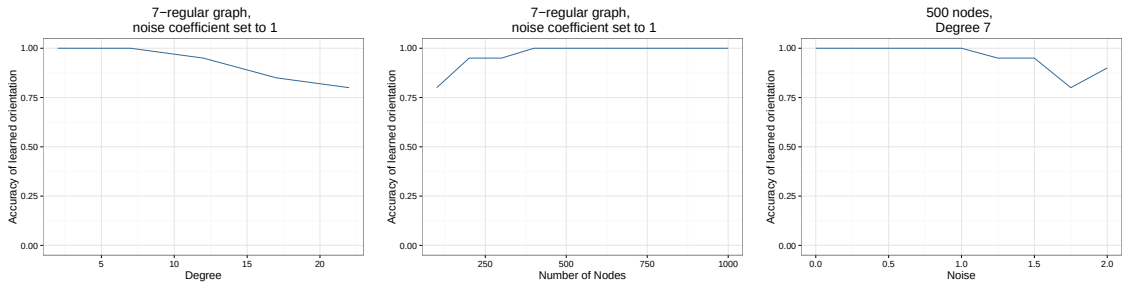


Figure 4.4: Orientation accuracy for various network types and functional forms, as the coefficient of the noise increases. The network size was kept constant at 1000 nodes.

and $\mathbf{y}$. We found that between all scenarios, RBO failed to induce dependence in 80-90% of cases. This has important ramifications for the RCD algorithm of Maier et al. [56]. As currently implemented, the RBO rule would have produced approximately %50 error rate, since it does not explicitly check both directions. Using our more conservative method, RBO would fire less frequently. In contrast, by incorporating the findings of the more direct marginal comparison presented here, vast numbers of edges would be accurately oriented. We plan on examining further integration of our findings into joint causal structure learning algorithms in future work.

4.5.4 Confounding Experiments

In addition to the experiments presented in the main text for determining the direction of dependence, we also empirically evaluated the efficacy of confounding detection. We replicated the experimental settings described in sections 6.1 and 6.2, except in this case both $\mathbf{x}$ and $\mathbf{y}$ are drawn using a direct dependence on a third variable $\mathbf{z}$. We then determined confounding by testing whether the covariance



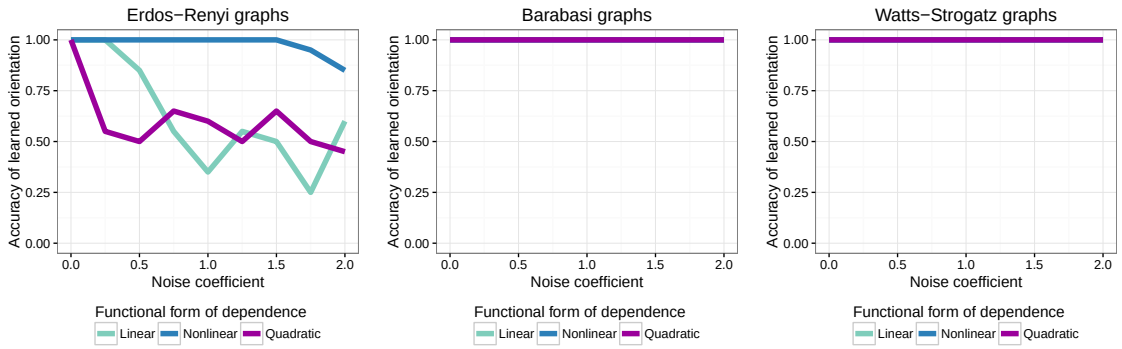
(a) (b) (c)

Figure 4.5: Accuracy detecting confounding for regular graphs for varying degree (4.6a), size of network (4.6b), and noise coefficient (4.6c).

between $A\mathbf{x}$ and $A\mathbf{y}$ was greater than both $Cov(A\mathbf{x}, \mathbf{y})$ and $Cov(A\mathbf{y}, \mathbf{x})$. The results for regular graphs can be seen in Figure 4.5. The confounding test is very robust across all of these dimensions. There is only a slight decrease in accuracy in even the most adversarial settings of large degree and high-noise generating scenarios. Figure 4.6 shows performance as the noise level is increased, across three non-regular graph generation algorithms. For two of the three graph generation procedures (Watts-Strogatz and Barabasi-Albert), there is near perfect performance. The Erdos-Renyi graph performance is considerably poorer. We conjecture that this is due to the high connectivity (each node is connected to approximately 20% of its neighbors), which greatly reduces the effective sample size. We plan on investigating methods to address causal inference on high-connectivity graphs as future work.

4.6 Real World Demonstration

In contrast to the propositional setting, where there is a number of labeled ground-truth data sets for testing novel methods of causal inference (e.g. [51]), to our knowledge, there are no known publicly available data sets which contain ground-truth relational causal relationships. In the absence of the ability to verify the relative efficacy of our findings on real-world data sets, we provide a demonstration of our



(a) (b) (c)

Figure 4.6: Accuracy detecting confounding for different types of networks graphs with varying noise.

method on a real-world data set. Specifically, we considered Stack Overflow, an online community where users pose and answer questions regarding software development. A user can post a question, which can be answered by anyone else within the community. Other users can then up/down vote questions and the given answers. These votes are tracked and the accrual of achieved points is displayed as the "reputation" of a user on the site. Moreover, users can comment on a question. Comments receive votes as well, but do not affect the reputation of a user. The data set consists of all users, questions, answers, comments, and votes from the inception of the site to 2014.

We tested three questions about user behavior on Stack Overflow. For every question we consider 100 sub-samples of 1000 data points. We computed KTA and Pearson's correlation in each direction. Significance of dependence was determined by performing permutation tests with 1000 permutations[5]. For all tests we set the significance threshold to be 0.01. When dependence was determined to be statistically significant, we also recorded how many times each direction was chosen by comparing test statistics in both directions.

[5]The consistency of dependence testing under these set of assumptions is provided as a chapter in the Appendix

The first question was: "Is there a relationship between the quality of a question and the quality of its subsequent answers?" To answer this, we used the scores of the questions and answers as proxies for their quality. All methods determined significance in both directions across all trials. However, the normalized statistics consistently determined the direction of dependence to be Question Quality $\rightarrow$ Answer Quality, while both of the un-normalized statistics consistently determined the direction of dependence to be Question Quality $\leftarrow$ Answer Quality. Clearly, the former conclusion matches intuition and temporal ordering far better than the latter.

The second question we considered was whether users with high reputation receive higher quality answers. This was quantified by using the reputation of a user and the score of the answers as a proxy for quality. In this case, we found that KTA and Pearson both detected significance for both directions. For direction, we found that both KTA and Pearson determined direction to be Reputation $\rightarrow$ Answer Quality for over 90% of the cases. This indicates that there may be bias in the Stack Overflow community towards questions asked by high reputation users. We caution that this does not take into account the possible latent confounder of question quality, i.e., higher reputation users may simply ask higher quality questions.

Finally, we looked at the efficacy of comments as a quality improvement mechanism, i.e., whether allowing users to comment on a question causes the poster to improve or clarify her post. We constructed this test with the comments posted for a question and whether revisions were subsequently made to the question. In this case we found that all of the methods inferred that there was *not* a significant relationship between the score of the comments and subsequent revisions to posts. This negative result indicates that the commenting system provided by Stack Overflow is not an effective mechanism for improving the quality of questions on the site.

4.7 Related Work

Relevant work to our investigation of methods for determining peer dependence in relational data falls into four basic categories. The most closely related work examines versions of this specific task with alternative methods. For example, Maier et al. [56], Rattigan [75], and Poole and Crowley [74] provide scenarios in which an asymmetry may arise similar to that observed in our tests for direction. However, in contrast to prior work, we study the phenomenon of asymmetric dependence directly and provide a formal examination which provides guarantees to the circumstances under which this asymmetry can be reliably leveraged. Further, we provide extensive simulation experiments that further show conditions under which direction can be found by considering the difference in dependence in both directions.

A second category of related work focuses on measuring causal dependence in non-relational (i.i.d.) data. For example, Peters , et al. [73] examine the problem of determining the direction of dependence with i.i.d. data by either assuming non-Gaussian noise and linear dependence or non-linear dependence and Gaussian noise. The problem of identifying causal direction in the case of deterministic, i.e., non-noisy data, was studied by Daniusis, et al. [19]. The setting considered was propositional data, and the proposed solution leverages properties of information geometry in order to find asymmetries between the conditional distributions of the two variables. In contrast, the relational setting considered provides a much more direct mechanism for determining direction.

A third thread of related work aims to detect non-causal dependence in relational data. This task has attracted attention in both statistical relational learning (SRL) community and in multiple areas of the social sciences. In SRL, Jensen and Neville [40] use a χ^2 test to detect autodependence in relational data and show its effect for feature selection. Angin, et al. [1] introduce a shrinkage estimator for autodependence in the presence of varying dependence strength. However, both of these rely on empirical

evaluation as evidence of correctness. Dhurandhar, et al. [24] and London, et al. [52] provide theoretical analysis for the inductive error of classification and regression in the relational setting.

In the social sciences, relational dependence has been examined under the monikers of peer influence, spillover, and interference. In the experimental setting, Eckles, et al. [25] characterize the threat to validity arising from the bias induced by relational dependence and provide experimental designs to reduce these effects. Manski [59], Vanderweele [91], and Samii and Aronow [6] examine methods for removing the bias associated with relational dependence, assuming discrete or linearly dependent data. Toulis, et al. [88] provide conditions for experimental design with binary treatments to identify peer influence. Ogburn and Vanderweele [68] characterize relational dependence in terms of graphical models, but do not present an explicit testing procedure. Work studying homophily and contagion (e.g., Christakis and Fowler [12], LaFonde, et al. [46]) is related but distinct in the task setup, as we do not assume the availability of temporal information.

Finally, our work is strongly connected and can serve as a complement to existing work on causal learning of relational domains. Maier, et al. [56] and Marazopoulou, et al. [62] present constraint-based algorithms to learn the structure of relational models from data. However, for their experiments they either rely on a d-separation oracle (without actual data), or use linear regression with mean-aggregation on synthetically generated data. As we showed in our synthetic experiments, these choices can lead to a large number of type II errors. This is especially troublesome for constraint-based structure learning algorithms where type II errors can lead to large deviations from the true causal model [16]. Such algorithms could leverage our test in order to improve results reported on data. Additionally, the directionality results presented in this chapter have implications for future work in constraint-based structure learn-

ing algorithms, since they imply a smaller Markov-equivalence class than what is commonly assumed.

4.8 Conclusions and Future Work

Inferring relational dependence is a task of general interest in a wide number of fields, from statistical relational learning to the social sciences. In this work, we have studied the problem of inferring causal direction in relational data. We have shown that, in contrast to the propositional setting, causal direction can be accurately inferred in relational data under the simplest functional forms such as linear deterministic dependence, without additional assumptions on the distribution of the underlying data. We then studied the problem of identifying confounding, showing the conditions when the presence of a relational confounding variable can be identified. Our experimental evaluation shows that these measures are robust, providing accurate inference under model and network mis-specification.

There are several promising avenues for future research. For causal learning, the ability to detect the direction of dependence in relational data implies that a different Markov equivalence class [83] holds for the relational setting than what is commonly assumed. Integration of the findings of this work into a causal learning algorithm could substantially improve the efficacy of existing methods such as RCD [56]. Further analysis of the interaction between the network structure and inference may further strengthen the robustness of the methods discussed here. Finally, the asymmetries shown to be inherent to relational data here may result in significant bias of conditional independence testing procedures. Incorporating this additional information is a first step in developing robust measures of conditional dependence in relational data to help determine causation, a problem which has broad application in both the statistical learning and social science communities.

CHAPTER 5

ESTIMATING EFFECTS

[1] A variety of methods have been devised for inferring causal effects from *observational* data. Classical methods for causal inference from observational data consist of two steps. First, an *adjustment set* [71] is identified, which consists of variables that are causally related to both the prospective cause variable (termed a *treatment*) and the potential effect variable (termed an *outcome*). Second, a procedure such as regression [70] or matching [76] is used to estimate the direct effect of treatment on outcome, correcting for the effects of the adjustment set. Extending this classical framework of estimation to relational data requires: (1) identifying adjustment sets in relational data, and (2) adjusting for the full range of the effects of those variables. Item 1 is primarily a structural question, and item 2 concerns estimation.

As a motivating example, consider the problem of estimating how a user-selected privacy setting influences the time that users spend interacting with an online social network. The privacy setting either requires users to explicitly approve others' posts to their page or it allows posting without such an approval process. Site administrators may be interested in changing the default privacy setting but want to ensure that such a change would not adversely affect site usage. Randomized experimentation on privacy settings may be controversial. Further, the propensity of users to share their posts with their friends could be influenced by characteristics of those friends. Figure 5.1 illustrates this example by indicating an implied correlation between so-

[1]Portions of this chapter previously appeared in KDD 2016 as Arbour, et al., "Inferring Network Effects from Observational Data."[3].

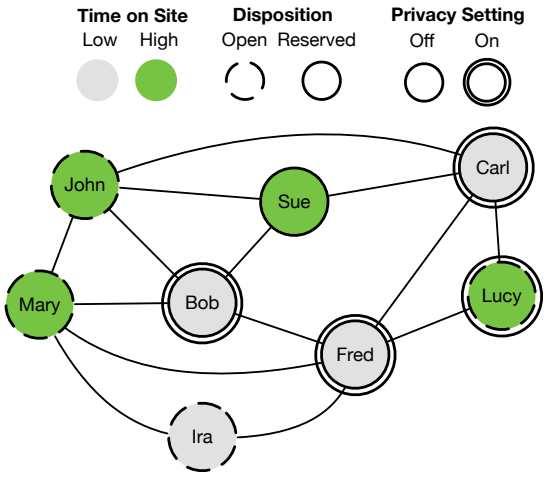


Figure 5.1: Social Network Privacy Example

cial disposition and use of the privacy setting as well as a correlation between social disposition and time spent on site.

The task of adjusting for this confounding is particularly challenging because some confounding variables can be properties of neighbors in the friendship network. In Figure 5.1, the social disposition and privacy settings of Lucy, Sue, John, and Fred could affect both the privacy settings of Carl and the amount of time he spends on the site. The task of deciding how to set privacy policy is an intrinsically causal question because it requires reasoning about the effect that intervening on the privacy setting would have on site usage. Additionally, modeling *network effects* is of central importance—time on site is a function of the privacy settings of an entire sub-network of friends rather than the privacy setting of an individual.

In this chapter, we present Relational Covariate Adjustment (RCA), the first reliable method for inferring arbitrary causal effects in networks from observational data. RCA uses a two-stage procedure. The first stage automatically identifies the set of variables that must be adjusted for. This stage uses relational d-separation [58], an extension of d-separation [71] to relational data. The second stage performs regression adjustment using relational non-parametric estimators. This adjustment procedure

makes limited assumptions about the nature of the causal relationship between treatment and outcome. We provide theoretical guarantees showing that RCA produces a consistent estimate of causal effect.

The rest of the chapter is structured as follows. Section 5.1 provides background for causal effect estimation and relational d-separation. Sections 5.2 and 5.3 introduce Relational Covariate Adjustment and discuss practical issues of implementation. Section 5.4 compares the estimates of RCA to estimates obtained via experimentation using multiple graph structures with data simulated under multiple functional forms, and shows that the performance of RCA can be competitive with experimental results.

5.1 Problem Setup

We assume that we are given an undirected graph $G = \langle V, E \rangle$. Let $N = |V|$, the number of vertices in the graph. Let T be a random variable composed of the treatment variables t_i of each node i in the network, so that $T = \langle t_1, t_2, \ldots, t_N \rangle$. Let $\boldsymbol{\pi}$ be an assignment to T, that is, $\boldsymbol{\pi} = \langle \pi_1, \pi_2, \ldots, \pi_N \rangle$, where π_i is an assignment to t_i. The *average causal effect* (ACE) is defined as the expected difference in outcome Y under treatment $\boldsymbol{\pi}$, contrasted with an alternate treatment $\boldsymbol{\pi'}$:

$$\text{ACE}(\boldsymbol{\pi}, \boldsymbol{\pi'}) = E[Y|do(T = \boldsymbol{\pi})] - E[Y|do(T = \boldsymbol{\pi'})]. \tag{5.1}$$

Throughout the chapter, we use the *do* operator [71] to refer to the interventional distribution, that is, the distribution that would arise due to manipulation of T rather than passive observation. Equation 5.1 may also be expressed in the potential outcomes framework [77] by regarding Y as a node-specific function of treatment. Ugander et al. [89] consider a special case of equation 5.1 where $\boldsymbol{\pi} = \vec{1}$ and $\boldsymbol{\pi'} = \vec{0}$. Hudgens and Halloran [38] refer to the above quantity as the *population average overall causal effect*.

Expressed directly in equation 5.1 is the notion that the outcome of subject i is a function of the entire treatment assignment vector, not only π_i. This distinction is critical for estimating network effects, as we now have a language to express interventions on multiple subjects. When dealing with causal quantities as in equation 5.1, it is common to assume that $E[Y|do(T = \boldsymbol{\pi})]$ is invariant with respect to treatment assignments to nodes which do not neighbor i. Let T_{nbr_i} denote the treatment variables of i's neighbors, and let $\boldsymbol{\pi}_{nbr_i} = \{\pi_j | \{i, j\} \in E\}$ and $\boldsymbol{\pi'}_{nbr_i}$ be multisets representing assignments to T_{nbr_i}. The neighborhood invariance assumption leads to the following reformulation of the average causal effect:

$$\begin{aligned}\text{ACE}(\boldsymbol{\pi}, \boldsymbol{\pi'}) = \frac{1}{N}\sum_{i=1}^{N} & E[Y|do(t_i = \pi_i, T_{nbr_i} = \boldsymbol{\pi'}_{nbr_i})] \\ & - E[Y|do(t_i = \pi'_i, T_{nbr_i} = \boldsymbol{\pi'}_{nbr_i})]. \end{aligned} \tag{5.2}$$

Equation 5.2 is consistent with the peer exposure models considered by Aronow et al. [5], Toulis and Kao [88], and the notion of effective treatments considered by Manski [59]. These causal quantities facilitate answering questions about interventional strategies including:

1. $E[Y|do(t_i = 1, T_{nbr_i} = \vec{1})] - E[Y|do(t_i = 0, T_{nbr_i} = \vec{0})]$: How would individual i's outcome change if i and its neighborhood were to be treated, as opposed to untreated? This quantity is the basis of $\text{ACE}(\vec{1}, \vec{0})$, the quantity considered by Ugander et al. [89] and Gui et al. [37].

2. $E[Y|do(t_i = 1, T_{nbr_i} = \vec{0})] - E[Y|do(t_i = 0, T_{nbr_i} = \vec{0})]$: How does subject i's expected outcome change if i is treated but no neighbors are treated? We might think of this effect as an "insulated" individual effect.

3. $E[Y|do(t_i = 0, T_{nbr_i} = \vec{1})] - E[Y|do(t_i = 0, T_{nbr_i} = \vec{0})]$: How does subject i's expected outcome change if i is left untreated but all neighbors are treated?

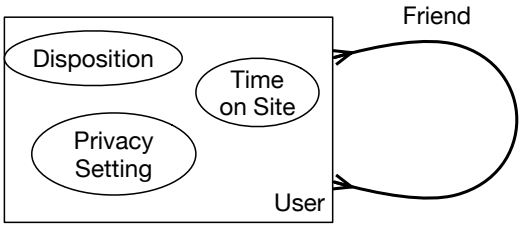


Figure 5.2: ER Diagram for Social Network Example

By considering different settings of $\boldsymbol{\pi}$ and $\boldsymbol{\pi}'$, we can examine a large number of possible intervention strategies, without being restricted to applying the same "type" of intervention to each node in the network. In practice, no single value of $\boldsymbol{\pi}$ could be used to apply interventions (2) and (3) in the list above to all nodes in the network. However, we can consider targeted interventions on specific individuals in the network, so it is useful to consider these effects.

5.2 Relational Adjustment Sets

We now briefly review the relational concepts necessary to describe Relational Covariate Adjustment, a more thorough explanation can be found in the background chapter.

5.2.1 Relational Causal Graphical Models

Let a *relational schema* $\mathcal{S} = (\mathcal{E}, \mathcal{R}, \mathcal{A}, \mathit{card})$ be the set of entity, relationship, and attribute classes of a domain. It includes a cardinality function that imposes constraints on the number of times an entity instance can participate in a relationship. Without loss of generality, we will focus our presentation on the case of a simple network, where there is a single entity, and a single many-to-many relationship, e.g. a social network. Continuing the example of Figure 5.1:

$$\mathcal{E} = \{\text{Users}\},$$

$$\mathcal{R} = \{\text{Friend}\},$$

$$\mathcal{A} = \{\text{time on site, disposition, privacy setting}\},$$

$$\mathit{card}(\text{Connected}) = \text{Many}.$$

Users are connected to potentially many other users, each of which has a time on site, disposition, and privacy setting attribute. Relational schemas are often visualized with entity-relationship diagrams as in Figure 5.2.

A *relational skeleton* is a partial instantiation of a relational schema that specifies the set of entity and relationship instances that exist in the domain. Using our online social network example, this corresponds to specific users and the friends that they connect to through the site. With a given schema, a *relational path* can be defined, which is a predicate that defines a path with respect to a schema. In our example, relational paths correspond to friendship paths, defined through the connectivity properties of the online social network. We will refer to variables with a trivial relational path (e.g., the immediate attributes of individuals), as *propositional variables*. *Relational variables* consist of a relational path and an attribute that can be reached through that path. For instance, the multiset of privacy settings for friends adjacent to user i is a relational variable. Relational variables can have causal dependencies defined between them, specified by a *relational model* $\mathcal{M} = (\mathcal{S}, \mathcal{D})$. This model consists of a collection of relational dependencies ($\mathcal{D}$) defined over a relational schema ($\mathcal{S}$). The relational model represents, as one example, the property that a user's time on site is affected by the privacy settings of adjacent users. $\mathcal{M}$ also specifies a parametrized conditional distribution of each relational variable given its parents. In the context of this work, we do not have access to these distributions and must estimate them from data.

Notation	Meaning
$U_i^0.ToS$	The value of variable ToS for instance i of entity U. For instance, this could represent the time on site of user i.
$U_i^1.\mathbf{ToS}$	A multiset representing the value of variable ToS on instances related to instance i of entity U through a path of length 1. For instance, this could represent the time that friends of user i spend on the site. We can represent users that are friends with i's friends with the notation U_i^2, and so on.

Table 5.1: Relational Notation

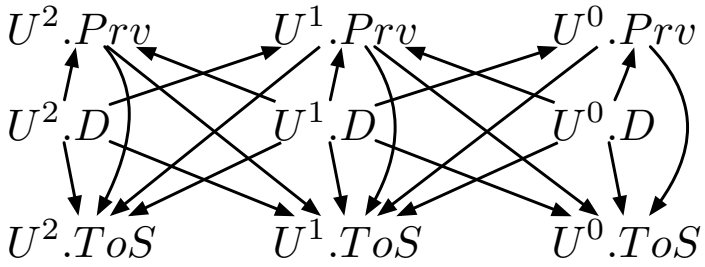


Figure 5.3: Abstract Ground Graph for the Social Network Example. In this example, each user's disposition ($U^0.D$) affects that user's privacy settings ($U^0.Prv$) and time on site ($U^0.ToS$). Further, the dispositions and privacy settings of a user's immediate peers ($U^1.D$ and $U^1.D$, respectively) affect that user's time on site. A user's privacy settings are also influenced by their peers' privacy settings. This structure repeats for U^2, representing friends of friends. Higher orders of U^p can be considered, but are not shown here.

To evaluate conditional independence queries on a model $\mathcal{M}$, we first construct an abstract ground graph (AGG) [58], a lifted representation that admits the computation of d-separation queries on multi-relational domains. Abstract ground graphs are defined from a given perspective, specifying a *base item* of the analysis, and include nodes that correspond to relational variables. In general, the construction of an AGG can involve creating auxiliary "intersection" variables. However, for the case of single-entity, single-relationship networks (e.g. social networks or simple communication networks) there exists a single AGG that can be represented without the use of auxiliary variables. That is:

Proposition 4. *Given a model with a single entity single, relationship schema, the complete set of d-separation facts can be determined by considering only propositional variables and relational variables.*

Proof. In constructing a conditional independence query with relational d-separation [56], paths composed of propositional variables, relational variables, and *intersection variables* must be considered. The set of propositional and relational variables to be considered for a perspective is directly identifiable from the relational model. Intersection variables, as defined by Maier et al. [56], are required for sound and complete reasoning of d-separation in relational domains whenever there exists two paths, $P_1 = [A, \ldots, B], P_2 = [A', \ldots, B']$ that are not subsets of each other and whose beginning and ending entity are the same, i.e., $A = A'$ and $B = B'$. We consider the case of the single entity, single relationship graph. Denote E to be the entity and R to be the relationship. Without loss of generality, we consider paths that begin at the entity. All possible path specifications then must be of the form $[A(BA)^*]$, where $*$ is the Kleene star. It follows directly that any two path specifications are either identical, or the shorter path is a sub-path of the other. This implies that for single entity, single relationship networks, intersection variables do not exist. □

Within our running example there is a single perspective (person) and relational variables are defined with respect to the relative distance to an individual (e.g. friends and friends of friends). One plausible abstract ground graph for this example is shown in Figure 5.3, in which the disposition and privacy settings of a person and her friends affect her time spent on site. Note that in Figure 5.3 there are two different types of variables present. Propositional variables are those preceded by U^0, and are measured on a single instance. Relational variables are named as U^i, for $i > 0$. These variables representing the values of a person's friends and the friends of her friends, respectively. Given the AGG, conditional independence facts can be computed directly using the same rules of d-separation used for Bayesian networks. For instance, from Figure 5.3,

we can see that $U^2.Prv \perp\!\!\!\perp U^0.Prv | U^1.Prv$, because $U^1.Prv$ blocks all d-connecting pathways between the privacy settings of *friends of friends* and a user's time spent on site. These d-separation properties are essential to identifying a sufficient set of conditioning variables for a given causal query, discussed in more detail in the following section.

5.2.2 Relational Backdoor Criterion

With a suitable representation in hand, we now turn to the core aim of this work: identifying interventional distributions. The approach taken here is to use an extension of the *back-door criterion* [71] to relational domains:

Definition 8. *(Relational Back-Door Criterion) A set of variables* $\mathbf{C}$ *satisfies the* relational back-door criterion *with respect to variable sets* (X_1, X_2) *in an AGG* G *if:*

1. *No node in* $\mathbf{C}$ *is a descendant of any node in* X_1 *in the AGG (equivalently, no node in* $\mathbf{C}$ *is a post-treatment variable); and*

2. $\mathbf{C}$ *blocks every back-door path between* X_1 *and* X_2 *in the AGG*

Note that here a *back-door path* refers to a path with an arrow into a member of X_1. Definition 8 is a direct extension to relational data of the back-door criterion presented by Pearl [71]. In the case of a single entity with no relationships, the definition reduces to the propositional case.

When such a set $\mathbf{C}$ can be identified, an estimate of the interventional distribution can be obtained through a simple application of the adjustment formula:

$$P(X_2 | do(X_1 = x)) = \int_c P(Y | X_1 = x, \mathbf{C} = c) \mathrm{d}P(\mathbf{C} = c) \tag{5.3}$$

Then, average causal effects can be computed as follows:

$$ACE = E[X_2|do(X_1 = x)] - E[X_2|do(X_1 = x')] \tag{5.4}$$

$$= \int_c yP(X_2|X_1 = x, \mathbf{C} = c)\mathrm{d}P(\mathbf{C} = c)$$

$$- \int_c yP(X_2|X_1 = x', \mathbf{C} = c)\mathrm{d}P(\mathbf{C} = c), \tag{5.5}$$

where P represent either a probability density or probability mass function. Semantically, because relational variables take on values that may be multisets, there is a notion of *exhangeability* encoded in this estimation framework. Consider once again the example of Figure 5.1. In this case, Sue has three neighbors, John, Bob and Carl. Let $U^1_{\text{Sue}}.\mathbf{Prv}$ represent the multiset of time on site values of these neighbors (see Table 5.1). As presented, $U^1_{\text{Sue}}.\mathbf{Prv}$ takes on the value {On, On, Off}. Intervention on Carl or Bob's privacy setting would yield the interventional regime $do(\, U^1_{\text{Sue}}.\mathbf{Prv} = \{\text{On, Off, Off}\})$. As such, our interventional language is invariant with respect to the *identities* of the instances under intervention, and focuses strictly on the variables measurable on those entities.

5.2.2.1 Connection to Network Experimentation

There is a close relationship between Relational Covariate Adjustment and the adjustments performed for peer-effects in the network experimentation literature (c.f., [5, 89, 37]). Given this connection, we discuss this relationship for readers familiar with network experimentation. Current work in network experimentation are described within the potential-outcomes framework and assume *strong ignorability*, i.e., that (1) the outcome is rendered independent of treatment given treatment status and (2) that all instances have a treatment probability, $p \in (0, 1)$. Within non-network experiments condition (1) is trivially satisfied via randomization. However, even in the simple network setting there is dependence between other treatments and an individual's outcome. Further, by virtue of network randomization designs (i.e., [89, 37]),

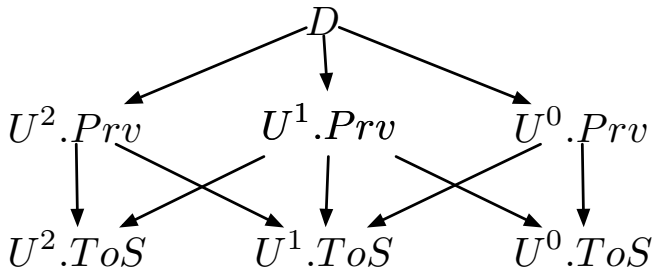


Figure 5.4: An abstract ground graph representing the dependence structure under network experiment. This structure is similar to 5.3, except that disposition no longer influences privacy settings, and is excluded from the diagram. A variable D representing the experimental design may induce marginal dependence between treatments. It is possible that the outcome of peers ($U^1.ToS$) affects $U^0.ToS$, but including $U^1.Prv$ in a conditioning set is sufficient to satisfy the back-door criterion for treatment $U^0.Prv$.

dependence is induced between the treatment status of instances. This dependence is depicted in Figure 5.4. The graphical view shows that simple use of Relational Covariate Adjustment can be applied to adjust for network bias, with $U^1.Prv$ constituting the adjustment set. Thus, the estimator of Gui et al. [37] can be seen as a special case of Relational Covariate Adjustment, with an assumed dependence structure of Figure 5.4 and adjustment performed with a linear model. However, in contrast to current network experimentation estimation methods, Relational Covariate Adjustment can be applied easily to observational data with multi-valued and continuous treatments and an arbitrary number of confounders without modification.

5.3 Empirical Estimation

We now discuss how to practically estimate the effects of interventions in relational domains. In contrast to the non-relational setting, computing the adjustment formula in equation 5.5 is not straightforward because the hypothetical values of X_1 could be multisets. We present a strategy for conditioning on multisets that does not make strong assumptions about functional form. Algorithm 1 presents the procedure. Step

1 identifies the adjustment set by using relational d-seperation to find the necessary set of variables $\mathbf{C}$ to block all back-door paths between T and Y.

The causal effect is then estimated as

$$\mathbb{E}[Y|do(T=t)] = \int_C yP(Y=y|T=t,\mathbf{C}=c)\mathrm{d}P(\mathbf{C}=c) \tag{5.6}$$

$$\approx \frac{1}{N}\sum_{i=1}^{N} y_i P(Y=y_i|T=t,\mathbf{C}=c_i) \tag{5.7}$$

$$= \frac{1}{N}\sum_{i=1}^{N} E[Y|T=t,\mathbf{C}=c_i], \tag{5.8}$$

where equation 5.7 is a Monte-Carlo approximation to the integral. $E[Y|T=t,\mathbf{C}=c_i]$ can be estimated from a regression of y on features T and $\mathbf{C}$.

Algorithm 1: `RelationalAdjustment`

Input: Relational model $\mathcal{M}$, outcome Y, treatment(s) X
Output: $h(x) = \sum_{i=1}^{N} E[Y|do(X=x)]$

1 Use relational d-separation to identify adjustment set $\mathbf{C}$ for causal effect of X on Y
2 Estimate $E[Y|X,\mathbf{C}]$ via regression or classification
3 $h(x) = \sum_{i=1}^{N} E[Y|X=x,\mathbf{C}=\mathbf{c}_i]$
4 **return** $h(x)$

5.3.1 Calculating Network Effects

Algorithm 1 can be applied to estimate a variety of causal effects derived from the definition presented in equation 5.2. In what follows, $U^0.T$ refers to a subject's treatment and $U^1.T$ refers to the treatments of immediate neighbors.

Marginal Individual Effect

$$h = \texttt{RelationalAdjustment}(\mathcal{M}, U^0.Y, U^0.T)$$

$$h(1) - h(0) = E[U^0.Y|do(U^0.T=1)]$$

$$- E[U^0.Y|do(U^0.T=0)] \tag{5.9}$$

This effect represents the expected change in an arbitrary subject's outcome, $U^0.Y$, when considering two alternate settings of that subject's treatment, $U^0.T$ (1 and 0). The function h represents the expected outcome when applying a hypothetical intervention to $U^0.T$, conditioning on $\mathbf{C}$. Additionally, the treatment assignment of peers, $U^1.T$, can influence both $U^0.T$ and $U^0.Y$, which requires including peer treatment values in the set of confounders, i.e, $U^1.T \in \mathbf{C}$.

Marginal Peer Effect

$$\begin{aligned} h &= \texttt{RelationalAdjustment}(\mathcal{M}, Y, U^1.T) \\ h(\theta) - h(\theta') &= E[U^0.Y | do(U^1.T = \theta)] \\ &\quad - E[U^0.Y | do(U^0.T = \theta')] \end{aligned} \tag{5.10}$$

The above case concerns the causal effect of settings of the treatment assignments of peers, $U^1.T$. θ and θ' are multisets consisting of the treatment values of neighbors. For instance, in the context of Figure 5.1, $\theta_{\text{Sue}} = \{\text{On}, \text{On}, \text{Off}\}$. We could consider altering the treatment of Sue's neighborhood to $\theta'_{\text{Sue}} = \{\text{Off}, \text{Off}, \text{On}\}$. The effect of the intervention is given by $h(\theta_{\text{Sue}}) - h(\theta'_{\text{Sue}})$. This formulation facilitates the estimation of arbitrary treatment settings of a node's neighborhood.

Total Effect

$$\begin{aligned} h &= \texttt{RelationalAdjustment}\big(\mathcal{M}, Y, (U^0.T, U^1.T)\big) \\ h(1, \vec{1}) - h(0, \vec{0}) &= E[U^0.Y | do(U^0.T = 1, U^1.T = \vec{1})] \\ &\quad - E[U^0.Y | do(U^0.T = 0, U^1.T = \vec{0})] \end{aligned} \tag{5.11}$$

This effect represents an intervention on both $U^0.T$ and $U^1.T$. The adjustment procedure is valid for simultaneous interventions on these variables because the back-door

criterion (Definition 8) applies to *sets* of variables. Now, h is a function of two variables, the hypothetical intervention to $U^0.T$ and the hypothetical intervention to $U^1.T$. The first argument to h is, in the case of binary treatments, 0 or 1. The second argument to h is a multiset. This class of effects is most applicable to estimation of applying an intervention to all individuals on a network, e.g., a site-wide feature roll-out.

5.3.2 Summarizing Relational Features

When any term in the adjustment equation is a relational variable, $E[Y|T, \mathbf{C}]$ cannot be directly estimated using regression or classification estimators designed for independent and identically distributed data because relational variables' instances consist of multisets rather than single observations. A common approach to address this is to create aggregations to succinctly represent the sets with a small number of real-valued features. There is a long history in statistical relational learning of using user-specified aggregation functions to model the distribution of a relational variable [43, 72]. While these approaches have yielded impressive results for the task of prediction, causal inference requires stronger guarantees about what is being captured by the aggregation functions. The aggregation function should be a sufficient statistic of the underlying distribution of the variable, rendering model parameters independent of the data. For instance, specifying the `mean` aggregation would be sufficient if the values of a relational variable are Poisson distributed, and in the case of a normal distribution, the `variance` aggregation must also be present. When sufficient statistics are employed, then we can be confident that *all relevant aspects* of the distribution of a set have been accounted for when marginalizing to compute the interventional distribution. When assumptions can be made about the marginal distribution of relational variables, a set of features can be constructed for regression by taking the sufficient statistics for each instance of a relational variable. Once this set is constructed, any consistent regression or classification model can be used to estimate $E[Y|T, \mathbf{C}]$. In the absence of known sufficient statistics, estimates of a number of the moments of a distribution can be used as an approximate solution. We assume that the sufficient statistics S_x of the true distribution can be described as a function of its k-th order moments:

$$S_x = f(M_1(X), \ldots, M_k(X))$$

where $M_k(X) = \frac{\sum_i^N X_i^k}{N} \approx \int x^k \hat{p}(x)dx$ is the empirical estimate of the k-th moment of X. This implies the following procedure: (1) for each relational variable generate a set of k aggregates of the $1, \ldots, k$ moments of the set, (2) use this new data set as the features to a non-linear regression or classification model to estimate $E[Y|T, \mathbf{C}]$.

5.4 Experiments

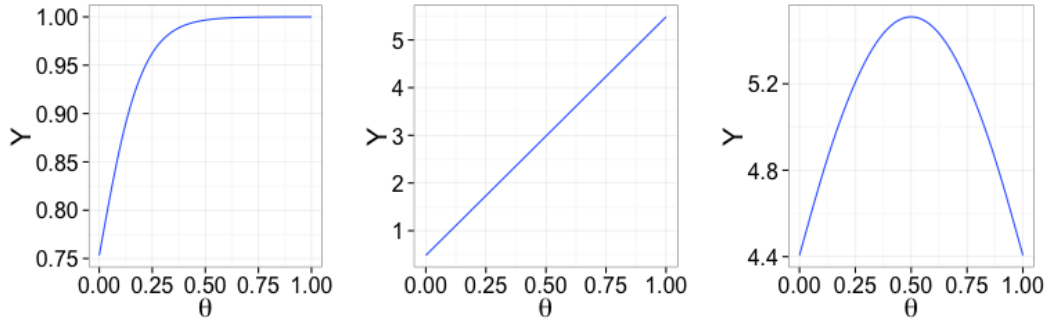
In this section we evaluate whether, and under which circumstances, Relational Covariate Adjustment can serve as a feasible alternative to experimentation for causal inference. To that end, we constructed an evaluation suite to compare RCA to state-of-the-art techniques for estimating causal effects from experiments. We provided experimental techniques with *experimental* data, and we provided RCAdata with more challenging data sets in which relational confounding variables are present. We examined a variety of real and synthetic networks, using simulated data with multiple functional relationships between treatment and outcome.

5.4.1 Synthetic Data Generation

Data generation process was performed as follows:

1. Generate a random network
2. Sample treatment using one of two regimes:
   (a) `Exp`: Sample treatment from an experimental context, in which treatment is assigned using a graph clustering technique
   (b) `Obs`: Sample treatment as a function of confounding variables and possibly treatments of neighbors in the network
3. Sample outcome according to the treatment assigned in step (2). In the `Obs` regime, outcome is a function of confounding variables and treatment. In the `Exp` regime, outcome is a function of treatment assignments.

In both the `Obs` regime and the `Exp` regime, the task is identical: estimate the relationship between treatments (individual and those of peers) and outcomes. We compared the performance of models learned from the observational data to estimates obtained by experimentation[2].



(a) Sigmoid, $\beta_P = 10$ (b) Linear, $\beta_P = 5$ (c) RBF, $\beta_P = 5$

Figure 5.5: Examples of outcome models considered in this work, shown here as a function of the proportion of treated friends.

5.4.1.1 Synthetic networks

We considered two network structures in our synthetic experiments: small-world networks and preferential attachment networks. For small-world networks, each node has degree (in+out) of 10 in the initial lattice. We varied the rewiring probability in $\{0, 0.01, 0.1, 0.15\}$. A rewiring probability of 0 results in a regular lattice, and a rewiring probability of 1 results in a random (Erdős-Rényi) network. For preferential attachment networks, we varied the power of the attachment in $\{0.1, 0.5, 1\}$. In all cases, the synthetic networks we consider have 1024 nodes.

Each network has a simple relational model consisting of a single entity (U) and relationship (adjacency). Each instance of U (i.e., a node in the network) has four

[2]Code used to reproduce these experiments is available at https://github.com/darbour/RelationalAdjustment

attributes, C_1, C_2, T, and Y. We are interested in estimating the effects of $U^0.T$ (intrinsic treatment) and $U^1.T$ (treatment of peers) on $U^0.Y$ (intrinsic outcome).

5.4.1.2 Treatment Models

In the `Obs` regime, propensity for treatment can be caused by intrinsic covariates, covariates of peers, and treatments of peers. To simulate data from that regime, we first constructed a confounding term L_i which is a linear combination of:

- $U^0.C_1$
- $U^0.C_2$
- $\texttt{mean}(U^1.C_1)$
- $\texttt{mean}(U^1.C_2)$
- $\texttt{var}(U^1.C_1)$
- $\texttt{var}(U^1.C_2)$
- $\texttt{mean}(U^1.C_1) * \texttt{var}(U^1.C_1)$
- $\texttt{mean}(U^1.C_2) * \texttt{var}(U^1.C_2)$

Then, treatment is sampled as a binomial random variable with success probability that is a logistic function of L_i. To simulate influence between the treatments of subject i and its neighbors, we use a Gibbs sampling technique inspired by Manski [59]. After initially assigning treatment, we resample treatment with an additional parameter $\theta_{nbr_i,s-1}$, the proportion of i's neighbors that are treated at the previous iteration. This process is repeated until $s = 3$.

$$T_{i,0} \sim Binom\left(logistic\left(\beta_L L_i + \epsilon\right)\right) \tag{5.12}$$

$$T_{i,s} \sim Binom\left(logistic\left(\beta_L L_i + \beta_T \theta_{nbr_i,s-1} + \epsilon\right)\right) \tag{5.13}$$

Here, $\epsilon \sim \mathcal{N}(0, 1)$. We vary the strength of the confounding coefficient, β_L, from 0 to 3. We vary the strength of dependence on peers' treatments, β_T, from 0 to 10. When $\beta_T = 2$, we find that the distribution of peer treatment proportions, θ_{nbr_i}, is roughly uniform. When $\beta_T = 10$, this distribution is bi-modal with peaks at 0 and 1.

In the `Exp` regime, treatment was assigned randomly (with probability 0.5) at the level of *graph clusters* rather than individuals using a technique outlined by Ugander *et al.* [89]. This clustering technique assigns treatment in such a way that nodes

are more likely to have completely treated or completely untreated neighborhoods. In other words, graph cluster randomization leads to bi-modal distributions of θ_{nbr_i} with peaks at 0 and 1. This randomization technique is employed by experimental estimators to estimate the total effect of equation 5.11.

5.4.1.3 Outcome Models

We explored the use of three distinct outcome forms. In the first case, outcome is a linear function of individual treatment, the proportion of treated peers, θ_{adj_i}, confounding variables L_i, with noise that is distributed as a standard normal. The general form of this function is shown below in equation 5.14. The relationship between the θ_{adj_i} and outcome is shown in Figure 5.5b for a specific parameter setting.

$$Y_i \sim \beta_I T_i + \beta_P \theta_{nbr_i} + \beta_L L_i + \epsilon \tag{5.14}$$

We also considered non-linear functions of treatment and covariates. The first of these is shown in equation 5.15, and is a sigmoid function of T_i, θ_{nbr_i}, and L_i. Figure 5.5a shows one instance of this function class. This function is bounded in the range $(0, 1)$. As β_I and β_P grow, the outcome approaches 1 more sharply.

$$Y_i \sim \left(1 + \exp\left(-\left(2\beta_I T_i + 2\beta_P \theta_{nbr_i} + \beta_L L_i + \epsilon\right)\right)\right)^{-1} \tag{5.15}$$

The final outcome model we use is linear in T_i and L_i, but depends on θ_{nbr_i} through a radial basis function about 0.5. An instance of this function can be seen in Figure 5.5c. In this case, the outcome peaks when $\theta_{nbr_i} = 0.5$.

$$Y_i \sim \beta_I T_i + \exp\left(-\left(\beta_P \theta_{nbr_i} - 0.5\right)^2\right) + \beta_L L_i + \epsilon \tag{5.16}$$

In what follows, we refer to the functions outlined in equations 5.14, 5.15, and 5.16 as *linear*, *sigmoid*, and *RBF*, respectively.

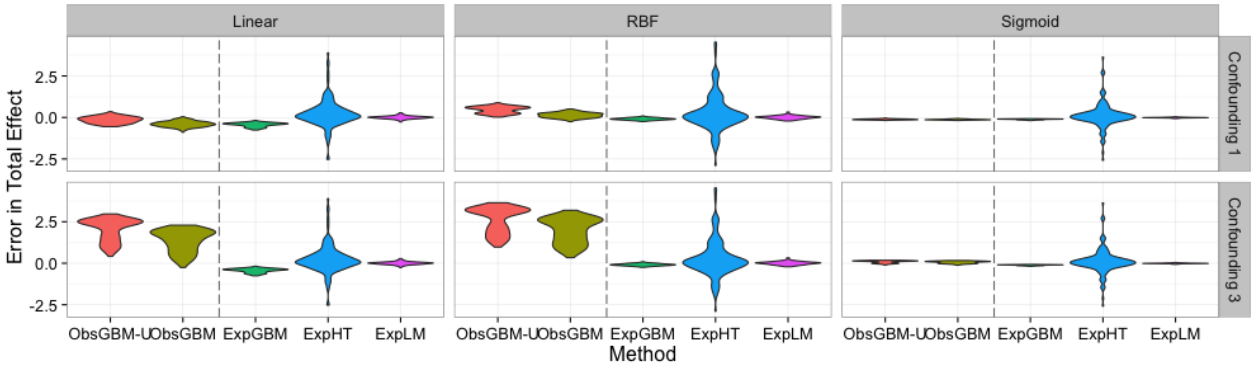


Figure 5.6: Accuracy of experimental and observational effect estimates across various outcome models as confounding strength is varied.

5.4.2 Estimators

In practice, any consistent conditional estimator of $E[Y|T, \theta_{nbr_i}, \mathbf{C}]$ will satisfy the requirements of the relational adjustment technique, provided $\mathbf{C}$ satisfies the relational back-door criterion. For our experiments, we used gradient boosted trees (GBMs) to model this expectation, where $\mathbf{C}$ consists of the means and variances of $U^1.C_1$ and $U^1.C_2$.

Gradient boosted trees [29] are a nonparametric ensemble where each base learner is a low-depth decision tree. At each iteration training samples are reweighted according to their predictive error on the previous iteration. The boosting procedure has been shown to be consistent [29], and provides near state-of-the-art results on a variety of tasks.

We employed two experimental effect estimators within the `Exp` regime, Horvitz-Thompson estimation [89] and a linear additive model [37]. The Horvitz-Thompson estimator can be written as a weighted sum of outcomes of nodes which fall into two distinct exposure categories. We defined a node i as "exposed" if $T_i = 1$ and $\theta_{nbr_i} > 0.75$. We defined a node as "non-exposed" if $T_i = 0$ and $\theta_{nbr_i} < 0.25$. Nodes which do not fall into one of these categories are not used in the estimation process.

$$\frac{1}{N}\sum_{i=1}^{N}\frac{Y_i\mathbb{I}(\theta_{nbr_i}>0.75,T_i=1)}{P(\theta_{nbr_i}>0.75,T_i=1)}-\frac{Y_i\mathbb{I}(\theta_{nbr_i}<0.25,T_i=0)}{P(\theta_{nbr_i}<0.25,T_i=0)}.$$

The probabilities in the denominator are estimated using the dynamic programming algorithm introduced by Ugander et al. [89]. This method is useful primarily when the effect of interest is the total effect and the distribution of θ_{nbr_i} is bimodal with peaks at 0 and 1. We refer to this estimation strategy as `ExpHT`.

The linear additive model introduced by Gui et al. [37] fits the conditional expectation $E[Y|T,\theta_{nbr_i}]$, which is appropriate when treatment is assigned experimentally and outcome is a linear model. We refer to this model as `ExpLM`.

It is important to note that, for both `ExpHT` and `ExpLM`, the results reported are with respect to a performed experiment. This is contrast to the setting of Relational Covariate Adjustment, which is given access *only* to observational data, without the benefit of randomization.

5.4.3 Findings

For each combination of parameter settings, spanning rewiring probability, β_I (individual effect), β_P (peer effect), β_L (confounding strength), and β_T (treatment autodependence), we performed 25 trials to assess variance. This resulted in $2269 \times 25 = 56{,}725$ data sets, some belonging to the experimental regime (`Exp`) and some belonging to the observational regime (`Obs`). Within the experimental regime, we estimated total effect using `ExpHT` and `ExpLM`, the current state-of-the-art techniques for effect estimation on networks. Within the observational regime, we used the Relational Covariate Adjustment procedure with gradient boosted trees. This is referred to as `ObsGBM` in what follows. We also include results for an unadjusted GBM which does not include any relational covariates. We refer to this model as `ObsGBM-U`.

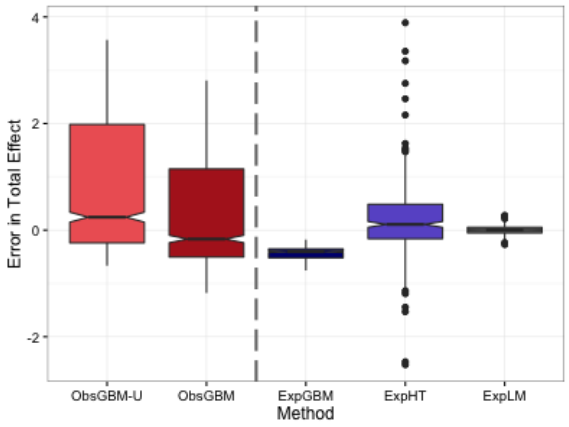


Figure 5.7: Comparison of estimates obtained from retrospective, confounded, observational data (left) and those from experimentation (right). An *overestimated* effect results in positive error, and an *underestimated* effect results in negative error. These methods almost always overestimate the true global effect.

Accuracy of the total causal effect estimates for the linear outcome model are shown in Figure 5.7. Each box in this figure represents the distribution of performance values across all network settings and model parameterizations. These results indicate that the `ExpLM` model performs best in this context, with `ExpHT` yielding slightly more bias and significantly more variance. However, in the observational case, which is a more complex estimation task, the Relational Covariate Adjustment implementation, `ObsGBM` is competitive with the linear model in terms of bias, and yields only slightly more variance than the HT estimator.

We also examined the performance of these models across a variety of outcome functions. The error in total causal effect estimates are shown in Figure 5.6. This demonstrates two dimensions of variability in our simulations. First, different functional forms lead to more or less challenging estimation tasks. Most significantly, as the strength of confounding (β_L) is increased from 1 to 3, the observational regime becomes more challenging. This matches intuition—in the extreme, where $\beta_L = 0$, any confounding between treatment and outcome disappears.

	Exp. LM	Obs. GLM
Linear	**0.0869 (0.0752)**	0.6527 (0.5936)
RBF	**0.102 (0.0877)**	0.2342 (0.1687)
Sigmoid	**0.0178 (0.0158)**	0.0269 (0.0157)

Table 5.2: Root mean squared error for marginal individual effects. One standard error is shown in parentheses.

	Exp. LM	Obs. GBM
Linear	**0.0535 (0.021)**	0.6241 (0.485)
RBF	0.4476 (0.248)	**0.403 (0.264)**
Sigmoid	0.0661 (0.015)	**0.0391 (0.027)**

Table 5.3: Root mean squared error for marginal peer effects. One standard error is shown in parentheses.

While the `ExpHT` model is specifically designed to estimate only total effects, `ExpLM` and `ObsGBM` can also compute marginal individual effects and marginal peer effects. We computed the root mean squared error between estimated individual effects and actual individual effects—this error is shown in Table 5.2. `ObsGBM` is competitive with `ExpLM` primarily for non-linear functional forms.

Finally, we examined the ability of the `ExpLM` and `ObsGBM` to model marginal peer effects. Unlike the total effect and the marginal individual effect, there is a *spectrum* of peer effects induced by varying θ_{nbr_i}. The possible functional relationships between θ_{nbr_i} and Y_i are shown in Figure 5.5. Figure 5.8 provides another concrete example of such a function along with the models estimated by `ExpLM` and `ObsGBM`. Table 5.3 shows the root mean squared error for peer effects $\theta_{nbr_i} \in \{0, 0.1, \ldots, 0.9, 1\}$. In the linear case, the `ExpLM` model has an advantage over `ObsGBM`. However, the importance of modeling non-linearity becomes clear in the RBF and sigmoid examples, for which the observational estimator is superior to the experimental estimator.

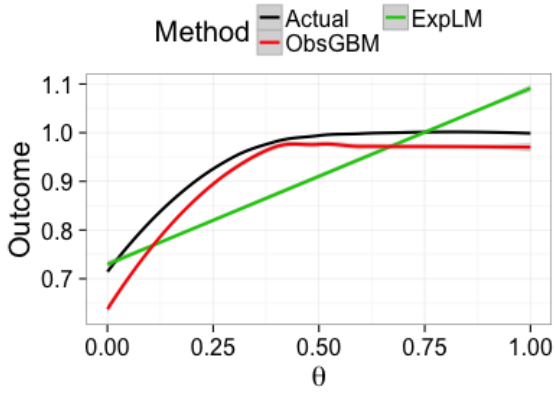


Figure 5.8: An example of the sigmoid outcome model. In this case, a model of the marginal peer effect is estimated from observational data using a boosted model and from experimental data with a linear model, with $\beta_I = \beta_P = 5$ and $\beta_L = 1$.

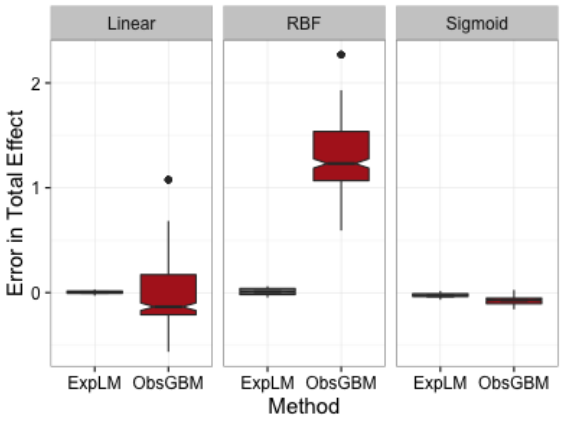


Figure 5.9: Estimated Total Effects in the Enron Data

5.4.4 Real Networks

To demonstrate the applicability of RCA to large networks for which the edge generation process is unknown, we also compared the performance of `ObsGBM` and `ExpLM` on the Enron graph [49]. The nodes of the Enron network are individuals, with an edge between them if either of them have sent an email to the other. The network is contains 36,692 nodes and 183,831 edges in total, with a clustering coefficient of approximately 0.5 and diameter of 11. In the absence of ground truth measures, we generated synthetic random variables following the procedure of Section 5.4.1, and

	ExpLM	ObsGBM
Linear	**0.0399 (0.0279)**	1.4038 (0.3772)
RBF	**0.6242 (0.2571)**	1.2883 (0.3778)
Sigmoid	0.3439 (0.2113)	**0.0561 (0.0101)**

Table 5.4: Root mean squared error for marginal individual effects in Enron data.

	`ExpLM`	`ObsGBM`
Linear	**0.0213 (0.007)**	0.4855 (0.195)
RBF	0.5255 (0.148)	**0.2278 (0.112)**
Sigmoid	0.2703 (0.14)	**0.0266 (0.025)**

Table 5.5: Root mean squared error for marginal peer effects in Enron data.

use the real graph topology to test scalability and efficacy. The form of the generative functions were identical to those used in the observational setting. We then measured the estimates of the total effect, marginal peer effects, marginal individual effects for each method.

Figure 5.9 shows the quality of estimated total effects across each outcome model. While the results from a synthesized network experiment are superior to the estimates from `ObsGBM`, the results are on comparable scales. Table 5.4 and Table 5.5 show the error in estimated individual and peer effects, respectively. Again, the results from `ObsGBM` are similar to `ExpLM`. `ExpLM` performs exceptionally well at experimental data with a linear outcome. However, `ObsGBM` has a clear advantage in estimating marginal peer effects under the RBF and Sigmoid models. We conjecture that the scale-free nature of the Enron network leads to particularly favorable circumstances for experimental approaches such as `ExpLM`. Scale-free networks have many nodes with only one or two neighbors, thus the probability that an entire neighborhood will be completely treated or completely untreated is relatively high.

5.5 Related Work

Ugander et al. [89] and Gui et al. [37] present methods which aim to measure the effect of placing the entire network under treatment versus control. In both cases, this is achieved by partitioning the graph into clusters, treating each cluster randomly and estimating the the average causal effect after adjustment for peer confounding. Toulis and Kao [88] consider experimental design and estimation to measure average peer effect as a quantity of interest. In both cases, the methods discussed within this chapter can be seen as complimentary work, providing interpretation within the causal graphical models framework. This interpretation aids the identification of threats to validity and provides a unified framework for estimation of a variety of causal effects. Importantly, the causal graphical model view of this work also admits inference in the non-experimental setting.

There have been numerous applications in recent year that seek to measure causal effects in real relational domains. Bakshy et al. [7] performed large scale experiments to understand the effect of social cues on consumers' receptiveness to advertisements. Aral and Walker [2] used experimentation to understand the process of social diffusion, or "virality" in large-scale social systems.

Within the observational setting, researchers have applied quasi-experimental designs (QEDs) to perform causal inference in relational data. QEDs exploit fortuitous circumstances in data that allow for the approximation of an experimental design post-hoc. For example, Oktay et al. [69] apply QEDs to Stack Overflow, an online question and answer site for programming, to understand the dynamics of users' behavior on the site. Krishnan and Sitaraman [45] consider a quasi-experimental design to determine the relationship between network quality and user engagement with online content. Kearns et al. [42] study patterns of network formation by performing an experiment where subjects were anonymously paired, and subsequently were asked to interactively complete a graph-coloring video game.

Manski [59] considers the problem of identifiability in the potential-outcomes framework in the presence of peer influence. Ogburn and VanderWheele [67] enumerate configurations of causal graphs that result in bias from social effects on single entity, single relationship networks. Maier et al. [58] considers the more general case of d-separation for multi-relational domains. Maier et al. [56] apply the rules implied by Maier et al. [58] to learn the causal structure of relational domains, but explicitly do not consider inference of individual causal effects.

5.6 Conclusion and Future Work

We have described and evaluated Relational Covariate Adjustment, an extension of nonparametric adjustment to relational data. Through the use of nonparametric regression estimators, RCA allows for estimation of a wide range of functional dependencies without modification. We showed the efficacy of this approach to causal inference with a set of experiments that examine Relational Covariate Adjustment and other experimental adjustment methods over a range of graph topologies.

This work represents one step toward a much larger goal of general causal inference in relational domains. Toward that end, we plan to extend RCA to the case of multiple entities and relationships and to extend the calculus of interventions by developing techniques that can estimate the effects of interventions that add or remove nodes or edges from the network.

CHAPTER 6

SUMMARY AND FUTURE DIRECTIONS

Systems that contain relational effects are ubiquitous in modern society from social networks, to transportation and the internet. The ability to reliably reason over these systems is critical for enabling better planning and policy for modern decision makers. Despite this, relatively little work exists which examine the statistical underpinnings of dependence and causation in relational domain. This thesis is one step towards closing this gap.

We have introduced methods for principled detection of autodependence in networks, as well as bivariate and conditional dependence. Previous work which addressed these problems relies on parametric assumptions [94, 75], discrete data [41, 75], or restricts the relational structure to automorphisms [98] which is known to not hold in the case of random graphs. With the exception of Zhang, et al. [98], these approaches have used measures of dependence as a heuristic in service of empirical evaluation, rather than as a theoretical focus. However, there are a number of tasks which require well developed statistical theory for dependence testing. For example, causal structure learning [83], relies on conditional independence tests in order to constrain the space of possible causal structures. Because of a lack of developed theory, existing work [55, 61] has relied on an conditional independence oracle. By providing this grounding we hope to enable further theoretical and empirical work in causal learning.

In order to reliably approximate the null distribution in domains that exhibit relational autodependence, we have introduced the first bootstrap for relational data.

This is accomplished by extending the dependent wild bootstrap for degenerate U and V statistics of Leuchte and Neumann [50] to relational domains. We believe that this same methodology can be extended for estimating confidence intervals and p-values for causal effects in networsk.

We then showed that relational dependence is inherently asymmetric, and that the asymmetry can be leveraged to provide a simple test of causal direction from observational relational data.

Finally, we introduced a general procedure for estimating causal effects in relational data by presenting an extension of Pearl's backdoor criterion to relational domains. There are a number of promising future directions along this line of research. These include reasoning about the effect of intervening on the network structure itself, i.e. adding or removing an edge, adapting to dynamic network structures, as well as considering more complicated intervention strategies.

APPENDIX

RELATIONAL DEPENDENCE TESTING IN THE ABSENCE OF AUTO DEPENDENCE

The chapter on relational dependence makes use of a kernelized measure of relational dependence. In this chapter, we will show that this measure is consistent, assuming ϑ-mixing, and that a simple test of permutation of values with respect to their node indices may be used to assess significance. While not a central contribution of this thesis, we believe these accompanying results may be of independent interest.

We will make use of the following definitions in order to provide guarantees regarding the convergence of the kernel mean.

Theorem 3. *(Generalized McDiarmid's Inequality)[52] Let $f : \mathcal{X} \mapsto \mathbb{R}$ be a measurable function with a constant c such that $\forall \mathbf{x}, \mathbf{x}' \in \mathcal{X}$ that differ only at a single coordinate, $|f(\mathbf{x}) - f(\mathbf{x}')| \leq \frac{c}{n}$. Then for all $\epsilon > 0$:*

$$P\{f(\mathcal{X}) - \mathbb{E}[f(\mathcal{X})] \geq \epsilon\} \leq \exp\left(\frac{-2n\epsilon^2}{c^2\|\Theta_n^{\pi}\|_{\infty}^2}\right)$$

We refer the reader to London et al. [52] for the proof.

A.0.1 Kernel Embeddings of Relational Data and Dependence Testing

Recall from the Background chapter that the *mean map* is a function $\mu : \mathscr{P} \to \mathcal{H}$ that defines a kernel embedding of a distribution into $\mathcal{H}$:

$$\mu_P = \mu(P) = \int_{\mathcal{X}} k(x, \cdot) dP(x)$$

In this work, the purpose of kernel mean is twofold. For propositional variables, it is used to represent the underlying distribution and, as we shall see, can be used directly in a test for dependence. For relational variables, the mean embedding serves as an aggregation function for observations. The advantage of using the kernel mean embedding is that, under the assumption that the underlying distribution belongs to the exponential family, the underlying distributions are represented completely.

To reason over the distance between distributions, we define a second kernel, K, over the kernel means. Christmann and Steinwart [13] showed that if the kernel inducing μ (k) is characteristic and K is the Gaussian kernel, then K is universal and thus, characteristic. This kernel is defined as:

$$K(\mu_x, \mu'_x) = e^{\frac{\|\mu_x - \mu'_x\|^2_{\mathcal{H}}}{2\theta}} \tag{A.1}$$

where $\sqrt{\theta}$ is the bandwidth of the kernel.

Lemma 3. *Under the assumptions that each kernel mean, $\hat{\mu}$, is close to their population values, and the degree of the network is bounded by some constant, d, and the random variable that gives rise to μ is independently distributed, the estimate $\hat{\mathcal{M}} = \frac{1}{N}\sum_i^N \hat{\mu}_i$ is a consistent estimator of the true embedded mean, $\mathcal{M}$.*

Lemma 4. *Under the assumptions that each kernel mean, $\hat{\mu}$, is close to their population values, and the degree of the network is bounded by some constant, d, and the random variable that gives rise to μ is independently distributed, the estimate $\hat{\mathcal{M}} = \frac{1}{N}\sum_i^N \hat{\mu}_i$ is a consistent estimator of the true embedded mean, $\mathcal{M}$.*

Let the mean of second-level mean embedding of $\mu \in \mathcal{M}(\Omega)$ into the RKHS provided by the kernel, $k\cdot$). We will assume $k(\cdot)$ is bounded by $k(\mu, \mu) \leq B_k(\forall \mu \in \Omega)$. We are given N samples, $mu_1, \ldots, \mu_N$, from a weakly dependent process, whose covariance matrix is given by Θ. Further, define the empirical mean embedding as $\mu_{\hat{\mu}} = \frac{1}{N}\sum_{n=1}^N k(\cdot, \mu_n)$. Then $\mathbb{P}(\|\mathcal{M}_{\hat{\mu}} - \mathcal{M}_{\mu}\|_H \geq \epsilon) \leq e^{-\frac{\epsilon^2 N}{2B_k\|\Theta\|_\infty}}$

Proof. Let $\phi(\mu) = k(\cdot, \mu)$, and by extension $k(\mu, \mu) = \|\phi(\mu)\|_H^2$. Let $g(S) = \|\mathcal{M}_{\hat{\mu}} - \mathcal{M}_\mu\|_H = \|\frac{1}{N}\sum_{n=1}^N \phi(\mu_n) - \mathcal{M}_x)\|_H$, with S being the set of samples, i.e., $S = \{\mu_1 \dots mu_N\}$. Also let $S' = \{\mu_1, \dots, \mu_{j-1}, \mu'_j, x_{j+1}, \dots, \mu_N\}$. We have

$$\begin{aligned}
|g(S) - g(S')| &= \Big| \big\| \frac{1}{N}\sum_{i=1}^N \phi(mu_n) - \mathcal{M}_\mu \big\|_H - \big\| \frac{1}{N}\sum_{i=1}^N \phi(\mu_n) + \frac{1}{N}\phi(\mu'_j) - \mathcal{M}_\mu \big\|_H \Big| \\
&\leq \frac{1}{N}\|\phi(\mu_j) - \phi(\mu'_j)\|_H \leq \frac{1}{N}(\|\phi(\mu_j)\|_H) + \|\phi(\mu'_j)\|_H) \\
&\leq \frac{1}{N}\Big[\sqrt{k(\mu_j, \mu_j)} + \sqrt{k(\mu'_j, \mu'_j)}\ \Big] \leq \frac{2\sqrt{B_k}}{N}
\end{aligned}$$

We can now use the generalized version of McDiarmid's inequality, yielding

$$\begin{aligned}
\mathbb{P}(g(S) - \mathbb{E}[g(S)] \geq \epsilon) &\leq \exp\left(-\frac{2\epsilon^2}{\sum_{n=1}^N \left(\frac{2\sqrt{B_k}}{N}\right)\|\Theta\|_\infty}\right) \\
&= \exp\left(-\frac{2\epsilon^2}{N\frac{4B_k}{N^2}\|\Theta\|_\infty}\right) \\
&= \exp\left(-\frac{\epsilon^2 N}{2B_k\|\Theta\|_\infty}\right)
\end{aligned}$$

Where $\|\Theta\|_\infty$ is the L_∞ norm of the covariance matrix between nodes as described earlier. We see now that convergence is governed not only by the maximum value of the kernel, but also the dependence amongst instances. Because we have assumed that the data from which each μ has arisen is i.i.d. the only source of bias is due to the overlap in nodes. This implies that if the degree is bounded by some finite constant, d as the network grows to infinity, there will be a maximum value of $\|\Theta\|_\infty < \infty$, which implies convergence. □

The Hilbert-Schmidt independence criterion [34] provides a test of dependence between two random variables, X and Y. Recall from previous chapter that HSIC is defined as

$$HSIC(X, Y) = \|P_{XY} - P_X P_Y\|_{\mathcal{H}}^2 = \|E[\phi(x) \otimes \psi(y)] - E[\phi(x)]E[\psi(y)]\|^2$$

Consistency is provided by the following theorem:

Theorem 4. *[33] Let k and l be characteristic kernels for the respective RKHSs $\mathcal{F}$ on $\mathcal{X}$ and $\mathcal{G}$ on $\mathcal{Y}$, with feature maps ϕ and ψ, respectively. Define the finite signed measure $\theta := P_{XY} - P_X P_Y$. Then, $C_{YX} = \int_{\mathcal{X}\times\mathcal{Y}} \psi(y) \otimes \phi(x) d\theta(x, y) = 0$ if and only if $\theta = 0$.*

The proof is given by Gretton [33]. The implication of Theorem 4 is that given *marginal* embeddings of P_X and P_Y that are characteristic, the statistic provided by HSIC is consistent, i.e., $HSIC(P_X, P_Y) \to 0$ if and only if $P_X \perp\!\!\!\perp P_Y$ as $n \to \infty$.

Given Lemma 4, HSIC is readily extended to the relational case. We do so with the following definition:

Definition 9 (Relational HSIC)**.** *In the relational setting, the test statistic is defined between a relational variable $C.\tau.X$ and a propositional variable $C.Y$ as:*

$$HSIC(C.\tau.X, C.Y) = \|P_{C.\tau.X,C.Y} - P_{C.\tau.X} P_{C.Y}\|^2_{\mathcal{H}} = \|E[\mu_x \otimes \psi(y)] - \mathcal{M}_x \otimes \mu_y\|^2$$

Note that the estimated statistic is now given by $T = \frac{1}{n^2}\mathbf{tr}\tilde{\mathbf{K}}_x\tilde{K}_y$, where $\mathbf{K}$ is the two-stage kernel defined in equation A.1, with each instance defined as the neighbors of node i. While this statistic is very similar to the traditional HSIC measure, it is worth taking a moment to clarify the difference in *what* is being tested. In the relational setting, rather than testing the joint distribution of a set of pairs against a null of the product of their marginals, we assess the relationship between a set of *distributions* and the marginal of a random variable. To demonstrate this, consider the more explicit definition of relational HSIC.

$$HSIC(C.\tau.X, C.Y) = \|\frac{1}{N}\sum_{i}^{N}\frac{1}{m_i}\sum_{j}^{m_1}\phi(x_{i,j})\otimes\psi(y) - \frac{1}{N}\sum_{i}^{N}\frac{1}{m_i}\sum_{j}^{m_1}\phi(x_{i,j})\otimes\frac{1}{N}\sum_{i}^{N}\psi(y_i)\|^2_{\mathcal{H}} \tag{A.2}$$

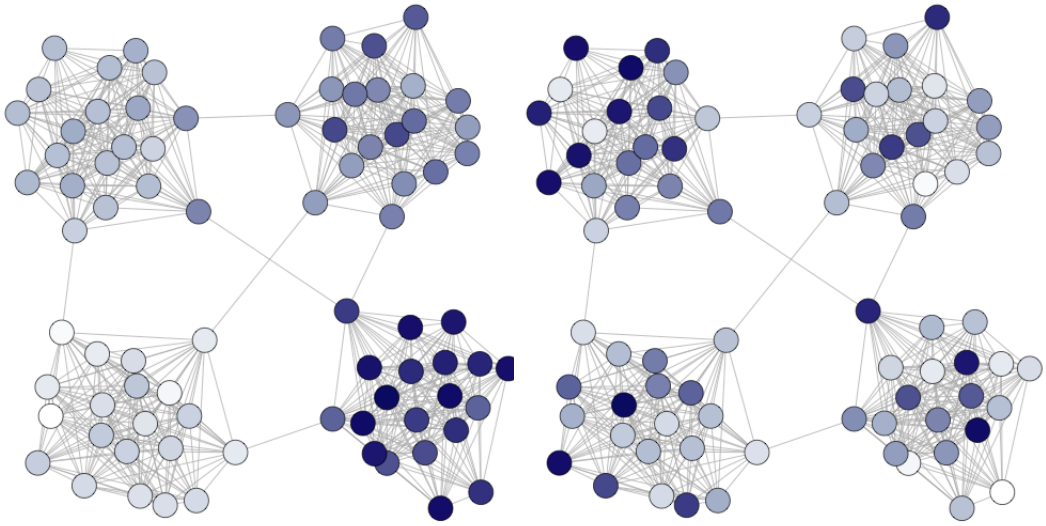
Here we can see that what is being considered is the *average* of the joint embeddings of y and each x which is contained in the set constituting μ_x.

Corollary 3 (Consistency of relational HSIC)**.** *Assume that (1) the kernel $k(C.Y, \cdot)$ is characteristic, (2) the kernel $k'(C.\tau.X, \cdot)$ is characteristic, (3) each $\hat{\mu} \in C.\tau.X$ is close to its population counterpart, (4) the second-level kernel $K(\mu_{C.\tau.X}, \cdot)$ is Gaussian, and (5) the degree of the relational structure is bounded by some constant, d. Then, $HSIC(C.\tau.X, C.Y)$ provides a consistent test of dependence.*

Proof. This is a direct consequence of Theorem 4 and Lemma 4. Lemma 4 is required in order to ensure convergence to the kernel mean, and by extension injectivity. □

While convergence is guaranteed, the asymptotic approximations provided for the propositional version of HSIC are no longer appropriate. However, the null distribution can easily be simulated via permutation.

Outside of the additional considerations necessary to the relational version, HSIC thus far looks very similar to its propositional counterpart. However, as we have shown in the previous section, relational dependence is inherently asymmetric. Because HSIC is the covariance of the data in feature space, simple extension provides that it will be symmetric in the case of regular network structure. We will investigate under which conditions asymmetry occurs in non-regular network structure, and the implications for both direction and dependence testing as part of this thesis.



(a) Original network (b) Permuted network

Figure A.1: An example of a network with data generated from an autodependent process and its permutation. On the left is the original network, where color represents the value of interest. On the right is a permutation of that network that preserves the structure while permuting the values. Shading of each node represents the value of the variable on that node.

A.0.1.1 Significance Testing

In this setting a simple approximation of the null distribution can be obtained empirical via Monte Carlo methods. The simplest of these methods is to permute the values of $\mathcal{X}$ by randomly reassigning each instance to a different node. For an intuition of the procedure, consider Figure A.1. On the left is our graph $\mathcal{G}$ and the values of a variable shown via the coloring of each vertex. The test is to determine whether the values of X depend on the structure of $\mathcal{G}$. By permuting the values (the result of one permutation is shown on the right), both the marginal distribution of X and all properties of $\mathcal{G}$ are preserved while ensuring independence between them. At each iteration, the bias induced from the graph structure remains constant as we have not altered it at all, but only the labels, or values, assigned to each node. However, any dependence between the values of $\mathcal{X}$ on neighboring nodes and an instance not due to aggregation bias will be destroyed. This is exactly the null distribution we

seek to simulate. Note that the naive procedure of permuting the values, treating the aggregates as fixed values will return a null distribution that represents the hypothesis of both no autodependence *and* no aggregation bias, and is likely to return a high number of type I errors.

BIBLIOGRAPHY

[1] Angin, Pelin, and Neville, Jennifer. A shrinkage approach for modeling non-stationary relational autocorrelation. In *Eighth IEEE International Conference on Data Mining* (2008), IEEE, pp. 707–712.

[2] Aral, Sinan, and Walker, Dylan. Creating social contagion through viral product design: A randomized trial of peer influence in networks. *Management Science 57*, 9 (2011), 1623–1639.

[3] Arbour, David, Garant, Dan, and Jensen, David. Inferring network effects from observational data. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (2016), ACM, pp. 715–724.

[4] Arbour, David, Marazopoulou, Katerina, and Jensen, David. Inferring causal direction from relational data. In *Proceedings of the Thirty-Second Conference on Uncertainty in Artificial Intelligence* (2016), AUAI Press, pp. 12–21.

[5] Aronow, Peter M, and Samii, Cyrus. Estimating average causal effects under general interference. In *Summer Meeting of the Society for Political Methodology, University of North Carolina, Chapel Hill, July* (2012), pp. 19–21.

[6] Aronow, Peter M., and Samii, Cyrus. Estimating average causal effects under interference between units. *arXiv preprint arXiv:1305.6156* (2013).

[7] Bakshy, Eytan, Eckles, Dean, Yan, Rong, and Rosenn, Itamar. Social influence in social advertising: evidence from field experiments. In *Proceedings of the 13th ACM Conference on Electronic Commerce* (2012), ACM, pp. 146–161.

[8] Blei, David M, Ng, Andrew Y, and Jordan, Michael I. Latent dirichlet allocation. *Journal of machine Learning research 3*, Jan (2003), 993–1022.

[9] Borsuk, Mark E, Stow, Craig A, and Reckhow, Kenneth H. A bayesian network of eutrophication models for synthesis, prediction, and uncertainty analysis. *Ecological Modelling 173*, 2 (2004), 219–239.

[10] Boyd, Stephen. Convex optimization of graph laplacian eigenvalues. In *Proceedings of the International Congress of Mathematicians* (2006), vol. 3, pp. 1311–1319.

[11] Chickering, David Maxwell. Learning equivalence classes of bayesian-network structures. *Journal of machine learning research 2*, Feb (2002), 445–498.

[12] Christakis, Nicholas, and Fowler, James. *Connected: The Surprising Power of Our Social Networks and How They Shape Our Lives*. Hachette Digital, Inc., 2009.

[13] Christmann, Andreas, and Steinwart, Ingo. Universal kernels on non-standard input spaces. In *Advances in neural information processing systems* (2010), pp. 406–414.

[14] Chwialkowski, Kacper, and Gretton, Arthur. A kernel independence test for random processes. In *Proceedings of the 31st International Conference on Machine Learning (ICML-14)* (2014), pp. 1422–1430.

[15] Chwialkowski, Kacper P., Sejdinovic, Dino, and Gretton, Arthur. A Wild Bootstrap for Degenerate Kernel Tests. In *Advances in Neural Information Processing Systems* (2014), pp. 3608–3616.

[16] Cornia, Nicholas, and Mooij, Joris M. Type-II errors of independence tests can lead to arbitrarily large errors in estimated causal effects: An illustrative example. In *Proceedings of the UAI 2014 Workshop Causal Inference: Learning and Prediction* (2014), pp. 35–42.

[17] Cornia, Nicholas, Mooij, Joris M, et al. Type-ii errors of independence tests can lead to arbitrarily large errors in estimated causal effects: An illustrative example. In *CEUR Workshop Proceedings* (2014), no. 1274, CEUR, pp. 35–42.

[18] Cortes, Corinna, Mohri, Mehryar, and Rostamizadeh, Afshin. Algorithms for learning kernels based on centered alignment. *The Journal of Machine Learning Research 13*, 1 (2012), 795–828.

[19] Daniusis, P, Janzing, D, Mooij, J, Zscheischler, J, Steudel, B, Zhang, K, Schölkopf, B, Spirtes, Grünwald P, et al. Inferring deterministic causal relations. In *26th Conference on Uncertainty in Artificial Intelligence (UAI 2010)* (2010), AUAI Press, pp. 143–150.

[20] Daudin, JJ. Partial association measures and an application to qualitative regression. *Biometrika 67*, 3 (1980), 581–590.

[21] Dawid, A Philip. Causal inference without counterfactuals. *Journal of the American Statistical Association 95*, 450 (2000), 407–424.

[22] Dedecker, Jérôme, Doukhan, Paul, Lang, Gabriel, Rafael, León R José, Louhichi, Sana, and Prieur, Clémentine. Weak dependence. In *Weak Dependence: With Examples and Applications*. Springer, 2007, pp. 9–20.

[23] Dehling, Herold, Durieu, Olivier, and Volny, Dalibor. New techniques for empirical processes of dependent data. *Stochastic Processes and their Applications 119*, 10 (2009), 3699–3718.

[24] Dhurandhar, Amit, and Dobra, Alin. Distribution-free bounds for relational classification. *Knowledge and Information Systems 31*, 1 (2012), 55–78.

[25] Eckles, Dean, Karrer, Brian, and Ugander, Johan. Design and analysis of experiments in networks: Reducing bias from interference. *arXiv preprint arXiv:1404.7530* (2014).

[26] Efron, Bradley. *The jackknife, the bootstrap and other resampling plans*, vol. 38. SIAM.

[27] Ferrucci, David, Levas, Anthony, Bagchi, Sugato, Gondek, David, and Mueller, Erik T. Watson: Beyond jeopardy! *Artificial Intelligence 199* (2013), 93–105.

[28] Flaxman, Seth R., Neill, Daniel B., and Smola, Alexander J. Gaussian processes for independence tests with non-iid data in causal inference. *ACM Transactions on Intelligent Systems and Technology (TIST)* (To Appear).

[29] Friedman, Jerome H. Greedy function approximation: A gradient boosting machine. *Annals of Statistics* (2001), 1189–1232.

[30] Fukumizu, Kenji, Gretton, Arthur, Sun, Xiaohai, and Schölkopf, Bernhard. Kernel measures of conditional dependence. In *Advances in Neural Information Processing Systems* (2007), pp. 489–496.

[31] Getoor, Lise, and Taskar, Ben. *Introduction to statistical relational learning*. MIT press, 2007.

[32] Gopnik, Alison, Glymour, Clark, Sobel, David M, Schulz, Laura E, Kushnir, Tamar, and Danks, David. A theory of causal learning in children: Mausal maps and bayes nets. *Psychological Review 111*, 1 (2004), 3.

[33] Gretton, A. A simpler condition for consistency of a kernel independence test. *ArXiv e-prints* (Jan. 2015).

[34] Gretton, A., Fukumizu, K., Teo, C.H., Song, L., Schölkopf, B., and Smola, A.J. A kernel statistical test of independence. In *Advances in Neural Information Processing Systems* (2008), pp. 585–592.

[35] Gretton, A., Herbrich, R., Smola, A., Bousquet, O., and Schoelkopf, B. Kernel methods for measuring independence. *Journal of Machine Learning Research 6* (2005), 2075–2129.

[36] Gretton, Arthur, Borgwardt, Karsten M., Rasch, Malte J., Schölkopf, Bernhard, and Smola, Alexander. A kernel two-sample test. *The Journal of Machine Learning Research 13*, 1 (2012), 723–773.

[37] Gui, Huan, Xu, Ya, Bhasin, Anmol, and Han, Jiawei. Network a/b testing: From sampling to estimation. In *Proceedings of the 24th International Conference on World Wide Web* (2015), International World Wide Web Conferences Steering Committee, pp. 399–409.

[38] Hudgens, Michael G, and Halloran, M Elizabeth. Toward causal inference with interference. *Journal of the American Statistical Association* (2012).

[39] Janzing, Dominik, Mooij, Joris, Zhang, Kun, Lemeire, Jan, Zscheischler, Jakob, Daniušis, Povilas, Steudel, Bastian, and Schölkopf, Bernhard. Information-geometric approach to inferring causal directions. *Artificial Intelligence 182* (2012), 1–31.

[40] Jensen, David, and Neville, Jennifer. Linkage and Autocorrelation Cause Feature Selection Bias in Relational Learning. In *Machine Learning, Proceedings of the Nineteenth International Conference* (2002), pp. 259–266.

[41] Jensen, David, and Neville, Jennifer. Autocorrelation and linkage cause bias in evaluation of relational learners. In *Inductive Logic Programming*. Springer, 2003, pp. 101–116.

[42] Kearns, Michael, Judd, Stephen, and Vorobeychik, Yevgeniy. Behavioral experiments on a network formation game. In *Proceedings of the 13th ACM Conference on Electronic Commerce* (2012), ACM, pp. 690–704.

[43] Koller, Daphne. Probabilistic relational models. In *Inductive logic programming*. Springer, 1999, pp. 3–13.

[44] Kondor, Risi Imre, and Lafferty, John. Diffusion kernels on graphs and other discrete input spaces. In *ICML* (2002), vol. 2, pp. 315–322.

[45] Krishnan, S Shunmuga, and Sitaraman, Ramesh K. Video stream quality impacts viewer behavior: Inferring causality using quasi-experimental designs. *Networking, IEEE/ACM Transactions on 21*, 6 (2013), 2001–2014.

[46] La Fond, Timothy, and Neville, Jennifer. Randomization tests for distinguishing Social Influence and Homophily Effects. In *Proceedings of the 19th international conference on World wide web* (2010), ACM, pp. 601–610.

[47] LeCun, Yann, Bengio, Yoshua, and Hinton, Geoffrey. Deep learning. *Nature 521*, 7553 (2015), 436–444.

[48] Lee, Sanghack, and Honavar, Vasant. On learning ccausal models from relational data. In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence* (2016), AAAI Press, pp. 3263–3270.

[49] Leskovec, Jure, and Krevl, Andrej. SNAP Datasets: Stanford large network dataset collection. http://snap.stanford.edu/data, June 2014.

[50] Leucht, Anne, and Neumann, Michael H. Dependent Wild Bootstrap for Degenerate U-and V-statistics. *Journal of Multivariate Analysis 117* (2013), 257–280.

[51] Lichman, M. UCI machine learning repository, 2013.

[52] London, Ben, Huang, Bert, Taskar, Benjamin, and Getoor, Lise. Collective Stability in Structured Prediction: Generalization from One Example. In *Proceedings of the 30th International Conference on Machine Learning (ICML-13)* (2013).

[53] Lopez-Paz, D., Muandet, K., and Recht, B. The randomized causation coefficient. *Journal of Machine Learning* (2015).

[54] Lopez-Paz, David, Hennig, Philipp, and Schölkopf, Bernhard. The randomized dependence coefficient. In *Advances in Neural Information Processing Systems* (2013), pp. 1–9.

[55] Maier, Marc, Marazopoulou, Katerina, Arbour, David, and Jensen, David. A Sound and Complete Algorithm for Learning Causal Models from Relational Data. In *Proceedings of the Twenty-Ninth Conference on Uncertainty in Artificial Intelligence* (2013), pp. 371–380.

[56] Maier, Marc, Marazopoulou, Katerina, Arbour, David, and Jensen, David. A sound and complete algorithm for learning causal models from relational data. In *Uncertainty in Artificial Intelligence* (2013), Citeseer, p. 371.

[57] Maier, Marc, Marazopoulou, Katerina, and Jensen, David. Reasoning About Independence in Probabilistic Models of Relational Data. *arXiv preprint arXiv:1302.4381* (2013).

[58] Maier, Marc E., Marazopoulou, Katarina, and Jensen, David D. Reasoning about independence in probabilistic models of relational data. arXiv preprint arXiv:1302.4381, 2014.

[59] Manski, Charles F. Identification of Treatment Response With Social Interactions. *The Econometrics Journal 16*, 1 (2013), S1–S23.

[60] Marazopoulou, Katerina, Arbour, David, and Jensen, David. Refining the Semantics of Social Influence. *Networks: From Graphs to Rich Data. NIPS Workshop* (2014).

[61] Marazopoulou, Katerina, Maier, Marc, and Jensen, David. Learning the Structure of Causal Models with Relational and Temporal Dependence. In *Proceedings of the Thirty-First Conference on Uncertainty in Artificial Intelligence* (2015).

[62] Marazopoulou, Katerina, Maier, Marc, and Jensen, David. Learning the structure of causal models with relational and temporal dependence. In *Proceedings of the 31st Conference on Uncertainty in Artificial Intelligence* (2015).

[63] Margaritis, Dimitris. Distribution-free learning of graphical model structure in continuous domains.

[64] Muchnik, Lev, Aral, Sinan, and Taylor, Sean J. Social influence bias: A randomized experiment. *Science 341*, 6146 (2013), 647–651.

[65] Neville, Jennifer, and Jensen, David. Relational Dependency Networks. *The Journal of Machine Learning Research 8* (2007), 653–692.

[66] Neville, Jennifer, Jensen, David, Friedland, Lisa, and Hay, Michael. Learning relational probability trees. In *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining* (2003), ACM, pp. 625–630.

[67] Ogburn, Elizabeth L., and VanderWeele, Tyler J. Causal diagrams for interference. *Statist. Sci. 29*, 4 (11 2014), 559–578.

[68] Ogburn, Elizabeth L., VanderWeele, Tyler J., et al. Causal Diagrams for Interference. *Statistical Science 29*, 4 (2014), 559–578.

[69] Oktay, Hüseyin, Taylor, Brian J., and Jensen, David. Causal discovery in social media using quasi-experimental designs. In *Proceedings of the SIGKDD/ACM Workshop on Social Media Analytics* (2010).

[70] Pearl, Judea. Graphs, causality, and structural equation models. *Sociological Methods & Research 27*, 2 (1998), 226–284.

[71] Pearl, Judea. *Causality*. Cambridge university press, 2009.

[72] Perlich, Claudia, and Provost, Foster. Distribution-based aggregation for relational learning with identifier attributes. *Machine Learning 62*, 1-2 (2006), 65–105.

[73] Peters, Jonas, Mooij, Joris M., Janzing, Dominik, and Schölkopf, Bernhard. Causal discovery with continuous additive noise models. *The Journal of Machine Learning Research 15*, 1 (2014), 2009–2053.

[74] Poole, David, and Crowley, Mark. Cyclic Causal Models with Discrete Variables: Markov Chain Equilibrium Semantics and Sample Ordering. In *Proceedings of the Twenty-Third international joint conference on Artificial Intelligence* (2013), AAAI Press, pp. 1060–1068.

[75] Rattigan, Matthew JH. *Leveraging Relational Representations for Causal Discovery*. PhD thesis, University OF Massachusetts Amherst, 2012.

[76] Rubin, Donald B. Matching to remove bias in observational studies. *Biometrics* (1973), 159–183.

[77] Rubin, Donald B. Causal inference using potential outcomes. *Journal of the American Statistical Association* (2011).

[78] Sejdinovic, Dino, Gretton, Arthur, and Bergsma, Wicher. A kernel test for three-variable interactions. In *Advances in Neural Information Processing Systems* (2013), pp. 1124–1132.

[79] Shao, Xiaofeng. The Dependent Wild Bootstrap. *Journal of the American Statistical Association 105*, 489 (2010), 218–235.

[80] Shpitser, Ilya, and VanderWeele, Tyler J. A complete graphical criterion for the adjustment formula in mediation analysis. *The International Journal of Biostatistics 7*, 1 (2011), 1–24.

[81] Smola, Alexander J, and Kondor, Risi. Kernels and regularization on graphs. In *Learning theory and kernel machines*. Springer, 2003, pp. 144–158.

[82] Song, Le. *Learning via Hilbert space embedding of distributions*. 2008.

[83] Spirtes, Peter, Glymour, Clark N, and Scheines, Richard. *Causation, Prediction, and Search*, vol. 81. MIT press, 2000.

[84] Sriperumbudur, Bharath, Fukumizu, Kenji, and Lanckriet, Gert. Universality, characteristic kernels and rkhs embedding of measures. *The Journal of Machine Learning Research 12* (2011), 2389–2410.

[85] Sriperumbudur, Bharath K, Gretton, Arthur, Fukumizu, Kenji, Lanckriet, Gert RG, and Schölkopf, Bernhard. Injective hilbert space embeddings of probability measures. In *COLT* (2008), vol. 21, pp. 111–122.

[86] Stegle, Oliver, Janzing, Dominik, Zhang, Kun, Mooij, Joris M, and Schölkopf, Bernhard. Probabilistic latent variable models for distinguishing between cause and effect. In *Advances in Neural Information Processing Systems* (2010), pp. 1687–1695.

[87] Sun, Hongwei. Mercer theorem for rkhs on noncompact sets. *Journal of Complexity 21*, 3 (2005), 337–349.

[88] Toulis, Panos, and Kao, Edward. Estimation of Causal Peer Influence Effects. In *Proceedings of The 30th International Conference on Machine Learning* (2013), pp. 1489–1497.

[89] Ugander, Johan, Karrer, Brian, Backstrom, Lars, and Kleinberg, Jon. Graph cluster randomization: Network exposure to multiple universes. In *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining* (2013), ACM, pp. 329–337.

[90] Urry, Matthew, and Sollich, Peter. Random walk kernels and learning curves for gaussian process regression on random graphs. *Journal of Machine Learning Research 14*, 1 (2013), 1801–1835.

[91] VanderWeele, Tyler J. Ignorability and Stability Assumptions in Neighborhood Effects Research. *Statistics in Medicine 27*, 11 (2008), 1934–1943.

[92] Wasserman, Larry. *All of statistics: a concise course in statistical inference*. Springer Science & Business Media, 2013.

[93] Wu, Chien-Fu Jeff. Jackknife, bootstrap and other resampling methods in regression analysis. *the Annals of Statistics* (1986), 1261–1295.

[94] Xiang, Rongjing, and Neville, Jennifer. Relational learning with one network: An asymptotic analysis. In *International Conference on Artificial Intelligence and Statistics* (2011), pp. 779–788.

[95] Zhang, K., Peters, J., Janzing, D., and Schölkopf, B. Kernel-Based Conditional Independence Test and Application in Causal Discovery. AUAI Press, pp. 804–813.

[96] Zhang, Kun, Peters, Jonas, and Janzing, Dominik. Kernel-based conditional independence test and application in causal discovery. In *In Uncertainty in Artificial Intelligence* (2011), Citeseer.

[97] Zhang, Xinhua, et al. *Graphical Models: Modeling, Optimization, and Hilbert Space Embedding*. PhD thesis, The Australian National University, 2010.

[98] Zhang, Xinhua, Song, Le, Gretton, Arthur, and Smola, Alex J. Kernel Measures of Independence for Non-IID Data. In *Advances in Neural Information Processing Systems* (2009), pp. 1937–1944.

[99] Zhu, Xiaojin, Kandola, Jaz, Ghahramani, Zoubin, and Lafferty, John D. Nonparametric transforms of graph kernels for semi-supervised learning. In *Advances in neural information processing systems* (2004), pp. 1641–1648.