technische universität
dortmund

fakultät
statistik

Probabilistic Graphical Models in the Manufacturing of Electric Vehicles

Dissertation zur Erlangung des Doktorgrades Dr. rer. nat. der Fakultät Statistik der Technischen Universität Dortmund

Vorgelegt von

Maximilian Kertel

geboren in Stuttgart

Dortmund, Mai 2024

Amtierender Dekan:
Prof. Dr. Philipp Doebler

Gutachter:
Prof. Dr. Markus Pauly (Technische Universität Dortmund)
Prof. Dr. Nadja Klein (Technische Universität Dortmund)

Abstract

Introduction: Battery electric vehicles (BEVs) are crucial to the reduction of greenhouse gas emissions in the transportation sector. The primary component of BEVs is the energy storage system, consisting of battery cells. However, its manufacturing process is intricate, characterized by numerous causal interdependencies across production steps. The limited understanding of these interdependencies contributes to high scrap rates, resulting in increased environmental, ethical, and economic costs associated with BEVs.

Methods: In order to address this challenge, this dissertation leverages the data and measurements collected by modern manufacturing facilities and production machines. Specifically, Probabilistic Graphical Models (PGMs), which can be grouped into directed and undirected models, are applied to this data to enhance the understanding of the production process, addressing the lack of knowledge. By utilizing PGMs, this research aims to uncover hidden relationships and dependencies within the manufacturing data, enabling more informed decision-making in battery cell manufacturing. In order to tackle the inherent complexity of the data, we focus on non-linear methods.

Specifically:

1. **Missing Data:** We address the challenge of estimating the joint distribution when parts of the data are missing, which is a common occurrence in manufacturing data that comprises sensor measurements. The proposed estimation procedure can be viewed as a learning algorithm for undirected PGMs.
2. **Real-Data Application:** Additionally, we apply a state-of-the-art learning procedure for directed PGMs to actual manufacturing data.
3. **Boosting:** Finally, we utilize the concept of boosting to learn directed PGMs from the data and investigate the theoretical and practical benefits.

We account for the circumstances in manufacturing scenarios. This includes leveraging prior knowledge in various aspects.

Results:

1. **Missing Data:** The proposed method effectively learns joint distributions semi-parametrically. A simulation study shows that the estimates improve with the sample size of the data, and the inclusion of expert knowledge in the estimation process leads to a holistic improvement in the accuracy of the estimates.
2. **Real-Data Application:** In contrast to other applications of PGMs, we observe large local variations in the number of relationships, challenging the assumption of sparsity. The integration of expert knowledge provides more reliable estimates in real-world manufacturing data applications.
3. **Boosting:** We demonstrate the consistency of a boosting-based learning algorithm for directed PGMs, which is a rare statistical guarantee for such algorithms. The practical adaptation of this algorithm proves to be competitive and, in some relevant cases, even outperforms state-of-the-art methods.

The results collectively demonstrate the significance and practical applicability of PGMs in the context of manufacturing applications.

Discussion & Outlook: We critically assess the derivation of causal relationships from data collected at the steady state of the production workflow. We propose a novel point of view on causal discovery as a recommendation system for potential causal relationships in manufacturing. Additionally, we sketch the idea of an iterative procedure involving PGM learning algorithms and experiments to derive causal relationships.

Danksagung

Viele Menschen haben Ihren Teil zu dieser Arbeit beigetragen. An erster Stelle möchte ich Herrn Professor Markus Pauly danken. Dein Wissen, Deine Kreativität und Deine Energie, gepaart mit Deiner sozialen Intelligenz, sind einzigartig.

Danken möchte ich ebenfalls Frau Professor Nadja Klein für den ermutigenden, bestärkenden und motivierenden Austausch, die Inspiration für das gemeinsame Papier und für die Erstellung des Zweitgutachtens.

Die Promotion war geprägt von der Pandemie und der großen Entfernung zwischen meinem Wohn- und Arbeitsort und dem Lehrstuhl. Umso schöner und wichtiger war es für mich, Euch als Kolleg*innen am Lehrstuhl zu haben. Ich habe mich vom ersten Tag an willkommen gefühlt und Eure Gastfreundschaft und Aufnahmebereitschaft waren überwältigend. Vielen Dank!

Des Weiteren möchte ich mich bei der BMW Group für die finanzielle Unterstützung, die Möglichkeiten und Freiheiten bedanken, die ich genießen durfte. Ein besonderer Dank gilt Thomas Kornas, der mir stets den Rücken freigehalten und die Ausrichtung meiner Arbeit unterstützt hat.

Meiner Mutter möchte ich danken, dass sie mich darin bestärkte trotz eines guten Jobs eine Promotion zu beginnen. Du hast meine Sorgen ernst genommen auch wenn diese oft schwer zu verstehen waren und mir vermittelt, dass mein Wert nicht vom Erfolg meiner Arbeit abhängt.

Meinem Vater danke ich für die Außenperspektive, wenn ich mich wieder in den Details verloren habe. Dein Vertrauen in mich war größer als mein eigenes. Am Ende hast Du wieder einmal Recht behalten.

Meiner Schwester möchte ich dafür danken, dass sie stets Empathie für meine Probleme aufbrachte, die mit Abstand betrachtet eigentlich unbedeutend waren.

Meinen Freundinnen und Freunden danke ich für ihre bedingungslose Unterstützung und ihr Verständnis. Ihr gebt mir den sicheren Hafen und Rückzugsort, von dem ich aus Wagnisse eingehen kann, von denen ich nicht weiß, ob sie gelingen werden.

Schließlich möchte ich Kristina danken, der ich in den letzten Jahren viel abverlangt habe. Du hast meinen Frust und meine Einsilbigkeit ertragen, wenn die Buchstaben auf den Papierstapeln nicht zusammenpassten. Du hast mich abgelenkt, aufgebaut und motiviert. Meinen Ehrgeiz und meine Verbissenheit hast Du mit großem Einsatz in geordnete Bahnen gelenkt. Du hast mich dazu angehalten Erfolgserlebnisse zu genießen und mir immer wieder gezeigt, dass die Welt nicht hinter dem Schreibtisch endet. Ich danke Dir für Dein Mitgefühl, Deine Lebensfreude und Deine Unternehmungslust, die Du in mein Leben bringst.

List of Publications

This cumulative thesis is based on the following three manuscripts:

Article 1: Kertel, M. & Pauly, M. (2022). Estimating Gaussian Copulas with Missing Data with and without Expert Knowledge. *Entropy*, 24(12), https://doi.org/10.3390/e24121849.

Contribution of the author:
The author conceptualized and wrote the manuscript under the guidance of Prof. Pauly. He implemented the method and designed and programmed the simulation study.

Article 2: Kertel, M., Harmeling, S. & Pauly, M. (2022). Learning Causal Graphs in Manufacturing Domains using Structural Equation Models. *IEEE International Conference on Artificial Intelligence for Industries*, pages 14–19, https://doi.org/10.1109/AI4I54798.2022.00010.

Contribution of the author:
The author conceptualized and wrote the manuscript under the guidance of Prof. Pauly. He implemented the method and the data analysis and designed and programmed the simulation study.

Article 3: Kertel, M. & Klein, N. (2024). Boosting Causal Additive Models. *arXiv*, https://doi.org/10.48550/arXiv.2401.06523

Contribution of the author:
Inspired by an idea of Prof. Klein, the author conceptualized and wrote the manuscript. He worked out the theoretical results, implemented the methods, and designed and programmed the simulation study.

Further publication:

(1) Wehner, C., Kertel, M. & Wewerka, J. (2023). Interactive and Intelligent Root Cause Analysis in Manufacturing with Causal Bayesian Networks and Knowledge Graphs. *IEEE Vehicular Technology Conference*, pages 1-7, https://doi.org/10.1109/VTC2023-Spring57618.2023.10199563

Contents

Notation

Throughout the thesis, vectors and random vectors are denoted by bold symbols, e.g., $\mathbf{x}$ or $\mathbf{X}$, respectively. Further specific notation is introduced in the respective sections.

$\mathbb{N}$	Natural numbers
$\mathbb{R}$	Real numbers
p	Number of dimensions, $p \in \mathbb{N}$
N	Sample size, $N \in \mathbb{N}$
$[k]$	Set $\{1, \ldots, k\}$, $k \in \mathbb{N}$
$\mathcal{B}(\mathbb{R})$	Borel σ-algebra on $\mathbb{R}$
$\mathcal{B}(\mathbb{R}^p)$	Borel product σ-algebra on $\mathbb{R} \times \ldots \times \mathbb{R} = \mathbb{R}^p$
$\sigma(\mathbf{X})$	σ-algebra generated by random vector $\mathbf{X}$
dx	Integral with respect to Lebesgue measure for univariate x ($d\mathbf{x}$ for multivariate $\mathbf{x}$, and analogously dy, $d\mathbf{y}, \ldots$)
P	Probability measure
$P(\mathbf{X})$	Push-forward measure of $\mathbf{X}$
θ	Statistical parameter
Θ	Parameter space, so that $\theta \in \Theta$

Part I

Introduction

1 Motivation

Battery electric vehicles (BEVs) play a crucial role in the transportation industry in mitigating greenhouse gas emissions. As a result, there is a projected exponential increase in both the adoption of BEVs and the manufacturing capabilities of battery cells. The manufacturing of battery cells is a **complex process**, characterized by a large number of production steps, which are **highly and causally interdependent**. Due to these interdependencies, which we call **cause-effect relationships (CERs)**, the process parameters must be coordinated between production steps. Unfortunately, many of these **interdependencies are unknown**, so that tuning the production workflow becomes challenging (Westermeier et al., 2014; Schnell and Reinhart, 2016; Örüm Aydin et al., 2023; Fitzner et al., 2023).

These challenges become apparent in the high scrap rate of battery cells, which is reported to be around 5% even for established cell manufacturers (Gaines et al., 2021). This is alarming, as the costs, the energy demand for mining and the social and environmental footprint of raw materials, such as cobalt, manganese, nickel and lithium, are significant (Örüm Aydin et al., 2023). Not surprisingly, given the growth of the battery sector, improving the efficiency of the production process is an active field of (interdisciplinary) research (Liu et al., 2021).

Simultaneously, the age of the Internet of Things in Industry (IIoT), Industry 4.0 and technologies such as OPC-UA (Drahoš et al., 2018) allow communication with machines and the **gathering of data**, that is, sensor measurements, images, sound, etc., **throughout the production workflow**. Therefore, a product arriving at the end of a production line can be characterized by dozens or hundreds of measurements, while the number of products is also large. Consequently, data-driven decision making is ubiquitous in modern manufacturing facilities. They are used to identify erroneous products and to detect trends using (semi-) supervised learning procedures. For example, technologies such as computer vision or audio analysis are used to detect failures during the production process (Rai et al., 2021).

However, obtaining CERs, in particular across production steps, is considerably more challenging, since it involves the **identification of causal relationships** instead of

correlational patterns (Vuković and Thalmann, 2022). The state-of-the-art methods for deriving CERs can be grouped into

- approaches for extracting expert knowledge (for example, Failure Mode and Effect Analysis; FMEA; Westermeier et al. 2013), and
- systematic experiment design (Design of Experiments; DOEs; Román-Ramírez and Marco 2022).

Although both approaches are widely applied, they lead to unsatisfactory results for battery cell production processes, as can be seen from the high scrap rate. While expert-based methods are unsuitable for deriving unknown CERs by construction, DOEs are challenging, as the **CERs are complex** and can have many influential factors. Thus, "[DOEs] soon reach their limits of applicability" (Schnell and Reinhart, 2016). Furthermore, it is typical for the production process that findings in one plant are not directly transferable to other plants, so experiments must be carried out in the main production facility (Grießl et al., 2022). Therefore, experiments cause an interruption in the production workflow and generate **high costs**. We emphasize that neither takes advantage of the data from the steady state of the manufacturing process. The structure of this data is intricate and the relationships between variables are hardly accessible to humans. Therefore, it is not used to deepen the understanding of the process workflow. This thesis aims to change that by applying and developing methods tailored for manufacturing that allow the visualization of complex data sets in an accessible fashion. We take into account the following situations.

In contrast to other data-driven situations, we recognize the following **favorable aspects**.

- **Big data:** The **volume**, that is the number of samples and measurements, is **large**.
- **Expert knowledge:** The production of battery cells is a highly active field of research. Although expert knowledge is limited in some respects, it is deep in other aspects of the production process.

On the other hand, we observe the following **obstructions**.

- **Missing data:** While IIoT data has a large volume it often comes with a low quality, that is, sensor measurements can be missing or implausible (Teh et al., 2020).
- **Complex data:** The data is complex due to intricate CERs (Fitzner et al., 2023).
- **Interdisciplinary teams:** Efficient battery cell manufacturing requires expertise in various fields. Therefore, the results need to be **communicated to process**

experts, who do not have extensive training in statistics, machine learning or mathematics (Kornas et al., 2019).

- **Precision vs. costs:** The manufacturing industry is strongly driven by the reduction of costs and, therefore, cycle times. Therefore, although certain measurement technologies effectively describe a feature of an intermediate product, they may not be integrated into the production workflow. For example, the acquisition costs of measurement technology can be too high or the measurement technology does not meet the cycle time requirements (Örüm Aydin et al., 2023).

Considering the insufficiency of the status quo approaches, we aim for a methodology that is capable of deriving connections between the measurements and making them accessible to humans using the **data of the steady state** of the production workflow. Deliberately, we leave open the exact definition of the kind of connection. If feasible, the objective is to derive the CERs.

Probabilistic graphical models (PGMs) represent multidimensional data with graphs consisting of nodes representing variables and edges representing relationships between those variables. Thus, this graph provides accessible information on how different variables are related. Learning algorithms to identify PGMs from data are an active field of research (Vowels et al., 2022). PGMs can be classified into directed and undirected graphs. For the former, the edges have a direction, while for the latter, this is not the case. The undirected and directed graphs describe different relationships between the variables.

The framework of PGMs is attractive, as the existing literature covers many of the already mentioned aspects. We briefly discuss them in the following.

A large proportion of the literature on PGMs investigates on how **causality**, that is CERs, can be deduced while avoiding experiments. Instead, these procedures utilize the observational distribution, which characterizes the **steady state** of the system. Observational data in the manufacturing sector comes at a **low cost**, as production interruptions for experiments are avoided. Surprisingly, the **complexity** of the relationships in a system can even promote the identification of relationships. The value of PGMs for **communication** is intrinsic. In many PGM learning algorithms, the inclusion of **expert knowledge** is possible. Due to early applications in fields such as genetics, research for high-dimensional data is well advanced and PGMs can be estimated from a machine learning perspective using gradient descent. Hence, many methods **scale well with the volume** of the data. Under specific assumptions, the consistency of the methods is shown (Kalisch and Bühlman, 2007; Raskutti et al., 2008; Bühlmann et al., 2014). Literature on **missing data** sparsely exists (Ding and Song, 2016; Tu et al., 2019).

PGMs are widely applied in fields such as biology and social sciences. The literature on the applications of PGMs to manufacturing is sparse and often does not consider state-of-the-art methods. This work contributes to this aspect and focuses on the following.

- The modelling approach in the case of **missing data** is often under-complex, Section 3.1.
- Although the benefit of incorporating **expert knowledge** is widely assumed, the literature on this topic is sparse (Spirtes and Zhang, 2016), Section 3.2.
- While there exist statistical consistency results for complex, non-linear PGMs, these are based on the maximum-likelihood regression, which is prone to overfitting. This leads to strict assumptions for the statistical consistency to hold. We replace the maximum-likelihood estimation by a boosting procedure and present assumptions under which the **PGM estimation is consistent**, Section 3.3.

A fourth publication investigates how expert knowledge and PGMs can be systematically combined for continuous knowledge discovery (Section 3.4).

The thesis is structured as follows. In Chapter 2, we introduce the methods used. Chapter 3 provides a summary of the three manuscripts included in this thesis, as well as the additional related article. Chapter 4 concludes with an analysis of the findings presented and offers some future research perspectives.

2 Statistical Methods

In this section, we outline the necessary statistical tools and methods. We will consider p-dimensional random vectors $\mathbf{X} = (X_1, \dots, X_p)$ with N i.i.d. realizations $\mathbf{x}^N = \{\mathbf{x}_1, \dots, \mathbf{x}_N\} \subset \mathbb{R}^p$, where $\mathbf{x}_\ell = (x_{\ell 1}, \dots, x_{\ell p})$ for $\ell = 1, \dots, N$. For a set $S = \{s_1, \dots, s_r\} \subset \{1, \dots, p\}$, we define $\mathbf{X}_S := (X_{s_1}, \dots, X_{s_r})$ and $s_1 < s_2 < \dots s_r$ to fix the order of $\mathbf{X}_S$. For $k \in \{1, \dots, p\}$ we define $[k] = \{1, \dots, k\}$. Our main goal is to find graphical representations of the distribution of $\mathbf{X}$, for which we assume throughout that the density with respect to the Lebesgue product measure exists. The resulting graph gives insights on the relationships between random variables $X_1, \dots, X_p$. The nodes of these graphs will be $X_1, \dots, X_p$, while the relationships will be represented by edges between these nodes.

This chapter is structured as follows. We start in Section 2.1 by introducing the concept of conditional distributions and conditional independence, which are essential for PGMs. In Section 2.2 we introduce copulas and describe how they characterize dependency structures. Section 2.3 introduces the graph terminology which is then leveraged in Section 2.4 for undirected PGMs and in Section 2.5 for directed PGMs. Section 2.6 presents some aspects of nonparametric regression, which is used to learn graphical models from data.

2.1 Conditional Distributions & Independence

This thesis investigates dependency structures in multivariate distributions through conditional distributions and independencies. In the following, we provide a foundation for these concepts.

Consider a probability space $(\Omega, \mathcal{F}, P)$, where Ω is the sample space and its elements are denoted by ω, $\mathcal{F}$ is a σ-algebra on Ω and P is a probability measure on $\mathcal{F}$. By $\mathcal{G} \subset \mathcal{F}$ we denote a sub-σ-algebra.

The random variable X and the random vectors $\mathbf{X}$ map to $\mathbb{R}$ and $\mathbb{R}^p$, respectively. They are assumed to be measurable with respect to the Borel σ-algebra $\mathcal{B}(\mathbb{R})$ or with respect

to the product Borel σ-algebra $\mathcal{B}(\mathbb{R}^p)$ on $\mathbb{R} \times \ldots \times \mathbb{R} = \mathbb{R}^p$. A set in these σ-algebras is called measurable. In addition, it is assumed that the implied distributions P_X and $P_{\mathbf{X}}$ of X and $\mathbf{X}$ are absolutely continuous with respect to the Lebesgue measure on $\mathbb{R}$ or $\mathbb{R}^p$. Thus, by the Radon-Nikodym theorem (Billingsley, 1995, Theorem 32.2) they have densities f_X, $f_{\mathbf{X}}$ so that, for example, for a measurable $A \subset \mathbb{R}^p$ it holds that

$$P_{\mathbf{X}}(A) = \int_A f_{\mathbf{X}}(\mathbf{x}) d\mathbf{x}.$$

2.1.1 Conditional Probability Functions

From introductory probability courses, it is well known that for any $A, B \in \mathcal{F}$ and $P(B) > 0$ the conditional probability is defined by

$$P(A|B) = \frac{P(A \cap B)}{P(B)}.$$

It describes the updated probability for the event A when it is known that the event B has taken place. For example, in poker, the event A could be the event of a royal flush, and B would be the event that the personal cards are ace and king in the same color. The function $A' \mapsto P(A'|B)$ for $A' \in \mathcal{F}$ constitutes a probability distribution.

On the other hand, in manufacturing, one might be interested in the conditional probability that a battery cell fulfills the quality requirements, knowing that the viscosity of the slurry takes on a specific value. The viscosity of the slurry is a production measurement and can be regarded as coming from a continuous distribution. In that case $P(B) = 0$ and the definition above fails. It is the purpose of the following definition to introduce conditional density functions that characterize $A' \mapsto P(A'|B)$. In all our applications, we consider continuously distributed random vectors, that is, random vectors with a density, for which we can use the following simple definition. For a broader introduction, see Billingsley (1995); Klenke (2013).

Definition 2.1.1. *Let $\mathbf{X} = (\mathbf{Y}, \mathbf{Z})$ be a random vector with density f, with $\mathbf{Y} \in \mathbb{R}^k$ and $\mathbf{Z} \in \mathbb{R}^{p-k}$. If $f(\mathbf{z}) := \int_{\mathbb{R}^k} f(\mathbf{t}, \mathbf{z}) d\mathbf{t} > 0$, the **conditional density** of $\mathbf{Y}$ given $\mathbf{Z} = \mathbf{z}$ is given by*

$$f(\mathbf{y}|\mathbf{z}) := \frac{f(\mathbf{y}, \mathbf{z})}{\int_{\mathbb{R}^k} f(\mathbf{t}, \mathbf{z}) d\mathbf{t}} = \frac{f(\mathbf{y}, \mathbf{z})}{f(\mathbf{z})}.$$

On the other hand, for $f(\mathbf{z}) = \int_{\mathbb{R}^k} f(\mathbf{t}, \mathbf{z}) d\mathbf{t} = 0$ the conditional density can be chosen arbitrarily.

For a fixed $\mathbf{z}$ with $f(\mathbf{z}) > 0$, Definition 2.1.1 corresponds to a density with respect to $\mathbf{y}$. Furthermore, the joint density of $\mathbf{X}$ can be decomposed into

$$f(x_1, \ldots, x_p) = f(x_1) \prod_{k=1}^{p-1} f(x_{k+1}|\mathbf{x}_{[k]}). \tag{2.1}$$

Definition 2.1.1 appears to be an arbitrary adaptation of the conditional probability for discrete random variables to continuous random variables. In the following, we embed it into a more general measure-theoretic context.

Definition 2.1.2. *Let B be a measurable set. We call a random variable $g_B(\mathbf{Z}) = P(B|\mathbf{Z})$ a **conditional probability of** B **given** $\mathbf{Z}$ if the following two conditions hold:*

1. *$g_B(\mathbf{z})$ is measurable with respect to $\sigma(\mathbf{Z})$,*
2. *$\mathbb{E}\left[1_A(\mathbf{Z}) g_B(\mathbf{Z})\right] = P(\mathbf{Z} \in A, B)$ for all measurable sets A.*

This definition is a special case of the one given, for example, in (Billingsley, 1995, Equation 33.8), which is sufficient for our purposes.

Proposition 2.1.3. *Any two conditional probabilities of B given $\mathbf{Z}$ are unique outside of a P null set. A specific choice is called a **version**.*

Proposition 2.1.4. *The random variable $g_{Y \in B}(\mathbf{z}) := \int_B f(\mathbf{y}|\mathbf{z}) d\mathbf{y}$ is a version of the conditional probability of $\{Y \in B\}$ given $\mathbf{Z}$.*

Definition 2.1.2 depends on the set B, so we can consider a family of random variables $g_{Y \in B}$ with the index B chosen in the measurable sets. Proposition 2.1.4 shows that every element of this family can be expressed by an integral of $f(\mathbf{y}|\mathbf{z})$ with respect to $\mathbf{y}$.

Example 2.1.5. *Let $\mathbf{X} = (\mathbf{Y}, \mathbf{Z})$ be a multivariate normally distributed with mean $\mu = (\mu_{\mathbf{Z}}, \mu_{\mathbf{Y}})$ and covariance matrix*

$$\Sigma = \begin{pmatrix} \Sigma_{\mathbf{Y},\mathbf{Y}} & \Sigma_{\mathbf{Y},\mathbf{Z}} \\ \Sigma_{\mathbf{Z},\mathbf{Y}} & \Sigma_{\mathbf{Z},\mathbf{Z}} \end{pmatrix},$$

where for example $\Sigma_{\mathbf{Y},\mathbf{Z}}$ is the covariance matrix of $\mathbf{Y}$ and $\mathbf{Z}$ and the other submatrices are defined analogously. Then the conditional density of $\mathbf{Y}$ given $\mathbf{Z} = \mathbf{z}$ is a Gaussian density with mean $\mu_{\mathbf{Y}} - \Sigma_{\mathbf{Y},\mathbf{Z}} \Sigma_{\mathbf{Z},\mathbf{Z}}^{-1} (\mathbf{z} - \mu_{\mathbf{Z}})$ and covariance $\Sigma_{\mathbf{Y},\mathbf{Y}} - \Sigma_{\mathbf{Y},\mathbf{Z}} \Sigma_{\mathbf{Z},\mathbf{Z}}^{-1} \Sigma_{\mathbf{Z},\mathbf{Y}}$.

Example 2.1.6. *Consider the relation*

$$Y = g(X, Z) + \varepsilon \tag{2.2}$$

and assume that g is continuously differentiable, ε is independent from X, Z with density f_ε and X, Z has the density f_{XZ}. Denote the joint density of (X, Z, ε) by $f(x, z, \varepsilon) = f_{XZ}(x, z)f_\varepsilon(\varepsilon)$. The function $H : \mathbb{R}^3 \to \mathbb{R}^3$ with $(x, z, \varepsilon) \mapsto (x, z, \varepsilon - g(x, z))$ is a diffeomorphism with the absolute value of the determinant of the Jacobi matrix constant to 1. Let $A = A_1 \times A_2 \times A_3$ belong to a family of subsets of $\mathbb{R}^3$, which is closed under finite intersections and generates $\mathcal{B}(\mathbb{R}^3)$. We have the following.

$$\begin{aligned}
P\left((X, Z, Y) \in A\right) &= \mathbb{P}\left(X \in A_1, Z \in A_2, \varepsilon + g(X, Z) \in A_3\right) \\
&= \mathbb{P}\left((X, Z, \varepsilon) \in H(A)\right) \\
&= \int_{H(A)} f(x, z, \varepsilon) dx dz d\varepsilon \\
&= \int_A f\left(H(x, z, \varepsilon)\right) dx dz d\varepsilon \\
&= \int_A f(x, z, \varepsilon - g(x, z)) dx dz d\varepsilon \\
&= \int_A f_{XZ}(x, z) f_\varepsilon\left(y - g(x, z)\right) dx dz dy.
\end{aligned}$$

Here, in the fourth equality we use the formula for changing variables and the fact that the absolute value of the determinant of the derivative of H is 1 and in the last equality the independence between X, Z and ε. As A was arbitrarily chosen, it follows for the joint density $f(x, z, y) = f_{XZ}(x, z)f_\varepsilon\left(y - g(x, z)\right)$ and thus $f(y|x, z) = f_\varepsilon\left(y - g(x, z)\right)$.

All statements are straightforward to generalize for multivariate $\mathbf{X}, \mathbf{Z}$ and ε. Furthermore, for additional restrictions on g, it can be generalized to $Y = g(X, Z, \varepsilon)$.

2.1.2 Conditional Independence

Using the conditional densities one can define conditional independence, which is a key concept in PGMs. See Dawid (1979) for further details.

Definition 2.1.7. *Let $\mathbf{X}, \mathbf{Y}, \mathbf{Z}$ be random vectors of dimension q, r, s with joint density*

f. *We say,* $\mathbf{X}$ *is* ***independent from*** $\mathbf{Y}$ ***given*** $\mathbf{Z}$ *denoted by*

$$\mathbf{X} \perp \mathbf{Y} | \mathbf{Z}$$

if the conditional density decomposes into

$$f(\mathbf{x}, \mathbf{y}|\mathbf{z}) = f(\mathbf{x}|\mathbf{z}) f(\mathbf{y}|\mathbf{z})$$

for all $\mathbf{x} \in \mathbb{R}^q, \mathbf{y} \in \mathbb{R}^r, \mathbf{z} \in \mathbb{R}^s$.

Proposition 2.1.8 (Properties of Conditional Independence)**.** *If* $\mathbf{X} \perp \mathbf{Y} | \mathbf{Z}$*, it holds that*

- $\mathbf{Y} \perp \mathbf{X} | \mathbf{Z}$*,*
- $f(\mathbf{x}, |\mathbf{y}, \mathbf{z}) = f(\mathbf{x}|\mathbf{z})$*,*
- $f(\mathbf{x}|\mathbf{y}, \mathbf{z}) = a(\mathbf{x}, \mathbf{z})$ *for some function* a*,*
- $g(\mathbf{X}) \perp \mathbf{Y} | \mathbf{Z}$ *for any measurable function* g*, and*
- $\mathbf{X} \perp \mathbf{Y} | h(\mathbf{Z})$ *for any diffeomorphism* h*.*

The last statement follows from a change of variables argument. For the other claims, see Dawid (1979).

Example 2.1.9. *Consider the normal distribution of Example 2.1.5 and let* $K = \Sigma^{-1}$ *be the symmetric precision matrix. The joint density can be written as*

$$f(\mathbf{x}) = C \exp\left(-\frac{1}{2}\mathbf{x}^\top K \mathbf{x}\right) = C \exp\left(-\frac{1}{2}\sum_{j,k=1}^{p} x_j x_k K_{j,k}\right), \tag{2.3}$$

where C *does not depend on* $\mathbf{x}$*. Clearly, iff* $K_{p,p-1} = 0$*, then*

$$f(\mathbf{x}) = C \exp\left(-\frac{1}{2}\mathbf{x}_{[p-1]}^\top K_{[p-1],[p-1]} \mathbf{x}_{[p-1]}\right) \\ \exp\left(-\frac{1}{2}\mathbf{x}_{[p]\setminus\{p-1\}}^\top K_{[p]\setminus\{p-1\},[p]\setminus\{p-1\}} \mathbf{x}_{[p]\setminus\{p-1\}}\right).$$

The term on the r.h.s. is proportional to the product $f(\mathbf{x}_{[p-2]}, x_{p-1}) f(\mathbf{x}_{[p]\setminus\{p-1\}})$*. Both*

terms constitute a density. Thus, for a $C' > 0$ it holds

$$\begin{aligned} f(x_p, x_{p-1}|\mathbf{x}_{[p-2]}) &= \frac{f(\mathbf{x})}{f(\mathbf{x}_{[p-2]})} \\ &= C' \frac{f(x_{p-1}, \mathbf{x}_{[p-2]}) f(\mathbf{x}_{[p]\setminus\{p-1\}})}{f(\mathbf{x}_{[p-2]})} \\ &= C' f(x_{p-1}|\mathbf{x}_{[p-2]}) f(\mathbf{x}_{[p]\setminus\{p-1\}}) \\ &= C' f(x_{p-1}|\mathbf{x}_{[p-2]}) f(x_p|\mathbf{x}_{[p-2]}) f(\mathbf{x}_{[p-2]}). \end{aligned}$$

Integrating the two sides with respect to x_{p-1}, x_p, we obtain $C' f(\mathbf{x}_{[p-2]}) = 1$ from which follows that

$$f(\mathbf{x}_{p-1}, \mathbf{x}_p|\mathbf{x}_{[p-2]}) = f(\mathbf{x}_{p-1}|\mathbf{x}_{[p-2]}) f(\mathbf{x}_p|\mathbf{x}_{[p-2]}).$$

Here, p and $p-1$ were chosen for convenience, and the same statement holds for any $j, k \in [p]$. Thus, the conditional independencies of two random components in $\mathbf{X}$ can be read off from the vanishing entries in the precision matrix.

Example 2.1.10. *Let us revisit Example 2.1.6, where $Y = g(X, Z) + \varepsilon$. If the regression function does not depend on Z, that is, $Y = h(X) + \varepsilon$, then applying the same steps as above, it follows for the joint distribution $f(y|x, z) = f_\varepsilon(y - h(x))$, which does not depend on z. From Proposition 2.1.8 it follows that $Y \perp Z|X$.*

Hence, under the relation (2.2)*, one can derive the conditional independencies from feature selection. That is, if Y is regressed on X and the regression function does not change by adding Z to the regressors, then Y and Z are independent given X.*

2.2 Copulas

This thesis investigates graphical representations of multivariate dependency structures. Copulas allow one to separate the marginal distributions from the dependency structure. Thorough introductions to copulas are given in Nelsen (2006); Joe (2014).

Definition 2.2.1 (Copula)**.** *We call a distribution function C with support $[0, 1]^p$ a p-dimensional copula if the marginal distribution functions of C are uniform.*

The set of p-dimensional copulas can describe any dependency structure, as shown in Sklar's theorem.

Theorem 2.2.2 (Sklar 1959)**.** *For a p-dimensional distribution function F with marginals $F_1, \dots, F_p$ there exists a copula C, such that*

$$F(x_1, \dots, x_p) = C\left(F_1(x_1), \dots, F_p(x_p)\right). \tag{2.4}$$

Furthermore, if $F_1, \dots, F_p$ are continuous, then C is unique.

In the following, we assume that the marginal distribution functions $F_1, \dots, F_p$ are continuous. Different joint distribution functions share the same copula C.

Example 2.2.3. *Let $\mathbf{X}$ follow a multivariate normal distribution $\mathbf{\Phi}_{\mu,\Sigma}$ with mean μ and covariance Σ. Denote by Φ_{m,s^2} the distribution function of the univariate normal distribution with mean m and variance s^2. Then by Theorem 2.2.2 it holds that*

$$\mathbf{\Phi}_{\mu,\Sigma}(x_1, \dots, x_p) = C_R\left(\Phi_{\mu_1,\Sigma_{11}}(x_1), \dots, \Phi_{\mu_p,\Sigma_{pp}}(x_p)\right).$$

*Here, C_R is called a **Gaussian copula** indexed by the correlation matrix R of $\mathbf{X}$. It is given by*

$$C_R\left(u_1, \dots, u_p\right) = \mathbf{\Phi}_R(\Phi^{-1}(u_1), \dots, \Phi^{-1}(u_p)),$$

where $\mathbf{\Phi}$ is the cumulative distribution function (cdf) of the multivariate normal distribution with mean 0 and covariance matrix R.

The class of distributions whose copula is Gaussian goes beyond the multivariate normal distributions, as one can replace the marginal distributions $\Phi_{\mu_k,\Sigma_{kk}}$ by any other continuous distribution function. As these distributions share the dependency structure with multivariate normal distributions, the conditional independencies can also be read off from the inverse of R.

Example 2.2.4. *Let the copula of F be Gaussian with the correlation matrix R where the marginal cdfs and their inverses are differentiable. Applying $x_k \mapsto \Phi^{-1} \circ F_k$ componentwise is a diffeomorphism, and the transformed random vector*

$$\mathbf{Z} = \left(\Phi^{-1}(F_1(X_1)), \dots, \Phi^{-1}(F_p(X_p))\right)$$

is Gaussian. It follows that the conditional independencies can be read off from the inverse of R by Proposition 2.1.8 and Example 2.1.9.

It is easy to sample from a multivariate distribution if sampling from its copula is feasible.

Remark 2.2.5. *Let F be a multivariate distribution with copula C and marginals $F_1, \ldots, F_p$. For the random vector $(U_1, \ldots, U_p)$ that follows the copula distribution C, it holds $\left(F_1^{-1}(U_1), \ldots, F_p^{-1}(U_p)\right) = F$. That is, data from the joint distribution F can be generated by*

1. *sampling $(u_1, \ldots, u_p)$ from C and*
2. *map u_j component-wise by $F_j^{-1}(u_j), j = 1, \ldots, p$.*

Here, F_j^{-1} is understood as the general inverse of $F_j, j = 1, \ldots, p$.

Estimating the Joint Distribution with Copulas

For large p, the non-parametric estimation of the joint distribution becomes difficult due to the curse of dimensionality (Scott and Sain, 2005). Thus, the parametric estimation becomes attractive in order to capture some aspects of the multivariate distribution. For example, assuming a multivariate normal distribution, the estimation captures the first two moments of the joint distribution.

On the other hand, Equation (2.4) suggests another approach. The marginal distribution functions $F_1, \ldots, F_p$ can be efficiently estimated by the empirical marginal distribution functions $\widehat{F_1}, \ldots, \widehat{F_p}$ (Van Der Vaart and Wellner, 1996). If the copula is assumed to lie in a parametric family of copulas $\{C_\theta : \theta \in \Theta\}$, then Genest et al. (1995) show that the copula parameter θ can be consistently estimated via maximum-likelihood estimation based on the transformed data

$$\left(\widehat{F_1}(x_{\ell 1}), \ldots, \widehat{F_p}(x_{\ell p})\right), \ell = 1, \ldots, N.$$

This two-step approach greatly increases the space of describable distributions.

Assuming that the copula is Gaussian, the second step in the two-step approach corresponds to the estimation of the rank correlations instead of covariances when a multivariate normal distribution is presumed. This robustifies the estimation procedure (Liu et al., 2012).

Unfortunately, if data is missing, then the two-step procedure is not guaranteed to be consistent. This is due to the fact that the empirical marginal distribution functions $\widehat{F_1}, \ldots, \widehat{F_p}$ based on the observed values do not necessarily converge to the true marginal distribution functions $F_1, \ldots, F_p$. Fixing the marginals can solve this problem. For example, if we assume the marginals to be normally distributed, then the joint distribution is multivariate normal and the parameters can be estimated by the

Expectation-Maximization (EM) algorithm (Dempster et al., 1977). However, this removes the flexibility of the copula model. In Section 3.1 we propose an EM algorithm in the case of missing data, that keeps the marginal distributions flexible.

2.3 Graphs

This work investigates the application of PGMs to multivariate manufacturing data. PGMs represent conditional independencies between the variables using **graphs** G, which are tuples (V, E) consisting of a set of **nodes** $V = \{v_1, \dots, v_p\}$ and a set of **edges** E. Depending on the type of elements in E, we call a graph undirected or directed. The different types of graph encode different conditional independencies. In Section 2.4 we investigate PGMs with undirected graphs, and in Section 2.5 we examine PGMs with directed graphs. For an introduction to graphs in the context of PGMs, consult Koller and Friedman (2009, Chapter 2.2).

If two graphs $G_1 = (V, E_1)$ and $G_2 = (V, E_2)$ contain the same nodes V and if $E_1 \subset E_2$, then we say that G_1 is a **subgraph** of G_2.

2.3.1 Undirected Graphs

A graph is called **undirected** if the set of edges consists of sets of two elements, that is, $E \subset \{\{v_1, v_2\} : v_1, v_2 \in V\}$. Two nodes v_1, v_2 are called **neighbors**, if $\{v_1, v_2\} \in E$. We denote the neighbors of v by $N_G(v)$. A **path** from v_{k_1} to v_{k_r} is a collection of edges

$$\{v_{k_1}, v_{k_2}\}, \{v_{k_2}, v_{k_3}\}, \dots, \{v_{k_{r-1}}, v_{k_r}\}$$

where $\{v_{k_{j-1}}, v_{k_j}\} \in E, j = 2, \dots, r$. The set $S \subset V$ **separates** v_j and v_k if any path between v_j and v_k passes through a node in S. If there is no path between the nodes, then any set S separates v_j and v_k by convention. The concepts are depicted in Figure 2.1.

2.3.2 Directed Acyclic Graphs (DAGs)

For a **directed** graph, the edges are ordered tuples, that is $E \subset \{(v_1, v_2) : v_1, v_2 \in V\}$. The **parents** of a node v in a graph G denoted by $pa_G(v)$ are defined as those nodes, which have an edge going to v, so $pa_G(v) = \{w : (w, v) \in E\}$. We further call $(v_{k_1}, v_{k_2}), (v_{k_2}, v_{k_3}), \dots, (v_{k_{r-1}}, v_{k_r})$ a **directed path** from v_{k_1} to v_{k_r} if $(v_{k_{j-1}}, v_{k_j}) \in E$

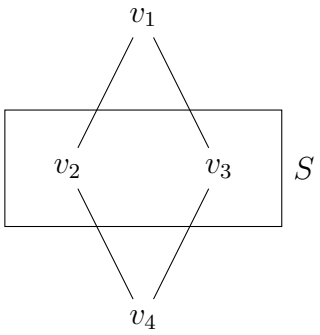

Figure 2.1: Undirected graph G with nodes $\{v_1, v_2, v_3, v_4\}$. The neighbors of v_2 are $N_G(v_2) = \{v_1, v_4\}$. The set S separates v_1 and v_4 as the two paths from v_1 to v_4 are $\{v_1, v_2\}, \{v_2, v_4\}$ and $\{v_1, v_3\}, \{v_3, v_4\}$.

for $j = 2, \ldots, r$. We call the set $desc_G(v) = \{w : \exists \text{ directed path from } v \text{ to } w\}$ the **descendents** of v_k. A path between two nodes is a path in the undirected version of G. Similarly, we call v_1 and v_2 neighbors if the edge $v_1 - v_2$ exists in the undirected version of G.

To slim down the notation, we introduce $pa_G(k) = \{j : v_j \in pa_G(v_k)\}, k = 1, \ldots, p$ which contains the indices of the parents of v_k, and $desc_G(k)$ and $N_G(k)$ are defined analogously.

A **Directed Acyclic Graph (DAG)** is a directed graph that contains no cycles, that is, there is no directed path from a node to itself. Three nodes (v_i, v_j, v_k) connected by $v_i \to v_j \leftarrow v_k$ and without an edge between v_i and v_k are called a **v-structure**. We say v_j and v_k are **d-separated** given $S \subset V$, denoted by $v_j \perp v_k|_G S$, if one of the following two conditions holds:

1. Any undirected path between v_j and v_k that has no v-structure goes through a node in S, and
2. For any undirected path between v_j and v_k that has a v-structure (v_q, v_r, v_t), neither the middle element v_r nor any of its descendants is in S.

A **topological ordering** for a DAG G is a permutation π on $(1, \ldots, p)$ such that for any $k = 1, \ldots, p$ it holds that $pa_G(k) \subset \{j : \pi(j) < \pi(k)\} =: \varpi_\pi(k)$. Such a topological ordering always exists although it is not necessarily unique. The set of topological orderings of G is denoted by $\Pi(G)$. The concepts are depicted in Figure 2.2. Consequently, for a topological ordering π there can only be a directed path from v_j to v_k if we have $j \in \varpi_\pi(k)$, that is, descendents appear later in the topological ordering.

In the following, we consider graphs with the vertices being the random variables

Figure 2.2: Two directed graphs, where G_1 contains the cycle $v_1 \to v_2 \to v_3 \to v_1$, while G_2 is a DAG. For G_2, the parents of v_2 are $pa_{G_2}(v_2) = \{v_1\}$. The descendants of v_1 are v_2, v_3, v_4, since there is a directed path to each of them. A directed path from v_1 to v_4 is highlighted in red. The set $S = \{v_1, v_4\}$ does not d-separate v_2 and v_3 as the undirected path $\{v_1, v_4\}, \{v_4, v_3\}$ contains a v-structure with v_4 in the middle and $v_4 \in S$. On the contrary, $S = \{v_1\}$ d-separates v_2 and v_3. The set of topological orderings is $\Pi(G_2) = \{(1, 2, 3, 4), (1, 3, 2, 4)\}$. The undirected version of G_2 is the graph in Figure 2.1.

$X_1, \ldots, X_p$. We can now clearly define the kind of relationships between $X_1, \ldots, X_p$ that PGMs embody.

2.4 Markov Random Fields

In this section, we describe the dependency structure of distributions using an undirected graph G. The conditional independencies encoded in undirected graphs are often simpler than those encoded in directed graphs. Still, the resulting graph depicts interesting aspects of the joint distribution and can be used for efficient inference. For further details, see Koller and Friedman (2009, Chapter 4).

For the random vector $\mathbf{X}$, we say that it fulfills the **global Markov property** given G if for any X_j and X_k and any $\mathbf{X}_S$ that separates X_j and X_k we have

$$X_j \perp X_k | \mathbf{X}_S.$$

In that case, we call the combination $(\mathbf{X}, G)$ a **Markov Random Field (MRF)**. On the contrary, $\mathbf{X}$ fulfills the **pairwise Markov property** w.r.t. G, if for any non-neighboring X_j, X_k it holds that

$$X_j \perp X_k | \mathbf{X}_{[p] \setminus \{j,k\}}.$$

The global Markov property implies the pairwise Markov property. The opposite direction holds if the distribution of $\mathbf{X}$ has a strictly positive density (Koller and Friedman, 2009, Theorem 4.4). Together with Example 2.1.9 this implies the following proposition.

Proposition 2.4.1. *Let* $\mathbf{X}$ *be normally distributed with precision matrix* K*. If for every pair* X_j, X_k *in* G *we have* $X_k \notin N_G(X_j) \Rightarrow K_{jk} = 0$*, then* $(\mathbf{X}, G)$ *is an MRF.*

Therefore, the derivation of the undirected graph G from the data reduces to finding the vanishing entries in the precision matrix. This is used by the graphical lasso algorithm proposed by Friedman et al. (2007). Its goal is to maximize the penalized log-likelihood with respect to the mean vector $\boldsymbol{\mu}$ and the precision matrix K, which is proportional to

$$L(\mathbf{x}^N; K, \boldsymbol{\mu}) = \log(det(K)) - \sum_{\ell=1}^{N} (\mathbf{x}_\ell - \boldsymbol{\mu})^\top K (\mathbf{x}_\ell - \boldsymbol{\mu}) - \lambda ||K||_1.$$

Here, $\lambda \geq 0$ is a hyperparameter and $||\cdot||_1$ is the 1-norm of a matrix penalizing non-zero entries similar to the LASSO. The sample mean of $\mathbf{x}^N$ maximizes L with respect to $\boldsymbol{\mu}$. Thus, for the sample covariance matrix of $\mathbf{x}^N$ denoted by S we obtain the expression

$$L(\mathbf{x}^N; K) = \log(det(K)) - tr(SK) - \lambda ||K||_1.$$

The optimization with respect to K is then efficiently solved by the graphical lasso. An increase in λ leads to sparser graphs.

For a distribution with a Gaussian copula, the pairwise Markov property is fulfilled if for any non-neighbors X_j and X_k, the corresponding entry of the precision matrix of the Gaussian copula is 0. The estimation of the Gaussian copula precision matrix is investigated by Liu et al. (2009). Applying a two-step approach, they start by estimating the marginal distributions by $\widehat{F}_1, \ldots, \widehat{F}_p$. Then, they maximize

$$\log(det(K)) - tr(\widetilde{S}K) - \lambda ||K||_1,$$

with respect to K using the graphical lasso. Here, $\widetilde{S}$ is the sample covariance matrix of the transformation

$$\left(\Phi^{-1} \circ \widehat{F}_1(x_{\ell 1}), \ldots, \Phi^{-1} \circ \widehat{F}_p(x_{\ell p})\right), \ell = 1, \ldots, N.$$

If the marginal distributions have strictly positive densities, then the joint distribution is also strictly positive. Then $(\mathbf{X}, G)$ is an MRF if for any non-neighboring X_j and X_k, it holds $K_{jk} = 0$.

2.5 Directed Acyclic Graphical Models

In the following we combine DAGs with random vectors $\mathbf{X}$ to derive another kind of graphical characterization of the distribution. We provide two perspectives on this, namely Bayesian Networks and Structural Equation Models. Eventually, Proposition 2.5.8 unifies both approaches and shows that they are equivalent.

Further references for Bayesian Networks and Structural Equation Models are Koller and Friedman (2009, Chapter 4) and Peters et al. (2017), respectively.

2.5.1 Bayesian Networks

Definition 2.5.1 (Bayesian Network)**.** *We call the tuple $(\mathbf{X}, G)$ consisting of a random vector and a DAG a **Bayesian Network (BN)** if the **Markov property** holds, that is,*

$$X_k \perp X_j |_G \mathbf{X}_S \Rightarrow X_k \perp X_j | \mathbf{X}_S.$$

Hence, if X_k and X_j are d-separated with respect to $\mathbf{X}_S$ in the graph G, then this implies the conditional independence of X_k and X_j given $\mathbf{X}_S$.

Example 2.5.2. *A complete DAG G, that is, all pairs of nodes are neighbors in G, is a BN for any random vector $\mathbf{X}$. This is due to the fact that for any pair of nodes X_j, X_k, there is no $\mathbf{X}_S$ that d-separates these nodes. Thus, G does not contain information on $\mathbf{X}$. Intuitively, graphs with fewer edges, that is, sparse graphs, contain more information on $\mathbf{X}$.*

We must possibly check a large number of conditional independencies to verify $(\mathbf{X}, G)$ is a BN. The following proposition shows that we need to investigate only a subset of conditional independencies.

Proposition 2.5.3 (Equivalence of Markov properties)**.** *If the distribution of $\mathbf{X}$ is absolutely continuous, that is, it has a density p, then the*

1. *Markov property,*
2. *local Markov property, that is, $X_k \perp X_{[p]\setminus(desc_G(k)\cup\{k\})} | \mathbf{X}_{pa_G(k)}$ for $k = 1, \ldots, p$, and*
3. *factorization w.r.t. G, that is, $p(\mathbf{x}) = \prod_{k=1}^{p} p(x_k | \mathbf{x}_{pa_G(k)})$*

are equivalent. Here, $p(x_k|\mathbf{x}_{pa_G(k)})$ *is the conditional density of* X_k *given the event* $\mathbf{X}_{pa_G(k)} = \mathbf{x}_{pa_G(k)}$*. If* $pa_G(k) = \emptyset$*, then* $p(x_k|\mathbf{x}_{pa_G(k)}) = p(x_k|\mathbf{x}_\emptyset) := p(x_k)$*, which is the marginal density of* X_k*.*

Proof. It is easy to see that the Markov property implies the local Markov property, which implies the factorization property. This is $1. \Rightarrow 2. \Rightarrow 3.$. For the remainder of the proof, see Lauritzen (1996, Theorem 3.27). □

Observe that it holds for any permutation π

$$p(\mathbf{x}) = \prod_{k=1}^{p} p(x_k|\mathbf{x}_{\varpi_\pi(k)}).$$

Of course, this formula holds also for a topological order of G. Hence, the factorization w.r.t. G is a simplification of the underlying joint distribution in the sense that the conditioning events are reduced.

Further, the implication $3. \Rightarrow 1.$ is somewhat surprising. The question arises whether all distributions whose density factorizes w.r.t. G, share more conditional independencies than those encoded in the d-separations of G. This can be denied, so that d-separation is the maximal graphical criterion for conditional independencies.

The type of conditional independencies in BNs is more intricate compared to MRFs. However, Proposition 2.5.3 shows that one can sample from a BN if the conditional distributions of any node given its parents are known. The conditional distributions are often easier to estimate or can be estimated in a more flexible way. Furthermore, if a BN and the conditional distributions are known, then inference on queries of the form $p(x_k|\mathbf{x}_S)$ is possible, which was one of the main motivations in the early days of BNs.

However, we still face the challenge that the graph in BNs is not necessarily unique. Consider a BN $(\mathbf{X}, G)$ and a graph G' that entails the same set of d-separations as G. That is, they contain the same undirected paths and v-structures. It is apparent from Definition 2.5.1, that this implies that $(\mathbf{X}, G')$ is also a BN. However, it can still be that $G' \neq G$ as the set of undirected edges and v-structures does not determine a DAG. Thus, the graph is not unique.

The second issue we need to address is indicated in Example 2.5.2. More generally, if one adds an edge to G resulting in graph G'', and that edge does not create a new v-structure, then G'' contains fewer d-separations than G. Consequently, it meets Definition 2.5.1. Such graphs with superfluous edges contain less information on $\mathbf{X}$. The following definition solves this issue.

Definition 2.5.4 (Minimal BN)**.** *Let $(\mathbf{X}, G)$ be a BN. G fulfills the minimality w.r.t. $\mathbf{X}$ if for any subgraph G' of G the Markov property does not hold, so $(\mathbf{X}, G')$ is not a BN.*

A BN $(\mathbf{X}, G)$ allows us to identify conditional independencies using the d-separation criterion. However, if X_j and X_k are not d-separated by $\mathbf{X}_S$ in G, this does not imply that they are conditionally dependent. Thus, we prefer graphs G where every conditional independence of $\mathbf{X}$ can be read off from G, as G would then contain the maximum information on $\mathbf{X}$. This intuition is captured in the concept of faithfulness.

Definition 2.5.5 (Faithfulness)**.** *For a BN $(\mathbf{X}, G)$, we say that $\mathbf{X}$ is **faithful w.r.t.** G if for any $X_j, X_k, \mathbf{X}_S$ we have*

$$X_j \perp X_k | \mathbf{X}_S \Rightarrow X_j \perp X_k |_G \mathbf{X}_S.$$

Hence, every (conditional) independence in $P(\mathbf{X})$ can be read off from the graph G. More generally, we say that $\mathbf{X}$ is faithful if there is a DAG G to which $\mathbf{X}$ is faithful.

We give a counterexample in the following.

Example 2.5.6. *Consider a distribution on (X_1, X_2, X_3) with the only (conditional) independencies being $X_1 \perp X_3$ and $X_1 \perp X_3 | X_2$. Such a distribution exists (Spirtes et al., 1993, Section 3.5.2) but there exists no DAG entailing this exact set of independencies.*

In the context of directed PGMs, the topological ordering of a DAG G is called the **causal order**.

2.5.2 Structural Equation Models

Structural Equation Models (SEMs) model the joint distribution $P(\mathbf{X})$ by the relationships between the random variables $X_1, \ldots, X_p$ and are among the most common causal models used, for example, in social sciences (Pearl, 2010; Peters et al., 2017; Kline, 2023).

Definition 2.5.7 (Structural Equation Model, SEM)**.** *Let G be a DAG and let $N_1, \ldots, N_p$ be jointly independent noise variables. The set of structural equations (SE) of the form*

$$X_k = f_k\left(\mathbf{X}_{pa_G(k)}, N_k\right), k = 1, \ldots, p$$

defines an SEM on $\mathbf{X}$ *with graph* G*. The SEs describe the conditional distribution of* X_k *given the non-descendants of* X_k *in* G*. That is, the distribution of* X_k *given* $\mathbf{X}_{[p]\setminus\{k\}\cup desc_G(k)} = \mathbf{x}_{[p]\setminus\{k\}\cup desc_G(k)}$ *is* $f_k\left(\mathbf{x}_{pa_G(k)}, N_k\right)$*, which is a transformation of the random variable* N_k*.*

The functional relationships between the variables in $\mathbf{X}$ seem to be restrictive compared to BNs, which models $\mathbf{X}$ by general conditional distributions, as can be seen from the factorization property in Proposition 2.5.3. However, the following proposition shows that the set of implied distributions is congruent.

Proposition 2.5.8 (Peters et al. 2017, Proposition 6.31 and Proposition 7.1)**.** *Let the distribution of* $\mathbf{X}$ *have a density with respect to the Lebesgue measure and* $(\mathbf{X}, G)$ *be a BN. Then there exists an SEM with graph* G *that induces the distribution of* $\mathbf{X}$*.*

On the contrary, if $\mathbf{X}$ *is induced by an SEM with graph* G*, then* $(\mathbf{X}, G)$ *is a BN.*

In the following, we provide an intuitive algorithm to sample from $P(\mathbf{X})$.

Remark 2.5.9. *A SEM provides a natural algorithm to draw samples from the joint distribution. Starting with the nodes without parents, one sets* $x_k = f_k(n_k)$*, where* n_k *is a realization of the noise distribution of* N_k*. Then one iterates through the graph following the causal order and assigns* $x_k = f_k(\mathbf{x}_{\mathbf{pa}_G(k)}, n_k)$*, where* $\mathbf{x}_{\mathbf{pa}_G(k)}$ *is already set and* n_k *is again a random draw from the noise distribution* N_k*.*

On the other hand, the following remark emphasizes that the intuitive sampling scheme is not the only possible data-generating process.

Remark 2.5.10. *There are other methods to generate the joint distribution of* $\mathbf{X}$*. For example, the joint distribution has a copula* C *by Sklar's theorem (Theorem 2.2.2). In this case, data following the joint distribution can be generated using the algorithm described in Remark 2.2.5. For this procedure, the underlying DAG is not involved.*

2.5.3 Learning DAGs

In this subsection, we discuss the derivation of the graph from $P(\mathbf{X})$ or from realizations thereof, which is called **causal discovery**. Under the assumption that $P(\mathbf{X})$ has a density, Proposition 2.5.8 shows that we can understand the tuple $(\mathbf{X}, G)$ as a BN and as a SEM. However, both definitions suggest different approaches to find G. For BNs, it seems

natural to use the Markov properties and to identify the conditional independencies. These methods are called **constrained-based**. In contrast, the definition of SEMs suggests identifying the nodes that contribute in the functional relationships. These methods are called **score-based**. In addition, there exist algorithms that combine both approaches as Tsamardinos et al. (2006); Ogarrio et al. (2016).

Constraint-based methods

Arguably the most established algorithm for causal discovery is the PC algorithm, named after its inventors Peter Spirtes and Clark Glymour. Starting from an undirected, complete graph, it gradually removes edges using the results of conditional independence tests. After the edge removal procedure, some of the edges can be oriented using the rules of Meek (1995).

As different DAGs entail the same set of conditional independencies, constraint-based methods can only search for a set of DAGs, which can be described as an equivalence class. All graphs in this equivalence class align in their undirected versions and v-structures. However, the direction of the edges of other edges can differ. It is emphasized that this problem is intrinsic to constraint-based methods and BNs and does not disappear even if the complete distribution $P(\mathbf{X})$ is known.

Furthermore, if $\mathbf{X}$ is faithful to G, then the PC algorithm is consistent (Spirtes et al., 1993). However, this consistency is not even uniform for all multivariate normal distributions that are faithful to G as shown by Robins et al. (2003). The uniform consistency for multivariate normal distributions can be shown by assuming a stronger version of faithfulness called λ-strong faithfulness. Kalisch and Bühlman (2007) show that their version of the PC algorithm is consistent even if p grows with N, assuming λ-strong faithfulness and that the maximal number of neighbors for a node in the graph G is limited. Unfortunately, Uhler et al. (2013) find that the set of distributions where λ-strong faithfulness holds is small.

In addition to these issues, the PC algorithm is highly dependent on the underlying conditional independence test. In the version of Kalisch and Bühlman (2007) this test is performed p^q times, where $q \in \mathbb{N}$ is the maximal number of neighbors of a node, which must be set as a hyperparameter and which in the worst case is $p - 1$. While testing for conditional independence in multivariate normal and multinomial distributions is straightforward, it becomes difficult for general distributions, as shown by Shah and Peters (2018). Nonparametric conditional independence tests were proposed, among others, by Gretton et al. (2007); Zhang et al. (2011); Strobl et al. (2019); Bellot and

van der Schaar (2019), but their computational complexity for each test grows at least quadratically in sample size. This is prohibitive for the applications that we have in mind.

Consequently, the vast majority of applications of the PC-algorithm implicitly assume that the data follows a multivariate normal distribution or it discretizes the data and applies conditional independence tests for the multinomial distribution. The first is likely a simplification with unknown consequences, while the latter loses information contained in the data and relies on the discretization procedure.

The incorporation of expert knowledge on conditional (in)dependencies and on the ordering of the nodes can be easily incorporated and is implemented in the R-package `pcalg` described in Kalisch et al. (2012).

Score-based methods

On the contrary, score-based method try to find the graph $\widehat{G}$ that minimizes the score function

$$S\left(G, (\mathbf{x}_1, \ldots, \mathbf{x}_N)\right),$$

where S is derived from Definition 2.5.7. However, Definition 2.5.7 is too general to find reasonable score functions S. Thus, we restrict the SEs and the noise terms. In the following, we consider Additive Noise Models.

Definition 2.5.11 (Additive Noise Model, ANM)**.** *In Definition 2.5.7 set the SEs to*

$$X_k = g_k(\mathbf{X}_{pa_G(k)}, N_k) = f_k(\mathbf{X}_{pa_G(k)}) + N_k, k = 1, \ldots, p.$$

*The resulting SEM is called an **Additive Noise Model (ANM)** with graph G.*

In Definition 2.5.11 we can clutter the input arguments of any f_k with additional input arguments that do not change f_k. We restrict ourselves to SEs, where every argument influences f_k.

Definition 2.5.12 (Minimality of ANMs)**.** *For ANMs with SEs $f_1, \ldots, f_p$ and graph G we assume that the ANM is **minimal**, that is, for any $k \in [p]$ and any $j \in pa_G(k)$ there exist two values $x_j \neq x_j'$ so that*

$$f_k(\mathbf{x}_{pa_G(k)\setminus\{j\}}, x_j) \neq f_k(\mathbf{x}_{pa_G(k)\setminus\{j\}}, x_j').$$

The following result states that the minimality for ANMs of Definition 2.5.12 is equivalent to the minimality for BNs of Definition 2.5.4. However, recall that other graphs constitute together with $\mathbf{X}$ a BN.

Proposition 2.5.13 (Peters et al. 2017, Proposition 7.4)**.** *Let $\mathbf{X}$ be generated by an ANM with graph G. Then $(\mathbf{X}, G)$ is a BN by Proposition 2.5.8. The ANM is minimal if and only if $(\mathbf{X}, G)$ satisfies the minimality of BNs in the sense of Definition 2.5.4.*

Remark 2.5.14. *Consider a joint density p implied by an ANM. By Proposition 2.5.3, Proposition 2.5.8 and Example 2.1.6 it holds*

$$p(\mathbf{x}) = \prod_{k=1}^{p} p(x_k|\mathbf{x}_{pa_G(k)}) = \prod_{k=1}^{p} p_{N_k}(x_k - f_k(\mathbf{x}_{pa_G(k)})),$$

where $p_{N_k}(\cdot)$ is the density of the noise N_k.

If we restrict the SEs and the noise within the ANMs, then we obtain interesting subclasses.

Linear Gaussian Model We begin by introducing a narrow restriction on ANMs, which is the linear Gaussian model. Although it does not meet the requirement that the model shall be able to capture complex relationships between the variables, we will see that several concepts can be transferred to non-linear models. However, for linear Gaussian models, these concepts can be more accessible.

Definition 2.5.15 (Linear Gaussian model)**.** *A linear Gaussian model is an ANM, where the noise $N_1, \ldots, N_p$ is Gaussian and the functions f_k are linear, so that for $\beta_{kj} \in \mathbb{R}$*

$$f_k(\mathbf{x}_{pa_G(k)}) = \sum_{j \in pa_G(k)} \beta_{kj} x_j, k = 1, \ldots, p.$$

A linear Gaussian model is minimal, if $\beta_{kj} \neq 0$ for any $k \in [p]$ and $j \in pa_G(k)$. Linear Gaussian models imply a multivariate normal distribution, which is the underlying assumption for most constraint-based methods.

Example 2.5.16. *Consider a linear Gaussian model for $p = 2$ and a DAG G of the form $X_1 \to X_2$. Let N_1, N_2 be independent standard normal distributions, and let $\beta_{21} \neq 0$.*

Then, the joint distribution of $\mathbf{X} = (X_1, X_2)$ *is multivariate normal with mean* 0 *and covariance matrix*

$$\Sigma = \begin{pmatrix} 1 & \beta_{21} \\ \beta_{21} & 1 + \beta_{21}^2 \end{pmatrix}.$$

On the other hand, the linear Gaussian model with G *set to* $X_2 \to X_1$*, where* $N_2 \sim \mathcal{N}(0, 1 + \beta_{21}^2)$*,* $N_1 \sim \mathcal{N}\left(0, \frac{1}{1+\beta_{21}^2}\right)$ *with parameter* $\beta'_{12} = \frac{\beta_{21}}{1+\beta_{21}^2}$ *leads to the same joint distribution. This observation can be generalized to any* p*.*

Observe that both graphs imply the same set of conditional independencies and thus both graphs constitute a BN together with $\mathbf{X}$*. More generally, every graph within the same equivalence class can lead to exactly the same set of distributions. These are all multivariate normal distributions that contain the conditional independencies implied by the equivalence class.*

Each graph G implies a set of multivariate normal distributions, which can be parameterized by $\theta_G \in \Theta_G$. A reasonable approach to define the score function is to choose $\widehat{\theta}_G$ as the parameter that minimizes the negative log-likelihood within Θ_G. Let $\widehat{\theta}_G$ imply the density $p_{\widehat{\theta}_G}$. Then, one can define the score for a graph G by

$$S\left(G, (\mathbf{x}_1, \ldots, \mathbf{x}_N)\right) = \min_{\theta_G \in \Theta_G} -\sum_{\ell=1}^{N} \log p_{\widehat{\theta}_G}(\mathbf{x}_\ell) + C(G),$$

where C is some complexity measure for graphs. A reasonable choice for C is to penalize the dimension of θ_G by an AIC score as proposed by Haughton (1988). We will propose a similar procedure for more complex models in Section 3.3.

Another string of algorithms applies Bayesian methods. Thereby, we first define a prior on the DAGs q_G, which is a discrete distribution. Then we define for every graph G a prior on the parameters, that is, $q(\theta_G|G)$. Note that together they imply a joint prior over θ_G, G.

Using Bayes law, we search for the maximum-a-posteriori (MAP) estimator, that is the maximizer of

$$\begin{aligned} -\log p(G|\left(\mathbf{x}_1, \ldots, \mathbf{x}_N\right)) &= -\log \int p(G, \theta_G|\left(\mathbf{x}_1, \ldots, \mathbf{x}_N\right)) d\theta_G \\ &\propto -\log \int p((\mathbf{x}_1, \ldots, \mathbf{x}_N)\,|G, \theta_G) q(\theta_G|G) q(G) d\theta_G \\ &= -\log q(G) - \log \int p((\mathbf{x}_1, \ldots, \mathbf{x}_N)\,|G, \theta_G) q(\theta_G|G) d\theta_G. \end{aligned}$$

Choosing the priors appropriately, the integral on the right has a closed solution (Geiger and Heckerman, 1994). The selection of the priors on the parameters can be tedious, in particular as they depend on the graph, but it allows us to incorporate expert knowledge.

The number of DAGs grows super-exponentially with p. Hence, beyond small p it becomes infeasible to calculate the score for all DAGs G. Luckily, the score S often decomposes nicely, so that if G and G' differ by only one edge or edge direction, then $S(G')$ can be quickly calculated from $S(G)$. This property is often used for greedy approaches, where one starts with an empty graph and continues to manipulate the graph with the addition, reversion, or deletion of one edge so that the score improves until no further improvement is found.

More recently, Zheng et al. (2018) proposed a gradient-based estimator for G. It uses a parametrization of all directed graphs using the edge matrix $W \in \mathbb{R}^{p\times p}$, where $W_{kj} = \beta_{kj}, j, k = 1, \ldots, p$ and β_{kj} is as in Definition 2.5.15. The key tool is a differentiable function $h(W)$ that is 0 if and only if W characterizes a DAG. Then they suggest using a score function on matrices in $\mathbb{R}^{p\times p}$ of the form

$$S\left(W, (\mathbf{x}_1, \ldots, \mathbf{x}_N)\right)) = L\left(W, (\mathbf{x}_1, \ldots, \mathbf{x}_N)\right) + \rho h(W)^2 + \lambda ||W||_1.$$

Here, $L\left(W, (\mathbf{x}_1, \ldots, \mathbf{x}_N)\right)$ is proportional to the negative log-likelihood, $||W||_1$ penalizes complex graphs, and $\lambda, \rho > 0$ are hyperparameters. They minimize with respect to W using gradient descent. The corresponding graph to the estimate $\widehat{W}$ is not necessarily a DAG, that is $h(W) \neq 0$, which they solve by thresholding.

It must be emphasized that all procedures are not guaranteed to converge to the global optimum as the search space over the DAGs is non-convex.

Instead of searching for the (equivalence class of the) DAG, Teyssier and Koller (2005) uses the fact that causal discovery drastically simplifies if the causal order π^0 is known. In that case, for node k it is necessary to identify the parents within the set $\varpi_{\pi^0}(k)$. This reduces causal discovery to a repeated feature selection problem that can be tackled for example by variants of the LASSO (Tibshirani, 1996; Zou, 2006), see also Shojaie and Michailidis (2010). Therefore, they search for a causal order of π^0 using the score

$$\widetilde{S}(\pi) = \min_{G:\pi\in\Pi(G)} S\left(G, (\mathbf{x}_1, \ldots, \mathbf{x}_N)\right).$$

Again, S is a score function on the graphs, for example, a Bayesian score. Teyssier and Koller (2005) further employ the method of Friedman et al. (1999), which restricts the search space of DAGs. It initially determines for every node a candidate set for the parents. This procedure results in a likely cyclic directed graph G_{super}. In the following

causal discovery, only DAGs G are considered, so that G is a subgraph of G_{super}. If the true DAG is sparse, this can significantly reduce the search space.

A great advantage of score-based methods is that one can consider models that imply distributions beyond multivariate normality, such as the Causal Additive Model, which we introduce below. In addition to providing a large and complex class of implied distributions, they also have advantageous properties for causal discovery.

Causal Additive Model We have seen that different linear Gaussian models can lead to the same joint distribution. The question arises whether one can restrict ANMs so that for different ANMs the implied distributions differ. We call such models **identifiable**, as they allow us to identify the ANM from the distribution. Shimizu et al. (2006) answers this question affirmatively if one restricts the SEs to linear functions and the noise to a non-Gaussian distribution. In that sense, the linear model with Gaussian noise is exceptional.

Peters et al. (2014) investigates identifiability more generally and derives explicit characterizations of non-identifiable models. Among others, the CAM of Definition 2.5.17 is an identifiable model. In Sections 3.2, 3.3 and 3.4 we investigate CAMs as they not only provide a flexible model, but the SEs can also be interpreted well.

Definition 2.5.17 (Causal Additive Model, CAM)**.** *If $N_1, \dots, N_p$ are centered Gaussians with variances $\sigma_1^2, \dots, \sigma_p^2$ and*

$$f_k(\mathbf{x}_{pa_G(k)}) = \sum_{j \in pa_G(k)} f_{kj}(x_j), k = 1, \dots, p,$$

*where every f_{kj} is three times differentiable and non-linear, then we call the ANM a **Causal Additive Model (CAM)**. A CAM can be characterized by the parameter tuple $(G, f_1, \dots, f_p, \sigma_1^2, \dots, \sigma_p^2)$.*

Theorem 2.5.18 (Peters et al. 2014, Corollary 31)**.** *The CAM is identifiable, which means that for any $\theta = (G, f_1, \dots, f_p, \sigma_1^2, \dots, \sigma_p^2)$ and $\theta' = (G', f_1', \dots, f_p', (\sigma_1')^2, \dots, (\sigma_p')^2)$ with $\theta \neq \theta'$ the distributions implied by θ and θ' differ.*

The identifiability of CAMs seems to contradict earlier results. In the notation of Theorem 2.5.18, consider a specific θ with graph G. This parameter leads to a set of conditional independencies in the implied distribution $P(\mathbf{X})$. Proposition 2.5.8 tells us that $(\mathbf{X}, G)$ is a BN. This BN is not necessarily unique, so we can consider another DAG

$G' \neq G$ for which $(\mathbf{X}, G')$ is still a BN. Applying again Proposition 2.5.8, we obtain an SEM w.r.t. G' that also leads to the same distribution $P(\mathbf{X})$. Thus, two SEMs with two different DAGs imply the same distribution $P(\mathbf{X})$. Why does this not contradict Theorem 2.5.18?

The reason is that the SEM w.r.t. G' that leads to the same distribution $P(\mathbf{X})$ is of the general form of Definition 2.5.7. Quite likely, the SEs corresponding to G' are more complex than those that we allow for CAMs. Therefore, the appropriate reduction in the search space for SEMs guarantees identifiability.

In the following, we assume that we are interested in identifying the distribution implied by $\theta^0 = (G^0, f_1^0, \ldots, f_p^0, (\sigma_1^0)^2, \ldots, (\sigma_p^0)^2)$.

Estimating the graph from the distribution A natural idea for causal discovery is to use the negative log-likelihood

$$L(\theta) = -\mathbb{E}_{\theta^0}\left[\log\left(p_\theta(\mathbf{X})\right)\right],$$

where $\mathbb{E}_{\theta^0}$ is the expectation with respect to the implied distribution of θ^0 and p_θ is the density of a candidate parameter $\theta = (G, f_1, \ldots, f_p, \sigma_1^2, \ldots, \sigma_p^2)$ corresponding to a minimal ANM. Observe that minimality for CAMs corresponds to $f_{kj} \neq 0$ for all additive components of SEs.

It follows from the identifiability that L attains its minimum exactly at θ^0. Denote by $\mathcal{G}(\theta)$ the graph within the parameter θ. As we are interested in finding G^0, we can define the score function

$$S(G) = \min_{\theta:\mathcal{G}(\theta)=G} L(\theta) = \min_{\theta:\mathcal{G}(\theta)=G} -\mathbb{E}_{\theta^0}\left[\log\left(p_\theta(\mathbf{X})\right)\right],$$

which attains its minimum at G^0. Further, the log-density decomposes into

$$\log\left(p_\theta(\mathbf{x})\right) = \sum_{k=1}^{p} \log\left(\frac{1}{\sigma_k}\phi\left(\frac{x_k - \sum_{j\in pa(k)} f_{kj}(x_j)}{\sigma_k}\right)\right),$$

where ϕ is the density of a univariate standard normal distribution (see Remark 2.5.14). If $G, f_1, \ldots, f_p$ within θ are fixed and we minimize $L(\theta)$ with respect to $\sigma_1^2, \ldots, \sigma_p^2$, we

obtain that the optimal choice for $\sigma_k^2, k = 1, \ldots, p$ is

$$\sigma_{k,\theta^0,f_k,G}^2 = \mathbb{E}_{\theta^0}\left[\left(X_k - \sum_{j \in pa_G(k)} f_{kj}(X_j)\right)^2\right].$$

Hence, we can simplify the score function by

$$S(G) = \min_{(f_1,\ldots,f_p)} -\mathbb{E}_{\theta^0}\left[\log\left(p_\theta(\mathbf{X})\right)\right],$$

where $f_1, \ldots, f_p$ must be chosen such that they align with G, that is, f_k is an additive function of $\mathbf{x}_{pa_G(k)}$ for $k = 1, \ldots, p$. In that case, it follows that

$$S(G) = \min_{(f_1,\ldots,f_p)} C + \sum_{k=1}^{p} \log\left(\sigma_{k,\theta^0,f_k,G}^2\right) \propto \min_{(f_1,\ldots,f_p)} \sum_{k=1}^{p} \log\left(\sigma_{k,\theta^0,f_k,G}^2\right).$$

It is emphasized that to calculate S we need to be aware of the true underlying parameter.

Estimating the graph from data Instead, we would like to use observations $\mathbf{x}^N = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$ from p_{θ^0} to find G^0. Thus, we replace the expectation with its empirical counterpart, that is,

$$\widehat{S}(G) = \widehat{S}\left(G, (\mathbf{x}_1, \ldots, \mathbf{x}_N)\right) = \sum_{k=1}^{p} \log\left(\sigma_{k,\widehat{f}_k,G}^2\right), \tag{2.5}$$

where

$$\sigma_{k,\widehat{f}_k,G}^2 = \frac{1}{N}\sum_{\ell=1}^{N}\left(x_{\ell k} - \sum_{j \in pa_G(k)} \widehat{f}_{kj}\left(x_{\ell j}\right)\right)^2$$

and $\widehat{f}_k = \sum_{j \in pa_G(k)} \widehat{f}_{kj}, k = 1, \ldots, p$ must be estimated from the data. The estimation of $\widehat{f}_k, k = 1, \ldots, p$ is discussed in Section 2.6.

Continuous learning algorithms Zheng et al. (2020) lever the differentiable DAG characterization for linear models to non-linear models and propose a continuous structure learning procedure. Here, the entries W_{kj} of a matrix $W \in \mathbb{R}^{p \times p}$ indicate whether the function f_k depends on X_j and $h(W) = 0$ if and only if the functions $f_1, \ldots, f_p$ correspond to a DAG. Then they consider arbitrary score functions $T(G, (\mathbf{x}_1, \ldots, \mathbf{x}_N)$

and find a minimum of

$$T\left(G, (\mathbf{x}_1, \ldots, \mathbf{x}_N)\right) + \rho h(W)^2 + \lambda ||W||_1$$

using gradient descent, where $\rho, \lambda > 0$ are again hyperparameters and $||W||_1$ penalizes dense graphs. Of course, $\widehat{S}$ is a reasonable choice for T. Other continuous structure learning approaches are Yu et al. (2019); Gao et al. (2021); Ng et al. (2022). Statistical properties of these procedures, such as consistency results, are not known.

On the contrary, Bühlmann et al. (2014) follow an order-based search as proposed by Teyssier and Koller (2005) for the linear Gaussian model and prove statistical consistency results. The details are outlined below.

Order-based search Bühlmann et al. (2014) define a score on the permutations by

$$\widehat{S}(\pi) = \widehat{S}(G_c(\pi)) = \sum_{k=1}^{p} \log\left(\sigma^2_{k,\widehat{f}_k,G_c(\pi)}\right),$$

where $\widehat{f}_k, k = 1, \ldots, p$ are chosen by ML estimation and $G_c(\pi)$ is the complete DAG with order π. Thus, $\widehat{f}_k$ is a regression estimate of X_k onto $\mathbf{X}_{\varpi_\pi(k)}$. This is a slight abuse of notation since $\widehat{S}$ is defined on permutations and graphs. In the following, it should be clear from the context. Under regularity assumptions on $f_1^0, \ldots, f_p^0$ (see Section 3.3 for details), they show that the score consistently prefers a $\pi^0 \in \Pi(G^0)$, that is, for any $\pi \notin \Pi(G^0)$

$$\lim_{N\to\infty} \widehat{S}(\pi) - \widehat{S}(\pi^0) > 0.$$

They additionally consider the case where p increases with N. As ML is prone to overfitting, they propose to initially find a set of candidates for the parents as in Friedman et al. (1999), which they call a **preliminary neighborhood selection (PNS)**. Under the additional assumptions,

- the number of parents for a node is uniformly bounded,
- the PNS correctly identifies a superset of bounded size of the true parents, and
- f_k^0 fulfills additional regularity conditions,

the score remains consistent. This result has little practical benefit, as for large p it is infeasible to calculate all scores $\widehat{S}(\pi)$ to determine the minimizer. Thus, for large p, they propose a greedy approach, which consists of the following steps:

1. **Preliminary Neighborhood Selection (PNS):** Every node X_k is regressed on

$\mathbf{X}_{[p]\setminus k}$. Nodes whose regression estimates are non-zero are considered as possible parents. Again, the result can be understood as a potentially cyclic directed graph G_{super}, which contains a superset of the final edges.

2. **Edge Orientation:** Initially, the graph G is empty. For every edge e in G_{super}, the reduction in score $\widehat{S}(G + e) - \widehat{S}(G)$ is calculated. Here, the score $\widehat{S}$ is defined in Equation (2.5) and the $\widehat{f}_{kj}$ are estimated by ML regression and $\widehat{G} + e$ is the graph when the edge e is added to G. The edge that reduces the score by the largest margin is added to G. Then, all edges that would cause a cycle in G are removed from G_{super}. The potential reduction in the score of Equation (2.5) is recalculated for every edge (only some score reductions for the edges have changed). The procedure is iterated until $G = G_{super}$.

3. **Pruning:** Using feature selection methods, for every node $X_k, k = 1, \ldots, p$ the influential parents are determined. The edges between the influential parents and X_k are kept, the other edges are discarded.

It is emphasized that PNS and pruning depend on the precise regression or feature selection method. Both usually come with additional hyperparameters that need to be tuned. The algorithm above is called MLCAM in this thesis. We revisit CAMs in Section 3.3 and propose boosting-based methods to derive the causal order and the DAG.

Multiple simulation studies suggest that MLCAM is superior to its continuous competitors (Lachapelle et al., 2019; Zheng et al., 2020; Charpentier et al., 2022). These findings are supported in our simulation study in Section 3.3. Furthermore, Reisach et al. (2021); Kaiser and Sipos (2022) indicate that the good performance of continuous structure learning procedures on simulated data sets results from artifacts from the simulation procedure. These findings question the reliability of continuous causal discovery methods for real-world data sets.

Evaluating Causal Discovery Algorithms

In this thesis, we use the Structural Hamming Distance (SHD) as a metric for DAGs G_1 and G_2. The SHD is the number of edge insertions, deletions, or flips to transform G_1 into G_2. It is popular for evaluating causal discovery algorithms (Tsamardinos et al., 2006; Kalisch et al., 2012; Bühlmann et al., 2014; Zheng et al., 2018).

2.5.4 Structural Equation Models and Causality

We already have used terminology such as *causal graph, causal discovery, causal order*, etc. borrowed from the field of causal inference. However, by inspecting the proposed methods more closely, we have applied probabilistic inference, that is, we derived a probability distribution or aspects thereof from the data. For example, for CAMs, we statistically estimated the parameter θ. In the field of PGMs, causal and probabilistic terminology and claims are frequently mixed. The purpose of this subsection is to disentangle them. Furthermore, Assumption 2.5.21 presents conditions such that SEMs derived from the observational distribution (or data) can be used to answer causal queries.

Interventions and Distributions

Causal inference tries among other things to investigate the effect of **interventions**. That is, if $\mathbf{X}_S$ is set externally at some fixed value $\mathbf{x}_S$, how does the distribution of the remaining variables $\mathbf{X}_{[p]\setminus S}$ change? This is in general different from the conditional distribution of $\mathbf{X}_{[p]\setminus S}|\mathbf{X}_S = \mathbf{x}_S$, which becomes apparent from the following example.

Example 2.5.19. *The study of Charig et al. (1986) investigates the success of two different surgeries* 0 *and* 1 *for kidney stones. Surgery* 1 *shows a lower probability of recovery compared to surgery* 0*. Thus,* $P(recovery|surgery = 0)$ *is larger than* $P(recovery|surgery = 1)$*. This seems to indicate that surgery* 1 *is inferior to surgery* 0*. However, the data also reveals that surgery* 1 *was applied more frequently to large kidney stones, which have a lower probability of recovery in general. In fact, considering small and large kidney stones separately, it is found that surgery* 1 *leads with a higher probability to good outcomes for both groups. Thus, the conditional probability does not reflect the causal relationships.*

Example 2.5.19 demonstrates that deriving causal relationships, that is, the effect of interventions, relies on assumptions and definitions beyond those of probabilistic investigations. Pearl (2009b) formalizes interventions with the do-operator, where

$$X_k|do(\mathbf{x}_s)$$

describes the distribution of X_k if $\mathbf{X}_S$ is intervened on and its value is set to $\mathbf{x}_S$. If there exists a DAG G called **causal graph**, so that the interventional distributions are represented by the **causal mechanisms**

$$X_k|do(\mathbf{x}_{\mathbf{pa}_G(k)}) = g_k(\mathbf{x}_{\mathbf{pa}_G(k)}, N_k) \tag{2.6}$$

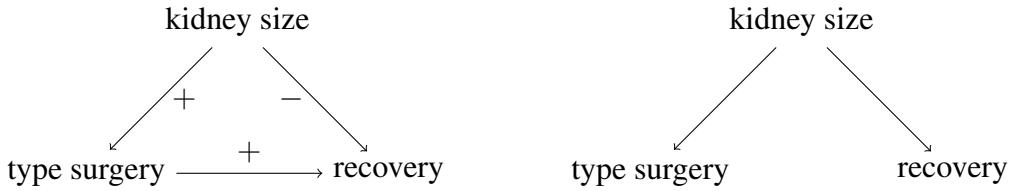

Figure 2.3: The causal graph behind Example 2.5.19 on the l.h.s. The probability of *recovery* conditioned on *type surgery* is higher for surgery 0 than for surgery 1. The reason is that large kidney sizes are more likely to be treated by surgery 1, while at the same time negatively affecting the chance of recovery. This contradicts the causal relationship that surgery 1 has a positive effect on *recovery*. The r.h.s. presents the causal graph if *type surgery* had no effect on *recovery*. By Propositions 2.5.8 and 2.5.20, the graph and the observational distribution constitute a BN. Furthermore, assuming faithfulness, *type surgery* and *recovery* are still dependent.

then the combination of the mechanisms and the graph is called a **causal model**. For a known causal graph, Pearl (1995) presents how the effects of interventions can be inferred from the observational distribution and thus from the data. The methodology can also be applied to Example 2.5.19 to derive the effect of the type of surgery on the outcome using the data at hand and assuming that the underlying causal graph follows Figure 2.3. It is emphasized that Equation (2.6) describes interventional distributions, while Definition 2.5.7 describes conditional distributions. The causal model contains a graphical representation of the CERs.

Causal Models and SEMs

The observational distribution of a causal model matches the distribution implied by the corresponding SEM. This is shown in the following proposition.

Proposition 2.5.20 (Peters et al. 2017, Proposition 6.3)**.** *If* $\mathbf{X}$ *is generated by a causal model, then its implied distribution* $P(\mathbf{X})$ *corresponds to the joint distribution implied by an SEM with the SEs set to causal mechanisms.*

SEMs describe the conditional distribution of X_k given its parents $\mathbf{X}_{\mathbf{pa}_G(k)}$. On the other hand, the causal mechanisms of Equation (2.6) give the distribution of X_k given a manipulation of its parents $\mathbf{X}_{\mathbf{pa}_G(k)}$. Both distributions do not necessarily align. However, if their equality is assumed and if a causal model corresponds to an SEM

in an identifiable class, then the causal model can be derived from the observational distribution. In the following, we collect the underlying assumptions.

Assumption 2.5.21. *To derive the causal graph over* $\mathbf{X}$ *from observational data using SEMs, the following assumptions must be satisfied.*

1. *There exists a causal model for* $\mathbf{X}$*,*
2. *its implied observational distribution (Proposition 2.5.20) corresponds to an SEM in an identifiable class and this class is known.*

Pearl (2009a) outlines that (1.) of Assumption 2.5.21 is not testable using observational data. SEM-based causal discovery methods derive the graph from a presumed class of SEMs.

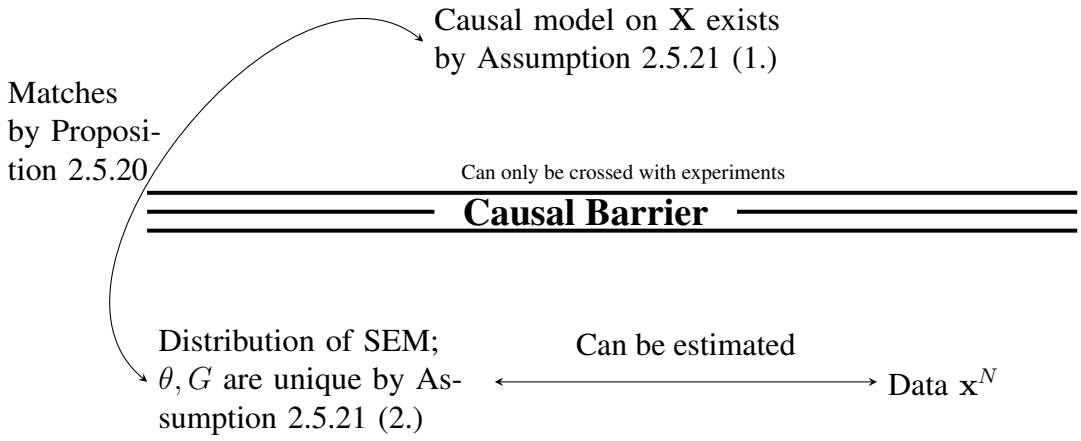

Figure 2.4: Schematic depiction of Section 2.5.4. The goal is to identify the interventional distributions from the data, that is, without interventions or experiments. Under the assumption that a causal model for $\mathbf{X}$ exists and belongs to an identifiable class, it can be characterized by the statistical parameter θ. This parameter can be estimated from the data. Without Assumption 2.5.21, the interventional distributions can only be derived by experiments, which allow one to go beyond the "Causal Barrier".

Confounding and Causal Discovery

Colombo et al. (2012) propose a constraint-based procedure called Really Fast Causal Inference (RFCI) to find the causal graph under unobserved confounding. That is,

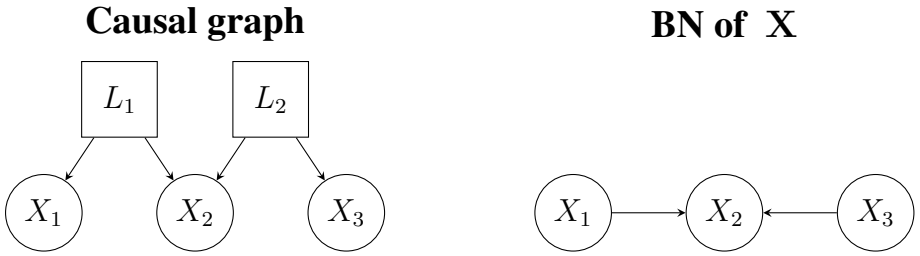

Figure 2.5: For the causal model on the left, the variables L_1, L_2 are unobserved while $\mathbf{X} = (X_1, X_2, X_3)$ are observed. The only (conditional) independence that exists within $\mathbf{X}$ is $X_1 \perp X_3$. Hence, the graph on the right is the unique graph that together with $\mathbf{X}$ constitutes a BN. Assume that there is a causal graph G on $\mathbf{X}$. G must be empty, as intervening on any of the variables has no effect. However, the implied distribution of the causal graph does not match the observational distribution, since, for example, X_1 and X_2 are dependent. This contradicts Proposition 2.5.20 and thus there cannot be a causal graph on $\mathbf{X}$. The example also shows once again, that although the observational distribution of $\mathbf{X}$ is implied by a CAM, it can not be used to identify the causal effects but one additionally relies on Assumption 2.5.21.

when there exist nodes in the causal graph for which no data is collected. In case of confounding, our goal is to derive the causal mechanisms within the observed variables whose marginal distribution we can observe. An example of their paper is reproduced in Figure 2.5. If there are confounders, then Assumption 2.5.21 (1.) is violated.

Unobserved confounding is one of the main challenges for causal discovery based on real-world data. Unfortunately, the interpretation of the resulting graph of RFCI is complex, as there exist six different kinds of edges. This induces a high complexity if one is investigating long chains in the graph. Similar to the PC algorithm, it relies on conditional independence tests, which are infeasible, but for multivariate normal distributions, multinomial distributions, or small data sets.

Score-based methods for causal discovery under confounding exist for special model classes (Wang and Drton, 2023).

Some Remarks for Applications in Manufacturing

Aspect 2 of Assumption 2.5.21 is questionable in manufacturing scenarios. The analysis of identifiable ANMs of Peters et al. (2014) is based on the fact that the noise

$(N_1, \ldots, N_p)$ has a strictly positive density. This implies that $P(\mathbf{X})$ also has a strictly positive density. However, the operator of a manufacturing process is interested in keeping the parameters within a specific interval for security reasons or to reduce the scrap rate. Therefore, this assumption is likely not met and causal effects outside of these parameter intervals can not be detected. The assumptions on the positivity of the density of $P(\mathbf{X})$ are strongly linked to the overlap assumptions in the potential outcome framework and, therefore, are not related to the graphical model approach.

Additionally, measurements along the production line lead to costs for acquisition and maintenance or increase the cycle time. Thus, some variables are not measured, even though they may be part of the causal mechanisms behind $\mathbf{X}$. Therefore, confounding is likely and raises questions about Assumption 2.5.21 (1.).

Although it is likely that in manufacturing applications Assumption 2.5.21 does not hold for ANMs and consequently for CAMs, causal discovery assuming CAMs provides an accessible visualization for high-dimensional complex data sets and can provide an approximation to the underlying causal graph. We hope that incorporation of prior knowledge into manufacturing compensates for the violation of Assumption 2.5.21 and the estimated graph is close to the underlying causal graph. We investigate this claim in Section 3.2 and Section 3.4.

2.6 Tools for Causal Discovery

Equation (2.5) shows that score-based causal discovery relies on regression estimators that regress X_k onto $\mathbf{X}_{pa_G(k)}$ for any $k \in [p]$ and some G. Intuitively, for the score to be expressive, it should hold for the true graph G^0 and increasing N that the function estimates $\widehat{f}_1, \ldots, \widehat{f}_p$ converge to the true SEs $f_1^0, \ldots, f_p^0$. This is formalized in Section 3.3.

Typically, little is known about $f_1^0, \ldots, f_p^0$ in addition to the fact that they are three times differentiable. It is unrealistic to assume that they lie in a known finite-dimensional function space. Therefore, parametric approaches such as linear regression or polynomial regression of bounded order typically fail to identify f_k^0.

Instead, nonparametric regression assumes that the functional relationship lies in a known infinite-dimensional space. Examples are the Generalized Additive Model (GAM, Wood 2006) and the Reproducing Kernel Hilbert Space (RKHS) regression. Similarly to polynomial regression, they also rely on a design matrix of a basis expansion. However, the size of the design matrix grows with N, and thus for $N \to \infty$ an increasing class

of functions can be approximated well. On the other hand, for noisy data, the risk of overfitting increases with the number of columns of the design matrix.

We briefly introduce the RKHS regression and then present how L^2-boosting combines small multiples of regression estimates to avoid overfitting. For a complete introduction to RKHS consult Wahba (1990); Schölkopf and Smola (2001); Wainwright (2019) and for boosting, see Bühlmann and Yu (2003); Schapire and Freund (2012).

Throughout this section, we consider a random variable Y and a random vector of p dimensions $\mathbf{X} = (X_1, \ldots, X_p)$ where the expectation of $g(\mathbf{X}, Y)$ with respect to their joint distribution is denoted by $\mathbb{E}_{\mathbf{X},Y}[g(\mathbf{X}, Y)]$. Furthermore, we denote the N i.i.d. samples from the joint distribution by $(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_N, y_N)$.

2.6.1 Reproducing Kernel Hilbert Space Regression

We start by introducing the kernel functions.

Definition 2.6.1. *We call a symmetric function $K : \mathbb{R}^p \times \mathbb{R}^p \to \mathbb{R}$ a positive definite (p.d.) kernel on $\mathbb{R}^p$ if*

$$\sum_{k=1}^{n} \sum_{\ell=1}^{n} \alpha_k \alpha_\ell K(\mathbf{z}_k, \mathbf{z}_\ell) \geq 0$$

for any $\{\alpha_1, \ldots, \alpha_n\} \subset \mathbb{R}$ and any $\{\mathbf{z}_1, \ldots, \mathbf{z}_n\} \subset \mathbb{R}^p$ and any $n \in \mathbb{N}$.

Example 2.6.2 (Gaussian kernel)**.** *For $\varsigma > 0$, the Gaussian kernel on $\mathbb{R}^p$ is defined by*

$$K(\mathbf{z}, \mathbf{z}') = \exp\left(-\frac{||\mathbf{z} - \mathbf{z}'||_2^2}{2\varsigma}\right).$$

We assume from now on, that K is p.d. Then it implies a unique Hilbert space of functions H with $K(\cdot, \mathbf{x}) \in H \, \forall \mathbf{x} \in \mathbb{R}^p$ and

$$f(\mathbf{x}) = \langle f, K(\cdot, \mathbf{x}) \rangle_H \tag{2.7}$$

for any $f \in H$ and $\mathbf{x} \in \mathbb{R}^p$. Equation (2.7) is the name-giving reproducing property. Depending on K, this function space H can be infinite-dimensional. From Equation (2.7) it follows for the inner product of $f = \frac{1}{\sqrt{N}} \sum_{k=1}^{N} \alpha_k K(\cdot, \mathbf{x}_k)$ and $g = \frac{1}{\sqrt{N}} \sum_{k=1}^{N} \beta_k K(\cdot, \mathbf{x}_k)$, that

$$\langle f, g \rangle_H = \alpha^T G \beta,$$

with $G_{jk} = \frac{K(\mathbf{x}_j, \mathbf{x}_k)}{N}, j, k = 1, \ldots, N$. Here, G is called the Gram matrix. By the representation theorem (Wainwright, 2019, Proposition 12.33) the minimizer of

$$\widehat{f} = \underset{f \in H}{\operatorname{argmin}} \frac{1}{N} \sum_{\ell=1}^{N} (y_\ell - f(\mathbf{x}_\ell))^2 + \lambda ||f||_H^2, \tag{2.8}$$

where $\lambda > 0$, can be expressed by

$$\widehat{f}(\cdot) = \frac{1}{\sqrt{N}} \sum_{\ell=1}^{N} \beta_\ell K(\cdot, \mathbf{x}_\ell),$$

for some $\beta \in \mathbb{R}^N$. Thus, the representation theorem shows that although the search space is infinite-dimensional, the minimizer of the sum of the empirical L^2-risk and a regularization term lies in a known finite-dimensional space. However, this does not hold for the population minimizer

$$f^* = \underset{f \in H}{\operatorname{argmin}} \, \mathbb{E}_{\mathbf{X},Y} \left[(Y - f(\mathbf{X}))^2 \right] + \lambda ||f||_H^2,$$

as f^* can be outside the N-dimensional subspace spanned by $K(\cdot, x_\ell), \ell = 1, \ldots, N$. For an increasing λ, the function estimate $\widehat{f}$ becomes more regular. The kind of regularity depends on the norm of H and thus on the kernel K.

If $K \in L^2(\mathbb{R}^p \times \mathbb{R}^p)$ and it is continuous, then it implies an integral operator $\mathcal{K} : L^2(\mathbb{R}^p) \to L^2(\mathbb{R}^p)$ by

$$f(\cdot) \mapsto (\mathcal{K}(f))(\mathbf{x}') = \int_{\mathbb{R}^p} K(\mathbf{x}, \mathbf{x}') f(\mathbf{x}) dP_{\mathbf{X}}(\mathbf{x}).$$

If the domain of $\mathbf{X}$ is compact, then Mercer's theorem (Wainwright, 2019, Theorem 12.20) shows that the operator $\mathcal{K}$ has eigenvalues $\mu_k \geq 0$ and eigenvectors $\phi_k \in L^2(\mathbb{R})$, so that $\mathcal{K}(\phi_k) = \mu_k \phi_k$ and

$$K(\mathbf{x}, \mathbf{x}') = \sum_{k=1}^{\infty} \mu_k \phi_k(\mathbf{x}) \phi_k(\mathbf{x}'),$$

where the convergence of the infinite series holds uniformly and absolutely. Sun (2005) generalizes Mercer's theorem to non-compact domains of $\mathbf{X}$. Let the eigenvalues be ordered non-increasingly. The decay rate of the eigenvalues determines the risk of overfitting for the RKHS regression and the appropriate choice of λ (Wainwright, 2019, Theorem 13.17).

Observe that, so far, the functions in the RKHS were p-dimensional but do not have an additive structure. Fortunately, for the kernels $K_1, \ldots, K_p$ defined on $X_1, \ldots, X_p$ and corresponding RKHSs $H_1, \ldots, H_p$, the sum

$$(K_1 + \ldots + K_p)(\mathbf{x}, \mathbf{x}') := \sum_{j=1}^{p} K_j(x_j, x_j')$$

is a p.d. kernel implying a Hilbert space H consisting of functions of the form

$$f(\mathbf{x}) = \sum_{j=1}^{p} f_j(x_j),$$

where $f_j \in H_j$. Its norm is defined by $||f||_H^2 = \sum_{j=1}^{p} ||f_j||_{H_j}^2$. In Section 3.3 we assume that SEs $f_k^0, k = 1, \ldots, p$ lie in the RKHS implied by the kernel $\sum_{j \in pa_{G^0}(k)} K_j(x_j, x_j')$, where K_j is a one-dimensional Gaussian kernel.

2.6.2 Boosting

Boosting is an ensemble learning method that combines different regression estimates. In this thesis, we focus on L^2-boosting which aims to find the minimizer of

$$\underset{f \in \mathcal{F}}{\operatorname{argmin}} \frac{1}{2} \mathbb{E}_{\mathbf{X},Y} \left[(Y - f(\mathbf{X}))^2 \right],$$

where $\mathcal{F}$ is a vector space of functions. Boosting builds on a base learner

$$S : ((\mathbf{x}_1, u_1), \ldots, (\mathbf{x}_N, u_N)) \mapsto \widehat{f} \in \mathcal{F}$$

that takes data and returns a regression estimate for arbitrary $\{u_1, \ldots, u_N\} \subset \mathbb{R}$. Starting with an initial learner $\widehat{f}^{(0)}$, it iteratively calculates the residuals for $m = 0, 1, \ldots$

$$u_\ell = y_\ell - \widehat{f}^{(m)}(\mathbf{x}_\ell) \tag{2.9}$$

and then learns $\widehat{f} = S((\mathbf{x}_1, u_1), \ldots, (\mathbf{x}_N, u_N))$. For a chosen step size $0 < \nu < 1$, the new boosting estimate is updated by

$$\widehat{f}^{(m+1)} = \widehat{f}^{(m)} + \nu \widehat{f}.$$

The procedure can be understood as a functional gradient descent, where Equation (2.9) is expressed by

$$u_\ell = -\frac{\partial \left(\frac{1}{2}(y_\ell - f(\mathbf{x}_\ell))^2\right)}{\partial f}|_{f=\widehat{f}^{(m)}}, \ell = 1, \ldots, N,$$

which is the derivative of the empirical version of the L^2-loss with respect to f. We stop the procedure after m_{stop} iterations, which is called early stopping. It is desirable that m_{stop} is chosen such that

$$\widehat{f}^{(m_{stop})} \approx \operatorname*{argmin}_{f \in \mathcal{F}} \frac{1}{2}\mathbb{E}_{\mathbf{X},Y}\left[(Y - f(\mathbf{X}))^2\right].$$

If m_{stop} is too small, then the residuals $u_1, \ldots, u_N$ are large and the boosting estimator is underfitting. Thus, the estimator has a large bias. On the other hand, if m_{stop} is too large, then the boosting estimator is overfitting and has a large variance. Hence, m_{stop} needs to trade off bias and variance. In this thesis, we consider two different classes of base learners S.

Symmetric base learners are such that for

$$\widehat{f} = S_{\mathbf{x}_1,\ldots,\mathbf{x}_N}(u_1, \ldots, u_N) := S((\mathbf{x}_1, u_1), \ldots, (\mathbf{x}_N, u_N))$$

the mapping $(u_1, \ldots, u_N) \mapsto \left(\widehat{f}(\mathbf{x}_1), \ldots, \widehat{f}(\mathbf{x}_N)\right)$ is a linear and symmetric mapping in $(u_1, \ldots, u_N)$. Thus, S can be diagonalized and its eigenvalues can be analyzed. An example is the RKHS regression for a kernel K. Here, the eigenvalues of S depend on the eigenvalues of the integral operator corresponding to K. Under the assumption that $Y = f(\mathbf{X}) + \varepsilon$ for a (sub-) Gaussian noise ε independent of $\mathbf{X}$, Bühlmann and Yu (2003); Raskutti et al. (2014) determine the optimal m_{stop}, which depends on the eigenvalues of S. It is emphasized that this does not hold for the score (2.5). That is, for $G \neq G^0$ it can be $X_k \neq f(\mathbf{X}_{pa_G(k)}) + \varepsilon$ for any Gaussian ε independent of $\mathbf{X}_{pa_G(k)}$ and three-times differentiable f. We consider symmetric base learners for their theoretical properties, in particular, the possibility to analyze their eigenvectors and eigenvalues.

Sparse additive learners are such that

$$S((\mathbf{x}_1, u_1), \ldots, (\mathbf{x}_N, u_N))(\mathbf{x}) = \widehat{f}(x_j),$$

where $\widehat{f}$ is a one-dimensional function in x_j. In that case, the boosting procedure is called componentwise and $\widehat{f}^{(m)}$ is an additive function. Sparse additive learners are non-linear and harder to analyze. However, they show excellent empirical properties

in high-dimensional settings (Tutz and Binder, 2006). Bühlmann and Hothorn (2007) propose an AIC score to determine m_{stop}. This makes the sparse additive learners attractive for causal discovery in high dimensions.

In summary, we consider symmetric base learners for the theoretical analysis, while sparse additive learners are employed because of their strong empirical performance in high dimensions.

3 Summary of the Articles

3.1 Estimating Gaussian Copulas with Missing Data with and without Expert Knowledge

Motivation

Missing data is ubiquitous in real-world manufacturing data sets. We rely on sensors, which are often error-prone and occasionally deliver either no or implausible values. At the same time, intermediate products of battery cells are sometimes taken out of the production line early because they do not meet the quality requirements. Hence, measurements that are recorded later in the production workflow are never collected. Still, one is interested in the multivariate joint distribution and in particular in the dependence structure of the production measurements for knowledge discovery. In this work, it is assumed that the distribution's copula is a Gaussian copula.

The appropriate method to infer statistical parameters from data with missing values depends on the mechanism that causes the data to be absent. Missing Completely At Random (MCAR) assumes that the probability of an entry to be missing is independent of any measurements. Under the MCAR assumption, the joint distribution can be consistently estimated by the two-step approach of Genest et al. (1995) applied to the complete observations. However, in the example above, MCAR assumes that the probability of the ejection of the product is independent of the already recorded measurements. This seems unrealistic.

Thus, in this work we assume Missing at Random (MAR), which is considerably less restrictive than MCAR. It assumes that the probability of entries to be missing depends only on the measurements that are observed. In the above example, the probability of the ejection of the intermediate product can be based on the measurements collected in the earlier steps. This seems reasonable. Under the MAR assumption, the two-step approach based on observed values is not consistent, as the estimates of the marginal distributions are biased. The paper provides an example. MAR allows for the application of the Expectation-Maximization (EM) algorithm and multiple imputation.

For the latter, one needs to choose the imputation procedure, which can be independent from the model assumption and its hyperparameters. On the other hand, the EM algorithm is merely based on the model assumption and guarantees the convergence of the log-likelihood towards a local optimum. As estimating the joint distribution with missing data is challenging, integration of prior knowledge is desirable.

Method

The method is based on the following observation. If the marginal distribution functions $F_1, \ldots, F_p$ are known, then the observed values can be assigned to their marginal quantiles $F_j(x_{\ell j}), j = 1, \ldots, p; \ell = 1, \ldots, N$. From these, an estimate for the copula can be derived using an EM algorithm.

On the other hand, if the copula is known, then one can estimate the location of the missing values. For example, if the copula shows that X_1 and X_2 are strongly and monotonically dependent, then a large value for X_1 implies a high probability that the value for X_2 is also large. If the value for X_1 is known but missing for X_2, then one can derive likely locations for the missing entry of X_2, that is, the conditional distribution of X_2 given the known value of X_1.

We use this intuition starting with the initial parameters for the marginal distributions, denoted by θ^0, and the Gaussian copula parameter, denoted by Σ^0. We apply a cyclic approach, in which we iteratively estimate Σ^{t+1} based on θ^t before we estimate θ^{t+1} based on Σ^{t+1}. In the paper we show that this intuition can be described as an EM algorithm, where the M-step consisting of updating Σ and θ is split in two. The EM algorithm relies on a parametric form for the marginal distributions. We preserve the flexibility of the copula model by parameterizing them as Gaussian mixtures. The estimation of the parameters of the marginals θ is based on a Monte Carlo integration and thus not exact. On the contrary, the estimation of Σ can be achieved without sampling and in closed form. The cyclic approach is depicted in Figure 3.1.

Prior knowledge can be leveraged, for example, by

1. fixing the entries of the precision matrix of the Gaussian copula to 0 if two variables are known to be conditionally independent given the remaining variables, and
2. choosing the parametric family of the marginal distributions.

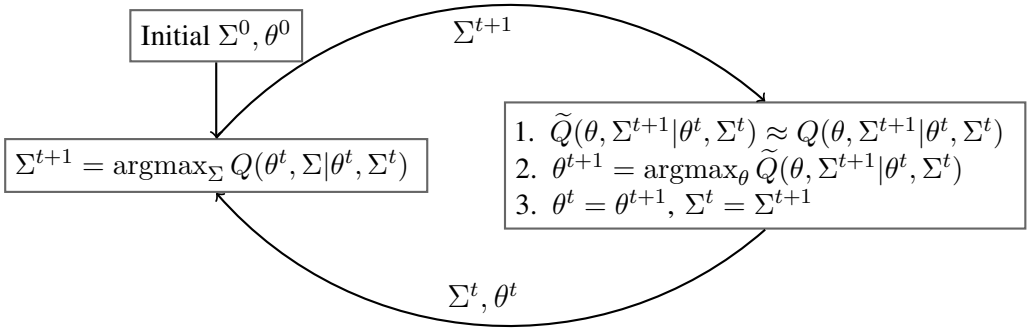

Figure 3.1: The proposed EM algorithm for the estimation of the copula and the marginal distributions. Here, $Q(\theta, \Sigma|\theta^t, \Sigma^t)$ is the expected log-likelihood function of θ, Σ with respect to θ^t, Σ^t. Starting with θ^0, Σ^0 the procedure updates Σ^{t+1} in closed form. For the new Σ^{t+1} an approximated E-Step is carried out, which is a Monte Carlo approximation $\widetilde{Q}$ to $\theta \mapsto Q(\theta, \Sigma^{t+1}|\theta^t, \Sigma^t)$. The maximizer of this function is assigned as θ^{t+1}. The algorithm stops if the increase of $Q(\theta^{t+1}, \Sigma^{t+1}|\theta^t, \Sigma^t)$ is below a threshold.

Simulation Study & Results

In the simulation study, we compare the proposed method, indicated by EM, with

- an EM algorithm which only estimates the copula parameter while estimating the marginals using the observed values only (corresponds to the procedure proposed by Ding and Song 2016, SCOPE), and
- a Markov Chain Monte Carlo (MCMC) approach proposed by Hoff (2007)

on synthetically generated data sets with data MAR following a distribution with a Gaussian copula. Here, the share of missing values, the correlation of the Gaussian copula, and the sample size is varied. The data is 2-dimensional. The estimates for the copula and the marginals are analyzed separately.

Estimates for Marginal Distributions It can be observed that EM provides better estimates for the marginal distributions than SCOPE. This effect is stronger when the share of missing values or the correlation parameter increases. MCMC provides the worst estimates. For an increasing sample size, the EM estimates approach the true marginal distributions. This does not hold for SCOPE and MCMC.

Estimates for Copula Estimator The copula parameter is slightly better estimated by SCOPE compared to EM. Again, MCMC performs worst. Increasing the sample size improves the estimates for EM and SCOPE towards the true parameter. This cannot be observed for MCMC.

The take-away is that it is sufficient to estimate the marginals only based on the observed values if one is merely interested in the copula parameters. On the other hand, if one is interested in the joint and hence in the marginal distributions, then it is advisable to apply EM, that is, the proposed algorithm.

Another simulation study shows that the incorporation of prior knowledge into the dependence structure improves not only the estimates for the copula parameter but also for the marginal distributions.

So far, the proposed procedure has been limited to small p. Parameterization of marginals as neural networks using differentiable sampling could leverage the approach to larger p.

3.2 Learning Causal Graphs in Manufacturing Domains using Structural Equation Models

In this work, we propose an adaptation of MLCAM that can incorporate existing prior knowledge in manufacturing. The goal is to derive CERs from data collected along a production line. The method was finally applied to battery modules, which combine battery cells and are the building blocks of vehicle energy storage.

Prior Knowledge & Objective

It is well known that the incorporation of prior knowledge provides a considerable benefit for causal discovery (de Campos and Castellano, 2007; Borboudakis and Tsamardinos, 2016; Hasan and Gani, 2022; Constantinou et al., 2023). Following de Campos and Castellano (2007), prior knowledge can be translated into one of the following three restrictions. All of them are relevant in manufacturing.

- **Existence relation**: An edge between two nodes is known to exist. For example, this is the case when DOEs have shown a CER between variables.
- **Absence relation**: An edge between two nodes is known to be absent. For example, products can consist of multiple sub-products for which measurements

also exist. If the sub-products are produced independently, then there cannot be a CER between the measurements of the sub-products. See Figure 1 in paper.

- **Temporal ordering**: Production steps are processed in a fixed order. Thus, there cannot be CERs from measurements of subsequent production steps to measurements of earlier steps.

All of the aforementioned analyses of the impact of expert knowledge on causal discovery are based on the linear Gaussian model or use constraint-based methods. In contrast, our work investigates CAMs and shows how the three steps of MLCAM as described in Section 2.5.3 can be adapted to incorporate absence relations and temporal ordering. We call the resulting algorithm the Temporal Causal Additive Model (TCAM). Existence relations can be included as described below.

Method

We adapt the steps of MLCAM as follows.

- **Preliminary Neighborhood Selection (PNS):** In the PNS we allow for super-DAGs that contain edges, which are neither known to be absent nor violate the known temporal ordering. That is, every node is regressed on these variables that were measured at the same or at an earlier production step and for which no absence relation is known. Afterwards, edges corresponding to existence relations are added.
- **Node Ordering:** The edges that cross the production steps are oriented according to the temporal ordering, that is, the ordering of the production steps. Subsequently, the edges within the same production step are oriented.
- **Pruning:** The edges are pruned as in the original algorithm. If existence relations exist, then pruning these edges can be forbidden.

The article describes causal discovery with the score function $\widehat{S}$ in Equation 2.5 chosen as the sum over the mean squared errors (MSEs) instead of choosing $\widehat{S}$ as the negative log-likelihood of the model. The MSE score was proposed, for example, by Zheng et al. (2020). However, the negative log-likelihood appears to be more appropriate. For a discussion of the different scores, see Reisach et al. (2021).

Application & Findings

In the article, we discuss the generation of the data set, as the data preparation workflow partially determines the existing prior knowledge. The final data set contains 459 measurements of 7254 battery modules.

The performance and applications of causal discovery algorithms are frequently discussed for high-dimensional data, where $p > N$ (Friedman et al., 1999; Kalisch et al., 2012; Bühlmann et al., 2014). In contrast, this is not a concern in manufacturing.

The graph estimated by TCAM contains some nodes with few neighbors, while other nodes have a large number of neighbors. This contradicts a frequent assumption in causal discovery that the number of neighbors is limited by a small number.

Many of the derived edges have been verified as plausible CERs by process experts. As an additional plausibility check, we investigate the patterns of identical sub-products that were produced independently of each other. As expected, these show similar patterns. Furthermore, the estimated graph reveals a CER from which experts could derive actionable insights that improve the quality of products.

Finally, we revisit the subgraph of the sub-products. For these, there exists a partial expert assessment of the causal relations. We bootstrap 500 data sets, each containing 500 sub-products and their measurements. Then, we apply TCAM, MLCAM, and an adaption of the PC algorithm (TPC) that incorporates the same prior knowledge as TCAM to each data set and compare the estimated graph with the expert assessment. We evaluate the quality of the estimated graph using an adapted SHD (aSHD) which adjusts to the uncertainty in the expert assessment. The results reveal that the mean and standard deviation of aSHD and the runtime of TCAM are significantly lower than those of MLCAM. The mean aSHD of TPC and TCAM is similar, while TCAM shows a lower standard deviation of the aSHD. This indicates that TCAM is more robust than TPC in manufacturing applications.

3.3 Boosting Causal Additive Models

This work investigates the learning of the DAG of a CAM from data and shows the consistency of the proposed procedure. Although there are numerous papers based on (conditional) independence tests, that is, constraint-based methods (Gretton et al., 2009; Mooij et al., 2009; Peters et al., 2014; Assaad et al., 2019; Lee et al., 2020) that demonstrate consistency, such results are sparse for score-based procedures. Here, the

work of Kpotufe et al. (2014) focuses on $p = 2$. For general p, the work of Nowzohour and Bühlmann (2016) employs penalized nonparametric regression. On the other hand, Bühlmann et al. (2014) show that their permutation score based on non-penalized ML regressions consistently prefers the correct causal ordering. Nonparametric ML regressions tend to overfit easily. Consequently, the consistency result relies on partially restrictive and non-tangible assumptions. We discuss them below.

On the other hand, there are theoretical results (Bühlmann and Yu, 2010; Raskutti et al., 2014) for boosting regression and its strong performance on real-world problems is well established. In this work, we

1. prove the statistical convergence of a permutation score based on boosting under less restrictive and more tangible assumptions than Bühlmann et al. (2014),
2. provide insights on the behavior of boosting under misspecification, and
3. develop a boosting-based method for causal discovery for data of larger dimension, when it becomes infeasible to calculate all permutation scores.

Theory & Methods

In the following, we assume that we have N observations $\mathbf{x}^N = (\mathbf{x}_1, \ldots, \mathbf{x}_N)$ from a CAM with parameter $\theta^0 = (f_1^0, \ldots, f_p^0, G^0, \sigma_1^0, \ldots, \sigma_p^0)$ and Π^0 is the set of causal orders of G^0. As before, our goal is to learn G^0 or Π^0 from the observations.

Low-dimensional data

Under the assumptions discussed below, we prove that when the $\widehat{f}_k, k = 1, \ldots, p$ in Equation (2.5) are estimated by a boosted RKHS regression with early stopping, where the number of boosting steps is chosen in the order $N^{\frac{C_u + C_d + 1/2}{4(C_d+1)}}$, then the corresponding score on the permutations is consistent. That is for all $\pi^0 \in \Pi^0$ and all $\pi \notin \Pi^0$ it holds that $\lim_{N\to\infty} \widehat{S}(\pi) - \widehat{S}(\pi^0) > 0$. Here, $C_u, C_d > 0$ are distribution-dependent constants.

This is shown in three steps.

1. We observe that Equation (2.5) defines a score function for any regression estimator.
2. In Proposition 1 we derive conditions on the regression estimator so that it consistently prefers $\pi^0 \in \Pi^0$.

3. We show that the conditions of Proposition 1 are met for the boosted RKHS regression estimator with the number of boosting steps chosen as above.

Proposition 1 If the regression estimator $((\mathbf{x}_{1S}, x_{1k}), \ldots, (\mathbf{x}_{NS}, x_{Nk})) \mapsto \widehat{f}$ is such that for any $k \in [p]$ and $S \subset [p] \setminus \{k\}$ it holds that

$$\left| \frac{1}{N} \sum_{\ell=1}^{N} \left(x_{\ell k} - \widehat{f}(\mathbf{x}_{\ell S}) \right)^2 - \mathbb{E}_{\theta^0} \left[\left(X_k - \widehat{f}(\mathbf{X}_S) \right)^2 \right] \right| \xrightarrow{\mathbb{P}} 0 \tag{3.1}$$

and for any $\pi^0 \in \Pi^0$ it holds that

$$\frac{1}{N} \sum_{\ell=1}^{N} \left(x_{\ell k} - \widehat{f}(\mathbf{x}_{\ell \varpi_{\pi^0}(k)}) \right)^2 \xrightarrow{\mathbb{P}} \underbrace{\mathbb{E}_{\theta^0} \left[\left(X_k - \sum_{j \in \mathbf{pa}_{G^0}(k)} f_{kj}^0(X_j) \right)^2 \right]}_{=(\sigma_k^0)^2}, \tag{3.2}$$

then the score of Equation (2.5), where the $\widehat{f}$ are estimated by the regression estimator, is consistent. Equation (3.2) is easier to show and corresponds to a consistency of the regression estimator. On the other hand, Equation (3.1) is harder to show and ensures that the estimator is not overfitting. It is emphasized that S and k are arbitrarily chosen. Thus, the conditional distribution of X_k given $\mathbf{X}_S$ can have any form.

RKHS Boosting fulfills the Conditions of Proposition 1 It is assumed that the SEs $f_1^0, \ldots, f_p^0$ lie in an RKHS with an additive Gaussian kernel. Then, for the number of boosting steps as above, Equation (3.2) is shown using techniques similar to those of Bühlmann and Yu (2010); Raskutti et al. (2014).

On the other hand, Equation (3.1) is upper bounded by the triangle inequality by

$$\underbrace{\left| \frac{1}{N} \sum_{\ell=1}^{N} \widehat{f}(\mathbf{x}_{\ell S})^2 - \mathbb{E}_{\theta^0} \left[\widehat{f}(\mathbf{X}_S)^2 \right] \right|}_{\text{Squared norm (I)}} + \underbrace{\left| \frac{1}{N} \sum_{\ell=1}^{N} x_{\ell k} \widehat{f}(\mathbf{x}_{\ell S}) - \mathbb{E}_{\theta^0} \left[X_k \widehat{f}(\mathbf{X}_S) \right] \right|}_{\text{Inner product (II)}}$$
$$+ \left| \frac{1}{N} \sum_{\ell=1}^{N} x_{\ell k}^2 - \mathbb{E}_{\theta^0} \left[X_k^2 \right] \right|$$

and we show the convergence to 0 in probability for all three terms. We assume that the fourth moment of X_k exists, so that the convergence of the last term follows. For

the convergence of the squared norm and the inner product term to 0 in probability for $N \to \infty$, we use results from empirical process theory.

Recall that $\widehat{f}$ depends on the data and therefore on N. Based on the work of Bartlett and Mendelson (2002), the convergence of term (I) towards 0 in probability can be shown if $\widehat{f}$ lies in a function class $\mathcal{F}_N$ whose *Rademacher complexity* grows slowly with N. On the other hand, the convergence of term (II) towards 0 in probability can be shown if the *covering number*[1] of $\mathcal{F}_N$ grows slowly with N. Further, the Rademacher complexity and the covering number of function classes contained in balls of some radius in the RKHS can be upper-bounded. We show that we can upper-bound the RKHS norm of the estimate $||\widehat{f}||_H$ by a number depending on the number of boosting steps and N with high probability. Both complexity measures are introduced in Wainwright (2019, Chapter 5). If the number of boosting steps is chosen as above, then it is ensured that, for increasing N the estimate $\widehat{f}$ lies in a function class whose Rademacher complexity and covering number grows sufficiently slowly to ensure the convergence of (I) and (II) towards 0. The idea is sketched in Figure 3.2.

Beyond small p

If p is such that it becomes infeasible to calculate the score for all permutations, we rephrase the learning problem of causal discovery by finding a function $F = (f_1, \ldots, f_p) : \mathbb{R}^p \to \mathbb{R}^p$ where $f_k(\mathbf{x}) = \sum_{j=1}^p f_{kj}(x_j), k = 1, \ldots, p$. Here, F shall represent a CAM. We aim to minimize the likelihood

$$L(F, \mathbf{x}^N) = L\left((f_1, \ldots, f_p), \mathbf{x}^N\right) = \sum_{k=1}^{p} \log\left(\sum_{\ell=1}^{N} \left(\mathbf{x}_{\ell k} - f_k(\mathbf{x}_\ell)\right)^2\right).$$

We follow an iterative approach. Starting with $F^{(0)} = 0$, we apply a component-wise boosting that adds one additive component f_{kj} at each step. Denote the function after m steps by $F^{(m)} = \left(f_1^{(m)}, \ldots, f_p^{(m)}\right)$.

1. For $j, k = 1, \ldots, p$ and $j \neq k$ the function $f(x_j)$ which reduces the unexplained noise in X_k after m steps $\sum_{\ell=1}^{N} \left(\left(x_{\ell k} - f_k^{(m)}(\mathbf{x}_\ell)\right) - f(x_{kj})\right)^2 + \lambda||f_{kj}||_{H_j}^2$ by the largest margin is identified. The term $\lambda||f_{kj}||_{H_j}^2$ penalizes complexity and $\lambda > 0$ is a hyperparameter, and $||\cdot||_{H_j}$ is the RKHS norm on X_j. These are the candidate functions $\widehat{f}_{kj}, j, k = 1, \ldots, p$.

[1]The Rademacher complexity and the covering numbers quantify the richness of a set of functions. When the functions are similar to each other, these measures are lower.

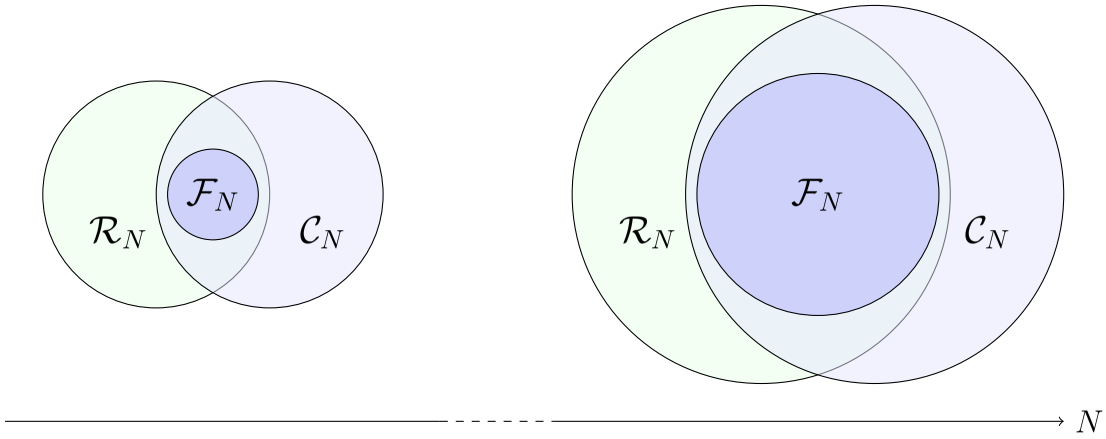

Figure 3.2: Visualization of proof idea. For $N = 1, \ldots$ one can define a growing sequence of function classes $\mathcal{R}_N$ that have Rademacher complexities depending on N. $\mathcal{R}_N$ is chosen such that, with probability going to 1 uniformly for all functions in $\mathcal{R}_N$ the empirical squared norm is close to the population version. Similarly, one can define a growing sequence of function classes $\mathcal{C}_N$ that have covering numbers depending on N so that uniformly for all functions in $\mathcal{C}_N$ the empirical inner product is close to the population version. We ensure that $\widehat{f}$ (which depends on N) is in a function class $\mathcal{F}_N \subset \mathcal{R}_N \cap \mathcal{C}_N$ by controlling the number of boosting iterations. Then the convergence of I and II can be ensured. To be precise, $\mathcal{F}_N \subset \mathcal{R}_N \cap \mathcal{C}_N$ holds with a probability going to 1 for $N \to \infty$. This is sufficient for the convergence of I and II to hold.

2. The index $(j^0, k^0) = \text{argmin}_{j \neq k} L(F^{(m)} + \widehat{f}_{kj})$ is determined. $F^{(m+1)}$ is updated by $f_k^{(m+1)}(\mathbf{x}) = f_k^{(m)}(\mathbf{x}) + \nu \widehat{f}_{k^0 j^0}(x_{j^0})$, where $0 < \nu < 1$ is a step size.
3. As we add every additive function one by one, it is easy to track which function, if added to $F^{(m+1)}$ would cause a cyclic graph. These functions are forbidden.
4. If $L(F^{(m+1)}, \mathbf{x}^N) + AIC(F^{(m+1)}) > L(F^{(m)}, \mathbf{x}^N) + AIC(F^{(m)})$, where AIC penalizes the complexity in $F^{(m+1)}$, then the procedure is stopped.

Finally, we apply a pruning step as for MLCAM. It is emphasized that this method does not rely on a PNS and thus needs fewer hyperparameters to be tuned.

Simulation Studies

We run simulation studies for $p = 5$ (low dimensions) and for $p = 100$ (high dimensions). Although, for the latter case we consider the sample size $N = 200$ and thus $p < N$, we call the data high-dimensional.

Low Dimensions We randomly construct CAMs and determine the minimal distance between the estimated permutation and the set of causal orders of the DAG for different sample sizes. One can see that the mean distance goes to 0 for increasing N, underlining the theoretical result. Even if the SEs are non-additive (a misspecification of the model), one observes a similar, although slower, pattern.

High Dimensions We randomly generate DAGs with on average 100 edges, where the edges are evenly (ER) or unevenly (SF) distributed among the nodes. The latter is relevant in manufacturing as shown in Section 3.2. The SEs are either additive or non-additive. The latter is a misspecification of the model. We learn the DAG using the proposed method, MLCAM, and NOTEARS (Zheng et al. (2020)) and compare the estimates with the underlying DAG using the SHD. For three settings, the proposed method and MLCAM perform similarly. However, for the most challenging scenario, which is relevant for many applications, that is, non-additive SEs with SF graphs, the proposed method outperforms MLCAM. NOTEARS performs worse in all scenarios.

Discussion

Low dimensions To our knowledge, in addition to the aforementioned results, the work of Bühlmann et al. (2014) and the strongly related work of van de Geer (2014), we

provide the only consistency result for causal discovery for CAMs. The convergence (3.1) gives insight into the behavior of L^2-boosting regression in quite general, misspecified scenarios.

Some assumptions of Bühlmann et al. (2014) are similar to ours. For example, both approaches assume an eigenvalue condition (Lemma 2 and Assumption 12) and moment and tail conditions (Assumption A2 and Assumption 9). However, Bühlmann et al. (2014) restrict the search space of $\widehat{f}_{kj}$ to a finite-dimensional subspace, which grows "sufficiently slowly" with N. Further, the search space "is deterministic and does not depend on the data". Our approach is different, as the RKHS functions depend on the data and our search space is infinite-dimensional. Instead of fixing the dimensionality of the search space by some vague number, we choose the number of boosting iterations asymptotically. This seems to be more elegant to us.

Both methods can be applied to higher-dimensional data if the search space of permutations can be drastically reduced. This can be the case if the variables are temporally ordered, as, for example, in manufacturing.

High dimensions Our method can be understood as a component-wise functional gradient-descent in the non-convex space of additive functions from $\mathbb{R}^p$ to $\mathbb{R}^p$ corresponding to a DAG. It extends the success of component-wise boosting for high-dimensional regression and classification to causal discovery.

3.4 Interactive and Intelligent Root Cause Analysis in Manufacturing with Causal Bayesian Networks and Knowledge Graphs

Deriving CERs from observational data is challenging and is based on assumptions that cannot be tested and are probably violated in manufacturing, as outlined in Section 2.5. At the same time, Section 3.2 presents an application of causal discovery to manufacturing using expert knowledge with promising results. Here, the search space of directed graphs could be considerably shrunken by expert knowledge.

This applied work presents how to acquire, represent, and update expert knowledge using a knowledge graph (KG), an interactive user interface (UI), and causal discovery. KGs are a flexible and state-of-the-art method to represent knowledge whose information can be queried using specialized languages.

In this way, we close the feedback loop between the process expert and the structure learning procedure. That is, process experts represent their state of knowledge in the KG using the UI. We propose to group the knowledge into different categories from which the constraints for the structure learning algorithm in the sense of Section 3.2 can be derived in an automated fashion. Under these constraints, the structure learning algorithm proposes a CER graph that is again stored in the KG. In turn, the CER graph is accessible by the process expert using the UI. She can challenge the proposed CERs with experiments or rely on her expertise. She updates the state of knowledge and feeds it back into the KG. At every point in time, the state of knowledge is standardized and accessible between multidisciplinary teams and automated reasoning systems. Thereby, we propose a solution to the fact that "[a]t the moment, causal discovery is mostly used on-demand for detection and analysis." We "tackle the topic of systematic integration of causal discovery in continuous process improvement" (Vuković and Thalmann, 2022).

From one perspective, the system can be understood as a recommendation system that proposes CERs from the current state of knowledge and process data. Another perspective is to consider the system as a human-in-the-loop reinforcement learning algorithm. The quality of the CER graphs is assessed by an expert and returned to the algorithm. A steady state is reached if the proposed CER graph does not contain edges that contradict the expert's knowledge. Along the way, the expert knowledge is challenged and possibly extended.

Finally, we show that an increase in incorporated expert knowledge reduces the computational complexity of the structure learning algorithm. Furthermore, the number of proposed CERs decreases, indicating that less spurious relations are proposed.

4 Discussion and Outlook

In this thesis, we addressed the insufficient understanding of CERs in the production of battery cells and storage systems, which results in high scrap rates. Since conducting experiments at the production plant can be challenging or expensive, we explored the use of data-driven techniques to discover knowledge from the manufacturing process data. The concept involves utilizing measurements of the (intermediate) products throughout the manufacturing process to characterize the production process of the end product. This results in a complex distribution. It was the goal to represent the complex distributions in an accessible fashion to enable multidisciplinary teams to interact with the derived representation.

As a framework for knowledge discovery, we identified PGMs because of the following benefits.

- PGMs **represent** complex data sets in an accessible visualisation, which can be used for exploratory analyses.
- The extensive **expert knowledge** in manufacturing can be integrated into PGMs.
- Directed PGMs can be used to derive **causal relationships**, that is, CERs, from observational data under assumptions. This avoids costly experiments.

To account for the complexity of the data-generating process, we focused on non-linear and flexible PGMs, which are the Gaussian copula model and the CAM. We combined them with important aspects of manufacturing.
For the former and undirected PGM, we investigated the estimation of the joint distribution when parts of the **data** are **missing** for example due to sensor breakdown.
For the latter, we explored how to efficiently **integrate manufacturing expertise** such as information on

1. the existence of CERs,
2. the absence of CERs, and
3. the temporal ordering of measurements

into the estimation algorithm and its benefit.
Finally, we present how **boosting** can be used for **causal discovery**. We show that the estimation is statistically consistent, which is a rare theoretical result in this field. A simulation study indicates that a practical adaptation of the algorithm outperforms state-of-the-art causal discovery methods based on maximum-likelihood estimation or gradient descent in relevant scenarios.

Although the results are promising, we remain cautious when it comes to deriving CERs from observational data due to the restrictive Assumption 2.5.21. In particular, the assumptions of no confounding, that is, the observational distribution aligns with the interventional distribution, and the joint distribution having a strictly positive density, are questionable in manufacturing applications.

We agree with Dawid (2010) that the representation as a DAG makes it difficult to resist the intuition of a causal or meaningful relationship. However, it is not clear why Assumption 2.5.21 holds. More precisely, why should a data generating process P_1 have the same interventional distributions as another data generating process P_2 when the two coincidentally share the same observational distribution? Throughout the thesis, we have presented several counterexamples for this assumption, and it seems likely that a given data generating process behind a manufacturing data set is violating Assumption 2.5.21. Furthermore, the assumption that the underlying causal model lies in an identifiable subclass is based on the intuition that models that can be described in simpler mathematical terms are more likely to be true (Peters et al., 2017, page 46). Although this perspective is popular in other fields such as physics, it is subject to criticism (Hossenfelder, 2018).

Therefore, we propose the utilization of causal discovery techniques in the manufacturing sector as a "CER recommendation tool" where suggestions must be validated by experiments or (deep) expert knowledge. In complex and high-dimensional scenarios, the reduction of potential CERs can already provide great benefit. This point of view is also shared by Dawid (2010); Vowels et al. (2022). In Section 3.4 we have elaborated on how such a tool can be designed and used for iterative knowledge discovery.

It can be argued that statistical models invariably simplify reality, yet they frequently offer dependable insights into real-world phenomena. However, one needs to assess empirically whether deviations from the assumptions are crucial for the conclusions drawn from the model. Thus, a major challenge for causal discovery is the lack of non-trivial data sets with known causal ground truth. The ground truth can be hard to determine as ambiguous concepts of causality and interventions exist concurrently and they can differ across research areas (Vowels et al., 2022). Nevertheless, these datasets

can be viewed as essential. For instance, in the field of computer vision, the availability of extensive labeled datasets has contributed significantly to advancements (O'Mahony et al., 2020).

Outlook We have evaluated the proposed Gaussian copula estimator for missing data with methods that directly estimate the joint distribution. In practice, however, multiple imputation is often applied with good results. Thus, it is desirable to compare the proposed approach with different imputation methods. The robustness with respect to a violation of the missing-at-random assumption can further be investigated. Using differentiable sampling could help to extend the method to higher dimensions.

For the CER recommendation tool, it would be desirable to have some uncertainty quantification of the estimated CERs using a Bayesian approach. Heckerman et al. (1995); Geiger and Heckerman (1994) show that for appropriate models and priors, the posterior probability of a graph can be calculated in closed form. Recent contributions such as Lorch et al. (2021) use variational inference to provide an approximate posterior over DAGs under less restrictive assumptions. The boosting-based procedure could be embedded in this framework. In the Bayesian context, the restriction of the potential graphs based on expert knowledge can be understood as assigning some graphs a vanishing prior probability. Using the uncertainty estimates, one could decide which experiments (whose costs potentially vary) should be run next. Furthermore, the posterior distribution over the DAGs can be used to provide probabilistic estimates or bounds for the CERs.

Furthermore, having a manufacturing data set with established causal relationships would be highly advantageous. However, the creation would be challenging and would involve defining the concept of causality, systematically gathering expert knowledge, and conducting experiments.

Bibliography

Assaad, C. K., Devijver, E., Gaussier, E., and Ait-Bachir, A. (2019). Scaling causal inference in additive noise models. In *ACM SIGKDD Workshop on Causal Discovery*, pages 22–33. PMLR.

Bartlett, P. L. and Mendelson, S. (2002). Rademacher and Gaussian complexities: Risk bounds and structural results. *The Journal of Machine Learning Research*, 3:463–482.

Bellot, A. and van der Schaar, M. (2019). Conditional independence testing using generative adversarial networks. In Wallach, H., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E., and Garnett, R., editors, *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc.

Billingsley, P. (1995). *Probability and Measure*. Wiley Series in Probability and Statistics. Wiley.

Borboudakis, G. and Tsamardinos, I. (2016). Towards robust and versatile causal discovery for business applications. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1435–1444.

Bühlmann, P. and Hothorn, T. (2007). Boosting algorithms: Regularization, prediction and model fitting. *Statistical Science*, 22(4):477–505.

Bühlmann, P., Peters, J., and Ernest, J. (2014). CAM: Causal additive models, high-dimensional order search and penalized regression. *The Annals of Statistics*, 42(6):2526–2556.

Bühlmann, P. and Yu, B. (2003). Boosting with the l2 loss. *Journal of the American Statistical Association*, 98(462):324–339.

Bühlmann, P. and Yu, B. (2010). Boosting. *WIREs Computational Statistics*, 2(1):69–74.

Charig, C. R., Webb, D. R., Payne, S. R., and Wickham, J. E. (1986). Comparison of treatment of renal calculi by open surgery, percutaneous nephrolithotomy, and extracorporeal shockwave lithotripsy. *British medical journal (Clinical research ed.)*, 292(6524):879.

Charpentier, B., Kibler, S., and Günnemann, S. (2022). Differentiable dag sampling. In *International Conference on Learning Representations*.

Colombo, D., Maathuis, M. H., Kalisch, M., and Richardson, T. S. (2012). Learning high-dimensional directed acyclic graphs with latent and selection variables. *The Annals of Statistics*, 40(1):294–321.

Constantinou, A. C., Guo, Z., and Kitson, N. K. (2023). The impact of prior knowledge on causal structure learning. *Knowledge and Information Systems*, 65:3385–3434.

Dawid, A. P. (1979). Conditional independence in statistical theory. *Journal of the Royal Statistical Society. Series B (Methodological)*, 41(1):1–31.

Dawid, A. P. (2010). Beware of the dag! In Guyon, I., Janzing, D., and Schölkopf, B., editors, *Proceedings of Workshop on Causality: Objectives and Assessment at NIPS 2008*, volume 6 of *The Proceedings of Machine Learning Research*, pages 59–86, Whistler, Canada. The Proceedings of Machine Learning Research.

de Campos, L. M. and Castellano, J. G. (2007). Bayesian network learning algorithms using structural restrictions. *International Journal of Approximate Reasoning*, 45(2):233–254.

Dempster, A., Laird, N., and Rubin, D. (1977). Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society: Series B (Methodological)*, 39(1):1–22.

Ding, W. and Song, P. (2016). EM algorithm in gaussian copula with missing data. *Computational Statistics & Data Analysis*, 101:1–11.

Drahoš, P., Kučera, E., Haffner, O., and Klimo, I. (2018). Trends in industrial communication and OPC UA. In *Cybernetics & Informatics (K&I)*, pages 1–5.

Fitzner, A., Abramowski, J.-P., Schmetz, A., Krauß, J., Borzutzki, K., Eckstein, M., Pouls, K., Baum, C., Schmitt, R., and Kampker, A. (2023). Cause-effect relationships in battery cell production - data based validation of expert knowledge in electrode production. *Procedia CIRP*, 120:469–474. 56th CIRP International Conference on Manufacturing Systems 2023.

Friedman, J., Hastie, T., and Tibshirani, R. (2007). Sparse inverse covariance estimation with the graphical lasso. *Biostatistics*, 9(3):432–441.

Friedman, N., Nachman, I., and Peér, D. (1999). Learning bayesian network structure from massive datasets: the "sparse candidate" algorithm. In *Proceedings of the Fifteenth Conference on Uncertainty in Artificial Intelligence*, pages 206–215. Morgan Kaufmann Publishers Inc.

Gaines, L., Dai, Q., Vaughey, J. T., and Gillard, S. (2021). Direct recycling R&D at the recell center. *Recycling*, 6(2).

Gao, Y., Shen, L., and Xia, S.-T. (2021). DAG-GAN: Causal structure learning with generative adversarial nets. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 3320–3324. IEEE.

Geiger, D. and Heckerman, D. (1994). Learning gaussian networks. In *Proceedings of the Tenth International Conference on Uncertainty in Artificial Intelligence*, pages 235–243. Morgan Kaufmann Publishers Inc.

Genest, C., Ghoudi, K., and Rivest, L.-P. (1995). A semiparametric estimation procedure of dependence parameters in multivariate families of distributions. *Biometrika*, 82(3):543–552.

Gretton, A., Fukumizu, K., Teo, C., Song, L., Schölkopf, B., and Smola, A. (2007). A kernel statistical test of independence. In Platt, J., Koller, D., Singer, Y., and Roweis, S., editors, *Advances in Neural Information Processing Systems*, volume 20, pages 585–592. Curran Associates, Inc.

Gretton, A., Spirtes, P., and Tillman, R. (2009). Nonlinear directed acyclic structure learning with weakly additive noise models. In Bengio, Y., Schuurmans, D., Lafferty, J., Williams, C., and Culotta, A., editors, *Advances in Neural Information Processing Systems*, volume 22, pages 1847–1855. Curran Associates, Inc.

Grießl, D., Adam, A., Huber, K., and Kwade, A. (2022). Effect of the slurry mixing process on the structural properties of the anode and the resulting fast-charging performance of the lithium-ion battery cell. *Journal of The Electrochemical Society*, 169(2).

Hasan, U. and Gani, M. O. (2022). Kcrl: A prior knowledge based causal discovery framework with reinforcement learning. In Lipton, Z., Ranganath, R., Sendak, M., Sjoding, M., and Yeung, S., editors, *Proceedings of the seventh Machine Learning*

for Healthcare Conference, volume 182 of *The Proceedings of Machine Learning Research*, pages 691–714. The Proceedings of Machine Learning Research.

Haughton, D. M. A. (1988). On the Choice of a Model to Fit Data from an Exponential Family. *The Annals of Statistics*, 16(1):342 – 355.

Heckerman, D., Geiger, D., and Chickering, D. M. (1995). Learning bayesian networks: The combination of knowledge and statistical data. *Machine Learning*, 20:197–243.

Hoff, P. (2007). Extending the rank likelihood for semiparametric copula estimation. *The Annals of Applied Statistics*, 1(1):265–283.

Hossenfelder, S. (2018). *Lost in Math: How Beauty Leads Physics Astray*. Basic Books.

Joe, H. (2014). *Dependence modeling with copulas*. CRC Press.

Kaiser, M. and Sipos, M. (2022). Unsuitability of notears for causal graph discovery when dealing with dimensional quantities. *Neural Processing Letters*, 54(3):1587–1595.

Kalisch, M. and Bühlman, P. (2007). Estimating high-dimensional directed acyclic graphs with the PC-algorithm. *The Journal of Machine Learning Research*, 8(3):613–636.

Kalisch, M., Mächler, M., Colombo, D., Maathuis, M. H., and Bühlmann, P. (2012). Causal inference using graphical models with the r package pcalg. *Journal of statistical software*, 47:1–26.

Kertel, M., Harmeling, S., and Pauly, M. (2022). Learning causal graphs in manufacturing domains using structural equation models. In *Fifth International Conference on Artificial Intelligence for Industries (AI4I)*, pages 14–19. IEEE.

Kertel, M. and Klein, N. (2024). Boosting causal additive models.

Kertel, M. and Pauly, M. (2022). Estimating gaussian copulas with missing data with and without expert knowledge. *Entropy*, 24(12).

Klenke, A. (2013). *Probability theory: a comprehensive course*. Springer Science & Business Media.

Kline, R. (2023). *Principles and Practice of Structural Equation Modeling*. Methodology in the Social Sciences Series. Guilford Publications.

Koller, D. and Friedman, N. (2009). *Probabilistic graphical models: principles and techniques*. MIT press.

Kornas, T., Daub, R., Karamat, M. Z., Thiede, S., and Herrmann, C. (2019). Data-and expert-driven analysis of cause-effect relationships in the production of lithium-ion batteries. In *15th International Conference on Automation Science and Engineering (CASE)*, pages 380–385. IEEE.

Kpotufe, S., Sgouritsa, E., Janzing, D., and Schölkopf, B. (2014). Consistency of causal inference under the additive noise model. In Xing, E. P. and Jebara, T., editors, *International Conference on Machine Learning*, volume 32 of *Proceedings of Machine Learning Research*, pages 478–486, Bejing, China. PMLR.

Lachapelle, S., Brouillard, P., Deleu, T., and Lacoste-Julien, S. (2019). Gradient-based neural DAG learning. In *International Conference on Learning Representations*.

Lauritzen, S. (1996). *Graphical Models*. Oxford Statistical Science Series. Clarendon Press.

Lee, K.-Y., Liu, T., Li, B., and Zhao, H. (2020). Learning causal networks via additive faithfulness. *Journal of Machine Learning Research*, 21(51):1–38.

Liu, H., Han, F., Yuan, M., Lafferty, J., and Wasserman, L. (2012). High-dimensional semiparametric gaussian copula graphical models. *The Annals of Statistics*, 40(4):2293–2326.

Liu, H., Lafferty, J., and Wasserman, L. (2009). The nonparanormal: Semiparametric estimation of high dimensional undirected graphs. *The Journal of Machine Learning Research*, 10(80):2295–2328.

Liu, Y., Zhang, R., Wang, J., and Wang, Y. (2021). Current and future lithium-ion battery manufacturing. *iScience*, 24(4):102332.

Lorch, L., Rothfuss, J., Schölkopf, B., and Krause, A. (2021). Dibs: Differentiable Bayesian structure learning. *Advances in Neural Information Processing Systems*, 34:24111–24123.

Meek, C. (1995). Causal inference and causal explanation with background knowledge. In *Proceedings of the Eleventh Conference on Uncertainty in Artificial Intelligence*, pages 403—-410. Morgan Kaufmann Publishers Inc.

Mooij, J., Janzing, D., Peters, J., and Schölkopf, B. (2009). Regression by dependence minimization and its application to causal inference in additive noise models. In *International Conference on Machine Learning*, pages 745—-752, New York, NY, USA. Association for Computing Machinery.

Nelsen, R. B. (2006). *Dependence*. Springer.

Ng, I., Zhu, S., Fang, Z., Li, H., Chen, Z., and Wang, J. (2022). Masked gradient-based causal structure learning. In *International Conference on Data Mining (SDM)*, pages 424–432. SIAM.

Nowzohour, C. and Bühlmann, P. (2016). Score-based causal learning in additive noise models. *Statistics*, 50(3):471–485.

Ogarrio, J. M., Spirtes, P., and Ramsey, J. (2016). A hybrid causal search algorithm for latent variable models. In *Proceedings of the Eighth International Conference on Probabilistic Graphical Models*, volume 52 of *The Proceedings of Machine Learning Research*, pages 368–379, Lugano, Switzerland. The Proceedings of Machine Learning Research.

O'Mahony, N., Campbell, S., Carvalho, A., Harapanahalli, S., Hernandez, G. V., Krpalkova, L., Riordan, D., and Walsh, J. (2020). Deep learning vs. traditional computer vision. In Arai, K. and Kapoor, S., editors, *Advances in Computer Vision*, pages 128–144, Cham. Springer International Publishing.

Pearl, J. (1995). Causal diagrams for empirical research. *Biometrika*, 82(4):669–688.

Pearl, J. (2009a). Causal inference in statistics: An overview. *Statistics Surveys*, 3:96–146.

Pearl, J. (2009b). *Causality: Models, Reasoning and Inference*. Cambridge University Press, Cambridge, 2nd edition.

Pearl, J. (2010). Causal inference. In Guyon, I., Janzing, D., and Schölkopf, B., editors, *Proceedings of Workshop on Causality: Objectives and Assessment at NIPS 2008*, volume 6 of *The Proceedings of Machine Learning Research*, pages 39–58, Whistler, Canada. The Proceedings of Machine Learning Research.

Peters, J., Janzing, D., and Schölkopf, B. (2017). *Elements of Causal Inference: Foundations and Learning Algorithms*. MIT Press.

Peters, J., Mooij, J. M., Janzing, D., and Schölkopf, B. (2014). Causal discovery with continuous additive noise models. *The Journal of Machine Learning Research*, 15(1):2009–2053.

Rai, R., Tiwari, M. K., Ivanov, D., and Dolgui, A. (2021). Machine learning in manufacturing and industry 4.0 applications. *International Journal of Production Research*, 59(16):4773–4778.

Raskutti, G., Wainwright, M. J., and Yu, B. (2014). Early stopping and non-parametric regression: an optimal data-dependent stopping rule. *The Journal of Machine Learning Research*, 15(1):335–366.

Raskutti, G., Yu, B., Wainwright, M. J., and Ravikumar, P. (2008). Model selection in gaussian graphical models: High-dimensional consistency of l1-regularized MLE. In Koller, D., Schuurmans, D., Bengio, Y., and Bottou, L., editors, *Advances in Neural Information Processing Systems*, volume 21. Curran Associates, Inc.

Reisach, A. G., Seiler, C., and Weichwald, S. (2021). Beware of the simulated dag! causal discovery benchmarks may be easy to game. *Advances in Neural Information Processing Systems*, 34:27772–27784.

Robins, J. M., Scheines, R., Spirtes, P., and Wasserman, L. (2003). Uniform consistency in causal inference. *Biometrika*, 90(3):491–515.

Román-Ramírez, L. and Marco, J. (2022). Design of experiments applied to lithium-ion batteries: A literature review. *Applied Energy*, 320:119305.

Schapire, R. E. and Freund, Y. (2012). *Boosting: Foundations and Algorithms*. MIT Press.

Schnell, J. and Reinhart, G. (2016). Quality management for battery production: A quality gate concept. *Procedia CIRP*, 57:568–573. Factories of the Future in the digital environment - Proceedings of the 49th CIRP Conference on Manufacturing Systems.

Schölkopf, B. and Smola, A. J. (2001). *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. MIT Press.

Scott, D. and Sain, S. (2005). Multidimensional density estimation. *Handbook of Statistics*, 24:229–261.

Shah, R. and Peters, J. (2018). The hardness of conditional independence testing and the generalised covariance measure. *Annals of Statistics*, 48(3):1514–1538.

Shimizu, S., Hoyer, P. O., Hyvärinen, A., and Kerminen, A. (2006). A linear non-gaussian acyclic model for causal discovery. *The Journal of Machine Learning Research*, 7(72):2003–2030.

Shojaie, A. and Michailidis, G. (2010). Penalized likelihood methods for estimation of sparse high-dimensional directed acyclic graphs. *Biometrika*, 97(3):519–538.

Sklar, A. (1959). Fonctions de repartition an dimensions et leurs marges. *Publ. inst. statist. univ. Paris*, 8:229–231.

Spirtes, P., Glymour, C., and Scheines, R. (1993). *Causation, Prediction, and Search*. MIT Press.

Spirtes, P. and Zhang, K. (2016). Causal discovery and inference: concepts and recent methodological advances. *Applied Informatics*, 3(1):1–28.

Strobl, E. V., Zhang, K., and Visweswaran, S. (2019). Approximate kernel-based conditional independence tests for fast non-parametric causal discovery. *Journal of Causal Inference*, 7(1):20180017.

Sun, H. (2005). Mercer theorem for RKHS on noncompact sets. *Journal of Complexity*, 21(3):337–349.

Teh, H. Y., Kempa-Liehr, A. W., and Wang, K. I.-K. (2020). Sensor data quality: A systematic review. *Journal of Big Data*, 7(1):1–49.

Teyssier, M. and Koller, D. (2005). Ordering-based search: a simple and effective algorithm for learning bayesian networks. In *Proceedings of the Twenty-First Conference on Uncertainty in Artificial Intelligence*, pages 584–590. AUAI Press.

Tibshirani, R. (1996). Regression shrinkage and selection via the LASSO. *Journal of the Royal Statistical Society. Series B (Methodological)*, 58(1):267–288.

Tsamardinos, I., Brown, L. E., and Aliferis, C. F. (2006). The max-min hill-climbing bayesian network structure learning algorithm. *Machine Learning*, 65:31–78.

Tu, R., Zhang, C., Ackermann, P., Mohan, K., Kjellström, H., and Zhang, K. (2019). Causal discovery in the presence of missing data. In Chaudhuri, K. and Sugiyama, M., editors, *Proceedings of the Twenty-Second International Conference on Artificial Intelligence and Statistics*, volume 89, pages 1762–1770. The Proceedings of Machine Learning Research.

Tutz, G. and Binder, H. (2006). Generalized additive modeling with implicit variable selection by likelihood-based boosting. *Biometrics*, 62(4):961–971.

Uhler, C., Raskutti, G., Bühlmann, P., and Yu, B. (2013). Geometry of the faithfulness assumption in causal inference. *The Annals of Statistics*, 41(2):436–463.

van de Geer, S. (2014). On the uniform convergence of empirical norms and inner products, with application to causal inference. *Electronic Journal of Statistics*, 8(1):543–574.

Van Der Vaart, A. W. and Wellner, J. A. (1996). *Weak convergence*. Springer.

Vowels, M. J., Camgoz, N. C., and Bowden, R. (2022). D'ya like dags? a survey on structure learning and causal discovery. *ACM Computing Surveys*, 55(4):1–36.

Vuković, M. and Thalmann, S. (2022). Causal discovery in manufacturing: A structured literature review. *Journal of Manufacturing and Materials Processing*, 6(1):1–10.

Wahba, G. (1990). *Spline Models for Observational Data*. Society for Industrial and Applied Mathematics.

Wainwright, M. J. (2019). *High-Dimensional Statistics: A Non-Asymptotic Viewpoint*. Cambridge Series in Statistical and Probabilistic Mathematics. Cambridge University Press.

Wang, Y. S. and Drton, M. (2023). Causal discovery with unobserved confounding and non-gaussian data. *The Journal of Machine Learning Research*, 24(271):1–61.

Wehner, C., Kertel, M., and Wewerka, J. (2023). Interactive and intelligent root cause analysis in manufacturing with causal bayesian networks and knowledge graphs. In *97th Vehicular Technology Conference (VTC2023-Spring)*, pages 1–7. IEEE.

Westermeier, M., Reinhart, G., and Steber, M. (2014). Complexity management for the start-up in lithium-ion cell production. *Procedia CIRP*, 20:13–19. Second International Conference on Ramp-Up Management.

Westermeier, M., Reinhart, G., and Zeilinger, T. (2013). Method for quality parameter identification and classification in battery cell production quality planning of complex production chains for battery cells. In *Third International Electric Drives Production Conference*, pages 1–10.

Wood, S. N. (2006). *Generalized additive models: an introduction with R*. Chapman and Hall/CRC, 2nd edition.

Yu, Y., Chen, J., Gao, T., and Yu, M. (2019). DAG-GNN: DAG structure learning with graph neural networks. In Chaudhuri, K. and Salakhutdinov, R., editors, *International Conference on Machine Learning*, volume 97, pages 7154–7163. The Proceedings of Machine Learning Research.

Zhang, K., Peters, J., Janzing, D., and Schölkopf, B. (2011). Kernel-based conditional independence test and application in causal discovery. In *Proceedings of the Twenty-Seventh Conference on Uncertainty in Artificial Intelligence*, pages 804–813. AUAI Press.

Zheng, X., Aragam, B., Ravikumar, P. K., and Xing, E. P. (2018). Dags with no tears: Continuous optimization for structure learning. In Bengio, S., Wallach, H., Larochelle, H., Grauman, K., Cesa-Bianchi, N., and Garnett, R., editors, *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc.

Zheng, X., Dan, C., Aragam, B., Ravikumar, P., and Xing, E. (2020). Learning sparse nonparametric dags. In *International Conference on Artificial Intelligence and Statistics*, pages 3414–3425. The Proceedings of Machine Learning Research.

Zou, H. (2006). The adaptive LASSO and its oracle properties. *Journal of the American statistical association*, 101(476):1418–1429.

Örüm Aydin, A., Zajonz, F., Günther, T., Dermenci, K. B., Berecibar, M., and Urrutia, L. (2023). Lithium-ion battery manufacturing: Industrial view on processing challenges, possible solutions and recent advances. *Batteries*, 9(11).

Part II

Publications

Article 1

Kertel and Pauly (2022)

entropy

IMPACT FACTOR 2.7

Indexed in: PubMed

CITESCORE 4.7

Article

Estimating Gaussian Copulas with Missing Data with and without Expert Knowledge

Maximilian Kertel and Markus Pauly

Special Issue
Improving Predictive Models with Expert Knowledge

Edited by
Prof. Dr. Markus Pauly

MDPI

https://doi.org/10.3390/e24121849

entropy

MDPI

Article

Estimating Gaussian Copulas with Missing Data with and without Expert Knowledge

Maximilian Kertel [1,2,*] and Markus Pauly [2,3]

[1] BMW Group, Battery Cell Competence Centre, 80788 Munich, Germany
[2] Department of Statistics, TU Dortmund University, 44227 Dortmund, Germany
[3] Research Center Trustworthy Data Science and Security, UA Ruhr, 44227 Dortmund, Germany
* Correspondence: maximilian.kertel@bmw.de

Abstract: In this work, we present a rigorous application of the Expectation Maximization algorithm to determine the marginal distributions and the dependence structure in a Gaussian copula model with missing data. We further show how to circumvent a priori assumptions on the marginals with semiparametric modeling. Further, we outline how expert knowledge on the marginals and the dependency structure can be included. A simulation study shows that the distribution learned through this algorithm is closer to the true distribution than that obtained with existing methods and that the incorporation of domain knowledge provides benefits.

Keywords: missing at random; expert knowledge; expectation maximization; semiparametric estimation

check for updates

Citation: Kertel, M.; Pauly, M. Estimating Gaussian Copulas with Missing Data with and without Expert Knowledge. *Entropy* **2022**, *24*, 1849. https://doi.org/10.3390/e24121849

Academic Editor: Boris Ryabko

Received: 27 October 2022
Accepted: 10 December 2022
Published: 19 December 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

1. Introduction

Even though the amount of data is increasing due to new technologies, big data are by no means good data. For example, missing values are ubiquitous in various fields, from the social sciences [1] to manufacturing [2]. For explanatory analysis or decision making, one is often interested in the joint distribution of a multivariate dataset, and its estimation is a central topic in statistics [3]. At the same time, there exists background knowledge in many domains that can help to compensate for the potential shortcomings of datasets. For instance, domain experts have an understanding of the causal relationships in the data generation process [4]. It is the scope of this paper to unify expert knowledge and datasets with missing data to derive approximations of the underlying joint distribution.

To estimate the multivariate distribution, we use copulas, where the dependence structure is assumed to belong to a parametric family, while the marginals are estimated nonparametrically. Genest et al. [5] showed that for complete datasets, a two-step approach consisting of the estimation of the marginals with an empirical cumulative distribution function (ecdf) and subsequent derivation of the dependence structure is consistent. This idea is even transferable to high dimensions [6].

In the case of missing values, the situation becomes more complex. Here, nonparametric methods do not scale well with the number of dimensions [7]. On the other hand, assuming that the distribution belongs to a parametric family, it can often be derived by using the EM algorithm [8]. However, this assumption is, in general, restrictive. Due to the encouraging results for complete datasets, there have been several works that have investigated the estimation of the joint distribution under a copula model. The authors of [9,10] even discussed the estimation in a missing-not-at-random (MNAR) setting. While MNAR is less restrictive than missing at random (MAR), it demands the explicit modeling of the missing mechanism [11]. On the contrary, the authors of [12,13] provided results in cases in which data were missing completely at random (MCAR). This strong assumption is rarely fulfilled in practice. Therefore, we assume an MAR mechanism in what follows [11].

Another interesting contribution [14] assumed external covariates, such that the probability of a missing value depended exclusively on them and not on the variables under

investigation. They applied inverse probability weighting (IPW) and the two-step approach of [5]. While they proved a consistent result, it is unclear how this approach can be adapted to a setting without those covariates. IPW for general missing patterns is computationally demanding, and no software exists [15,16]. Thus, IPW is mostly applied with monotone missing patterns that appear, for example, in longitudinal studies [17]. The popular work of [18] proposed an EM algorithm in order to derive the joint distribution in a Gaussian copula model with data MAR [11]. However, their approach had weaknesses:

1. The presented algorithm was inexact. Among other things, the algorithm simplified by assuming that the marginals and the copula could be estimated separately (compare Equation (6) in [18] and Equation (11) in this paper).
2. If there was no a priori knowledge of the parametric family of all marginals, Ref. [18] proposed using the ecdf of the observed data points. Afterwards, they exclusively derived the parameters of the copula. This estimator of the marginals was biased [19,20], which is often overlooked in the copula literature, e.g., [21] (Section 4.3), [22] (Section 3), [23] (Section 3), or [24] (Section 3).
3. The description of the simulation study was incomplete and the results were not reproducible.

The aim of this paper is to close these gaps, and our contributions are the following:

1. We give a rigorous derivation of the EM algorithm under a Gaussian copula model. Similarly to [5], it consists of two separate steps, which estimate the marginals and the copula, respectively. However, these two steps alternate.
2. We show how prior knowledge about the marginals and the dependency structure can be utilized in order to achieve better results.
3. We propose a flexible parametrization of the marginals when a priori knowledge is absent. This allows us to learn the underlying marginal distributions; see Figure 1.
4. We provide a Python library that implements the proposed algorithm.

The structure of this paper is as follows. In Section 2, we review some background information about the Gaussian copula. We proceed by presenting the method (Section 3). In Section 4, we investigate its performance and the effect of domain knowledge in simulation studies. We conclude in Section 5. All technical aspects and proofs in this paper are given in Appendices A and B.

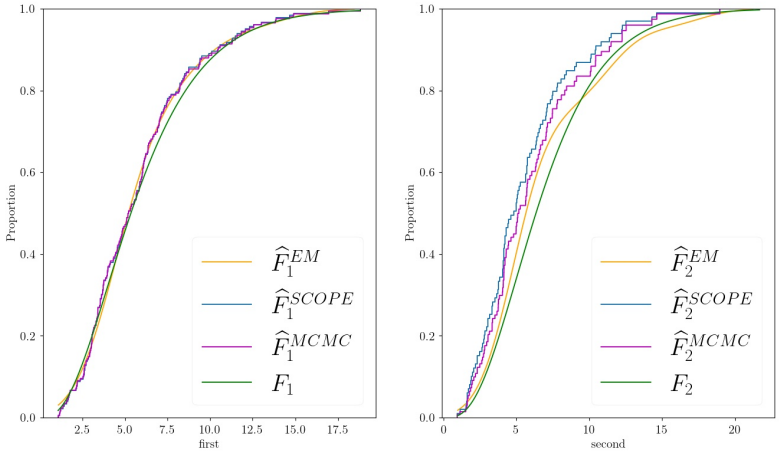

Figure 1. Estimates of the proposed EM algorithm ($\widehat{F}_i^{EM}$, orange line), the Standard Copula Estimator ($\widehat{F}_i^{SCOPE}$, blue line, corresponds to ecdf), the Markov chain–Monte Carlo approach ($\widehat{F}_i^{MCMC}$, purple line) for the marginals X_i, $i = 1, 2$, and the truth (F_i, green line) of a two-dimensional example dataset generated as described in Section 4.2 with $N = 200$, $\rho = 0.5$, and $\beta = (0, 2)$.

2. The Gaussian Copula Model

2.1. Notation and Assumptions

In the following, we consider a p-dimensional dataset $\{\mathbf{x}^1, \dots, \mathbf{x}^N\} \subset \mathbb{R}^p$ of size N, where $\mathbf{x}^1, \dots, \mathbf{x}^N$ are i.i.d. samples from a p-dimensional random vector $\mathbf{X} = (X_1, \dots, X_p)$ with a joint distribution function F and marginal distribution functions $F_1, \dots, F_p$. We denote the entries of $\mathbf{x}^\ell$ by $\mathbf{x}^\ell = \left(x_1^\ell, \dots, x_p^\ell\right) \forall \ell = 1, \dots, N$. The parameters of the marginals are represented by $\theta = (\theta_1, \dots, \theta_p)$, where θ_j is the parameter of F_j, so we write $F_j^{\theta_j}$, where θ_j can be a vector itself.

For $\ell \in \{1, \dots, p\}$, we define $\mathbf{obs}(\ell) \subset \{1, \dots, p\}$ as the index set of the observed and $\mathbf{mis}(\ell) \subset \{1, \dots, p\}$ as the index set of the missing columns of $\mathbf{x}^\ell$. Hence, $\mathbf{mis}(\ell) \cup \mathbf{obs}(\ell) = \{1, \dots, p\}$ and $\mathbf{mis}(\ell) \cap \mathbf{obs}(\ell) = \emptyset$. $\mathbf{R} = (R_1, \dots, R_p) \in \{0,1\}^p$ is a random vector for which $R_i = 0$ if X_i is missing and $R_i = 1$ if X_i can be observed. Further, we define ϕ to be the density function and Φ to be the distribution function of the one-dimensional standard normal distribution. $\Phi_{\boldsymbol{\mu},\Sigma}$ stands for the distribution function of a p-variate normal distribution with covariance $\Sigma \in \mathbb{R}^{p \times p}$ and mean $\boldsymbol{\mu} \in \mathbb{R}^p$. To simplify the notation, we define $\Phi_\Sigma := \Phi_{0,\Sigma}$. For a matrix $A \in \mathbb{R}^{p \times p}$, the entry of the i-th row and the j-th column is denoted by A_{ij}, while for index sets $\mathbf{S}, \mathbf{T} \subset \{1, \dots, p\}$, $A_{\mathbf{S},\mathbf{T}}$ is the submatrix of A with the row number in $\mathbf{S}$ and column number in $\mathbf{T}$. For a (random) vector $\mathbf{x}$ ($\mathbf{X}$), $\mathbf{x_S}$ ($\mathbf{X_S}$) is the subvector containing entries with the index in $\mathbf{S}$.

Throughout, we assume F to be strictly increasing and continuous in every component. Therefore, F_j is strictly increasing and continuous for all $j \in \{1, \dots, p\}$, and so is the existing inverse function F_j^{-1}. For $\mathbf{S} = \{s_1, \dots, s_k\} \subset \{1, \dots, p\}$, we define $F_\mathbf{S} : \mathbf{R}^{|S|} \to \mathbf{R}^{|S|}$ by

$$F_\mathbf{S}(x_{s_1}, \dots, x_{s_k}) = \big(F_{s_1}(x_{s_1}), \dots, F_{s_k}(x_{s_k})\big).$$

This work assumes that data are Missing at Random (MAR), as defined by [11], i.e.,

$$\mathbb{P}_{\mathbf{X},\mathbf{R}}(\mathbf{R} = \mathbf{r} | \mathbf{X_r} = \mathbf{x_{-r}}, \mathbf{X_r} = \mathbf{x_r}) = \mathbb{P}_{\mathbf{X},\mathbf{R}}(\mathbf{R} = \mathbf{r} | \mathbf{X_r} = \mathbf{x_r}), \tag{1}$$

where $\mathbf{X_r} := \mathbf{X}_{\{i:\, \mathbf{r}_i = 1\}}$ are the observed and $\mathbf{X_{-r}} := \mathbf{X}_{\{i:\, \mathbf{r}_i = 0\}}$ are the missing entries of $\mathbf{X}$.

2.2. Properties

Sklar's theorem [25] decomposes F into its marginals $F_1, \dots, F_p$ and its dependency structure C with

$$F(x_1, \dots, x_p) = C\big(F_1(x_1), \dots, F_p(x_p)\big). \tag{2}$$

Here, C is a copula, which means it is a p-dimensional distribution function with support $[0,1]^p$ whose marginal distributions are uniform. In this paper, we focus on Gaussian copulas, where

$$C_\Sigma(u_1, \dots, u_p) = \Phi_\Sigma\Big(\Phi^{-1}(u_1), \dots, \Phi^{-1}(u_p)\Big) \tag{3}$$

and Σ is a covariance matrix with $\Sigma_{jj} = 1 \; \forall j \in \{1, \dots, p\}$. Beyond all multivariate normal distributions, there are distributions with non-normal marginals whose copula is Gaussian. Hence, the Gaussian copula model provides an extension of the normality assumption. Consider a random vector $\mathbf{X}$ whose copula is C_Σ. Under the transformation

$$\mathbf{Z} := \Phi^{-1} \circ F(\mathbf{X}) := \Big(\Phi^{-1} \circ F_1(X_1), \dots, \Phi^{-1} \circ F_p\big(X_p\big)\Big),$$

it holds that

$$\begin{aligned} F_{\mathbf{Z}}(z_1 \dots, z_p) &= \mathbb{P}(Z_1 \leq z_1, \dots, Z_p \leq z_p) \\ &= \mathbb{P}\Big(X_1 \leq F_1^{-1}(\Phi(z_1)), \dots, X_p \leq F_p^{-1}(\Phi(z_p))\Big) \\ &= \Phi_\Sigma\Big(\Phi^{-1}\Big(F_1\Big(F_1^{-1}(\Phi(z_1))\Big)\Big), \dots, \Phi^{-1}\Big(F_p\Big(F_p^{-1}(\Phi(z_p))\Big)\Big)\Big) \\ &= \Phi_\Sigma(z_1, \dots, z_p) \end{aligned} \tag{4}$$

and hence, $\mathbf{Z}$ is normally distributed with mean 0 and covariance Σ. The two-step approaches given in [5,6] use this property and apply the following scheme:

1. Find consistent estimates $\hat{F}_1, \dots, \hat{F}_p$ for the marginal distributions $F_1, \dots, F_p$.
2. Find Σ by estimating the covariance of the random vector

$$\mathbf{Z} = \Big(\Phi^{-1}\Big(\widehat{F}_1(X_1)\Big), \dots, \Phi^{-1}\Big(\widehat{F}_p(X_p)\Big)\Big).$$

From now on, we assume that the marginals of $\mathbf{X}$ have existing density functions $f_1, \dots, f_p$. Then, by using Equation (4) and a change of variables, we can derive the joint density function

$$f_{F_1,\dots,F_p,\Sigma}(x_1, \dots, x_p) = f(x_1, \dots, x_p) = |\Sigma|^{-\frac{1}{2}} \exp\Big(-\frac{1}{2}\mathbf{z}^T\Big(\Sigma^{-1} - I\Big)\mathbf{z}\Big) \prod_{j=1}^{p} f_j(x_j), \tag{5}$$

where $\mathbf{z} := \big(\Phi^{-1}(F_1(x_1)), \dots, \Phi^{-1}(F_p(x_p))\big)$. As for the multivariate normal distribution, we can identify the conditional independencies ([6]) from the inverse of the covariance matrix $K := \Sigma^{-1}$ by using the property

$$K_{jk} = K_{kj} = 0 \iff X_j \perp X_k | \{X_i : i \in \{1, \dots, p\} \setminus \{j, k\}\}. \tag{6}$$

K is called the precision matrix. In order to slim down the notation, we define

$$\Phi^{-1}(F_{\mathbf{S}}(\mathbf{x}_{\mathbf{S}})) := \Big(\Phi^{-1}(F_{s_1}(x_{s_1})), \dots, \Phi^{-1}(F_{s_k}(x_{s_k}))\Big)$$

and similarly

$$F_{\mathbf{S}}^{-1}(\Phi(\mathbf{z}_{\mathbf{S}})) := \Big(F_{s_1}^{-1}(\Phi(z_{s_1})), \dots, F_{s_1}^{-1}(\Phi(z_{s_k}))\Big).$$

The former function transforms the data of a Gaussian copula distribution to be normally distributed. The latter mapping takes multivariate normally distributed data and returns data following a Gaussian copula distribution with marginals $F_{s_1}, \dots, F_{s_k}$. The conditional density functions have a closed form.

Proposition 1 (Conditional Distribution of Gaussian Copula)**.** *Let* $\mathbf{S} = \{s_1, \dots, s_k\}$ *and* $\mathbf{T} = \{t_1, \dots, t_{k'}\}$ *be such that* $\mathbf{T} \dot{\cup} \mathbf{S} = \{1, \dots, p\}$.

1. *The conditional density of* $\mathbf{X}_{\mathbf{T}} | \mathbf{X}_{\mathbf{S}} = \mathbf{x}_{\mathbf{S}}$ *is given by*

$$f(\mathbf{x}_{\mathbf{T}} | \mathbf{X}_{\mathbf{S}} = \mathbf{x}_{\mathbf{S}}) = |\Sigma'|^{-\frac{1}{2}} \exp\Big(-\frac{1}{2}(\mathbf{z}_{\mathbf{T}} - \boldsymbol{\mu})^T \Sigma'^{-1}(\mathbf{z}_{\mathbf{T}} - \boldsymbol{\mu})\Big) \exp\Big(\frac{1}{2}\mathbf{z}_{\mathbf{T}}^T \mathbf{z}_{\mathbf{T}}\Big) \prod_{j \in \mathbf{T}} f_j(x_j),$$

   where $\boldsymbol{\mu} = \Sigma_{\mathbf{T},\mathbf{S}}\Sigma_{\mathbf{S},\mathbf{S}}^{-1}\mathbf{z}_{\mathbf{S}}$, $\Sigma' = \Sigma_{\mathbf{T},\mathbf{T}} - \Sigma_{\mathbf{T},\mathbf{S}}\Sigma_{\mathbf{S},\mathbf{S}}^{-1}\Sigma_{\mathbf{S},\mathbf{T}}$, $\mathbf{z}_{\mathbf{T}} = \Phi^{-1}(F_{\mathbf{T}}(\mathbf{x}_{\mathbf{T}}))$ *and* $\mathbf{z}_{\mathbf{S}} = \Phi^{-1}(F_{\mathbf{S}}(\mathbf{x}_{\mathbf{S}}))$.
2. $\Phi^{-1}(F_{\mathbf{T}}(\mathbf{X}_{\mathbf{T}}))|\mathbf{X}_{\mathbf{S}} = \mathbf{x}_{\mathbf{S}}$ *is normally distributed with mean* $\boldsymbol{\mu}$ *and covariance* Σ'.
3. *The expectation of* $h(\mathbf{X}_{\mathbf{T}})$ *with respect to the density* $f(\mathbf{x}_{\mathbf{T}}|\mathbf{X}_{\mathbf{S}} = \mathbf{x}_{\mathbf{S}})$ *can be expressed by*

$$\int h(\mathbf{x}_{\mathbf{T}}) f(\mathbf{x}_{\mathbf{T}} | \mathbf{X}_{\mathbf{S}} = \mathbf{x}_{\mathbf{S}}) d\mathbf{x}_{\mathbf{T}} = \int h\Big(F_{\mathbf{T}}^{-1}(\Phi(\mathbf{z}_{\mathbf{T}}))\Big) \phi_{\boldsymbol{\mu},\Sigma'}(\mathbf{z}_{\mathbf{T}}) d\mathbf{z}_{\mathbf{T}}.$$

Proposition 1 shows that the conditional distribution's copula is Gaussian as well. More importantly, we can derive an algorithm for sampling from the conditional distribution.

Algorithm 1: Sampling from the conditional distribution of a Gaussian copula

Input: $\mathbf{x_S}, \Sigma, F_1, \ldots, F_p$
Result: m samples of $\mathbf{X_T}|\mathbf{X_S} = \mathbf{x_S}$
Calculate $\mathbf{z_S} := \Phi^{-1}(F_\mathbf{S}(\mathbf{x_S}))$;
Calculate $\boldsymbol{\mu}$ and Σ' as in Proposition 1 using $\mathbf{z_S}$ and Σ;
Draw samples $\{\mathbf{z}^1, \ldots, \mathbf{z}^m\}$ from $\mathcal{N}(\mu, \Sigma')$;
return $\{F_\mathbf{T}^{-1}(\Phi(\mathbf{z}^1)), \ldots, F_\mathbf{T}^{-1}(\Phi(\mathbf{z}^m))\}$

The very last step follows with Proposition 1, as it holds for any measurable $A \subset \mathbb{R}^{k'}$:

$$\mathbb{P}(\mathbf{X_T} \in A|\mathbf{X_S} = \mathbf{x_S}) = \int 1_A(\mathbf{x_T}) f(\mathbf{x_T}|\mathbf{X_S} = \mathbf{x_S}) d\mathbf{x_T} = \int 1_A\left(F_\mathbf{T}^{-1}(\Phi(\mathbf{z_T}))\right) \phi_{\boldsymbol{\mu},\Sigma'}(\mathbf{z_T}) d\mathbf{z_T}.$$

3. The EM Algorithm in the Gaussian Copula Model

3.1. The EM Algorithm

Let $\{\mathbf{y}^1, \ldots, \mathbf{y}^N\} \subset \mathbb{R}^p$ be a dataset following a distribution with parameter ψ and corresponding density function $g_\psi(\cdot)$, where observations are MAR. The EM algorithm [8] finds a local optimum of the log-likelihood function

$$\begin{aligned}\sum_{\ell=1}^N \ln\left(g_\psi\left(\mathbf{y}^\ell_{\mathbf{obs}(\ell)}\right)\right) &= \sum_{\ell=1}^N \int \ln\left(g_\psi\left(\left(\mathbf{y}^\ell_{\mathbf{obs}(\ell)}, \mathbf{y}_{\mathbf{mis}(\ell)}\right)\right)\right) \\ &\qquad g_\psi\left(\mathbf{y}_{\mathbf{mis}(\ell)}|\mathbf{Y}^\ell_{\mathbf{obs}(\ell)} = \mathbf{y}^\ell_{\mathbf{obs}(\ell)}\right) d\mathbf{y}_{\mathbf{mis}(\ell)} \\ &= \sum_{\ell=1}^N \mathbb{E}_\psi\left(\ln\left(g_\psi\left(\left(\mathbf{y}_{\mathbf{obs}(\ell)}, \mathbf{y}_{\mathbf{mis}(\ell)}\right)\right)\right)|\mathbf{Y}^\ell_{\mathbf{obs}} = \mathbf{y}^\ell_{\mathbf{obs}(\ell)}\right).\end{aligned}$$

After choosing a start value ψ^0, it does so by iterating the following two steps.

1. E-Step: Calculate

$$\begin{aligned}\lambda(\psi|\mathbf{y}^1, \ldots, \mathbf{y}^N, \psi^t) &:= \sum_{\ell=1}^N \mathbb{E}_{\psi^t}\left(\ln\left(g_\psi\left(\left(\mathbf{y}_{\mathbf{obs}(\ell)}, \mathbf{y}_{\mathbf{mis}(\ell)}\right)\right)\right)|\mathbf{Y}^\ell_{\mathbf{obs}} = \mathbf{y}^\ell_{\mathbf{obs}(\ell)}\right) \\ &= \sum_{\ell=1}^N \lambda(\psi|\mathbf{y}^\ell, \psi^t).\end{aligned} \tag{7}$$

2. M-Step: Set

$$\psi^{t+1} = \operatorname*{argmax}_\psi \lambda(\psi|\mathbf{y}^1, \ldots, \mathbf{y}^N, \psi^t) \tag{8}$$

and $t = t + 1$.

For our purposes, there are two extensions of interest:

- If there is no closed formula for the right-hand side of Equation (7), one can apply Monte Carlo integration [26] as an approximation. This is called the Monte Carlo EM algorithm.
- If $\psi = (\psi_1, \ldots, \psi_v)$ and the joint maximization of (8) with respect to ψ is not feasible, Ref. [27] proposed a sequential maximization. Thus, we optimize (8) with respect to ψ_i while holding $\psi_1 = \psi_1^{t+1}, \ldots, \psi_{i-1} = \psi_{i-1}^{t+1}, \psi_{i+1} = \psi_{i+1}^t, \ldots, \psi_v = \psi_v^t$ fixed before we continue with ψ_{i+1}. This is called the Expectation Conditional Maximization (ECM) algorithm.

3.2. Applying the ECM Algorithm on the Gaussian Copula Model

As we need a full parametrization of the Gaussian copula model for the EM algorithm, we assume parametric marginal distributions $F_1^{\theta_1}, \ldots, F_p^{\theta_p}$ with densities $f_1^{\theta_1}, \ldots, f_p^{\theta_p}$. According to Equation (5), the joint density with respect to the parameters $\theta = (\theta_1, \ldots, \theta_p)$ and Σ has the form

$$f_{\theta,\Sigma}(x_1, \ldots, x_p) = |\Sigma|^{-\frac{1}{2}} \exp\left(-\frac{1}{2}\mathbf{z}_\theta^T\left(\Sigma^{-1} - I\right)\mathbf{z}_\theta\right)\prod_{j=1}^{p} f_j^{\theta_j}(x_j), \tag{9}$$

where $\mathbf{z}_\theta := \left(\Phi^{-1}\left(F_1^{\theta_1}(x_1)\right), \ldots, \Phi^{-1}\left(F_p^{\theta_p}(x_p)\right)\right)$. Section 3.3 will describe how we can keep the flexibility for the marginals despite the parametrization. However, first, we outline the EM algorithm for general parametric marginal distributions.

3.2.1. E-Step

Set $K := \Sigma^{-1}$ and $K^t := \Sigma^{t^{-1}}$. For simplicity, we pick one summand in Equation (7). By Equation (7) and (9), it holds with $\psi = (\theta, \Sigma)$ and $\mathbf{x}^\ell$ taking the role of $\mathbf{y}^\ell$:

$$\begin{aligned}\lambda(\theta, \Sigma|\mathbf{x}^\ell, \theta^t, \Sigma^t) &= \mathbb{E}_{\theta^t,\Sigma^t}\left(\ln\left(f_{\theta,\Sigma}\left(\left(\mathbf{x}_{\mathbf{obs}(\ell)}, \mathbf{x}_{\mathbf{mis}(\ell)}\right)\right)\right)|\mathbf{X}_{\mathbf{obs}(\ell)} = \mathbf{x}^\ell_{\mathbf{obs}(\ell)}\right)\\ &= -\frac{1}{2}\ln(|\Sigma|)\\ &\quad - \frac{1}{2}\mathbb{E}_{\Sigma^t,\theta^t}\left({\mathbf{z}_\theta}^T(K - I)\mathbf{z}_\theta|\mathbf{X}_{\mathbf{obs}(\ell)} = \mathbf{x}^\ell_{\mathbf{obs}(\ell)}\right)\\ &\quad + \sum_{j=1}^{p}\mathbb{E}_{\Sigma^t,\theta^t}\left(\ln\left(f_j^{\theta_j}(x_j)\right)|\mathbf{X}_{\mathbf{obs}(\ell)} = \mathbf{x}^\ell_{\mathbf{obs}(\ell)}\right).\end{aligned} \tag{10}$$

The first and last summand depend only on Σ and θ, respectively. Thus, of special interest is the second summand, for which we obtain the following with Proposition 1:

$$\mathbb{E}_{\Sigma^t,\theta^t}\left(\mathbf{z}_\theta^T(K - I)\mathbf{z}_\theta|\mathbf{X}_{\mathbf{obs}(\ell)} = \mathbf{x}^\ell_{\mathbf{obs}(\ell)}\right) = \int\left(\mathbf{z}^T_{\theta,\theta^t}(K - I)\mathbf{z}_{\theta,\theta^t}\right)\phi_{\boldsymbol{\mu},\Sigma^{t'}}\left(\mathbf{q}_{\mathbf{mis}(\ell)}\right)d\mathbf{q}_{\mathbf{mis}(\ell)}, \tag{11}$$

where

$$\mathbf{z}_{\theta,\theta^t} := \left(\Phi^{-1}\left(F_1^{\theta_1}\left(F_1^{\theta_1^t{}^{-1}}(\Phi(q_1))\right)\right), \ldots, \Phi^{-1}\left(F_p^{\theta_p}\left(F_p^{\theta_p^t{}^{-1}}(\Phi(q_p))\right)\right)\right).$$

Here,

$$\boldsymbol{\mu} = \Sigma_{\mathbf{mis}(\ell),\mathbf{obs}(\ell)}\Sigma^{-1}_{\mathbf{obs}(\ell),\mathbf{obs}(\ell)}\Phi^{-1}\left(F^{\theta^t}_{\mathbf{obs}(\ell)}\left(x^\ell_{\mathbf{obs}(\ell)}\right)\right)$$

and

$$\Sigma^{t'} = \Sigma^t_{\mathbf{mis}(\ell),\mathbf{mis}(\ell)} - \Sigma^t_{\mathbf{mis}(\ell),\mathbf{obs}(\ell)}\left(\Sigma^t_{\mathbf{obs}(\ell),\mathbf{obs}(\ell)}\right)^{-1}\Sigma^t_{\mathbf{obs}(\ell),\mathbf{mis}(\ell)}.$$

At this point, the authors of [18] neglected that, in general,

$$F_j^{\theta_j^t} \neq F_j^{\theta_j}, j = 1, \ldots, p$$

holds, and hence, (11) depends not only on Σ, but also on θ. This let us reconsider their approach, as we describe below.

3.2.2. M-Step

The joint optimization with respect to θ and Σ is difficult, as there is no closed form for Equation (10). We circumvent this problem by sequentially optimizing with respect to Σ and θ by applying the ECM algorithm. The maximization routine is the following.

1. Set $\Sigma^{t+1} = \text{argmax}_\Sigma \sum_{l=1}^{N}\lambda(\theta^t, \Sigma|\mathbf{x}^\ell, \theta^t, \Sigma^t)$.

2. Set $\theta^{t+1} = \text{argmax}_\theta \sum_{l=1}^{N} \lambda(\theta, \Sigma^{t+1} | \mathbf{x}^\ell, \theta^t, \Sigma^t)$.

This is a two-step approach consisting of estimating the copula first and the marginals second. However, both steps are executed iteratively, which is typical for the EM algorithm.

Estimating Σ

As we are maximizing Equation (10) with respect to Σ with a fixed $\theta = \theta^t$, the last summand can be neglected. By a change-of-variables argument, we show the following in Theorem A1:

$$\mathbb{E}_{\Sigma^t, \theta^t}\left(\mathbf{z}_{\theta^t}{}^T (K - I) \mathbf{z}_{\theta^t} | \mathbf{X}_{\mathbf{obs}(\ell)} = \mathbf{x}^\ell_{\mathbf{obs}(\ell)} \right) = tr\left(\Sigma^{-1} V_\ell \right),$$

where V_ℓ depends on Σ^t and $\mathbf{z}_{\theta^t, \mathbf{obs}(\ell)} = \Phi^{-1}\left(F^{\theta^t}_{\mathbf{obs}(\ell)} \left(\mathbf{x}^\ell_{\mathbf{obs}(\ell)} \right) \right)$. Thus, considering all observations, we search for

$$\begin{aligned} \Sigma^{t+1} &= \underset{\Sigma, \Sigma_{\ell\ell}=1 \forall \ell=1,\ldots,p}{\text{argmax}} \frac{1}{N} \sum_{l=1}^{N} \lambda(\theta^t, \Sigma | \mathbf{x}^\ell, \theta^t, \Sigma^t) \\ &= \underset{\Sigma, \Sigma_{\ell\ell}=1 \forall \ell=1,\ldots,p}{\text{argmax}} \frac{1}{N} \sum_{\ell=1}^{N} -\frac{1}{2} \ln(|\Sigma|) - \frac{1}{2} tr\left(\Sigma^{-1} V_\ell \right) \\ &= \underset{\Sigma, \Sigma_{\ell\ell}=1 \forall \ell=1,\ldots,p}{\text{argmax}} -\frac{1}{2} \ln(|\Sigma|) - \frac{1}{2} tr\left(\Sigma^{-1} \frac{1}{N} \sum_{\ell=1}^{N} V_\ell \right), \end{aligned} \tag{12}$$

which only depends on the statistic $S := \frac{1}{N} \sum_{\ell=1}^{N} V_\ell$. Generally, this maximization can be formalized as a convex optimization problem that can be solved by a gradient descent. However, the properties of this estimator are not understood (for example, a scaling of S by $a \in \mathbb{R}_{>0}$ leads to a different solution; see Appendix A.3). To overcome this issue, we instead approximate the solution with the correlation matrix

$$\underset{\Sigma, \Sigma_{\ell\ell}=1 \forall \ell=1,\ldots,p}{\text{argmax}} -\frac{1}{2} \ln(|\Sigma|) - \frac{1}{2} tr\left(\Sigma^{-1} S \right) \approx PSP,$$

where $P \in \mathbb{R}^p$ is the diagonal matrix with entries $P_{jj} = \frac{1}{\sqrt{S_{jj}}}, \forall j = 1, \ldots, p$. This was also proposed in [28] (Section 2.2).

In cases in which there is expert knowledge on the dependency structure of the underlying distribution, one can adapt Equation (12) accordingly. We discuss this in more detail in Section 4.4.

Estimating θ

We now focus on finding θ^{t+1}, which is the maximizer of

$$\begin{aligned} \sum_{\ell=1}^{N} \lambda(\theta, \Sigma^{t+1} | \mathbf{x}^\ell, \theta^t, \Sigma^t) &= \sum_{\ell=1}^{N} \mathbb{E}_{\theta^t, \Sigma^t}\left(\ln\left(f_{\theta, \Sigma^{t+1}}\left(\mathbf{x}_{\mathbf{obs}(\ell)}, \mathbf{x}_{\mathbf{mis}(\ell)} \right) \right) | \mathbf{X}_{\mathbf{obs}(\ell)} = \mathbf{x}^\ell_{\mathbf{obs}(\ell)} \right) \\ &= \sum_{\ell=1}^{N} \int \ln\left(f_{\theta, \Sigma^{t+1}}\left(\mathbf{x}^\ell_{\mathbf{obs}(\ell)}, \mathbf{x}_{\mathbf{mis}(\ell)} \right) \right) \\ &\qquad f_{\theta^t, \Sigma^t}\left(\mathbf{x}_{\mathbf{mis}(\ell)} | \mathbf{X}_{\mathbf{obs}(\ell)} = \mathbf{x}^\ell_{\mathbf{obs}(\ell)} \right) d\mathbf{x}_{\mathbf{mis}(\ell)} \end{aligned}$$

with respect to θ. As there is, in general, no closed formula for the right-hand side, we use Monte Carlo integration. Again, we start by considering a single observation $\mathbf{x}^\ell$ to simplify terms. Employing Algorithm 1, we receive M samples $\mathbf{x}^\ell_{\mathbf{mis}(\ell),1}, \ldots, \mathbf{x}^\ell_{\mathbf{mis}(\ell),M}$

from the distribution of $X_{\mathbf{mis}(\ell)}|X_{\mathbf{obs}(\ell)} = x^{\ell}_{\mathbf{obs}(\ell)}$ given the parameters θ^t and Σ^t. We set $\mathbf{x}^{\ell}_{\mathbf{obs}(\ell),m} = \mathbf{x}^{\ell}_{\mathbf{obs}(\ell)}$ $\forall m = 1, \ldots, M$. Then, by Equation (9),

$$\begin{aligned}\lambda(\theta, \Sigma^{t+1}|\mathbf{x}^{\ell}, \theta^t, \Sigma^t) \approx C + \frac{1}{M}\sum_{m=1}^{M} &-\frac{1}{2}\left(\Phi^{-1}\left(F_1^{\theta_1}(x^{\ell}_{1,m})\right), \ldots, \Phi^{-1}\left(F_p^{\theta_p}(x^{\ell}_{p,m})\right)\right)^T \\ &\left(K^{t+1} - I\right) \\ &\left(\Phi^{-1}\left(F_1^{\theta_1}(x^{\ell}_{1,m})\right), \ldots, \Phi^{-1}\left(F_p^{\theta_p}(x^{\ell}_{p,m})\right)\right) \\ &+ \sum_{j=1}^{p} \ln\left(f_j^{\theta_j}(x^{\ell}_{j,m})\right).\end{aligned} \tag{13}$$

Hence, considering all observations, we set

$$\begin{aligned}\theta^{t+1} = \underset{\theta}{\operatorname{argmax}} \frac{1}{M}\sum_{\ell=1}^{N}\sum_{m=1}^{M} &-\frac{1}{2}\left(\Phi^{-1}\left(F_1^{\theta_1}(x^{\ell}_{1,m})\right), \ldots, \Phi^{-1}\left(F_p^{\theta_p}(x^{\ell}_{p,m})\right)\right)^T \\ &\left(K^{t+1} - I\right) \\ &\left(\Phi^{-1}\left(F_1^{\theta_1}(x^{\ell}_{1,m})\right), \ldots, \Phi^{-1}\left(F_p^{\theta_p}(x^{\ell}_{p,m})\right)\right) \\ &+ \sum_{j=1}^{p} \ln\left(f_j^{\theta_j}(x^{\ell}_{j,m})\right).\end{aligned} \tag{14}$$

Note that we only use the Monte Carlo samples to update the parameters of the marginal distributions θ. We would also like to point out some interesting aspects about Equations (13) and (14):

- The summand $\sum_{\ell=1}^{N}\sum_{m=1}^{M} \ln\left(f_j^{\theta_j}(x^{\ell}_{j,m})\right)$ describes how well the marginal distributions fit the (one-dimensional) data.
- The estimations of the marginals are interdependent. Hence, in order to maximize with respect to θ_j, we have to take into account all other components of θ.
- The first summand adjusts for the dependence structure in the data. If all observations at step $t+1$ are assumed to be independent, then $K^{t+1} = I$, and this term is 0.
- More generally, the derivative $\frac{\partial\lambda(\theta,\Sigma^{t+1}|\mathbf{x}^{\ell},\theta^t,\Sigma^t)}{\partial\theta_j}$ depends on θ_k if and only if $K^{t+1}_{jk} \neq 0$. This means that if Σ^{t+1} implies the conditional independence of column j and k given all other columns (Equation (6)), the optimal θ_j can be found without considering θ_k. This, e.g., is the case if we set entries of the precision matrix to 0. Thus, the incorporation of prior knowledge reduces the complexity of the identification of the marginal distributions.

The intuition behind the derived EM algorithm is simple. Given a dataset with missing values, we estimate the dependency structure. With the identified dependency structure, we can derive likely locations of the missing values. Again, these locations help us to find a better dependency structure. This leads to the proposed cyclic approach. The framework of the EM algorithm guarantees the convergence of this procedure to a local maximum for $M \to \infty$ in Equation (14).

3.3. Modelling with Semiparametric Marginals

In the case in which the missing mechanism is MAR, the estimation of the marginal distribution using only complete observations is biased. Even worse, any moment of the distribution can be distorted. Thus, one needs a priori knowledge in order to identify the parametric family of the marginals [19,20]. If their family is known, one can directly apply

the algorithm of Section 3.2. If this is not the case, we propose the use of a mixture model parametrization of the form

$$F_j^{\theta_j}(x_j) = \frac{1}{g}\sum_{k=1}^{g}\Phi\left(\frac{x_j-\theta_{jk}}{\sigma_j}\right), \theta_{j1} \leq \ldots \leq \theta_{jg}, \forall j = 1,\ldots,p, \tag{15}$$

where σ_j is a hyperparameter and the ordering of the θ_{jk} ensures the identifiability.

Using mixture models for density estimation is a well-known idea (e.g., [29–31]). As the authors of [31] noted, mixture models vary between being parametric and being non-parametric, where flexibility increases with g. It is reasonable to choose Gaussian mixture models, as their density functions are dense in the set of all density functions with respect to the L^1-norm [29] (Section 3.2). This flexibility and the provided parametrization make the mixture models a natural choice.

3.4. A Blueprint of the Algorithm

The complete algorithm is summarized in Algorithm 2. For the Monte Carlo EM algorithm, Ref. [26] proposed the stabilization of the parameters with a rather small number of samples M and to increase this number substantially in the latter steps of the algorithm. This seems to be reasonable for line 8 of Algorithm 2 as well.

If there is no a priori knowledge about the marginals, we propose that we follow Section 3.3. We choose the initial θ^0 such that the cumulative distribution function of the mixture model fits the ecdf of the observed data points. For an empirical analysis of the role of g, see Section 4.3.3. For $\sigma_1,\ldots,\sigma_p$, we use a rule of thumb inspired by [3] and set

$$\sigma_j = 1.06\frac{\widehat{\sigma_j}}{g^{1/5}},$$

where $\widehat{\sigma_j}$ is the standard deviation of the observed data points in the j-th component.

Algorithm 2: Blueprint for the EM algorithm for the Gaussian copula model

Input: $\{\boldsymbol{x^1},\ldots,\boldsymbol{x^N}\}, \Sigma^0, \theta^0, n_{max}, \epsilon_{converged}, M$
Result: Σ, θ

1 $n_{iter} \leftarrow 0$;
2 $\epsilon \leftarrow \infty$;
3 $\Sigma^t \leftarrow \Sigma^0$;
4 $\theta^t \leftarrow \theta^0$;
5 **while** $n_{iter} \leq n_{max}$ *and* $\epsilon > \epsilon_{converged}$ **do**
6 $\Sigma^{t+1} \leftarrow$ solution of Equation (12);
7 **for** $\ell \in \{1,\ldots,N\}$ **do**
8 Draw M samples of $\boldsymbol{X}|\mathbf{X}_{\mathbf{obs}(\ell)} = \mathbf{x}^{\ell}_{\mathbf{obs}(\ell)}$, under $\left(\theta^t, \Sigma^t\right)$;
9 **end**
10 $\theta^{t+1} \leftarrow$ solution of Equation (14);
11 $\epsilon \leftarrow \|\Sigma^{t+1} - \Sigma^t\| + \|\theta^{t+1} - \theta^t\|$;
12 $\theta^t \leftarrow \theta^{t+1}$;
13 $\Sigma^t \leftarrow \Sigma^{t+1}$;
14 $n_{iter} \leftarrow n_{iter} + 1$;
15 **end**
16 **return** Σ^t, θ^t

4. Simulation Study

We analyze the performance of the proposed estimator in two studies. First, we consider scenarios for two-dimensional datasets and check the potential of the algorithm.

In the second part, we explore how expert knowledge can be incorporated and how this affects the behavior and performance. The proposed procedure, which is indexed with EM in the figures below, is compared with:

1. **Standard COPula Estimator (SCOPE)**: The marginal distributions are estimated by the ecdf of the observed data points. This was proposed by [18] if the parametric family is unknown, and it is the state-of-the art approach. Thus, we apply an EM algorithm to determine the correlation structure on the mapped data points

$$z_j^\ell = \Phi^{-1}\left(\widehat{F}_j(x_j^\ell)\right), \ell = 1, \ldots, N, j = 1, \ldots, p,$$

   where $\widehat{F}_j$ is the ecdf of the observed data points in column j. Its corresponding results are indexed with SCOPE in the figures and tables.
2. **Known marginals**: The distribution of the marginals is completely known. The idea is to eliminate the difficulty of finding them. Here, we apply the EM algorithm for the correlation structure on

$$z_j^\ell = \Phi^{-1}\left(F_j(x_j^\ell)\right), \ell = 1, \ldots, N, j = 1, \ldots, p,$$

   where F_j is the real marginal distribution function. Its corresponding results are indexed with a 0 in the figures and tables.
3. **Markov chain–Monte Carlo (MCMC) approach [21]**: The author proposed an MCMC scheme to estimate the copula in a Bayesian fashion. Therefore, Ref. [21] derived the distribution of the multivariate ranks. The marginals are treated as nuisance parameters. We employed the R package `sbgcop`, which is available on `CRAN`, as it provides not only a posterior distribution of the correlation matrix Σ, but also imputations for missing values. In order to compare the approach with the likelihood-based methods, we set

$$\widehat{\Sigma}_{MCMC} = \frac{1}{M}\sum_{m=1}^{M}\Sigma^m,$$

   where $\{\Sigma^m : m = 1, \ldots, M\}$ are samples of the posterior distribution of the correlation matrix. For the marginals, we defined

$$\widehat{F}_{j,MCMC}(x) = \frac{1}{MN}\sum_{\ell=1}^{N}\sum_{m=1}^{M} 1_{\{x_{j,m}^\ell \leq x\}},$$

   where $x_{j,m}^\ell$ is the m-th of the total of M imputations for x_j^ℓ and $x_{j,m}^\ell = x_j^\ell \ \forall m = 1, \ldots, M$ if x_j^ℓ can be observed. The samples were drawn from the posterior distribution. The corresponding results were indexed with the MCMC approach in the figures and tables.

Sklar's theorem shows that the joint distribution can be decomposed into the marginals and the copula. Thus, we analyze them separately.

4.1. Adapting the EM Algorithm

In Sections 4.3 and 4.4, we chose $g = 15$, for which we saw a sufficient flexibility. A sensitivity analysis of the procedure with respect to g can be found in Section 4.3.3. The initial θ^0 was chosen by fitting the marginals to the existing observations, and Σ^0 was the identity matrix. For the number of Monte Carlo samples M, we observed that with $M = 20$, θ stabilized after around 10 steps. Cautiously, we ran 20 steps before we increased M to 1000, for which we run another five steps. We stopped the algorithm when the condition $\|\Sigma^{t+1} - \Sigma^t\|_1 < 10^{-5}$ was fulfilled.

4.2. Data Generation

We considered a two-dimensional dataset (we would have liked to include the setup of the simulation study of [18]; however, neither could the missing mechanism be extracted from the paper nor did the authors provide it on request) with a priori unknown marginals F_1 and F_2, whose copula was Gaussian with the correlation parameter $\rho \in [-1, 1]$. The marginals were chosen to be χ^2 with six and seven degrees of freedom. The data matrix $D \in \mathbb{R}^{N\times 2}$ kept N (complete) observations of the random vector. We enforced the following MAR mechanism:

1. Remove every entry in D with probability $0 \leq p_{MCAR} < 1$. We denote the resulting data matrix (with missing entries) as $D^{MCAR} = \left(D^{MCAR}_{\ell j}\right)_{\ell=1,\dots,N, j=1,2}$.
2. If $D^{MCAR}_{\ell 1}$ and $D^{MCAR}_{\ell 2}$ are observed, remove $D^{MCAR}_{\ell 2}$ with probability

$$\begin{aligned}\mathbb{P}(R_2 = 0 | X_1 = D_{\ell 1}, X_2 = D_{\ell 2}) &= \mathbb{P}(R_2 = 0 | X_1 = D_{\ell 1}) \\ &= \frac{1}{1 + \exp(-\beta_0 - \beta_1 \Phi^{-1}(F_1(D_{\ell 1})))}\end{aligned}$$

We call the resulting data matrix D^{MAR}.

The missing patterns were non-monotone. Aside from p_{MCAR}, the parameters β_0 and β_1 controlled how many entries were absent in the final dataset. Assuming that $\rho > 0$, $\beta_1 > 0$, and $|\beta_0|$ was not too large, the ecdf of the observed values of X_2 was shifted to the left compared to the true distribution function (changing the signs of β_1 and/or ρ may change the direction of the shift, but the situation is analogous). This can be seen in Figure 1, where we chose $N = 200$, $\rho = 0.5$, $\beta = (\beta_0, \beta_1) = (0, 2)$. The marginal distribution of X_1 could be estimated well by the ecdf of the observed data.

4.3. Results

This subsection explores how different specifications of the data-generating process presented in Section 4.2 influenced the estimation of the joint distribution. First, we investigate the influence of the share of missing values (controlled via β) and the dependency (controlled via ρ) by fixing the number of observations (denoted by N) to 100. Then, we vary N to study the behavior of the algorithms for larger sample sizes. Afterwards, we carry out a sensitivity analysis of the EM algorithm with respect to g, the number of mixtures. Finally, we study the computational demands of the algorithms.

4.3.1. The Effects of Dependency and Share of Missing Values

We investigate two different choices for the setup in Section 4.2 by setting the parameters to $\rho = 0.1$, $\beta = (-1, 1)$ and $\rho = 0.5$, $\beta = (0, 2)$. For both, we draw 1000 datasets with $N = 100$ each and apply the estimators. To evaluate the methods, we look at two different aspects.

First, we compare the estimators for ρ with respect to bias and standard deviations. The results are depicted in the corresponding third columns of Table 1 and are summarized as boxplots in Figure A1 in Appendix B.3. We see that no method is clearly superior. While the EM algorithm has a stronger bias for $\rho = 0.5$ than that of SCOPE, it also has a smaller standard deviation. The MCMC approach shows the largest bias. As even known marginals (ρ_0) do not lead to substantially better estimators compared to SCOPE (ρ_{SCOPE}) or the proposed (ρ_{EM}) approach, we deduce that (at least in this setting) the estimators for the marginals are almost negligible. MCMC performs notably worse.

Second, we investigate the Cramer–von Mises statistics ω between the estimated and the true marginal distribution (ω^1 statistic for the first marginal, ω^2 statistic for the second marginal). The results are shown in Table 1 (corresponding first two columns) and are summarized as boxplots in Figure A2 in Appendix B.3. While for $\rho = 0.1$, the proposed estimator behaves only slightly better than SCOPE, we see that the benefit becomes larger in the case of high correlation and more missing values, especially when estimating the

second marginal. This is in line with the intuition that if the correlation is vanishing, the two random variables X_1 and X_2 become independent. Thus, R_2, the missing value indicator, and X_2 become independent. (Note that there is a difference from the case in which $\rho \neq 0$, and hence, the missingness probability R_2 isconditionally independent from X_2 given X_1.) In that case, we can estimate the marginal of X_2 using the ecdf of the observed data points. Hence, SCOPE's estimates of the marginals should be good for small values of ρ. An illustration can be found in Figure 2. Again, the MCMC approach performs the worst.

Table 1. Comparison of the algorithms with respect to the Cramer–von Mises distance between the estimated and the true first (ω^1) and true second marginal distributions (ω^2), as well as the correlation (ρ). Shown are the mean and standard deviation of the proposed EM algorithm (EM), the method based on known marginals (0), the Standard Copula Estimator (SCOPE), and the Markov chain–Monte Carlo approach (MCMC) for 1000 datasets generated as described in Section 4.3.1.

		Mean			**Standard Deviation**		
Setting	**Method**	ω^1	ω^2	ρ	ω^1	ω^2	ρ
$\rho = 0.1, \beta = (-1, 1)$	EM	8.55	10.41	0.107	9.30	11.67	0.139
	0	-	-	0.109	-	-	0.144
	SCOPE	9.13	12.25	0.105	8.47	11.00	0.144
	MCMC	18.21	24.99	0.094	16.62	21.89	0.127
$\rho = 0.5, \beta = (0, 2)$	EM	8.03	16.48	0.455	8.68	19.47	0.139
	0	-	-	0.498	-	-	0.138
	SCOPE	9.06	45.25	0.486	8.25	36.11	0.143
	MCMC	17.90	59.34	0.393	16.13	57.15	0.131

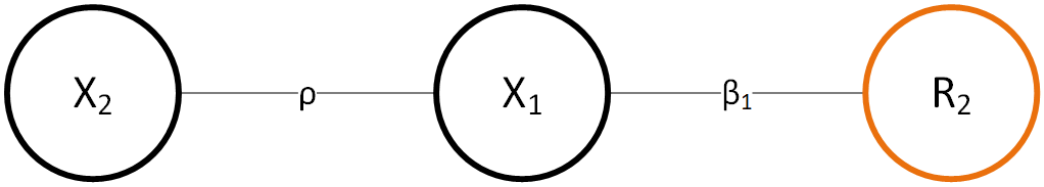

Figure 2. Dependency graph for X_1, X_2, and R_2. X_2 is independent of R_2 if either X_1 and X_2 are independent ($\rho = 0$) or if X_1 and R_2 are independent ($\beta_1 = 0$).

4.3.2. Varying the Sample Size N

To investigate the behavior of the methods for larger sample sizes, we repeat the experiment from Section 4.2 with $\rho = 0.5, \beta = (0, 2)$ for the sample sizes $N = 100, 200, 500, 1000$. The results are depicted in Table 2 and Figures A3–A5 in Appendix B.3. The bias of SCOPE and EM algorithm for ρ seem to vanish for large N, while the MCMC approach remains biased. Studying the estimation of the true marginals, the approximation of the second marginal via MCMC and SCOPE improves only slowly and is still poor for the largest sample sizes $N = 1000$. In contrast, the EM algorithm performs best in small sample sizes, and the mean (of ω^1 and ω^2) and standard deviations (of all three values) move towards 0 for increasing N.

Table 2. Comparison of the algorithms with respect to the Cramer–von Mises distance between the estimated and the true first (ω^1) and true second marginal distributions (ω^2), as well as the correlation (ρ). Shown are the mean and standard deviation of the proposed EM algorithm (EM), the method based on known marginals (0), the Standard Copula Estimator (SCOPE), and the Markov chain–Monte Carlo approach (MCMC) for 1000 datasets generated as described in Section 4.2 with $\rho = 0.5$ and $\beta = (0, 2)$ and varying sample sizes $N = 100, 200, 500, 1000$.

		Mean			**Standard Deviation**		
N	**Method**	ω^1	ω^2	ρ	ω^1	ω^2	ρ
$N = 100$	EM	8.03	16.48	0.455	8.68	19.47	0.139
	0	-	-	0.498	-	-	0.138
	SCOPE	9.06	45.25	0.486	8.25	36.11	0.143
	MCMC	17.90	59.34	0.393	16.13	57.15	0.131
$N = 200$	EM	4.91	8.53	0.469	5.46	8.88	0.098
	0	-	-	0.500	-	-	0.094
	SCOPE	4.76	37.38	0.493	4.18	25.35	0.096
	MCMC	9.27	42.91	0.370	8.01	36.23	0.089
$N = 500$	EM	3.01	3.83	0.480	2.92	3.59	0.063
	0	-	-	0.499	-	-	0.060
	SCOPE	2.05	31.92	0.497	1.85	14.95	0.060
	MCMC	4.01	31.41	0.0360	3.49	20.51	0.051
$N = 1000$	EM	2.25	2.74	0.486	1.92	2.40	0.047
	0	-	-	0.500	-	-	0.042
	SCOPE	1.08	30.60	0.499	0.93	11.13	0.043
	MCMC	1.99	28.13	0.365	1.84	14.49	0.037

4.3.3. The Impacts of Varying the Number of Mixtures g

The proposed EM algorithm relies on the hyperparameter g, the number of mixtures in Equation (15). To analyze the behavior of the EM algorithm with respect to g, we additionally run the EM algorithm with $g = 5$ and $g = 30$ on the 1000 datasets of Section 4.2 for $\rho = 0.5$, $\beta = (0, 2)$, and $N = 100$. We did not adjust the number of steps in the EM algorithm to keep the results comparable. The results can be found in Table 3. We see that the choice of g does not have a large effect on the estimation of ρ. However, an increased g leads to better estimates for X_1. This is in line with the intuition that the ecdf of the first components is an unbiased estimate for the distribution function of X_1, and setting g to the number of samples corresponds to the kernel density estimator. On the other hand, the estimator for X_2 benefits slightly from $g = 5$, as ω^2_{EM} has a lower mean and standard deviation compared to the choice $g = 30$. However, this effect is small and almost non-existent when we compare $g = 5$ with $g = 15$. As the choice $g = 15$ leads to better estimates of the first marginal compared to $g = 5$, we see this choice as a good compromise for our setting. For applications without prior knowledge, we recommend considering g as additional tuning parameter (via cross-validation).

Table 3. Comparison of the proposed EM algorithm with respect to the Cramer–von Mises distance between the estimated and the true first (ω^1) and true second marginal distributions (ω^2), as well as the correlation (ρ), for different numbers of mixtures g in Equation (15). Shown are the mean and standard deviation for $g = 5, 15, 30$ and for 1000 datasets generated as described in Section 4.2 with $\rho = 0.5$ and $\beta = (0, 2)$.

	Mean			**Standard Deviation**		
# Mixtures	ω^1	ω^2	ρ	ω^1	ω^2	ρ
$g = 5$	13.82	16.38	0.469	14.17	19.69	0.145
$g = 15$	8.03	16.48	0.455	8.68	19.47	0.139
$g = 30$	7.17	18.73	0.454	7.48	20.98	0.140

4.3.4. Run Time

We analyze the computational demands of the different algorithms by comparing their run times in the study of Section 4.3.1 with $\rho = 0.5$ and $\beta = (0, 2)$ (the settings $\rho = 0.1$ and $\beta = (-1, 1)$ lead to similar results and are omitted). The run times of all presented algorithms depend not only on the dataset, but also on the parameters (e.g., convergence criterion and Σ^0 for SCOPE). Thus, we do not aim for an extensive study, but focus on the magnitudes. We compare the proposed EM algorithm with a varying number of mixtures ($g = 5, 15, 30$) with MCMC and SCOPE. The results are shown in Table 4. We see that the EM algorithm has the longest run time, which depends on the number of mixtures g. The MCMC approach and the proposed EM algorithm have a higher computational demand than SCOPE, as they are trying to model the interaction between the copula and the marginals. As mentioned in the onset, we could reduce the run time of the EM algorithm by going down to only 10 steps instead of 20.

Table 4. Comparison of the algorithms with respect to the run time in seconds. Shown are the mean and standard deviation of the proposed EM algorithm (EM) with the number of mixtures g set to $5, 15, 30$, the Standard Copula Estimator (SCOPE), and the Markov chain–Monte Carlo approach (MCMC) for 1000 datasets generated as described in Section 4.2 with $\rho = 0.5$ and $\beta = (0, 2)$.

	Run Time in Seconds	
Method	**Mean**	**Standard Deviation**
EM ($g = 5$)	21.78	3.27
EM ($g = 15$)	55.94	11.39
EM ($g = 30$)	161.57	38.00
SCOPE	0.45	0.11
MCMC	12.98	0.87

4.4. Inclusion of Expert Knowledge

In the presence of prior knowledge on the dependency structure, the presented EM algorithm is highly flexible. While information on the marginals can be used to parametrize the copula model, expert knowledge on the dependency structure can be incorporated by adapting Equation (12). In the case of soft constraints on the covariance or precision matrix, one can replace Equation (12) with a penalized covariance estimation, where the penalty reflects the expert assessment [32,33]. Similarly, one can define a prior distribution on the covariance matrices and set Σ^{t+1} as the mode of the posterior distribution (the MAP estimate) of Σ given the statistic S of Equation (12).

Another possibility could be that we are aware of conditional independencies in the data-generating process. This is, for example, the case when causal relationships are known [4]. To exemplify the latter, we consider a three-dimensional dataset $\mathbf{X}$ with the

Gaussian copula C_Σ and marginals X_1, X_2, X_3, which are χ^2 distributed with six, seven, and five degrees of freedom. The precision is set to

$$K = \Sigma^{-1} = \Delta^{1/2} \begin{pmatrix} 1 & 0.5 & 0.5 \\ 0.5 & 1 & 0 \\ 0.5 & 0 & 1 \end{pmatrix} \Delta^{1/2},$$

where $\Delta^{1/2}$ is a diagonal matrix, which ensures that the diagonal elements of Σ are 1. We see that X_2 and X_3 are conditionally independent given X_1. The missing mechanism is similar to the one in Section 4.2. The missingness of X_3 depends on X_1 and X_2, while the probability of a missing X_1 or X_2 is independent of the others. The mechanism is, again, MAR. Details can be found in Appendix B.2. We compare the proposed method with prior knowledge on the zeros in the precision matrix (indexed by KP, EM in the figures) with the EM, SCOPE, and MCMC algorithms without background knowledge. We again sample 1000 datasets with 50 observations each from the real distribution. The background knowledge on the precision is used by restricting the non-zero elements in Equation (12). Therefore, we apply the procedure presented in [34] (Chapter 17.3.1) to find Σ^{t+1}. The means and standard deviations of the estimates are presented in Table 5.

First, we evaluate the estimated dependency structures by calculating the Frobenius norm of the estimation error $\Sigma - \widehat{\Sigma}$. The EM algorithm with background knowledge (KP, EM) performs best and is more stable than its competitors. Apart from MCMC, the other procedures behave similarly, which indicates again that the exact knowledge of the marginal distributions is not too relevant for identifying the dependency structure. MCMC performs the worst.

Table 5. Comparison of the algorithms with respect to the Cramer–von Mises distance between the estimated and the true first marginal distribution (ω^1), true second marginal distribution (ω^2), and true third marginal distribution (ω^3), as well as the correlation (ρ). Shown are the mean and standard deviation of the proposed EM algorithm (EM), the proposed EM algorithm with prior knowledge on the conditional independencies (KP, EM), the method based on known marginals (0), the Standard Copula Estimator (SCOPE), and the Markov chain–Monte Carlo approach (MCMC) for 1000 datasets generated as described in Section 4.4.

	Mean				**Standard Deviation**			
Method	ω^1	ω^2	ω^3	$\|\|\widehat{\Sigma} - \Sigma\|\|_2$	ω^1	ω^2	ω^3	$\|\|\widehat{\Sigma} - \Sigma\|\|_2$
EM	12.12	13.38	21.15	0.229	13.89	14.25	22.44	0.113
KP, EM	12.04	13.28	19.66	0.182	13.93	14.37	20.88	0.111
0	-	-	-	0.227	-	-	-	0.108
SCOPE	17.57	17.55	26.69	0.232	16.75	15.55	24.84	0.113
MCMC	36.85	35.70	80.22	0.263	32.82	33.24	78.57	0.140

Second, we see that the proposed EM estimators return marginal distributions that are closer to the truth, while the estimate with background knowledge (KP, EM) performs the best. Thus, the background knowledge on the copula also transfers into better estimates for the marginal distribution—in particular, for X_3. This is due to Equation (14) and the comments thereafter. The zeros in the precision structure indicate which other marginals are relevant in order to identify the parameter of a marginal. In our case, X_2 provides no additional information for X_3. This information is provided to the EM algorithm through the restriction of the precision matrix.

Finally, we compare the EM estimates of the joint distribution. The relative entropy or Kullback–Leibler divergence is a popular tool for estimating the difference between two distributions [35,36], where one of them is absolutely continuous with respect to the other. A lower number indicates a higher similarity. Due to the discrete structure of the marginals of SCOPE and MCMC, we cannot calculate their relative entropy with respect to the truth.

However, we would like to analyze how the estimate of the proposed procedure improves if we include expert knowledge. The results are depicted in Table 6. Again, we observe that the incorporation of extra knowledge improves the estimates. This is in line with Table 5, as the estimation of all components in the joint distribution of Equation (3) is improved by the domain knowledge.

Table 6. Comparison of the algorithms with respect to the Kullback–Leibler divergence (D_{KL}) between the true joint distribution (F) and the estimates. Shown are the mean and standard deviation of the proposed EM algorithm (EM) and the proposed EM algorithm with prior knowledge on the conditional independencies (KP, EM) for 1000 datasets generated as described in Section 4.4.

	Mean$(D_{KL}(F,\cdot))$	**Standard Deviation**$(D_{KL}(F,\cdot))$
EM	1.37	0.53
KP, EM	1.26	0.32

5. Discussion

In this paper, we investigated the estimation of the Gaussian copula and the marginals with an incomplete dataset, for which we derived a rigorous EM algorithm. The procedure iteratively searches for the marginal distributions and the copula. It is, hence, similar to known methods for complete datasets. We saw that if the data are missing at random, a consistent estimate of a marginal distribution depends on the copula and other marginals.

The EM algorithm relies on a complete parametrization of the marginals. The parametric family of the marginals is, in general, a priori unknown and cannot be identified through the observed data points. For this case, we presented a novel idea of employing mixture models. Although this is practically always a misspecification, our simulation study revealed that the combination of our EM algorithm and marginal mixture models delivers better estimates for the joint distribution than currently used procedures do. In principle, uncertainty quantification of the parameters derived by the proposed EM algorithm can be achieved by bootstrapping [37].

There are different possibilities for incorporating expert knowledge. Information on the parametric family of the marginals can be used for their parametrization. However, causal and structural understandings of the data-generating process can also be utilized [4,38,39]. For example, this can be achieved by restricting the correlation matrix or its inverse, the precision matrix. We presented how one can restrict the non-zero elements of the precision, which enforces conditional independencies. Our simulation study showed that this leads not only to an improved estimate for the dependency structure, but also to better estimates for the marginals. This translates into a lower relative entropy between the real distribution and the estimate. We also discussed how soft constraints on the dependency structure can be included.

We note that the focus of this paper is on estimating the joint distribution without precise specification of its subsequent use. Therefore, we did not discuss imputation methods (see, e.g., [40–43]). However, Gaussian copula models were employed as a device for multiple imputation (MI) with some success [22,24,44]. The resulting complete datasets can be used for inference. All approaches that we are aware of estimate the marginals by using the ecdf of the observed data points. The findings in Section 4 translate into better draws for the missing values.

Additionally, the joint distribution can be utilized for regressing a potentially multivariate $\mathbf{Y}$ on $\mathbf{Z}$ even if data are missing. By applying the EM algorithm on $\mathbf{X} := (\mathbf{Y}, \mathbf{Z})$ and by Proposition 1, one even obtains the whole conditional distribution of $\mathbf{Y}$ given $\mathbf{Z} = \mathbf{z}$.

We have shown how to incorporate a causal understanding of the data-generating process. However, in the potential outcome framework of [45], the derivation of a causal relationship can also be interpreted as a missing data problem in which the missing patterns are "misaligned" [46]. Our algorithm is applicable for this.

Author Contributions: Conceptualization, M.K.; methodology, M.K.; software, M.K.; validation, M.P. and M.K.; formal analysis, M.K.; investigation, M.K.; resources, M.P. and M.K.; data curation, M.K.; writing—original draft preparation, M.K.; writing—review and editing, M.P. and M.K; visualization, M.K.; supervision, M.P.; project administration, M.P. All authors have read and agreed to the published version of the manuscript.

Funding: Maximilian Kertel's work is funded by the BMW Group.

Data Availability Statement: The data generation procedures of the simulation studies and the proposed algorithm are availabe at https://github.com/mkrtl/misscop, accessed on 26 October 2022.

Conflicts of Interest: The authors declare no conflict of interest.

Appendix A. Technical Results

Appendix A.1. Proof of Conditional Distribution

Proof of Proposition 1. We prove in the order of the proposition, which is a multivariate generalization of [47].

1. We inspect the conditional density function:

$$\begin{aligned} f(\mathbf{x_T}|\mathbf{X_S}=\mathbf{x_S}) &= \frac{|\Sigma|^{-\frac{1}{2}}\exp\left(-\frac{1}{2}\mathbf{z}^T(\Sigma^{-1}-I)\mathbf{z}\right)\prod_{j=1}^p f_j(x_j)}{|\Sigma_{\mathbf{S},\mathbf{S}}|^{-\frac{1}{2}}\exp\left(-\frac{1}{2}\mathbf{z}_\mathbf{S}^T\left(\Sigma_{\mathbf{S},\mathbf{S}}^{-1}-I\right)\mathbf{z_S}\right)\prod_{j\in\mathbf{S}} f_j(x_j)} \\ &= \frac{|\Sigma|^{-\frac{1}{2}}\exp\left(-\frac{1}{2}\mathbf{z}^T\Sigma^{-1}\mathbf{z}\right)\exp(\frac{1}{2}\mathbf{z}^T\mathbf{z})\prod_{j=1}^p f_j(x_j)}{|\Sigma_{\mathbf{S},\mathbf{S}}|^{-\frac{1}{2}}\exp\left(-\frac{1}{2}\mathbf{z}_\mathbf{S}^T\Sigma_{\mathbf{S},\mathbf{S}}^{-1}\mathbf{z_S}\right)\exp(\frac{1}{2}\mathbf{z}_\mathbf{S}^T\mathbf{z_S})\prod_{j\in\mathbf{S}} f_j(x_j)} \\ &= \frac{|\Sigma|^{-\frac{1}{2}}\exp\left(-\frac{1}{2}\mathbf{z}^T\Sigma^{-1}\mathbf{z}\right)\exp(\frac{1}{2}\mathbf{z}_\mathbf{T}^T\mathbf{z_T})\prod_{j\in\mathbf{T}} f_j(x_j)}{|\Sigma_{\mathbf{S},\mathbf{S}}|^{-\frac{1}{2}}\exp\left(-\frac{1}{2}\mathbf{z}_\mathbf{S}^T\Sigma_{\mathbf{S},\mathbf{S}}^{-1}\mathbf{z_S}\right)} \end{aligned}$$

Using well-known factorization lemmas and using the Schur complement (see, for example, [48] (Section 4.3.4)) applied on Σ^{-1}, we encounter

$$f(\mathbf{x_T}|\mathbf{X_S}=\mathbf{x_S}) = |\Sigma'|^{-\frac{1}{2}}\exp\left(-\frac{1}{2}(\mathbf{z_T}-\boldsymbol{\mu})^T\Sigma'^{-1}(\mathbf{z_T}-\boldsymbol{\mu})\right)\exp\left(\frac{1}{2}\mathbf{z}_\mathbf{T}^T\mathbf{z_T}\right)\prod_{j\in\mathbf{T}} f_j(x_j). \quad \text{(A1)}$$

2. The distribution of

$$\Phi^{-1}(F_\mathbf{T}(\mathbf{X_T}))|\mathbf{X_S}=\mathbf{x_S}$$

follows with a change-of-variable argument. Using Equation (A1), we observe for any measurable set A that

$$\begin{aligned} &\mathbb{P}\Big(\Big(\Phi^{-1}(F_\mathbf{T}(\mathbf{X_T}))|\mathbf{X_S}=\mathbf{x_s}\Big)\in A\Big) \\ &= \int_{F^{-1}(\Phi(A))} |\Sigma'|^{-\frac{1}{2}}\exp\left(-\frac{1}{2}(\mathbf{z_T}-\boldsymbol{\mu})^T\Sigma'^{-1}(\mathbf{z_T}-\boldsymbol{\mu})\right)\exp\left(\frac{1}{2}\mathbf{z}_\mathbf{T}^T\mathbf{z_T}\right)\prod_{j\in\mathbf{T}} f_j(x_j)d\mathbf{x_T} \\ &= \int_A \phi_{\boldsymbol{\mu},\Sigma'}(\mathbf{q_T})d\mathbf{q_T}, \end{aligned}$$

where, in the second equation, we used the transformation $\mathbf{q_T} = \Phi^{-1}(F_\mathbf{T}(\mathbf{x_T}))$ and the fact that

$$\left|D\left(\phi^{-1}(F_\mathbf{T}(\mathbf{x_T}))\right)\right| = 2\pi^{\frac{|\mathbf{T}|}{2}}\exp\left(\frac{1}{2}\left(\Phi^{-1}(F_\mathbf{T}(\mathbf{x_T}))\right)^T\left(\Phi^{-1}(F_\mathbf{T}(\mathbf{x_T}))\right)\right)\prod_{j\in\mathbf{T}} f_j(x_j).$$

3. This proof is analogous to the one above, and we finally obtain

$$\int h(\mathbf{x_T}) f(\mathbf{x_T}|\mathbf{X_S} = \mathbf{x_S}) d\mathbf{x_T} = \int h\Big(F^{-1}(\Phi(\mathbf{z_T}))\Big)\phi_{\boldsymbol{\mu},\Sigma'}(\mathbf{z_T}) d\mathbf{z_T}.$$

The result can be generalized to the case in which $\mathbf{S} \cup \mathbf{T} \neq \{1, \ldots, p\}$. □

Appendix A.2. Closed-Form Solution of the E-Step for $\theta = \theta^t$

Theorem A1. *We assume w.l.o.g. that* $\mathbf{x}^\ell = (\mathbf{x}^\ell_{\mathbf{obs}(\ell)}, \mathbf{x}^\ell_{\mathbf{mis}(\ell)})$ *and set*

$$\mathbf{z}_{\theta^t} := \Big(\mathbf{z}_{\mathbf{obs}(\ell),\theta^t}, \mathbf{z}_{\mathbf{mis}(\ell)}\Big) := \Big(\Phi^{-1}\Big(F^{\theta^t}_{\mathbf{obs}(\ell)}(\mathbf{x}^\ell_{\mathbf{obs}(\ell)})\Big), \mathbf{z}_{\mathbf{mis}(\ell)}\Big).$$

Then, it holds that

$$\mathbb{E}_{\Sigma^t,\theta^t}\Big({\mathbf{z}_{\theta^t}}^T \Sigma^{-1} \mathbf{z}_{\theta^t} | \mathbf{X}_{\mathbf{obs}(\ell)} = \mathbf{x}^\ell_{\mathbf{obs}(\ell)}\Big) = tr\Big(\Sigma^{-1} V\Big),$$

where $V = \begin{pmatrix} \mathbf{z}_{\mathbf{obs}(\ell),\theta^t} {\mathbf{z}_{\mathbf{obs}(\ell),\theta^t}}^T & \mathbf{z}_{\mathbf{obs}(\ell),\theta^t} \boldsymbol{\mu}^T \\ \boldsymbol{\mu} {\mathbf{z}_{\mathbf{obs}(\ell),\theta^t}}^T & \Sigma' + \boldsymbol{\mu}\boldsymbol{\mu}^T \end{pmatrix}$, $\boldsymbol{\mu} = \Sigma^t_{\mathbf{mis}(\ell),\mathbf{obs}(\ell)} {\Sigma^t_{\mathbf{obs}(\ell),\mathbf{obs}(\ell)}}^{-1} \mathbf{z}_{\mathbf{obs}(\ell),\theta^t}$
and $\Sigma' = \Sigma^t_{\mathbf{mis}(\ell),\mathbf{mis}(\ell)} - \Sigma^t_{\mathbf{mis}(\ell),\mathbf{obs}(\ell)} {\Sigma^t_{\mathbf{obs}(\ell),\mathbf{obs}(\ell)}}^{-1} \Sigma^t_{\mathbf{obs}(\ell),\mathbf{mis}(\ell)}$.

Proof. We define $F_{\theta^t}(\mathbf{x}_{\mathbf{mis}(\ell)}) := F_{\theta^t}(\mathbf{x}^\ell_{\mathbf{obs}(\ell)}, \mathbf{x}_{\mathbf{mis}(\ell)})$. Then,

$$\begin{aligned}
&\mathbb{E}_{\Sigma^t,\theta^t}\Big(\mathbf{z}_{\theta^t}^T \Sigma^{-1} \mathbf{z}_{\theta^t} | \mathbf{X}_{\mathbf{obs}(\ell)} = \mathbf{x}^\ell_{\mathbf{obs}(\ell)}\Big) \\
&= \mathbb{E}_{\Sigma^t,\theta^t}\Bigg(\Big(\Phi^{-1}\Big(F_{\theta^t}(\mathbf{x}_{\mathbf{mis}(\ell)})\Big)\Big)^T \Sigma^{-1} \Big(\Phi^{-1}\Big(F_{\theta^t}(\mathbf{x}_{\mathbf{mis}(\ell)})\Big)\Big) | \mathbf{X}_{\mathbf{obs}(\ell)} = \mathbf{x}^\ell_{\mathbf{obs}(\ell)}\Bigg) \\
&= \int \Big(\Phi^{-1}\Big(F_{\theta^t}(\mathbf{x}_{\mathbf{mis}(\ell)})\Big)\Big)^T \Sigma^{-1} \Big(\Phi^{-1}\Big(F_{\theta^t}(\mathbf{x}_{\mathbf{mis}(\ell)})\Big)\Big) \\
&\qquad f_{\theta^t,\Sigma^t}\Big(\mathbf{x}_{\mathbf{mis}(\ell)} | \mathbf{X}_{\mathbf{obs}(\ell)} = \mathbf{x}^\ell_{\mathbf{obs}(\ell)}\Big) d\mathbf{x}_{\mathbf{mis}(\ell)}.
\end{aligned}$$

We now apply Proposition 1 and encounter

$$\begin{aligned}
&\int \Big(\Phi^{-1}\Big(F_{\theta^t}(\mathbf{x}_{\mathbf{mis}(\ell)})\Big)\Big)^T \Sigma^{-1} \Phi^{-1}\Big(F_{\theta^t}(\mathbf{x}_{\mathbf{mis}(\ell)})\Big) \\
&\qquad f_{\theta^t,\Sigma^t}\Big(\mathbf{x}_{\mathbf{mis}(\ell)} | \mathbf{X}_{\mathbf{obs}(\ell)} = \mathbf{x}^\ell_{\mathbf{obs}(\ell)}\Big) d\mathbf{x}_{\mathbf{mis}(\ell)} \\
&= \int \mathbf{z}_{\theta^t}^T \Sigma^{-1} \mathbf{z}_{\theta^t} \phi_{\Sigma',\boldsymbol{\mu}}(\mathbf{z}_{\mathbf{mis}(\ell)}) d\mathbf{z}_{\mathbf{mis}(\ell)} \\
&= \int tr(\mathbf{z}_{\theta^t} \mathbf{z}_{\theta^t}^T \Sigma^{-1}) \phi_{\Sigma',\boldsymbol{\mu}}(\mathbf{z}_{\mathbf{mis}(\ell)}) d\mathbf{z}_{\mathbf{mis}(\ell)} \\
&= tr\Big(\Sigma^{-1} \int \mathbf{z}_{\theta^t} \mathbf{z}_{\theta^t}^T \phi_{\Sigma',\boldsymbol{\mu}}(\mathbf{z}_{\mathbf{mis}(\ell)}) d\mathbf{z}_{\mathbf{mis}(\ell)}\Big).
\end{aligned}$$

The last integral is understood element-wise. By the first and second moment of $\Phi_{\Sigma',\boldsymbol{\mu}}$, it follows that

$$\begin{aligned}
\int \mathbf{z}_{\theta^t} \mathbf{z}_{\theta^t}^T \phi_{\Sigma',\boldsymbol{\mu}}(\mathbf{z}_{\mathbf{mis}(\ell),\theta^t}) d\mathbf{z}_{\mathbf{mis}(\ell),\theta^t} &= \int \Big(\mathbf{z}_{\mathbf{obs}(\ell),\theta^t}, \mathbf{z}_{\mathbf{mis}(\ell),\theta^t}\Big) \Big(\mathbf{z}_{\mathbf{obs}(\ell),\theta^t}, \mathbf{z}_{\mathbf{mis}(\ell),\theta^t}\Big)^T \\
&\qquad \phi_{\Sigma',\boldsymbol{\mu}}(\mathbf{z}_{\mathbf{mis}(\ell),\theta^t}) d\mathbf{z}_{\mathbf{mis}(\ell),\theta^t} \\
&= \begin{pmatrix} \mathbf{z}_{\mathbf{obs}(\ell),\theta^t} \mathbf{z}^T_{\mathbf{obs}(\ell),\theta^t} & \mathbf{z}_{\mathbf{obs}(\ell),\theta^t} \boldsymbol{\mu}^T \\ \boldsymbol{\mu} \mathbf{z}^T_{\mathbf{obs}(\ell),\theta^t} & \Sigma' + \boldsymbol{\mu}\boldsymbol{\mu}^T \end{pmatrix}.
\end{aligned}$$

□

Appendix A.3. Maximizer of $\operatorname{argmax}_{\Sigma,\Sigma_{jj}=1\forall j=1,\ldots,p} \lambda(\theta^t, \Sigma|\theta^t, \Sigma^t)$

We are interested in

$$\underset{\Sigma_{jj}=1\forall j=1,\ldots,p}{\operatorname{argmax}} \; l(\Sigma) := \underset{\Sigma_{jj}=1\forall j=1,\ldots,p}{\operatorname{argmax}} \; -\log(|\Sigma|) - tr\Big(\Sigma^{-1}S\Big),$$

where $\Sigma, S \in \mathbb{R}^{p\times p}$ are positive definite matrices. Clearly,

$$\Sigma_{jj} = 1 \iff 1 = e_j^T \Sigma e_j = tr\Big(e_j^T \Sigma e_j\Big) = tr\Big(e_j e_j^T \Sigma\Big).$$

Using the Lagrangian, we obtain the following function to optimize

$$L(\Sigma, \lambda) = -\log(|\Sigma|) - tr\Big(\Sigma^{-1}S\Big) + \sum_{j=1}^{p} \lambda_j \Big(tr\Big(e_j e_j^T \Sigma\Big) - 1\Big).$$

Applying the identities $\frac{\partial tr(AX)}{\partial X} = A$, $\frac{\partial tr(AX^{-1})}{\partial X} = -X^{-1}AX^{-1}$, and $\frac{\partial \log(|X|)}{\partial X} = X^{-1}$, we obtain the derivative with respect to Σ:

$$\frac{\partial L}{\partial \Sigma} = -\Sigma^{-1} + \Sigma^{-1}S\Sigma^{-1} - \left(\sum_{j=1}^{p} \lambda_j \Big(e_j e_j^T\Big)\right) \stackrel{!}{=} 0.$$

This is equivalent to

$$-K + KSK = D_\lambda,$$

where D_λ is the diagonal matrix with entries $\lambda = (\lambda_1, \ldots, \lambda_p)$ and $K := \Sigma^{-1}$. We see that the scaling of S by $a \in \mathbb{R}_{>0}$ leads, in general, to a different solution K, and hence, the estimator is not invariant under strictly monotone linear transformations of S.
We can also formulate the task as a convex optimization problem:

$$\underset{(K^{-1})_{ii}=1 \; \forall i=1,\ldots,p}{\operatorname{argmin}} \; -\log(|K|) + tr(KS).$$

Appendix B. Details of the Simulation Studies

Appendix B.1. Drawing Samples from the Joint Distributions

Appendix B.1.1. Estimators of the Percentile Function

- In the case of SCOPE, consider the marginal observed data points, which we assume to be ordered $y_1 \leq \ldots \leq y_N$. We use the following linearly interpolated estimator for the percentile function:

$$\widehat{F^{-1}}(u) = \left\{ \begin{array}{ll} y_1 & \text{for } u \leq \frac{1}{N+1} \\ y_N, & \text{for } u > \frac{N}{N+1} \\ \frac{u-\frac{i}{N+1}}{\frac{i+1}{N+1}-\frac{i}{N+1}}(y_{i+1}-y_i)+y_i, & \text{for } u \in \left(\frac{i}{N+1}, \frac{i+1}{N+1}\right] \end{array} \right\}$$

- To estimate the percentile function for the mixture models, we choose with equal probability (all Gaussians have equal weight) one component of the mixture and then draw a random number with its mean θ_{jk} and standard deviation σ_j, $j = 1, \ldots, p$, $k = 1, \ldots, g$. In this manner, we generate N' samples $y'_1, \ldots, y'_{N'}$. The estimator for the percentile function is then chosen to be analogous to the one above. A higher N' leads to a more exact result. We choose N' to be 10,000.

Appendix B.1.2. Sampling

Given an estimator $\widehat{\rho}$ and estimators for the percentile functions $\widehat{F_1^{-1}}, \widehat{F_2^{-1}}$, we obtain M samples from the learned joint distribution with

$$y_\ell = (y_{\ell 1}, y_{\ell 2}) = \left(\widehat{F_1^{-1}}(u_{\ell 1}), \widehat{F_2^{-1}}(u_{\ell 2})\right) = \left(\widehat{F_1^{-1}}(\Phi(z_{\ell 1})), \widehat{F_2^{-1}}(\Phi(z_{\ell 2}))\right), \ell = 1, \ldots, M,$$

where $z_\ell = (z_{\ell 1}, z_{\ell 2}), \ell = 1, \ldots, M$ are draws from a bivariate normal distribution with mean 0 and covariance $\begin{pmatrix} 1 & \widehat{\rho} \\ \widehat{\rho} & 1 \end{pmatrix}$. In the case of the gold standard, we set $\widehat{F_j^{-1}} = F_j^{-1}, j = 1, 2$. We obtain samples of the real underlying distribution by using the correct percentile functions, as in the gold standard, and, additionally, $\widehat{\rho} = \rho$. The procedure for three dimensions is analogous.

Appendix B.2. Missing Mechanism for Section 4.4

The missing mechanism is similar to the two-dimensional case. The marginals are chosen to be χ^2 with six, seven, and five degrees of freedom. The data matrix $D \in \mathbb{R}^{N \times 3}$ keeps N (complete) observations of the random vector. We enforce the following missing data mechanism:

1. Again, we remove every entry in the data matrix D with probability $0 \leq p_{MCAR} < 1$. The resulting data matrix (with missing entries) is denoted as

$$D^{MCAR} = \left(D_{\ell j}^{MCAR}\right)_{\ell=1,\ldots,N, j=1,2,3}.$$

2. If $D_{\ell 1}^{MCAR}$, $D_{\ell 2}^{MCAR}$, and $D_{\ell 3}^{MCAR}$ are observed, we remove $D_{\ell 3}^{MCAR}$ with probability

$$\mathbb{P}(R_3 = 0 | X_1 = D_{\ell 1}, X_2 = D_{\ell 2}) = h(D_{\ell 1}, D_{\ell 2}; \beta),$$

where

$$h(D_{\ell 1}, D_{\ell 2}; \beta) = \frac{1}{1 + \exp(-(\beta_0 + \beta_1 \Phi^{-1}(F_1(D_{\ell 1})) + \beta_2 \Phi^{-1}(F_2(D_{\ell 2}))))}$$

and $\beta = (\beta_0, \beta_1, \beta_2)$.

We call the resulting data matrix D^{MAR}. Its missing patterns are, again, non-monotone, and the data are MAR, but not MCAR. In Section 4.4, we set $\beta = (0, 2, 2)$.

Appendix B.3. Complementary Figures

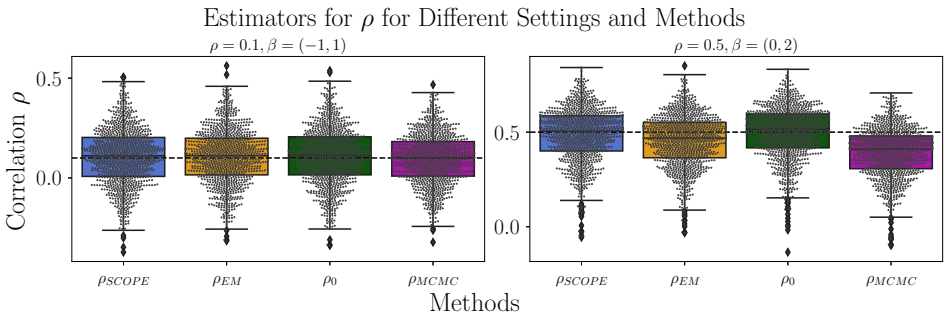

Figure A1. Comparison of the algorithms with respect to the correlation ρ. Shown are the boxplots for the Standard Copula Estimator (SCOPE), the proposed EM algorithm (EM), the method based on known marginals (0), and the Markov chain–Monte Carlo approach (MCMC) for 1000 datasets generated as described in Section 4.2, where $\rho = 0.1, \beta = (-1, 1)$ are depicted in the left canvas and $\rho = 0.5, \beta = (0, 2)$ are depicted in the right canvas. The true correlations are indicated by the dashed line.

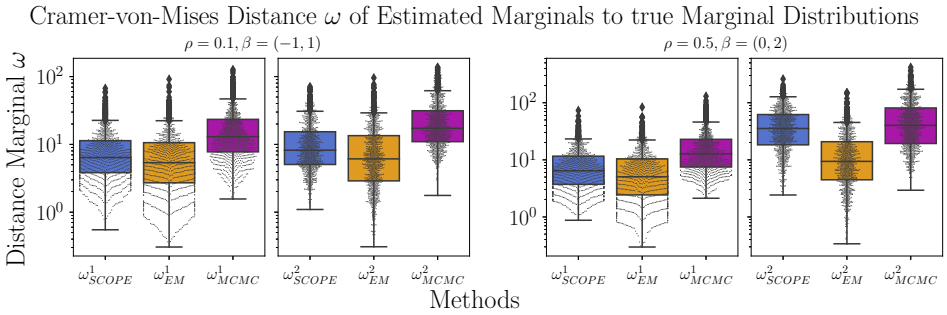

Figure A2. Comparison of the algorithms with respect to the Cramer–von Mises distance between the estimated and the true first (ω^1) and true second marginal distributions (ω^2). Shown are the boxplots on a logarithmic scale for the proposed EM algorithm (EM), the Standard Copula Estimator (SCOPE), and the Markov chain–Monte Carlo approach (MCMC) for 1000 datasets generated as described in Section 4.2, where $\rho = 0.1, \beta = (-1, 1)$ are depicted in the left canvas and $\rho = 0.5, \beta = (0, 2)$ are depicted in the right canvas.

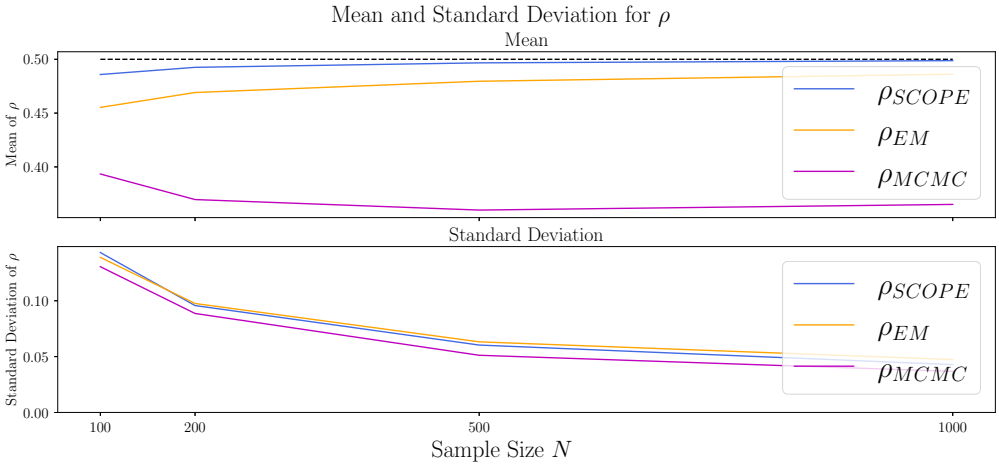

Figure A3. Comparison of the algorithms with respect to the correlation (ρ). Shown are the mean (upper canvas) and standard deviation (lower canvas) of the Standard Copula Estimator (SCOPE), the proposed EM algorithm (EM), and the Markov chain–Monte Carlo approach (MCMC) for 1000 datasets generated as described in Section 4.2 with $\rho = 0.5$ and $\beta = (0, 2)$ and for varying sample sizes $N = 100, 200, 500, 1000$, where the true ρ is 0.5 (dashed line in the upper canvas).

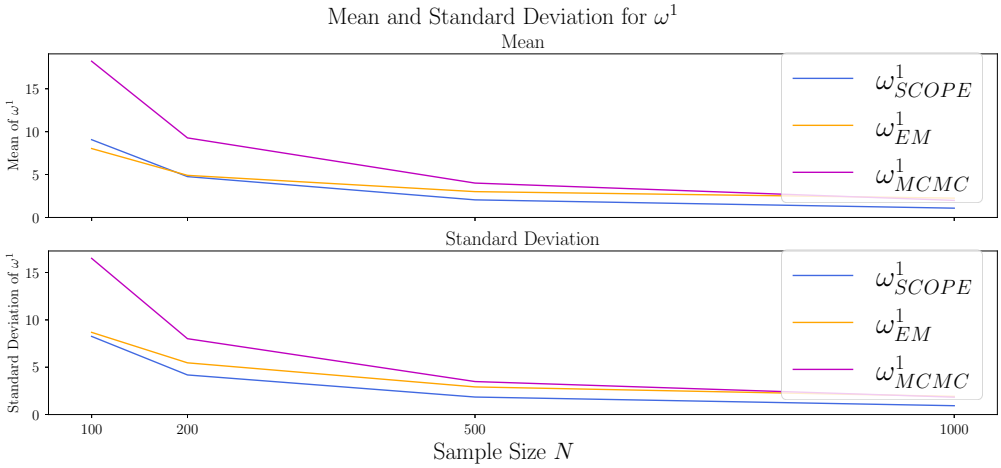

Figure A4. Comparison of the algorithms with respect to the Cramer–von Mises statistic ω^1 between the estimated and the true first marginal distribution. Shown are the mean (upper canvas) and standard deviation (lower canvas) of the Standard Copula Estimator (SCOPE), the proposed EM algorithm (EM), and the Markov chain–Monte Carlo approach (MCMC) for 1000 datasets generated as described in Section 4.2 with $\rho = 0.5$ and $\beta = (0, 2)$ and for varying sample sizes of $N = 100, 200, 500, 1000$.

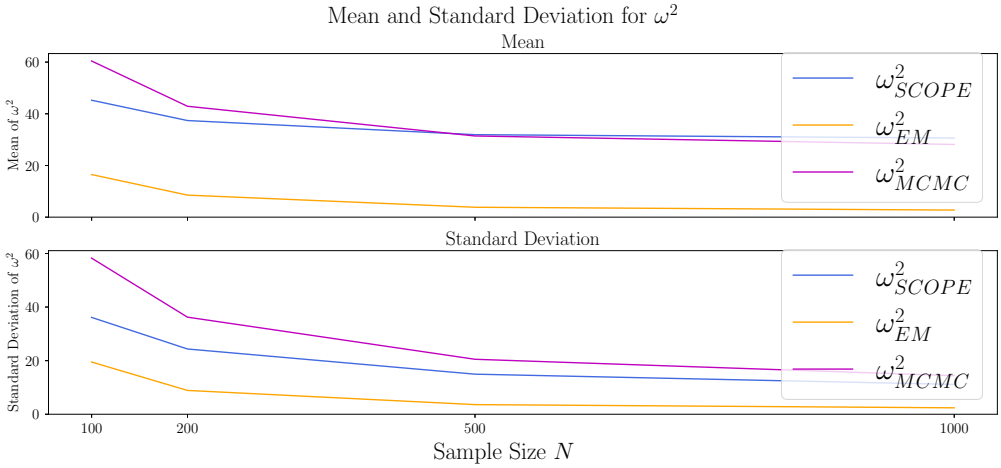

Figure A5. Comparison of the algorithms with respect to the Cramer–von Mises statistic ω^2 between the estimated and the true second marginal distribution. Shown are the mean (upper canvas) and standard deviation (lower canvas) of the Standard Copula Estimator (SCOPE), the proposed EM algorithm (EM), and the Markov chain–Monte Carlo approach (MCMC) for 1000 datasets generated as described in Section 4.2 with $\rho = 0.5$ and $\beta = (0, 2)$ and for varying sample sizes of N = 100, 200, 500, 1000.

References

1. Thurow, M.; Dumpert, F.; Ramosaj, B.; Pauly, M. Imputing missings in official statistics for general tasks–our vote for distributional accuracy. *Stat. J. IAOS* **2021**, *37*, 1379–1390. [CrossRef]
2. Liu, Y.; Dillon, T.; Yu, W.; Rahayu, W.; Mostafa, F. Missing value imputation for industrial IoT sensor data with large gaps. *IEEE Internet Things J.* **2020**, *7*, 6855–6867. [CrossRef]
3. Silverman, B. *Density Estimation for Statistics and Data Analysis*; Routledge: London, UK, 2018.
4. Kertel, M.; Harmeling, S.; Pauly, M. Learning causal graphs in manufacturing domains using structural equation models. *arXiv* **2022**, arXiv:2210.14573. https://doi.org/10.48550/ARXIV.2210.14573.
5. Genest, C.; Ghoudi, K.; Rivest, L.P. A semiparametric estimation procedure of dependence parameters in multivariate families of distributions. *Biometrika* **1995**, *82*, 543–552. [CrossRef]
6. Liu, H.; Han, F.; Yuan, M.; Lafferty, J.; Wasserman, L. High-dimensional semiparametric gaussian copula graphical models. *Ann. Stat.* **2012**, *40*, 2293–2326. [CrossRef]

7. Titterington, D.; Mill, G. Kernel-based density estimates from incomplete data. *J. R. Stat. Soc. Ser. B Methodol.* **1983**, *45*, 258–266. [CrossRef]
8. Dempster, A.; Laird, N.; Rubin, D. Maximum likelihood from incomplete data via the EM algorithm. *J. R. Stat. Soc. Ser. B Methodol.* **1977**, *39*, 1–22.
9. Shen, C.; Weissfeld, L. A copula model for repeated measurements with non-ignorable non-monotone missing outcome. *Stat. Med.* **2006**, *25*, 2427–2440. [CrossRef]
10. Gomes, M.; Radice, R.; Camarena Brenes, J.; Marra, G. Copula selection models for non-Gaussian outcomes that are missing not at random. *Stat. Med.* **2019**, *38*, 480–496. [CrossRef]
11. Rubin, D.B. Inference and missing data. *Biometrika* **1976**, *63*, 581–592. [CrossRef]
12. Cui, R.; Groot, P.; Heskes, T. Learning causal structure from mixed data with missing values using Gaussian copula models. *Stat. Comput.* **2019**, *29*, 311–333. [CrossRef]
13. Wang, H.; Fazayeli, F.; Chatterjee, S.; Banerjee, A. Gaussian copula precision estimation with missing values. In Proceedings of the Seventeenth International Conference on Artificial Intelligence and Statistics, Reykjavik, Iceland, 22–25 April 2014; PMLR: Reykjavik, Iceland, 2014; Volume 33, pp. 978–986.
14. Hamori, S.; Motegi, K.; Zhang, Z. Calibration estimation of semiparametric copula models with data missing at random. *J. Multivar. Anal.* **2019**, *173*, 85–109. [CrossRef]
15. Robins, J.M.; Gill, R.D. Non-response models for the analysis of non-monotone ignorable missing data. *Stat. Med.* **1997**, *16*, 39–56. [CrossRef]
16. Sun, B.; Tchetgen, E.J.T. On inverse probability weighting for nonmonotone missing at random data. *J. Am. Stat. Assoc.* **2018**, *113*, 369–379. [CrossRef] [PubMed]
17. Seaman, S.R.; White, I.R. Review of inverse probability weighting for dealing with missing data. *Stat. Methods Med. Res.* **2013**, *22*, 278–295. [CrossRef]
18. Ding, W.; Song, P. EM algorithm in gaussian copula with missing data. *Comput. Stat. Data Anal.* **2016**, *101*, 1–11. [CrossRef]
19. Efromovich, S. Adaptive nonparametric density estimation with missing observations. *J. Stat. Plan. Inference* **2013**, *143*, 637–650. [CrossRef]
20. Dubnicka, S.R. Kernel density estimation with missing data and auxiliary variables. *Aust. N. Z. J. Stat.* **2009**, *51*, 247–270. [CrossRef]
21. Hoff, P. Extending the rank likelihood for semiparametric copula estimation. *Ann. Appl. Stat.* **2007**, *1*, 265–283. [CrossRef]
22. Hollenbach, F.; Bojinov, I.; Minhas, S.; Metternich, N.; Ward, M.; Volfovsky, A. Multiple imputation using gaussian copulas. *Sociol. Methods Res.* **2021**, *50*, 1259–1283. [CrossRef]
23. Di Lascio, F.; Giannerini, S.; Reale, A. Exploring copulas for the imputation of complex dependent data. *Stat. Methods Appl.* **2015**, *24*, 159–175. [CrossRef]
24. Houari, R.; Bounceur, A.; Kechadi, T.; Tari, A.; Euler, R. A new method for estimation of missing data based on sampling methods for data mining. *Adv. Intell. Syst. Comput.* **2013**, *225*, 89–100. [CrossRef]
25. Sklar, A. Fonctions de repartition an dimensions et leurs marges. *Publ. Inst. Statist. Univ. Paris* **1959**, *8*, 229–231.
26. Wei, G.; Tanner, M. A monte carlo implementation of the EM algorithm and the poor man's data augmentation algorithms. *J. Am. Stat. Assoc.* **1990**, *85*, 699–704. [CrossRef]
27. Meng, X.L.; Rubin, D. Maximum likelihood estimation via the ECM algorithm: A general framework. *Biometrika* **1993**, *80*, 267–278. [CrossRef]
28. Guo, J.; Levina, E.; Michailidis, G.; Zhu, J. Graphical models for ordinal data. *J. Comput. Graph. Stat.* **2015**, *24*, 183–204. [CrossRef]
29. McLachlan, G.; Lee, S.; Rathnayake, S. Finite mixture models. *Annu. Rev. Stat. Its Appl.* **2019**, *6*, 355–378. [CrossRef]
30. Hwang, J.; Lay, S.; Lippman, A. Nonparametric multivariate density estimation: A comparative study. *IEEE Trans. Signal Process.* **1994**, *42*, 2795–2810. [CrossRef]
31. Scott, D.; Sain, S. Multidimensional density estimation. *Handb. Stat.* **2005**, *24*, 229–261.
32. Zuo, Y.; Cui, Y.; Yu, G.; Li, R.; Ressom, H. Incorporating prior biological knowledge for network-based differential gene expression analysis using differentially weighted graphical LASSO. *BMC Bioinform.* **2017**, *18*, 99. [CrossRef]
33. Li, Y.; Jackson, S.A. Gene network reconstruction by integration of prior biological knowledge. *G3 Genes Genomes Genet.* **2015**, *5*, 1075–1079. [CrossRef] [PubMed]
34. Hastie, T.; Tibshirani, R.; Friedman, J. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*; Springer Series in Statistics; Springer: Berlin/Heidelberg, Germany, 2009.
35. Joyce, J.M. Kullback-Leibler divergence. In *International Encyclopedia of Statistical Science*; Springer: Berlin/Heidelberg, Germany, 2011; pp. 720–722.
36. Contreras-Reyes, J.E.; Arellano-Valle, R.B. Kullback–Leibler divergence measure for multivariate skew-normal distributions. *Entropy* **2012**, *14*, 1606–1626. [CrossRef]
37. Honaker, J.; King, G.; Blackwell, M. Amelia II: A program for missing data. *J. Stat. Softw.* **2011**, *45*, 1–47. [CrossRef]
38. Holzinger, A.; Langs, G.; Denk, H.; Zatloukal, K.; Müller, H. Causability and explainability of artificial intelligence in medicine. *WIREs Data Min. Knowl. Discov.* **2019**, *9*, e1312. [CrossRef] [PubMed]
39. Dinu, V.; Zhao, H.; Miller, P.L. Integrating domain knowledge with statistical and data mining methods for high-density genomic SNP disease association analysis. *J. Biomed. Inform.* **2007**, *40*, 750–760. [CrossRef]

40. Rubin, D. Multiple imputation after 18+ years. *J. Am. Stat. Assoc.* **1996**, *91*, 473–489. [CrossRef]
41. Van Buuren, S. *Flexible Imputation of Missing Data*; CRC Press: Boca Raton, FL, USA, 2018.
42. Ramosaj, B.; Pauly, M. Predicting missing values: A comparative study on non-parametric approaches for imputation. *Comput. Stat.* **2019**, *34*, 1741–1764. [CrossRef]
43. Ramosaj, B.; Amro, L.; Pauly, M. A cautionary tale on using imputation methods for inference in matched-pairs design. *Bioinformatics* **2020**, *36*, 3099–3106. [CrossRef]
44. Zhao, Y.; Udell, M. Missing value imputation for mixed data via gaussian copula. In Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, Virtual Event, CA, USA, 6–10 July 2020; Association for Computing Machinery: New York, NY, USA, 2020; pp. 636–646. [CrossRef]
45. Rubin, D.B. Causal Inference Using Potential Outcomes: Design, Modeling, Decisions. *J. Am. Stat. Assoc.* **2005**, *100*, 322–331. [CrossRef]
46. Ding, P.; Li, F. Causal inference: A missing data perspective. *Stat. Sci.* **2017**, *33*. [CrossRef]
47. Käärik, E.; Käärik, M. Modeling dropouts by conditional distribution, a copula-based approach. *J. Stat. Plan. Inference* **2009**, *139*, 3830–3835. [CrossRef]
48. Murphy, K. *Machine Learning: A Probabilistic Perspective*; The MIT Press: Cambridge, MA, USA, 2012.

Article 2

Kertel et al. (2022)

Learning Causal Graphs in Manufacturing Domains using Structural Equation Models

Maximilian Kertel
Technology Development Battery Cell
BMW Group
Munich, Germany
maximilian.kertel@bmw.de

Stefan Harmeling
Department of Computer Science
TU Dortmund University
Dortmund, Germany
stefan.harmeling@tu-dortmund.de

Markus Pauly
Department of Statistics
TU Dortmund University
Dortmund, Germany
Research Center Trustworthy Data Science and Security
UA Ruhr, Germany
pauly@statistik.tu-dortmund.de

***Abstract*—Many production processes are characterized by numerous and complex cause-and-effect relationships. Since they are only partially known they pose a challenge to effective process control. In this work we present how Structural Equation Models can be used for deriving cause-and-effect relationships from the combination of prior knowledge and process data in the manufacturing domain. Compared to existing applications, we do not assume linear relationships leading to more informative results.**

***Index Terms*—Causal Discovery, Bayesian Networks, Industry 4.0**

I. Introduction

Complex manufacturing processes as, e.g. for battery cells show high scrap rates and thus high production costs and large environmental footprints. One of the driving factors is the missing knowledge on the interdependencies between the process parameters, intermediate product properties and the quality characteristics [1]. Together we call this the cause-and-effect relationships (CERs). CERs can be visualized as a network with the process and product characteristics as nodes and the CERs as directed edges [1], [2]. It is the goal of our paper to unify expert knowledge and process data to derive such a network, which allows the visual identification of

- root-causes of erroneous products,
- relevant parameters for process control during successive production steps and
- important characteristics to predict the quality of the final product.

In complex manufacturing domains, CERs form a linked mesh of hundreds of involved factors [1]. Typically, CERs are derived by running Designs of Experiments (DOEs). However, DOEs can be time-demanding and the production line has to be stopped in the meantime leading to prohibitively high costs. Moreover, if there are many potential CERs, the number of experiments can become infeasible.

At the same time, the Internet of Things (IoT) allows data processing and storage along the whole production line, leading to a vast amount of accessible information. It is thus desirable to derive the CERs from the existing observational (or non-experimental) data. For this purpose, Bayesian Networks can be used to unify expert knowledge and data. From these, CERs can be derived under the assumption of causal sufficiency [3]. This approach is called *causal discovery* or *structure learning*. The most common example in the manufacturing domain [4]–[6], is the PC algorithm [3]. This algorithm relies on the assumption of faithfulness and on efficient statistical tests for conditional independence. In principle the PC algorithm can be applied with any test for conditional independence. However, existing nonparametric tests do not scale well [7], [8]. Most of the applications of the PC algorithm either discretize the measurements, or researchers approximate the joint distribution of the variables by a multivariate normal distribution. For discrete data and normally distributed data fast tests for conditional independence exist. However, the former leads to a loss of information, while the latter requires a linear dependency between the variables to be exact. In case of manufacturing data this is most likely a misspecification [9]. Simulation studies show, that the performance of the PC algorithm can be poor in case of non-linearity [10]. This questions the application of the PC algorithm for large or high-dimensional manufacturing data.

In recent years, Structural Equation Models (SEM), which can incorporate arbitrary functional relationships, were increasingly proposed to derive Causal Bayesian Networks. They replace the assumption on faithfulness by a functional form of the conditional distributions (see Equation (1)). While the PC algorithm returns a set of graphs, methods based on SEMs often derive a single graph. To the best of our knowledge, we are the first to apply SEMs to derive such graphical models in the manufacturing domain.

The paper is structured as follows. In Section II we present potential prior knowledge and available data in manufacturing domains. We continue in Section III by reviewing Bayesian Networks and SEMs and explain Causal Additive Models (CAM). In Section IV we present an extension of CAM, called TCAM, which efficiently incorporates prior knowledge. We apply our method in Section V to process data of the assembly of battery modules at BMW. We conclude in Section VI.

II. Data and Challenges in Complex Manufacturing Domains

In this section we describe the data sources and propose a preprocessing of the data. Then, we explain the broad prior knowledge in manufacturing domains. Finally, we mention common challenges with production data.

A. Data Sources along the Production Line

The assembly of products consists of production lines, which again contain several stations, which are passed in a fixed order and where process steps are carried out. During those process steps the piece is transformed or it is combined with other parts in order to achieve a predefined outcome. All involved parts are assigned to unique identifiers. Data of different types is collected along the production process:

- Process data: the stations take measurements of the involved parts (e.g. thickness of the piece) and the parameters of the machine (e.g. weight of applied glue).
- End-of-Line (EoL) tests take additional quality measurements of the intermediate or final products.
- Station information: at some production steps the pieces are spread out to identical stations, such that parts can be processed in parallel and every piece is assigned to one of the stations.
- Bill of Material (BoM): the BoM contains the information which pieces were merged together and on which position they have been worked in.
- Supplier data: suppliers transmit data on provided goods.

The preprocessing of the data, which is depicted in Figure 1, consists of the following steps:

1) Collect the data for every intermediate product.
2) Iteratively merge the data of all subcomponents of a final product.

Measurements of identical subcomponents, which are placed in the same position, can be found in the same column. Eventually, the final tabular data set contains all measurements that can be associated with a final product.

B. Prior Knowledge

As the stations are passed in a fixed order, we know that CERs across different stations can only act forward in time. Additonally, in many manufacturing organizations, tools as the Failure Mode and Effect Analysis (FMEA) [11] are implemented to extract expert knowledge on CERs in the production process and to provide the information in a structured form.

C. Challenges of Data Analysis in Manufacturing

Often, similar information is recorded multiple times along the production line, leading to multicollinearity [4]. Also, sensors might deliver non-informative data by recording implausible values. Industrial data is also reported to be drifting over time. However, even in shorter time intervals, data of a series production contains thousands of observations. This distinguishes the manufacturing domain from other applications of causal discovery as medicine, genetics or the social sciences.

III. Structure Learning of Graphical Models

A. Some Preliminaries on Graphical Models

Let $G = (\mathbf{V}, \mathbf{E})$ be a directed acyclic graph (DAG) [12, Chapter 6] with nodes $\mathbf{V} = (V_1, \dots, V_p)$ and edges $\mathbf{E}$. The node V_i is called a parent of V_j if the edge $V_i \to V_j$ is in $\mathbf{E}$. We denote the set of all parents of V_j as $pa(V_j)$. A tuple of nodes $(V_{j_1}, \dots, V_{j_\ell})$, such that V_{j_k} is a parent of $V_{j_{k+1}}$ for all $k = 1, \dots, (\ell - 1)$, is called a *directed path*. Nodes that can be reached from X_j through a directed path are called the *descendants* of X_j.

In the following we denote random vectors with bold letters as $\mathbf{Z}$ and random variables as Z. Let $\mathbf{X} = (X_1, \dots, X_p)$ be a random vector representing the data generating process. For a graph G with nodes $X_1, \dots, X_p$, we call $(\mathbf{X}, G)$ a Bayesian network if the local Markov property holds, i.e.

$$X_i \perp X_j | pa(X_i)$$

for any X_j that is not a descendant of X_i in G. Here, $X \perp Y | \mathbf{Z}$ denotes the conditional independence of X and Y given $\mathbf{Z}$. In that case, we can deduce additional conditional independencies for $\mathbf{X}$ from the graph G using the concept of *d-separation* [12]. For a Bayesian Network $(\mathbf{X}, G)$, it then holds that $X_i \perp X_j | \mathbf{S}$ if X_i and X_j are d-separated by $\mathbf{S}$ in G. On the other hand, if there is a graph G, such that $X_i \perp X_j | \mathbf{S}$ implies that X_i and X_j are d-separated given $\mathbf{S}$ in G, then $\mathbf{X}$ is called *faithful with respect to* G. As multiple graphs can contain the same d-separations, this graph G is in general not unique.

To promote the intuition, assume that $\mathbf{X}$ has a joint density f. Then $X_i \perp X_j | \mathbf{S}$ can be characterized by

$$f\left(x_i | X_j = x_j, \mathbf{S} = \mathbf{s}\right) = f\left(x_i | \mathbf{S} = \mathbf{s}\right),$$

where $f\left(x_i | \mathbf{Z} = \mathbf{z}\right)$ denotes the conditional density function of X_i given $\mathbf{Z} = \mathbf{z}$. Thus, if we already know $\mathbf{S}$, then X_j does not provide additional information on X_i. Assume that we are interested which variable in $\{X_j, \mathbf{X_S}\}$ causes the variable X_i to be out of the specification limits. Then we know, that the root causes can be found within $\mathbf{S}$.

B. Graph Learning with Structural Equation Models

While the PC algorithm is the classic approach for deriving a Causal Bayesian Network, recent research focused on identifying it using acyclic SEMs [10], [13]–[15]. They assume that there exists a permutation $\Pi^0(1, \dots, p) = \left(\pi^0(1), \dots, \pi^0(p)\right)$ and functions $\{f_\ell, \ell = 1, \dots, p\}$, such that

$$X_\ell = f_\ell(X_{\ell_1}, \dots, X_{\ell_v}, \varepsilon_\ell), \ \ell = 1, \dots, p, \tag{1}$$

where $\pi^0(\ell_k) < \pi^0(\ell)$ for all $k = 1, \dots, v$ and $\varepsilon_1, \dots, \varepsilon_p$ are i.i.d. noise terms. As the estimation of f_ℓ in Equation (1) is difficult in high dimensions, one typically restricts the function class and the distribution of the noise terms. In this work, we assume that the functions follow the additive form

$$f_\ell(X_{\ell_1}, \dots, X_{\ell_v}, \varepsilon_\ell) = c_\ell + \sum_{k:\pi^0(k)<\pi^0(\ell)} f_{k,\ell}(X_k) + \varepsilon_\ell, \tag{2}$$

where $\varepsilon_\ell \sim \mathcal{N}(0, \sigma_\ell)$ and $c_\ell \in \mathbb{R}$. To ensure the uniqueness of the $f_{k,\ell}$ and without loss of generality, we set $\mathbf{E}\left(X_\ell\right) = 0$

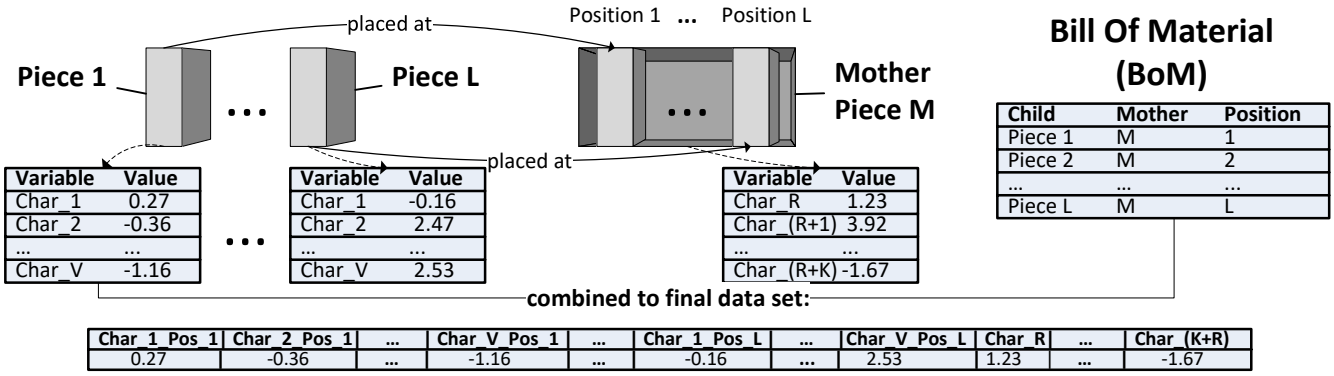

Fig. 1. Visualization of the data preparation described in Section II. The same measurements are collected for piece 1 to piece L. Then they are placed in their mother piece with identifier M. Finally, the resulting data set consists of all measurements of M and those from piece 1 to piece L, where the positioning of the measurements of the child pieces within the data frame depends on their placement according to the BoM. This step is carried out repeatedly, if M itself is positioned in another mother piece.

and $\mathbf{E}\left(f_{k,\ell}(X_k)\right) = 0$, for all $\ell = 1, \ldots, p, \pi^0(k) < \pi^0(\ell)$. From Equations (1) and (2) we derive that

$$X_\ell \perp X_k | \left(X_{v_1}, \ldots, X_{v_j}\right),$$

with $\pi^0(k) < \pi^0(\ell), \pi^0(v_1) < \pi^0(\ell), \ldots, \pi^0(v_j) < \pi^0(\ell)$ if and only if $f_{k,\ell} = 0$. Let G^0 be the graph on $X_1, \ldots, X_p$, that contains the edge $X_i \to X_j$ if and only if $f_{i,j} \neq 0$ for $\pi^0(i) < \pi^0(j)$. Then $(\mathbf{X}, G^0)$ is a Bayesian network, as it is fulfilling the Markov property.

If we assume that the functions $f_{k,\ell}$ in Equation (2) are non-linear and smooth, then [15] show that G^0 is identifiable from observational data. This is in contrast to the PC algorithm, which typically returns a class of graphs. Note that we do not presume that the distribution is faithful to some DAG, which is a central assumption of the PC algorithm. We emphasize that for the PC algorithm non-linearity is an obstacle as efficient conditional independence testing is just feasible for multivariate normal data. In contrast, we can utilize the non-linearity for identifying SEMs to receive more informative results (under the assumption of Equation (2)).

An example of a learning algorithm for SEMs is the Causal Additive Model (CAM, [10]). We will focus on CAM due to its applicability to high-dimensional data, its ability to capture non-linearity and due to the theoretical justification that G^0 can be identified, if the functions on the right-hand side of Equation 2 are nonlinear and smooth. [10] propose to find G^0 with the following steps:

1) Find the underlying node ordering Π^0 of $X_1, \ldots, X_p$.
2) Identify the influential functions $f_{k,\ell}$ with feature selection methods.

To make things more precise, consider N observations $(x_{i1}, \ldots, x_{ip}), i = 1, \ldots, N$ from $\mathbf{X}$ and call the data matrix $\mathbf{D} \in \mathbb{R}^{N \times p}$.

1) Finding the node ordering: [10] show that if

- the functions $f_{k,\ell}$ are smooth and non-linear and can be approximated well and
- the derivatives of $f_{k,\ell}$ and the fourth moments of $f_{k,\ell}(X_k)$ and X_k are bounded.

then the following estimator for Π^0 is consistent as $N \to \infty$:

$$\hat{\Pi} = \underset{\Pi}{\operatorname{argmin}} \sum_{\ell=1}^{p} ||x_\ell - \sum_{\pi(k)<\pi(\ell)} \widehat{f}_{k,\ell}(x_k)||_{2,N}^2 \qquad (3)$$

Here, we define $||x_k||_{2,N}^2 := \frac{1}{N}\sum_{k=1}^{N} x_{k\ell}^2$ and $\widehat{f}_{k,\ell}$ is found by running an additive model regression [16] of X_ℓ on $\{X_k : \pi(k) < \pi(\ell)\}$.

For large p, [10] propose a greedy method to find $\hat{\Pi}$. Let G be a DAG on $\mathbf{X}$ with edges $E(G)$. For simplicity we denote the edge $X_k \to X_\ell$ by (k, ℓ). A score for G is defined by

$$S(G) = \sum_{\ell=1}^{p} ||x_\ell - \sum_{(k,l)\in E(G)} \widehat{f}_{k,\ell}(x_k)||_{2,N}^2.$$

The functions $\widehat{f}_{k,\ell}$ are estimated by running an additive model regression of X_ℓ on its parents in G. Intuitively, $S(G)$ indicates how much variation of $\mathbf{D}$ is captured by G. The edges that can be added to G without causing cycles are denoted by

$$A(G) := \{(i,j) \in \{1,\ldots,p\} \times \{1,\ldots,p\} : (\mathbf{X}, E(G) \cup \{(i,j)\}) \text{ is DAG}\}.$$

Starting with the empty graph G_0, [10] iteratively add the edge $(k^0, \ell^0) = \operatorname{argmax}_{(k',\ell')\in A(G_t)} M_t(k', \ell')$, where

$$\begin{aligned} M_t(k',\ell') = &||x_\ell - \sum_{(k,\ell)\in E(G_t)} \widehat{f}_{k,\ell}(x_k)||_{2,N}^2 \\ &- ||x_\ell - \sum_{(k,l)\in E(G_t)\cup\{(k',\ell')\}} \widetilde{f}_{k,\ell}(x_k)||_{2,N}^2. \end{aligned} \qquad (4)$$

The functions $\widehat{f}_{k,\ell}$ are found by regressing X_ℓ on its parents in G_t, while $\widetilde{f}_{k,\ell}$ are found by regressing X_ℓ on its parents in $G' = (\mathbf{X}, E(G_t) \cup \{(k',\ell')\})$. Thus, the edge (k^0, ℓ^0) maximally reduces the unexplained variance. We set $G_{t+1} = (\mathbf{X}, E(G_t) \cup \{(k^0, \ell^0)\})$ and continue until we obtain

a complete DAG, which implies the node ordering.
This greedy method is still computationally intense for large p. Thus, [10] propose to take advantage of sparse structures, where p is large but the number of edges in the graph is assumed to be small: to this end they start by a preliminary neighborhood selection (PNS) step. Here, initially for every $\ell \in \{1, \dots, p\}$ a superset of neighbors of X_ℓ in G^0 is identified. In the subsequent node ordering step, one only considers the superset of the neighbors, when greedily adding new edges. This reduces the computation time of the algorithm significantly, if the sizes of the supersets are considerably smaller than p.

2) Identifying edges: After the node ordering is set, we need to identify the influential characteristics for every X_ℓ among those X_k for which $\hat{\pi}(k) < \hat{\pi}(\ell)$. The idea is to detect those $f_{k,\ell}$ which are not 0, using feature selection methods [16], [17]. For those k, a change in X_k has an effect on X_ℓ. For a comparison of CAM and the PC algorithm based on simulated data sets with known ground truth, see [10].

IV. Methodology

The goal of this section is to derive a method that combines the current results on structure learning of SEMs with the features of the manufacturing domain in Section II.

A. Recap of Common Prior Knowledge

Compared to other applications of causal discovery, it is typical for the manufacturing domain, that there exists prior knowledge, see Section II. In particular, there is a partial and transitive ordering of the variables implied by the stations' ordering. Additionally, we include expertise on the absence of edges. Both facets shall improve the algorithm's runtime.

B. Adaptions to CAM

The data generating process behind manufacturing data sets often leads to a low number of conditional independencies in $\mathbf{X}$, when compared to p. Thus, the Causal Bayesian Network of $\mathbf{X}$ is not sparse. This poses a challenge to many structural learning algorithms in higher dimensions. We show in this subsection how prior knowledge on the node ordering and the existence of edges can be incorporated so that structure learning remains feasible. To formalize our prior knowledge, let $t : \{1, \dots, p\} \to \{1, \dots, T\}$, so that $t(k) < t(\ell)$ means that there can only be edges from X_k to X_ℓ but not vice versa. Further, let F be a boolean matrix, where $F_{k,\ell} = \texttt{True}$ if the edge from X_k to X_ℓ is known to be absent.

1) Preliminary Neighborhood Selection: For every measurement X_ℓ, we determine a set of possible parents among those k, where $F_{k,\ell} = \texttt{False}$ and $t(k) \leq t(\ell)$. Denote that set for index ℓ by P_ℓ.

2) Node Ordering: We start by adding all potential edges that go across stations and add them to the initial graph G_0, as those can not cause any cycle. The score of G_0 hence is

$$S(G_0) = \sum_{\ell=1}^{p} \sum_{k \in P_\ell, t(k) < t(\ell)} ||x_\ell - \widehat{f}_{k,\ell}(x_k)||^2_{2,N}. \quad (5)$$

We continue by determining the node ordering as in Section III-B. Note that we only need to determine the node ordering for indices k, ℓ so that $t(k) = t(\ell)$. The initial inclusion of across-station-edges saves update steps of M (Equation 4). This makes the algorithm feasible even for non-sparse high-dimensional settings, if the number of tiers T or the number of edges known to be absent is sufficiently large.

3) Pruning: The pruning step is identical to CAM.
In the manufacturing industry, the prior knowledge on $t(k) < t(l)$ is often given by the temporal nature of the production process. We therefore call our adaption *TCAM* (*Temporal* Causal Additive Models). It is sketched in Algorithm 1.

Algorithm 1: TCAM Algorithm

Input: D, F, t as in Section IV-B
Result: DAG G
// Preliminary Neighborhood Selection (PNS)
SupersetNeighbors $= \texttt{list}()$;
for $\ell = 1, \dots, p$ **do**
  $I = \{k \; s.t. \; t(k) \leq t(\ell) \;\&\; F(k, \ell) = \texttt{False}\}$;
  SupersetNeighbors$[\ell] = \texttt{PNS}\,(X_\ell, \mathbf{X}_I)$;
  // Other edges are now forbidden
  for $k = 1, \dots, p$ **do**
    if $k \notin$ SupersetNeighbors$[\ell]$ **then**
      $F(k, \ell) = \texttt{True}$;
    end
  end
end
// Add across-tier edges
Set G as empty graph on $X_1, \dots, X_p$;
for $k, \ell = 1, \dots, p$ **do**
  if $k \in$ SupersetNeighbors$[\ell]$ $\&$ $t(k) < t(\ell)$ **then**
    Append (k, ℓ) to edges of G;
  end
end
$M(k, \ell) =$ right-hand side of (5) ;
for $k, \ell = 1, \dots, p$ **do**
  if $((t(k) > t(\ell)) \,|\, (F(k, \ell) = \texttt{True}))$ **then**
    $M(k, \ell) = -\infty$;
  end
end
// Add within-tier edges
while $\max(M) > -\infty$ **do**
  Find $(k_0, \ell_0) = \operatorname{argmax}_{(k,\ell) \in A(G)} M(k, \ell)$;
  Append (k_0, ℓ_0) to edges of G;
  $M(k_0, \ell_0) = -\infty$;
  Update $M(\cdot, \ell_0)$;
  Set $M(k, \ell) = -\infty$ for all
   $\{1, \dots, p\} \times \{1, \dots, p\} \ni (k, \ell) \notin A(G)$;
end
// Pruning like CAM (details omitted)
return G

V. Application

The energy storage of electric vehicles is called a battery pack which is composed of battery modules, which in turn contain a fixed number of battery cells. A battery module connects the battery cells in series or parallel and it protects those cells against shock, vibration and heat. Thus, the battery module is a key component for the safety of battery-electric vehicles. We apply TCAM to data collected at the assembly at BMW. The data set under investigation contains 7254 battery modules with 738 variables each.

A. Data Preparation

As the missing values rate is low (around 2.4%) we apply naive mean imputation instead of more sophisticated method as [18]–[20]. We then continue by removing features that have only one distinct value and hence provide no information. As this data set also shows multicollinearity, we apply an expert-based approach. We asked experts to identify clusters of variables containing similar information and to define representatives for them. For a purely data-driven approach in manufacturing, see [5]. Those steps reduced the number of characteristics from 738 to 491. Finally, we standardize the data so the variables' empirical mean and standard deviation is 0 and 1 respectively.

Beyond the temporal ordering of the stations, it is reasonable that the production measurements of identical intermediate products as depicted in Figure 1 are independent. Thus, it is possible to restrict the potential edges that have to be considered. Additionally, we assume that some of the recorded measurements as the facility temperature and the selection of the stations are not affected by other measurements. We can mark those values as *root nodes*, meaning that they have no incoming edges. This further restricts the number and orientation of possible edges.

B. Choice of Software and Hyperparameters

1) Preliminary Neighborhood Selection: For our application of TCAM, we find supersets of the neighbors by applying the LASSO. For $\ell \in \{1, \dots, p\}$, we run a regression of X_ℓ on those components of $\mathbf{X}$, which are possible parents according to our prior knowledge. Going forward we mark those variables as potential parents of X_ℓ, where the corresponding regression coefficient is above 10^{-2}. The penalty parameter λ is chosen via cross-validation. Let λ_{min} be the penalty parameter that minimizes the mean squared cross-validation error. Then we choose the maximal λ such that the mean cross-validation error is within one standard deviation of the minimum λ_{min}.

2) Node Ordering and Pruning: For the node ordering we employ the package `mgcv` by [16]. Let us call the graph after node ordering G_{NO}. In the pruning step we run a sparse additive regressions of X_ℓ on its parents in G_{NO} for $\ell = 1, \dots, p$. This step returns p-values for the parents of X_ℓ in G_{NO}. We follow [10] and set the regressands as parents of X_ℓ in the final graph, whose p-values are below the threshold of 10^{-3}.

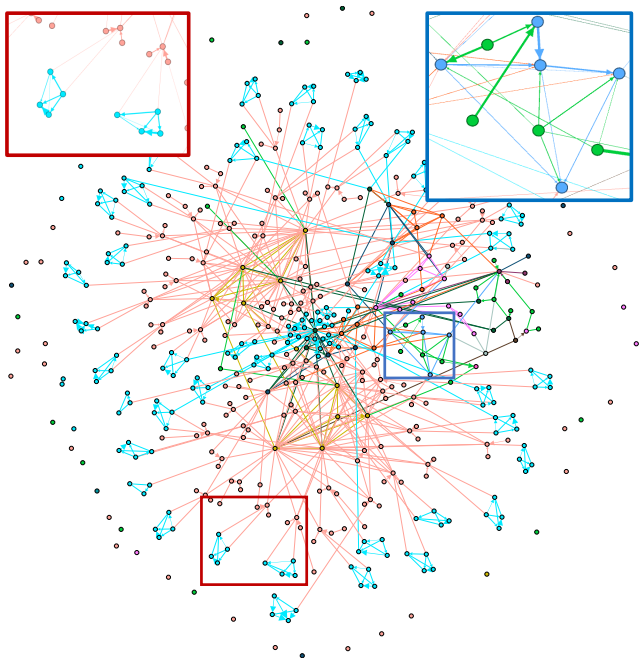

Fig. 2. Resulting graph of TCAM where the nodes correspond to characteristics of the product and edges correspond to detected CERs. The node coloring is according to the station, where the variable was measured. Edge colors are according to respective source node's color. The blue box highlights the detected relationship between the choice of the stations (green nodes) and the product quality (blue nodes). The red box depicts the similarities between structures of identical subcomponents.

C. Results

The resulting graph is depicted in Figure 2 and contains 491 nodes and 859 edges. We observe that there are a few nodes that have a large number of neighbors. In general this poses a difficulty for most structure learning algorithms and CAM did not finish in reasonable time. For details on the runtime for a low-dimensional special case, see Section V-D. With TCAM and the inclusion of prior knowledge we were able to overcome those obstacles.

Further, substructures of identical parts show similar patterns. The red box in Figure 2, highlights patterns consisting of two linked clusters, where one cluster consists of four nodes, while the other one consists of three nodes. Together with process experts we could further verify that many CERs detected by TCAM are plausible.

This application is confidential, but we would still like to share one of the insights. TCAM discovered a CER between one station that processed the part and the part's quality. Experts derived that the maintenance of that station was overdue and the CER can be used to find better maintenance intervals. This is one example how graphical models can contribute to an effective and proactive process control.

D. Evaluation against Expert Knowledge

For the characteristics of one of the subcomponents, we derived an expert-based graph, which is depicted in Figure 3. Here, the blue CERs potentially exist, while green CERs surely

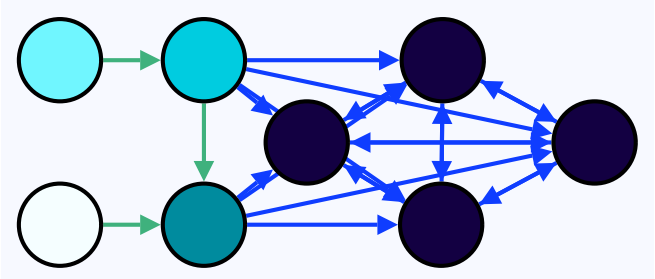

Fig. 3. Expert-based graph on measurements for subcomponents. The green edges are known to exist, while the blue edges potentially exist. Edges beyond the ones depicted are known to be absent. The darker the node, the later the corresponding variable is measured in the production process.

	$\overline{\text{aSHD}}$	sd(aSHD)	$\overline{\#\text{edges}}$	sd(#edges)	$\overline{\text{time (s)}}$
CAM	3.496	1.442	9.464	0.948	1.342
TCAM	1.120	0.343	8.084	0.778	1.000
TPC	1.108	0.866	7.463	1.623	0.013

TABLE I

THE AVERAGE ASHD ($\overline{\text{ASHD}}$), THE STANDARD DEVIATION OF THE ASHD (SD(ASHD)), THE AVERAGE NUMBER OF EDGES ($\overline{\#\text{EDGES}}$) AND THE STANDARD DEVIATION OF THE NUMBER OF EDGES (SD(#EDGES)) FOR ALL THREE METHODS OF SECTION V-D AND FOR 500 REPLICATIONS.

exist. Other CERs can be ruled out. We compare the estimated graphs and runtimes of TCAM, CAM and a variant of the PC algorithm called TPC [21], which allows the inclusion of temporal background knowledge. The significance level is set to 0.01. We run 500 experiments, where we randomly draw 500 subcomponents, while each of them appears in at most one of the runs. We define an adapted Structural Hamming Distance (aSHD) [22] between an estimated graph G_{est} and the one in Figure 3 by the sum over the number of green edges that are not in G_{est} and the number of edges G_{est} that do not appear in Figure 3. The results are depicted in Table I. TPC and TCAM perform better than CAM, which shows the advantage of the inclusion of prior knowledge. Additionally, even in this low-dimensional setting the average runtime for TCAM is smaller than for CAM. Further, we observe that the aSHD of TCAM and TPC is on average quite similar. However, the standard deviation of the aSHD and the standard deviation of the number of edges is smaller for TCAM. This indicates that TCAM delivers more stable and informative results in the manufacturing domain. The original PC algorithm performed worse than TPC and is omitted.

VI. CONCLUSION

We have presented a method to derive the graphical representation of CERs of manufacturing processes based on SEMs. While existing approaches for causal discovery in the manufacturing domain assumed linear relationships between the process characteristics, we applied CAM to find arbitrary additive functional relationships in data. We showed how existing prior domain knowledge can be included and improves the computational burden of CAM. A case study on manufacturing data reveals that the learned graph detects unknown root-causes, delivers more informative results and paves the way to an efficient and proactive process control.

REFERENCES

[1] T. Kornas, R. Daub, M. Z. Karamat, S. Thiede, and C. Herrmann, "Data- and expert-driven analysis of cause-effect relationships in the production of lithium-ion batteries," in *2019 IEEE 15th International Conference on Automation Science and Engineering (CASE)*. IEEE, 2019, pp. 380–385.

[2] T. Wuest, C. Irgens, and K.-D. Thoben, "An approach to monitoring quality in manufacturing using supervised machine learning on product state data," *Journal of Intelligent Manufacturing*, vol. 25, no. 5, pp. 1167–1180, 2014.

[3] P. Spirtes, C. N. Glymour, R. Scheines, and D. Heckerman, *Causation, prediction, and search*. MIT press, 2000.

[4] M. Vuković and S. Thalmann, "Causal discovery in manufacturing: A structured literature review," *Journal of Manufacturing and Materials Processing*, vol. 6, no. 1, p. 10, 2022.

[5] K. Marazopoulou, R. Ghosh, P. Lade, and D. Jensen, "Causal discovery for manufacturing domains," 2016. [Online]. Available: https://arxiv.org/abs/1605.04056

[6] J. Li and J. Shi, "Knowledge discovery from observational data for process control using causal bayesian networks," *IIE transactions*, vol. 39, no. 6, pp. 681–690, 2007.

[7] K. Zhang, J. Peters, D. Janzing, and B. Schölkopf, "Kernel-based conditional independence test and application in causal discovery," in *Proceedings of the Twenty-Seventh Conference on Uncertainty in Artificial Intelligence*, ser. UAI'11. Arlington, Virginia, USA: AUAI Press, 2011, p. 804–813.

[8] K. Fukumizu, A. Gretton, X. Sun, and B. Schölkopf, "Kernel measures of conditional dependence," *Advances in neural information processing systems*, vol. 20, 2007.

[9] P. Lade, R. Ghosh, and S. Srinivasan, "Manufacturing analytics and industrial internet of things," *IEEE Intelligent Systems*, vol. 32, no. 3, pp. 74–79, 2017.

[10] P. Bühlmann, J. Peters, and J. Ernest, "CAM: Causal additive models, high-dimensional order search and penalized regression," *The Annals of Statistics*, vol. 42, no. 6, pp. 2526 – 2556, 2014. [Online]. Available: https://doi.org/10.1214/14-AOS1260

[11] D. H. Stamatis, *Failure mode and effect analysis: FMEA from theory to execution*. Quality Press, 2003.

[12] J. Peters, D. Janzing, and B. Schölkopf, *Elements of Causal Inference: Foundations and Learning Algorithms*. Cambridge, MA, USA: MIT Press, 2017.

[13] X. Zheng, B. Aragam, P. K. Ravikumar, and E. P. Xing, "Dags with no tears: Continuous optimization for structure learning," *Advances in Neural Information Processing Systems*, vol. 31, 2018.

[14] S. Shimizu, T. Inazumi, Y. Sogawa, A. Hyvärinen, Y. Kawahara, T. Washio, P. O. Hoyer, and K. Bollen, "Directlingam: A direct method for learning a linear non-gaussian structural equation model," *The Journal of Machine Learning Research*, vol. 12, pp. 1225–1248, 2011.

[15] J. Peters, J. M. Mooij, D. Janzing, and B. Schölkopf, "Causal discovery with continuous additive noise models," *The Journal of Machine Learning Research*, vol. 15, no. 1, p. 2009–2053, 01 2014.

[16] S. N. Wood, *Generalized additive models: an introduction with R*. Chapman and Hall/CRC, 2006, vol. 2.

[17] J. Huang, J. L. Horowitz, and F. Wei, "Variable selection in nonparametric additive models," *Annals of statistics*, vol. 38, no. 4, p. 2282, 2010.

[18] S. van Buuren, *Flexible Imputation of Missing Data*. Chapman and Hall/CRC, 2012, vol. 2.

[19] B. Ramosaj, J. Tulowietzki, and M. Pauly, "On the relation between prediction and imputation accuracy under missing covariates," *Entropy*, vol. 24, no. 3, p. 386, 2022.

[20] M. Kertel and M. Pauly, "Estimating gaussian copulas with missing data," 2022. [Online]. Available: https://arxiv.org/abs/2201.05565

[21] R. M. Andrews, R. Foraita, V. Didelez, and J. Witte, "A practical guide to causal discovery with cohort data," 2021. [Online]. Available: https://arxiv.org/abs/2108.13395

[22] I. Tsamardinos, L. E. Brown, and C. F. Aliferis, "The max-min hill-climbing bayesian network structure learning algorithm," *Machine learning*, vol. 65, no. 1, pp. 31–78, 2006.

Article 3

Kertel and Klein (2024)

Boosting Causal Additive Models

Maximilian Kertel[1,2] and Nadja Klein[2,3]

[1] Technology Development Battery Cell, BMW Group, Munich, Germany
[2]Department of Statistics, TU Dortmund University, Dortmund, Germany
[3]Chair of Uncertainty Quantification and Statistical Learning,
Research Center Trustworthy Data Science and Security (UA Ruhr)

March 6, 2024

Abstract

We present a boosting-based method to learn additive Structural Equation Models (SEMs) from observational data, with a focus on the theoretical aspects of determining the causal order among variables. We introduce a family of score functions based on arbitrary regression techniques, for which we establish necessary conditions to consistently favor the true causal ordering. Our analysis reveals that boosting with early stopping meets these criteria and thus offers a consistent score function for causal orderings. To address the challenges posed by high-dimensional data sets, we adapt our approach through a component-wise gradient descent in the space of additive SEMs. Our simulation study underlines our theoretical results for lower dimensions and demonstrates that our high-dimensional adaptation is competitive with state-of-the-art methods. In addition, it exhibits robustness with respect to the choice of the hyperparameters making the procedure easy to tune.

Keywords: causal discovery; directed acyclic graph; boosting; high-dimensional data; reproducing kernel Hilbert space

1 Introduction

Causal discovery is the process of deriving causal relationships between variables in a system. These help in improving decisions, predictions and interventions [Kyono et al., 2020]. With the rapid growth of large-scale data sets in fields as healthcare, genetics [Aibar et al., 2017], or manufacturing [Kertel et al., 2023], causal discovery has become increasingly important. Traditionally, the researcher designs an experiment and intervenes on certain variables, that is, assigns for example a drug or placebo, which allows to estimate the causal effect of the manipulated variables. Often however, it is less time, and effort consuming or more practical or ethical to collect data of a steady state of the system, where no variables are manipulated. For instance, in complex manufacturing domains, the number of input variables might be too large to conduct a design of experiments, and additionally those experiments would lead to a production downtime generating high costs.

Thus, although observational data is often complex, noisy, or has confounding variables [Bhattacharya et al., 2021], it is desirable to derive causal relationships from observational data rather than having to rely on impractical experimental studies.

In this work, we follow the assumption that the causal relationships between variables can be modeled as a Directed Acyclic Graph (DAG). This assumption implies that the impact between two variables flows in at most one direction, and there are no cyclic or self-reinforcing pathways. It is the goal of the present work to identify the DAG from observational data. Existing algorithms for this task can be broadly classified into two categories. First, constraint-based methods rely on statistical tests for conditional independence, but these approaches become computationally expensive and difficult apart from multivariate normal or multinomial distributions [Zhang et al., 2011, Shah and Peters, 2018]. Additionally, they assume faithfulness [Peters et al., 2017], and the identified graph is typically not unique [Spirtes et al., 1993, Kalisch and Bühlman, 2007].

Instead, we focus on the second category for identifying DAGs from observational data, namely on Structural Equation Models (SEMs). SEMs rely on the assumption that each variable is a function of other variables in the system and a perturbing noise term. Peters et al. [2014] show that if one restricts the functional relationships and the noise, called the Additive Noise Model (ANM), then the implied distribution is unique. Specifically, this is the case when the functional relationships are non-linear, and the noise term is Gaussian, as we assume throughout this work. This allows to identify the SEM from data.

Causal discovery using ANMs is an active field of research [Vowels et al., 2022]. Many recent works leverage continuous acyclity characterisations [Lachapelle et al., 2019, Yu et al., 2019, Zheng et al., 2020, Kalainathan et al., 2022, Ng et al., 2022a,b] and find the DAG using gradient-descent. However, opposed to earlier works [as for example, Shimizu et al., 2011, Bühlmann et al., 2014] for most machine learning methods the statistical behaviour is unknown [Kaiser and Sipos, 2022]. In this context, a popular contribution is that of Aibar et al. [2017], which successfully derives the cyclic graphical representation of a gene regulatory network using boosting. Our work is an extension of this approach towards acyclic graphs.

In this paper, we leverage the success of gradient-based methods towards statistical boosting for causal discovery in ANMs and investigate the underlying statistical behaviour. In particular, we show consistency and robustness of our proposed method.

Our main contributions are the following.

- We propose a generic method for causal discovery based on unspecified regression techniques. We then state assumptions on the regression technique that lead to consistent causal discovery, see Proposition 1.
- We show in Theorem 5 that L^2-boosting with early stopping fulfills the conditions of Proposition 1, that is, L^2-boosting can be employed for consistent causal discovery.
- We show in Theorem 6 that L^2-boosting with early stopping avoids overfitting even under misspecification. Further, if the model is correctly specified, then L^2-boosting with early stopping is consistent, as we show in Theorem 7.
- We propose a variant of the boosting procedure for high-dimensional settings, when the number of variables p is large.
- We conduct a simulation study demonstrating that our approach is competitive with state-of-the-art methods. We furthermore show that it is robust with respect to the choice of the hyperparameters and thus easy to tune.

This paper is structured as follows. In Section 2 we discuss SEMs and required assumptions. Section 3 reviews boosting and Reproducing Kernel Hilbert Spaces (RKHSs). Section 4 proposes the novel causal discovery method and shows its consistency. Section 5 explores the results of Section 4 empirically and benchmarks different state-of-the-art algorithms with our approach, while Section 6 concludes. Technical details and proofs can be found in the Appendix.

2 Causal Discovery

Let $\mathbf{X} = (X_1, \ldots, X_p)^\top$ be a p-dimensional random vector. For any $S = \{s_1, \ldots, s_T\} \subset \{1, \ldots, p\}$ and a vector $\mathbf{x} \in \mathbb{R}^p$ we define $\mathbf{x}_S := (x_{s_1}, \ldots, x_{s_T})^\top$. Analogously, for the random vector $\mathbf{X}$ we set $\mathbf{X}_S := (X_{s_1}, \ldots, X_{s_T})^\top$. We order the elements in S to make the representations unique, such that $s_1 < s_2 < \ldots < s_T$. For a Directed Acyclic Graph (DAG) G on $X_1, \ldots, X_p$ we define the parents of $k \in \{1, \ldots, p\}$ denoted by $\mathbf{pa}_G(k)$ as those $j \in \{1, \ldots, p\}$ for which the edge $X_j \to X_k$ exists in G. We assume that there exists a DAG G so that $\mathbf{X}$ follows a SEM with additive noise, that is

$$X_k = f_k(\mathbf{X}_{\mathbf{pa}_G(k)}) + \varepsilon_k. \tag{1}$$

Here, ε_k are i.i.d. noise terms for $k = 1, \ldots, p$.

Every DAG has at least one topological ordering π (that is, a permutation) on $\{1, \ldots, p\}$. We denote the nonempty set of topological orderings for G by $\Pi(G)$. With $\pi \in \Pi(G)$ there can only be a directed path in G from X_j to X_k if $\pi(j) < \pi(k)$ but not vice versa; see Figure 1 for an illustrating example.

$$\begin{aligned} X_1 &= f_1(X_2) + \varepsilon_1 \\ X_2 &= \varepsilon_2 \\ X_3 &= f_3(X_2) + \varepsilon_3 \end{aligned}$$

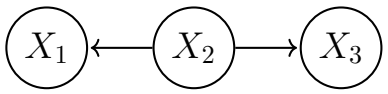

Figure 1: An example of a SEM on the left-hand side and its corresponding graph G on the right-hand-side for $p = 3$. The set of possible topological orderings is $\{(2, 1, 3), (2, 3, 1)\}$. For $\pi^0 = (2, 3, 1)$ it holds $X_1 = f_1(X_2) + \varepsilon_1 = f_{12}(X_2) + f_{13}(X_3) + \varepsilon_1$ with $f_{12} = f_1$ and $f_{13} = 0$.

2.1 Identifiability

The goal of our analysis is to identify the graph G from the distribution of $\mathbf{X}$. In general there is no one-to-one correspondence between the distribution $P(\mathbf{X})$ and the underlying SEM or G. However, if we impose appropriate restrictions on the noise terms $\{\varepsilon_k : k = 1, \ldots, p\}$ and the functions $\{f_k : k = 1, \ldots, p\}$, then the desired one-to-one correspondence between a distribution and a SEM exists [see Peters et al., 2014, Corollary 31]. In this case, we call the SEM identifiable. We consider SEMs with the following assumptions, which guarantee identifiability.

Assumption 1. *For the SEM of Equation* (1) *assume that* f_k *has the additive decomposition*

$$f_k(\mathbf{x}_{\mathbf{pa}_G(k)}) = \sum_{j \in \mathbf{pa}_G(k)} f_{kj}(x_j), \quad k = 1, \ldots, p,$$

where the f_{kj} *are three times differentiable, non-linear and non-constant for any* $k = 1, \ldots, p$ *and* $j \in \mathbf{pa}_G(k)$*. Further, let* $(\varepsilon_1, \ldots, \varepsilon_p)$ *be a random vector of independent components, which are normally distributed with mean zero and standard deviations* $\sigma_1, \ldots, \sigma_p > 0$*. We call SEMs of this form Causal Additive Models [CAMs; Bühlmann et al., 2014].*

Define $\varpi_\pi(k) = \{j : \pi(j) < \pi(k)\}$ as the predecessors of k with respect to a permutation π. Thus for $p = 3$ and $\pi = (2, 1, 3)$ it holds $\varpi_\pi(1) = \{2\}, \varpi_\pi(2) = \emptyset, \varpi_\pi(3) = \{1, 2\}$.

Consider a CAM, which is characterized by functions $f_1, \ldots, f_p$, standard deviations $\sigma_1, \ldots, \sigma_p$, and a graph G with topological ordering π. The implied density is

$$p(\mathbf{x}) = \prod_{k=1}^{p} p\left(x_k | \mathbf{x}_{\mathbf{pa}_G(k)}\right) = \prod_{k=1}^{p} p\left(x_k | \mathbf{x}_{\varpi_\pi(k)}\right). \tag{2}$$

Here, $p\left(x_k | \mathbf{x}_S\right)$ is the density of the conditional distribution of X_k given $\mathbf{X}_S = \mathbf{x}_S$, which we assume to exist throughout this work. For $S = \emptyset$, we set $p\left(x_k | \mathbf{x}_S\right) = p\left(x_k\right)$. The second equality in Equation (2) holds since X_k is independent from its predecessors in π given its parents [see Peters et al., 2017, Proposition 6.31], that is

$$X_k \perp X_{\varpi_\pi(k) \setminus \mathbf{pa}_G(k)} | \mathbf{X}_{\mathbf{pa}_G(k)}. \tag{3}$$

For any combination of $f_1, \ldots, f_p$, G and any topological ordering π of G, it trivially holds

$$f_k = \sum_{j \in \mathbf{pa}_G(k)} f_{kj}(X_j) = \sum_{j \in \varpi_\pi(j)} \widetilde{f_{kj}}(X_j),$$

where $\widetilde{f_{kj}} = f_{kj}$ if $j \in \mathbf{pa}_G(k)$ and $\widetilde{f_{kj}} = 0$ if $j \notin \mathbf{pa}_G(k)$. Consequently any CAM characterized by $f_1, \ldots, f_p$, G, $\sigma_1, \ldots, \sigma_p$ can be re-parameterized by $f_1, \ldots, f_p$, π, $\sigma_1, \ldots, \sigma_p$, where π is a topological ordering of G. Note that the latter parametrization is not unique, since π can be chosen arbitrarily from the set of topological orderings of G. However, once the topological ordering π is known, G can be found by identifying the parents of X_k (those j for which $f_{kj} \neq 0$) within $\varpi_\pi(k)$ for any $k = 1, \ldots, p$. This is straightforward using pruning or feature selection methods [Teyssier and Koller, 2005, Shojaie and Michailidis, 2010, Bühlmann et al., 2014]. Thus, we simplify our objective and instead of searching for G, we aim to identify its topological ordering π.

Consider a CAM characterized by $f_1, \ldots, f_p, G, \sigma_1, \ldots, \sigma_p$. It can be re-parameterized by the parameter tuple $\theta = (f_1, \ldots, f_p, \pi, \sigma_1, \ldots, \sigma_p)$. Using Equation (3) it follows that the implied conditional distribution of $X_k | \mathbf{X}_{\varpi_\pi(k)} = \mathbf{x}_{\varpi_\pi(k)}$ is Gaussian with mean $f_k\left(\mathbf{x}_{\varpi_\pi(k)}\right)$ and standard deviation σ_k. The implied density p_θ is given by

$$\log(p_\theta(\mathbf{x})) = \sum_{k=1}^{p} \log\left(\frac{1}{\sigma_k} \phi\left(\frac{x_k - f_k(\mathbf{x}_{\varpi_\pi(k)})}{\sigma_k}\right)\right),$$

where ϕ is the density function of a univariate standard normal distribution. From now on let $\mathbf{X}$ follow a CAM characterized by $\theta^0 = \left(f_1^0, \ldots, f_p^0, \pi^0, \sigma_1^0, \ldots, \sigma_p^0\right)$. To identify θ^0 we define the population score function

$$\theta \mapsto \mathbb{E}_{p_{\theta^0}}\left[-\log\left(p_\theta(x)\right)\right].$$

It holds

$$\mathbb{E}_{p_{\theta^0}}\left[-\log\left(p_{\theta^0}(x)\right)\right] \leq \mathbb{E}_{p_{\theta^0}}\left[-\log\left(p_{\theta}(x)\right)\right],$$

and equality holds if and only if $p_{\theta^0} = p_\theta$ by the properties of the Kullback-Leibler divergence. For these minimal θ, their ordering π must be in $\Pi(G^0)$ by the identifiability.

Let us consider the problem of minimizing the score function with respect to θ. Fixing π and $f_1, \ldots, f_p$ in θ and minimzing with respect $\sigma_1, \ldots, \sigma_p$ leads to the minimizers

$$\sigma^2_{k,p_{\theta^0},f_k,\pi} := \mathbb{E}_{p_{\theta^0}}\left[\left(X_k - f_k(\mathbf{x}_{\varpi_\pi(k)})\right)^2\right] = \mathbb{E}_{p_{\theta^0}}\left[\left(X_k - \sum_{j\in\varpi_\pi(k)} f_{kj}(X_j)\right)^2\right]$$

for $k = 1, \ldots, p$. Hence, when minimizing the score function we only need to consider the subset of the parameter space $(f_1, \ldots, f_p, \pi, \sigma_1, \ldots, \sigma_p)$, where $\sigma_k = \sigma_{k,p_{\theta^0},f_k,\pi}, k = 1, \ldots, p$, that is, the relevant parameter space reduces to $(f_1, \ldots, f_p, \pi)$. Thus, it holds

$$\begin{aligned}
\arg\min_\theta \mathbb{E}_{p_{\theta^0}}\left[-\log\left(p_\theta(x)\right)\right] &= \underset{\theta=(f_1,\ldots,f_p,\pi,\sigma_1=\sigma_{1,p_{\theta^0},f_1,\pi},\ldots,\sigma_p=\sigma_{p,p_{\theta^0},f_p,\pi})}{\arg\min} \mathbb{E}_{p_{\theta^0}}\left[-\log\left(p_\theta(x)\right)\right] \\
&= \underset{(f_1,\ldots,f_p,\pi)}{\arg\min} \sum_{k=1}^p \log(\sigma^2_{k,p_{\theta^0},f_k,\pi}) + C \\
&= \underset{(f_1,\ldots,f_p,\pi)}{\arg\min} \sum_{k=1}^p \log(\sigma^2_{k,p_{\theta^0},f_k,\pi}),
\end{aligned}$$

with C only depending on p. Denote the functions aligning with π by

$$\vartheta(\pi) = \left\{ (f_1, \ldots, f_p) : f_k = \sum_{\pi(j)<\pi(k)} f_{kj}, f_{kj} : \mathbb{R} \to \mathbb{R}, f_{kj} \text{ is } \begin{array}{l} \text{three times differentiable,} \\ \text{non-linear, and} \\ \text{non-constant.} \end{array} \right\}.$$

We fix π and optimize with respect to $f_1, \ldots, f_p$ to define a population score on the orderings

$$S(\pi) = \min_{(f_1,\ldots,f_p)\in\vartheta(\pi)} \sum_{k=1}^p \log(\sigma^2_{k,p_{\theta^0},f_k,\pi}). \tag{4}$$

By identifiability, $S(\pi)$ is minimal if and only if $\pi^0 \in \Pi(G^0)$, that is

$$\sum_{k=1}^p \log\left(\left(\sigma^0_k\right)^2\right) = S(\pi^0) < S(\pi)\, \forall \pi^0 \in \Pi(G^0), \pi \notin \Pi(G^0). \tag{5}$$

Intuitively, the score $S(\pi)$ measures how much variance remains when any X_k is regressed on using its predecessors $\mathbf{X}_{\varpi_\pi(k)}$. An example for $p = 2$ is depicted in Figure 2.

2.2 Estimation of the Ordering

In practice we are unaware of the true parameter tuple θ^0 but observe N realizations $\mathbf{x}^N := (\mathbf{x}_1, \ldots, \mathbf{x}_N)$ of $\mathbf{X}$ with density p_{θ^0}, where $\mathbf{x}_\ell \in \mathbb{R}^p, \ell = 1, \ldots, N$ and $\mathbf{x}_\ell = (x_{\ell 1}, \ldots, x_{\ell p})^\top$. It is natural to propose the empirical version of the population score function (4)

$$\widehat{S}(\pi) = \sum_{k=1}^p \log(\widehat{\sigma}^2_{k,\widehat{f}_{k,\pi}}), \tag{6}$$

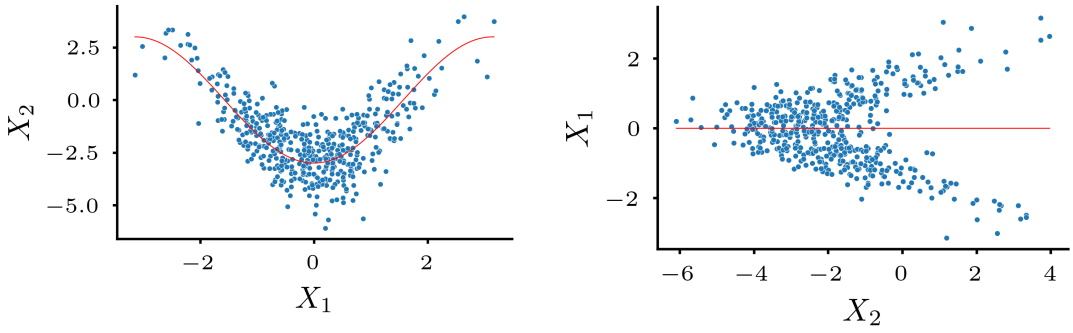

Figure 2: The blue dots represent 500 realizations of a distribution following a SEM with $p = 2$ and $X_1 = \varepsilon_1 \sim \mathcal{N}(0,1)$ and $X_2 = -3\cos(X_1) + \varepsilon_2$ with $\varepsilon_2 \sim \mathcal{N}(0,1)$. On the left-hand-side we plot X_2 on the y-axis and X_1 on the x-axis, while on the right-hand-side it is vice versa. The red lines give the conditional mean functions. We see that $\arg\min_{(f_1,f_2)\in\vartheta((1,2))} \sum_{k=1}^{2} \log(\sigma^2_{k,p_{\theta^0},f_k,(1,2)}) = (0, -3\cos(x_1))$ and $\arg\min_{(f_1,f_2)\in\vartheta((2,1))} \sum_{k=1}^{2} \log(\sigma^2_{k,p_{\theta^0},f_k,(2,1)}) = (0,0)$. The distribution $X_1 - \mathbf{E}[X_1|X_2 = x_2]$ becomes bi-modal for larger values of x_2. The unexplained noise (distance of blue dots to red line) is smaller on the left, which is the correct ordering, thus $S((1,2)) < S((2,1))$.

where

$$\widehat{\sigma}^2_{k,\widehat{f}_{k,\pi}} = \frac{1}{N}\sum_{\ell=1}^{N}\left(x_{\ell k} - \widehat{f}_{k,\pi}(\mathbf{x}_{\ell\varpi_\pi(k)})\right)^2 = \frac{1}{N}\sum_{\ell=1}^{N}\left(x_{\ell k} - \sum_{j\in\varpi_\pi(k)} \widehat{f}_{kj,\pi}(x_{\ell j})\right)^2.$$

Here $\widehat{f}_{k,\pi} = \sum_{j\in\varpi_\pi(k)} \widehat{f}_{kj,\pi}$ is a regression function estimate using data $(\mathbf{x}_{\ell\varpi_\pi(k)}, x_{\ell k}), \ell = 1,\dots,N$ with the convention $\mathbf{x}_{\ell S} := \mathbf{x}_{\ell_S} = (\mathbf{x}_{\ell s_1},\dots,\mathbf{x}_{\ell s_T})$ for $S = \{s_1,\dots,s_T\}$ introduced before. Although the regression estimates depend on the data and thus N, we omit the additional index for better readability.

Remark 1. *In contrast to the population version* (4)*, it is unclear whether* (6) *is also minimized at* $\pi^0 \in \Pi(G^0)$ *even for* $N \to \infty$*. This is due to the fact, that the regression functions* $\widehat{f}_{k,\pi}$ *and the prediction errors* $\widehat{\sigma}^2_{k,\widehat{f}_{k,\pi}}$ *are estimated from a finite sample.*

Since any regression function estimator leads to a score function $\widehat{S}$, the following remark gives some intuition on necessary properties through two "extreme" examples.

Remark 2.

1. *Let the regression estimator interpolate the data, that is,* $\widehat{f}_{k,\pi}(x_{\ell,\varpi_\pi(k)}) = x_{\ell k}$ *for all* $\ell = 1,\dots,N$*,* $k = 1,\dots,p$ *and all* π*. Then the regression estimator is overfitting. In that case any permutation* π *has a score diverging to* $-\infty$*.*

2. *Let* $\widehat{f}_{k,\pi}(x_{\ell,\varpi_\pi(k)}) = 0$ *for all* $\ell = 1,\dots,N$*,* $k = 1,\dots,p$ *and all* π*. Then the regression estimator is underfitting. Again, any ordering* π *has the same score, which is* $\sum_{k=1}^{p} \log\left(\frac{1}{N}\sum_{\ell=1}^{N} x^2_{\ell k}\right)$*.*

In both cases, $\widehat{S}$ *cannot identify an optimal ordering* $\pi^0 \in \Pi(G^0)$ *even with infinite data. Intuitively, we need a regression estimator, that*

1. *is not overfitting and preserves the non-explainable noise, and*

2. *provides estimates that are close to the true regression functions* $f_1^0, \ldots, f_p^0$.

In this work, we apply L^2-boosting regression in conjunction with early stopping [Bühlmann and Yu, 2003, Raskutti et al., 2014] and show that the resulting score of Equation (6) prefers consistently a $\pi^0 \in \Pi(G^0)$. Proposition 1 below formalizes the intuition of Remark 2 and states necessary conditions on the regression function estimator. In the following Definition 1, Y takes the role of X_k, while $\widetilde{\mathbf{X}}$ takes the role of $\mathbf{X}_{\varpi_\pi(k)}$.

Definition 1 (Non-overfitting)**.** *Let* $\widehat{f}$ *be a regression estimate based on* N *i.i.d. samples* $(\widetilde{\mathbf{x}}_1, y_1), \ldots, (\widetilde{\mathbf{x}}_N, y_N)$ *from* $(\widetilde{\mathbf{X}}, Y)$*. We say the estimator is not overfitting w.r.t.* $(\widetilde{\mathbf{X}}, Y)$*, if*

$$\left| \frac{1}{N} \sum_{\ell=1}^{N} \left(y_\ell - \widehat{f}(\widetilde{\mathbf{x}}_\ell) \right)^2 - \mathbb{E}_{\widetilde{\mathbf{X}}, Y} \left[\left(Y - \widehat{f}(\widetilde{\mathbf{X}}) \right)^2 \right] \right|$$

converges to 0 *in probability for* $N \to \infty$.

Proposition 1. *Let Assumption 1 hold. Then, if the regression estimator is such that*

1. $\widehat{f}_{k,S}$ *is not overfitting with respect to* $(\mathbf{X}_S, X_k)$ *for any combination of* $k = 1, \ldots, p$ *and* $S \subset \{1, \ldots, p\} \setminus \{k\}$ *according to Definition 1 and*

2. $\widehat{\sigma}^2_{k,\widehat{f}_{k,\pi^0}} \xrightarrow{\mathbb{P}} \sigma^2_{k,p_{\theta^0},f_k^0,\pi^0} = \left(\sigma_k^0\right)^2$ *for all* $\pi^0 \in \Pi(G^0)$ *and* $k = 1, \ldots, p$*, that is*

$$\frac{1}{N} \sum_{\ell=1}^{N} \left(x_{\ell k} - \widehat{f}_{k,\varpi_{\pi^0}(k)}(\mathbf{x}_{\ell \varpi_{\pi^0}(k)}) \right)^2 \xrightarrow{\mathbb{P}} \left(\sigma_k^0\right)^2 = \mathbb{E}\left[\left(X_k - \sum_{j \in \mathbf{pa}_{G^0}(k)} f_{kj}^0(X_j) \right)^2 \right].$$

Then it holds for the derived score function $\widehat{S}$ *that*

$$\widehat{S}(\pi^0) < \widehat{S}(\pi)$$

for any $\pi^0 \in \Pi(G^0)$ *and* $\pi \notin \Pi(G^0)$ *with probability going to* 1 *for* $N \to \infty$.

Sketch of the proof of Proposition 1 Our goal is to show that for any $\pi \notin \Pi(G^0)$ and $\pi^0 \in \Pi(G^0)$ it holds asymptotically

$$\sum_{k=1}^{p} \log(\widehat{\sigma}^2_{k,\widehat{f}_{k,\pi^0}}) = \widehat{S}(\pi^0) < \widehat{S}(\pi) = \sum_{k=1}^{p} \log(\widehat{\sigma}^2_{k,\widehat{f}_{k,\pi}}).$$

By inequality (5) this is fulfilled if for $N \to \infty$ and $\pi \notin \Pi(G^0)$

$$\lim_{N \to \infty} \widehat{S}(\pi) \geq S(\pi) \tag{7}$$

and if at the same time for $\pi^0 \in \Pi(G^0)$ it holds that

$$\lim_{N \to \infty} \widehat{S}(\pi^0) = S(\pi^0). \tag{8}$$

Applying the continuous mapping theorem, Relation (8) is ensured by Condition 2.

Contrariwise for relation (7) when $\pi \notin \Pi(G^0)$, the non-overfitting Condition 1. of Proposition 1 ensures that for any $k = 1, \ldots, p$ and for $N \to \infty$

$$\widehat{\sigma}^2_{k,\widehat{f}_{k,\pi}} = \frac{1}{N}\sum_{\ell=1}^{N}\left(x_{\ell k} - \sum_{j\in\varpi_\pi(k)} \widehat{f}_{kj,\pi}\left(x_{\ell j}\right)\right)^2 \geq \mathbb{E}_{p_{\theta 0}}\left[\left(X_k - \sum_{j\in\varpi_\pi(k)} \widehat{f}_{kj,\pi}\left(X_j\right)\right)^2\right].$$

It thus follows that for $N \to \infty$

$$\begin{aligned}
\widehat{S}(\pi) &= \sum_{k=1}^{p} \log\left(\widehat{\sigma}^2_{k,\widehat{f}_{k,\pi}}\right) = \sum_{k=1}^{p} \log\left(\frac{1}{N}\sum_{\ell=1}^{N}\left(x_{\ell k} - \sum_{j\in\varpi_\pi(k)} \widehat{f}_{kj,\pi}\left(x_{\ell j}\right)\right)^2\right) \\
&\geq \sum_{k=1}^{p} \log\left(\mathbb{E}_{p_{\theta 0}}\left[\left(X_k - \sum_{j\in\varpi_\pi(k)} \widehat{f}_{kj,\pi}\left(X_j\right)\right)^2\right]\right) \\
&\geq \min_{(f_1,\ldots,f_p)\in\vartheta(\pi)} \sum_{k=1}^{p} \log(\sigma^2_{k,p_{\theta 0},f_k,\pi}) = S(\pi).
\end{aligned}$$

A detailed proof can be found in the Appendix A.

3 Background and Preliminaries

As our main result in Theorem 5 relies on boosting Kernel Hilbert space regressions, we briefly introduce necessary concepts and results on boosting (Section 3.1) and Reproducing Kernel Hilbert Spaces (RKHSs) (Section 3.2) next. Further details on the two concepts can be found in Bühlmann and Yu [2003], Schapire and Freund [2012], as well as Wahba [1990], Schölkopf and Smola [2001], Wainwright [2019], respectively.

3.1 Boosting

L^2-boosting addresses the problem of finding a function f in some function space H that minimizes the expected L^2-loss

$$\frac{1}{2}\mathbb{E}_{\mathbf{X},Y}\left[\left(Y - f(\mathbf{X})\right)^2\right]. \tag{9}$$

In practice, (9) is replaced by the empirical minimizer of

$$\frac{1}{2N}\sum_{\ell=1}^{N}\left(y_\ell - f(\mathbf{x}_\ell)\right)^2 \tag{10}$$

based on N i.i.d. samples $(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_N, y_N)$. L^2-boosting employs a functional gradient descent approach using a base learner S that maps $y^N := (y_1, \ldots, y_N)^\top$ to the estimates $\widehat{f}$ and $\widehat{y}^N = \widehat{f}(\mathbf{x}^N)$, where $\widehat{f}(\mathbf{x}^N) := \left(\widehat{f}(\mathbf{x}_1), \ldots, \widehat{f}(\mathbf{x}_N)\right)^\top$. More precisely, after initializing $\widehat{f}^{(0)}$, in each boosting step $m = 1, \ldots, m_{\text{stop}}$ the residuals at the current $\widehat{f}^{(m)}$

$$u_\ell = \frac{\partial}{\partial f}\frac{1}{2N}\left(y_\ell - f(x_\ell)\right)^2 |_{f=\widehat{f}^{(m)}} = \frac{1}{N}\left(y_\ell - \widehat{f}^{(m)}(x_\ell)\right)$$

are computed and $\widehat{f} = S(u) = S(u_1, \ldots, u_N)$ is determined. The solution is then used to update the estimate of the regression function

$$\widehat{f}^{(m+1)} = \widehat{f}^{(m)} + \upsilon \widehat{f},$$

where $0 < \upsilon \leq 1$ is the step size commonly fixed at a small value [Bühlmann and Yu, 2003]. For many base learners S this leads, for fixed N to an overfitting if $m_{\text{stop}} \to \infty$. Stopping earlier is thus desired and *early stopping* often applied [Schapire and Freund, 2012]. Following Bühlmann and Yu [2003], Raskutti et al. [2014], we consider linear and symmetric base learners S, that is, $S : y^N \mapsto \widehat{y}^N$ is a linear and symmetric mapping. Spline regression and linear regression, even in the generalized ridge regression sense, fall under this definition. So does the (additive) kernel ridge regression [Raskutti et al., 2014, Kandasamy and Yu, 2016], which we will consider in the following. In contrast, the popular choice of decision trees is not linear. Proposition 1 of Bühlmann and Yu [2003] shows that the estimate $\widehat{\mathbf{y}}$ after m boosting steps for $\mathbf{y}$ is given by

$$\widehat{f}^{(m)}(\mathbf{x}^N) = \widehat{\mathbf{y}}^N = B^{(m)}\mathbf{y} = (I - (I - S)^m)\mathbf{y}.$$

As we assume S to be symmetric, there exists an orthogonal $U \in \mathbb{R}^{N \times N}$ containing the eigenvectors of S, such that for the diagonal matrix D with the eigenvalues of S on the diagonal, we have $S = UDU^T$. It follows that $B^{(m)} = U(I - (I - D)^m)U^T$.

3.2 Reproducing Kernel Hilbert Spaces

We choose the function estimates $\widehat{f}$ from a Reproducing Kernel Hilbert Space (RKHS) H, while S is a kernel regression estimator. We start by introducing kernel functions.

Definition 2. *We call a symmetric function $K : \mathbb{R}^p \times \mathbb{R}^p \to \mathbb{R}$ a positive definite (p.d.) kernel on $\mathbb{R}^p$ if*

$$\sum_{k=1}^{N}\sum_{\ell=1}^{N} \alpha_k \alpha_\ell K(\mathbf{x}_k, \mathbf{x}_\ell) \geq 0$$

for any $\{\alpha_1, \ldots, \alpha_N\} \subset \mathbb{R}$ and any $\{\mathbf{x}_1, \ldots, \mathbf{x}_N\} \subset \mathbb{R}^p$.

For any p.d. kernel K, there exists a unique Hilbert space H with $K(\cdot, \mathbf{x}) \in H \, \forall \mathbf{x} \in \mathbb{R}^p$ and where it holds for any $f \in H$ and $\mathbf{x} \in \mathbb{R}^p$

$$f(\mathbf{x}) = < f, K(\cdot, \mathbf{x}) >_H . \tag{11}$$

Equation (11) is called the reproducing property. Consequently H is called an RKHS. By the reproducing property, it holds for $f = \frac{1}{\sqrt{N}} \sum_{k=1}^{N} \alpha_k K(\cdot, \mathbf{x}_k)$ and $g = \frac{1}{\sqrt{N}} \sum_{k=1}^{N} \beta_k K(\cdot, \mathbf{x}_k)$, that

$$< f, g >_H = \alpha^T G \beta, \tag{12}$$

where $G \in \mathbb{R}^{N \times N}$ is symmetric with $G_{jk} = \frac{K(\mathbf{x}_j, \mathbf{x}_k)}{N}$. We call G the Gram matrix. By the representation theorem [Wainwright, 2019, Proposition 12.33] the minimizer of

$$\widehat{f} = \underset{f \in H}{\arg\min} \frac{1}{N} \sum_{\ell=1}^{N} (y_\ell - f(\mathbf{x}_\ell))^2 + \gamma ||f||_H^2 \tag{13}$$

can be expressed by

$$\widehat{f} = \frac{1}{\sqrt{N}} \sum_{\ell=1}^{N} \beta_\ell K(\cdot, \mathbf{x}_\ell),$$

where $\beta = \frac{1}{\sqrt{N}}(G + \gamma N I)^{-1} y^N$. By Equation (12), $||\widehat{f}||_H^2 = \beta^T G \beta$ holds. Clearly, the mapping $S : y^N \mapsto G(G + \lambda I)^{-1} y^N = \widehat{f}(\mathbf{x}^N)$ is linear and symmetric. It can be derived that S has the eigenvalues $d_\ell = \frac{\widehat{\mu_\ell}}{\widehat{\mu_\ell} + \gamma N}$, where $\widehat{\mu_1}, \ldots, \widehat{\mu_N}$ are the eigenvalues of G. The regularization parameter $\lambda = \gamma N$ shall be constant in N in this work.

The boosting estimate $\widehat{f}^{(m)}$ is built sequentially by adding small amounts of current estimates. These current estimates are of the form $\frac{1}{\sqrt{N}} \sum_{\ell=1}^{N} \widehat{\alpha}_\ell K(\cdot, \mathbf{x}_\ell)$. Thus, if S is the base learner used for boosting, it holds by the construction of the boosting estimator, that there exists a $\widehat{\beta} \in \mathbb{R}^N$ with

$$\widehat{f}^{(m)} = \frac{1}{\sqrt{N}} \sum_{\ell=1}^{N} \widehat{\beta}_\ell K(\cdot, \mathbf{x}_\ell). \tag{14}$$

Here, $\widehat{f}^{(m)}$ is the boosting regression estimate after m boosting steps. In this context, we define

$$\mathcal{F}_N := \left\{ f \in H : f = \frac{1}{\sqrt{N}} \sum_{\ell=1}^{N} \beta_\ell K(\cdot, \mathbf{x}_\ell), ||f||_H = 1, \{\mathbf{x}_1, \ldots, \mathbf{x}_N\} \subset \mathbb{R}^p \right\}$$

and $\widehat{f}^{(m)} \in h\mathcal{F}_N$ for some $h > 0$. For a continuous kernel $K \in L^2(\mathbb{R} \times \mathbb{R})$, we can define an integral operator $\mathcal{K} : L^2(\mathbb{R}) \to L^2(\mathbb{R})$ by

$$f(\cdot) \mapsto (\mathcal{K}(f))(y) = \int_{\mathbb{R}} K(x, y) f(x) d\mathbb{P}(x).$$

Under assumptions on $\mathbb{P}$ and K, Sun [2005] shows that the operator $\mathcal{K}$ has eigenvalues $\mu_k \geq 0$ and eigenvectors $\phi_k \in L^2(\mathbb{R})$, so that $\mathcal{K}(\phi_k) = \mu_k \phi_k$. If the eigenvalues are ordered non-increasingly, then μ_k goes to 0. The decay rate of the eigenvalues will be important in our analysis. We close this subsection with two examples.

Example 1 (Kernel functions)**.**

1. ***Gaussian kernels on*** $\mathbb{R}$*:* $K : \mathbb{R} \times \mathbb{R}$ *is defined for some* $\varsigma > 0$ *by*

$$K(x, x') = \exp\left(\frac{|x - x'|^2}{2\varsigma} \right).$$

Its eigenvalues exist under mild assumptions on $P(\mathbf{X})$ *[Sun, 2005, Section 4] and follow an exponential decay of the form*

$$\mu_k \leq \exp(-Ck)$$

for some $C > 0$*. For further details, see Section C of Bach and Jordan [2002] and Example 3 of Cucker and Smale [2002].*

2. ***Additive Kernel***: *Let $H_1, \ldots, H_p$ be RKHSs with kernels K_k on X_k, $k = 1, \ldots, p$. The space $H_1 \oplus \cdots \oplus H_p := \{f : \mathbb{R}^p \to \mathbb{R} : f(x_1, \ldots, x_p) = \sum_{k=1}^p f_k(x_k), f_k \in H_k\}$ is a RKHS with kernel $K = \sum_{k=1}^p K_k$. Its norm is defined by $||f||_H^2 = \sum_{k=1}^p ||f_k||_{H_k}^2$ [see Wainwright, 2019, Proposition 12.27]. For Gaussian kernels $K_1, \ldots, K_p$, we assume that the eigenvalues of K can be upper bounded by*

$$\mu_k \leq p \exp(-Ck) \tag{15}$$

for some $C > 0$. Note that for p fixed, this is of type

$$\mu_k \leq \exp(-C'k)$$

for some $C' > 0$. The solution of (13) can then be written by $\widehat{f}(x_1, \ldots, x_p) = \sum_{k=1}^p \widehat{f}_k(x_k)$, where

$$\widehat{f}_k = \sum_{\ell=1}^N \widehat{\beta}_\ell K_k(\cdot, x_{\ell k})$$

and $\widehat{\beta}$ is shared among the different components, that is, does not depend on k [see Kandasamy and Yu, 2016, for further details].

The idea behind inequality (15) is the following. Each K_k is a self-adjoint and compact operator for any $k = 1, \ldots, p$. Let A, B be linear and self-adjoint operators with non-increasing eigenvalues $\lambda_1, \lambda_2, \ldots$ and $\mu_1, \mu_2, \ldots$, respectively. It holds by Zwahlen [1965/66] for the non-increasing eigenvalues $\gamma_1, \gamma_2, \ldots$ of the self-adjoint and linear operator $A + B$ that for any $1 \leq r, s \leq p$

$$\gamma_{r+s} \leq \gamma_{r+s-1} \leq \lambda_r + \mu_s.$$

Now consider the non-increasing eigenvalues μ_ℓ^k of the operator $A_1 + \ldots + A_k$. Let $\lambda_\ell^j, \ell = 1, 2, \ldots$ be the eigenvalues of A_j. Then it holds that

$$\mu_{\ell p}^p \leq \mu_{(p-1)\ell}^{p-1} + \lambda_\ell^p \leq \ldots \leq \lambda_\ell^1 + \ldots + \lambda_\ell^p.$$

Inequality (15) follows under the assumption that $\lambda_\ell^j \leq \exp(-C\ell)$ for $j = 1, \ldots, p$ for some $C > 0$.

The Gaussian (and its additive counterpart) kernel is bounded, which allows to uniformly upper-bound the supremums norm of the unit ball on H.

Remark 3. *If H is a RKHS with kernel K such that $K(\mathbf{x}, \mathbf{x}) \leq B$ for some $B > 0$, then it holds*

$$\sup_{||f||_H \leq 1} ||f||_\infty \leq B < \infty.$$

4 Boosting DAGs

In this section we prove Theorem 5 which is our main result. It states that if we choose the regression procedure in Section 2 as L^2-boosting with early stopping, then the estimator for the topological ordering is consistent. This holds for an uniform and asymptotic number of boosting iterations. We provide the assumptions in Section 4.1 and prove the statement in Section 4.2. In Section 4.3 we propose an adaption of the procedure which is effective in high dimensions.

4.1 Assumptions

Proposition 1 has shown that in order to consistently estimate the causal ordering, we have to avoid overfitting whenever we regress X_k onto $\mathbf{X}_S$ for any $k = 1, \ldots, p$ and $S \subset \{1, \ldots, p\} \setminus \{k\}$. This poses the main challenge in applying Proposition 1. We assume that $X_k - \mathbb{E}[X_k | \mathbf{X_S} = \mathbf{x_S}]$ is sub-Gaussian, which later allows to control the regression estimates.

Definition 3. *We call a random variable ε sub-Gaussian if its Orlicz norm defined by*

$$s(\varepsilon) = \inf \left\{ r \in (0, \infty) : \mathbb{E}\left[\exp\left(\frac{\varepsilon^2}{r^2} \right) \right] \leq 2 \right\}$$

is finite.

Assumption 2. *For any k and $S \subset \{1, \ldots, p\} \setminus \{k\}$ consider the decomposition*

$$X_k = \mu_{k,S}(\mathbf{X}_S) + \varepsilon_{k,S},$$

where

$$\mu_{k,S}(\mathbf{X}_S) = \mathbf{E}[X_k | \mathbf{X}_S] \text{ and } \varepsilon_{k,S} = X_k - \mathbf{E}[X_k | \mathbf{X}_S].$$

We assume that

1. *$\varepsilon_{k,S} | \mathbf{X}_S = \mathbf{x}_S$ is sub-Gaussian with Orlicz norm $s_{k,S}(\mathbf{x}_S)$, $0 < s_{k,S}(\mathbf{x}_S) \leq s_{max}$ for all $\mathbf{x}_S \in \mathbb{R}^{|S|}$, and*
2. *$||\mu_{k,S}||_\infty \leq \mu_{max} < \infty$.*

The constants shall hold uniformly for any $k \in \{1, \ldots, p\}$ and $S \subset \{1, \ldots, p\} \setminus \{k\}$.

We will prove that the regression estimate will lie in the function class $h\mathcal{F}_N$ for some radius $h > 0$. We denote the ball of radius h in H by B_h. In Theorem 5 we use the radius $h > 0$ and the function complexity measures Rademacher complexity and covering numbers to derive Condition 1. of Proposition 1. Both measures quantify the richness of a function class. While the Rademacher complexity of $\mathcal{F}_N$ can be upper-bounded by Theorem 13 given in the Appendix, we also need to upper bound the covering numbers of $\mathcal{F}_N$. The complexity measures for $h\mathcal{F}_N$ can then be upper-bounded using scaling arguments. We thus make the following assumption:

Assumption 3. *For any $k = 1, \ldots, p$, let H_k be a RKHS on X_k and $B_1^k := \{f \in H : ||f||_{H_k} \leq 1\}$. Then it shall hold for any $z > 0$ and $k = 1, \ldots, p$*

$$\int_0^1 \sqrt{\log\left(\mathcal{N}\left(\frac{uz}{2}, B_1^k \right) \right)} du < \infty.$$

Here, $\mathcal{N}(\cdot, B_1^k)$ is the covering number with respect to $||\cdot||_\infty$, which is defined in Section B.2.

Remark 4. *Let $H_S = H_{s_1} \oplus \ldots \oplus H_{s_k}$ and denote the unit ball of H_{s_k} by $B_1^{s_k}$ and the unit ball of H_S by B_1^S. Assume that for any $j = 1, \ldots, k$ it holds that*

$$\int_0^1 \sqrt{\log\left(\mathcal{N}\left(\frac{uz}{2}, B_1^j \right) \right)} du \leq C(z) < \infty$$

for some $0 < C(z) < \infty$. *Further, one can derive that*

$$\mathcal{N}\left(u, B_1^S\right) \leq \prod_{j=1}^{k} \mathcal{N}\left(\frac{u}{k}, B_1^{s_j}\right).$$

Thus, using Jensen's inequality and that $\mathcal{N}(\cdot, B_1^j)$ *is non-increasing, it holds that*

$$\int_0^1 \sqrt{\log\left(\mathcal{N}\left(\frac{uz}{2}, B_1^S\right)\right)} du \leq \int_0^1 \sqrt{\sum_{j=1}^{k} \log\left(\mathcal{N}\left(\frac{uz}{2k}, B_1^{s_j}\right)\right)} du \leq \sqrt{p} C\left(\frac{z}{2p}\right) < \infty.$$

We assume that the eigenvalues of the random Gram matrix G vanish with the same rate as the eigenvalues of the operator $\mathcal{K}$. For details on the connection between the eigenvalues of G and $\mathcal{K}$, consult Section C of Bach and Jordan [2002].

Assumption 4. *For* $S = \{s_1, \ldots, s_k\} \subset \{1, \ldots, p\}$ *let* $H = H_{s_1} \oplus \ldots \oplus H_{s_k}$ *be the RKHS on* $\mathbf{X}_S$ *generated by the additive kernel* $K_{s_1} + \ldots + K_{s_k}$. *There shall exist events* $\mathcal{B}_N$ *with* $\lim_{N\to\infty} \mathbb{P}(\mathcal{B}_N) = 1$ *so that on* $\mathcal{B}_N$ *and for* $K_0 = \lfloor \frac{1}{2C_d+1} \ln(N) \rfloor$ *it holds for the empirical eigenvalues of the Gram matrix* G

$$\sum_{k=K_0}^{N} \widehat{\mu_k} \leq \sum_{k=K_0}^{N} \exp(-C_u k),$$

and additionally

$$\widehat{\mu}_{K_0} \geq \exp(-C_d K_0)$$

for some $C_d > C_u > 0$. *The constants hold uniformly for any* $S \subset \{1, \ldots, p\}$.

To give some intuition, recall that for the kernel regression estimator $S : y^N \mapsto \widehat{y}^N$ it holds that $S = UDU^\top$, where U contains the eigenvectors of G and D is a diagonal matrix with entries $d_\ell = \frac{\widehat{\mu}_\ell}{\lambda + \widehat{\mu}_\ell}$. Clearly, d_ℓ has a similar decay rate as $\widehat{\mu}_\ell$. For simplicity, consider $w^N = U^\top y^N$. Assumption 4 then ensures that only few entries of w^N largely influence $\widehat{y}^N = UDw^N$, while most entries of w^N contribute little. Thus, $\widehat{y}^N$ is mostly influenced by a small subspace of $\mathbb{R}^N$.

4.2 Main Theorem

We state now the main theorem.

Theorem 5. *Let* $H_{k'}$ *be a RKHS on* $X_{k'}$ *with Gaussian kernel* $K_{k'}$ *for any* $k' = 1, \ldots, p$. *Assume that* $\mathbf{X} = (X_1, \ldots, X_p)$ *follows a CAM as in Assumption 1 for functions* $f_1^0 \in H_1, \ldots, f_p^0 \in H_p$, *where* $H_k = H_{k_1} \oplus \ldots \oplus H_{k_q}$, $\{k_1, \ldots, k_q\} = \mathbf{pa}(k)$. *Given Assumptions 2, 3, and 4 and assuming we estimate* $\widehat{f}_{k,\pi}^{(m_{stop})} = \widehat{f}_{k,\pi} = \sum_{j \in \varpi_\pi(k)} \widehat{f}_{kj,\pi}$ *using* L^2*-boosting with* X_k *as response,* $\mathbf{X}_{\varpi_\pi(k)}$ *as predictors and with number of boosting steps chosen as* $m_{stop} = N^{\frac{1}{4}\frac{C_u + C_d + 1/2}{C_d + 1}}$, *then it holds that*

$$\widehat{S}(\pi^0) < \widehat{S}(\pi)$$

for $N \to \infty$ *with probability going to* 1 *for any* $\pi^0 \in \Pi^0$ *and* $\pi \notin \Pi^0$.

Remark 5. *Theorem 2 of Minh [2010] ensures that $f_1^0, \ldots, f_p^0$ are smooth, non-constant, and non-linear and thus meet Assumption 1.*

Proof. We apply Proposition 1 and show the following two conditions.

1. *Boosting under Misspecification:* $\widehat{f}_{k,S}^{(m_{stop})}$ is not overfitting with respect to $(\mathbf{X}_S, X_k)$ for any $k \in \{1, \ldots, p\}$ and $S \subset \{1, \ldots, p\} \setminus \{k\}$, and
2. *Consistency of Variance Estimation:* For any $k = 1, \ldots, p$ and $\pi^0 \in \Pi^0$ it holds

$$\left| \frac{1}{N} \sum_{\ell=1}^{N} \left(\mathbf{x}_{\ell k} - \widehat{f}^{(m_{stop})}(\mathbf{x}_{\ell \varpi_{\pi^0}(k)} \right)^2 - \mathbb{E}\left[\left(X_k - f_k^0(\mathbf{X}_{\mathbf{pa}_{G^0}(k)} \right)^2 \right] \right| \to 0$$

in probability.

We will see that 1. follows from Theorem 6 in Section 4.2.1 and 2. is shown in Theorem 7 in Section 4.2.2. □

4.2.1 Boosting under Misspecification

We now show Condition 1. of Theorem 5. We fix $S = \{s_1, \ldots, s_d\}$ and k and define $\left(\widetilde{X}_1, \ldots, \widetilde{X}_d\right) = \widetilde{\mathbf{X}} = \mathbf{X}_S$ and $Y = X_k$. Let $\widetilde{\mathbf{X}}^N$ be the random element containing N i.i.d. observations of $\widetilde{\mathbf{X}}$ and denote the realizations of $\widetilde{\mathbf{X}}^N$ by $\widetilde{\mathbf{x}}^N = (\widetilde{\mathbf{x}}_1, \ldots, \widetilde{\mathbf{x}}_N)$. Analogously we define Y^N and y^N. Our goal is to prove, that

$$\left| \frac{1}{N} \sum_{\ell=1}^{N} \left(\widehat{f}^{(m_{stop})}(\widetilde{\mathbf{x}}_\ell) - y_\ell \right)^2 - \mathbb{E}_{\widetilde{\mathbf{X}},Y} \left[\left(\widehat{f}^{(m_{stop})}(\widetilde{\mathbf{X}}) - Y \right)^2 \right] \right| \xrightarrow{\mathbb{P}} 0 \tag{16}$$

for $N \to \infty$ for the boosting estimate $\widehat{f}^{(m_{stop})}$. The left term is an expectation with respect to the empirical distribution P_N (and thus depends on the realizations), whereas the right term is under the population distribution induced by $\left(\widetilde{\mathbf{X}}, Y\right)$ denoted by P. Thus, the l.h.s. of Equation (16) becomes

$$(P_N - P) \left[\left(\widehat{f}^{(m_{stop})}(\widetilde{\mathbf{X}}) - Y \right)^2 \right]. \tag{17}$$

Assumptions 2′, 3′, and 4′ are reformulated versions of Assumptions 2, 3, and 4 using the notation introduced above.

Assumption 2′. *Let $Y = \mu(\widetilde{\mathbf{X}}) + \varepsilon$, for which we define the random variables*

$$\mu(\widetilde{\mathbf{X}}) = \mathbb{E}\left[Y | \widetilde{\mathbf{X}} \right] \text{ and } \varepsilon = Y - \mathbb{E}\left[Y | \widetilde{\mathbf{X}} \right].$$

We assume that $\varepsilon | \widetilde{\mathbf{X}} = \widetilde{\mathbf{x}}$ is sub-Gaussian with Orlicz norm $s(\widetilde{\mathbf{x}})$ and $0 < s(\widetilde{\mathbf{x}}) \leq s_{max}$ for all $\widetilde{\mathbf{x}} \in \mathbb{R}^d$ and $||\mu||_\infty \leq \mu_{max} < \infty$. Let $\sigma^2_{max} := \max_{\widetilde{\mathbf{x}} \in \mathbb{R}^d} \mathbb{E}\left[\varepsilon^2 | \widetilde{\mathbf{X}} = \widetilde{\mathbf{x}} \right]$.

Assumption 3′. *Let $B_1 := \{f \in \mathcal{F} : ||f||_H \leq 1\}$, where H is the additive RKHS on $\widetilde{\mathbf{X}} = \left(\widetilde{X}_1, \ldots, \widetilde{X}_d\right)$. Then for any $z > 0$ it holds*

$$\int_0^1 \sqrt{\log\left(\mathcal{N}\left(\frac{uz}{2}, B_1 \right) \right)} du < \infty.$$

Assumption 4′. *There exist events $\mathcal{B}_N$ on $\widetilde{\mathbf{X}}^N$ with $\lim_{N\to\infty}\mathbb{P}(\mathcal{B}_N)=1$ so that on $\mathcal{B}_N$ and for $K_0=\lfloor\frac{1}{2C_d+1}\ln(N)\rfloor$ it holds for the empirical eigenvalues of the Gram matrix G for $\widetilde{\mathbf{x}}^N\in\mathcal{B}_N$*

$$\sum_{k=K_0}^{N}\widehat{\mu_k}\leq\mu_k\leq\sum_{k=K_0}^{N}\exp(-C_uk),$$

and additionally

$$\widehat{\mu}_{K_0}\geq\exp(-C_dK_0)$$

for some $C_d>C_u>0$.

Note that all constants in the Assumptions 2′, 3′, 4′ are independent of the choice of k,S. It is the purpose of this subsection to prove the following theorem (and thus Condition 1).

Theorem 6. *Under the Assumptions 2′, 3′ and 4′ it holds for $m_{stop}(N)=N^{\frac{1}{4}\frac{C_u+C_d+1/2}{C_d+1}}$*

$$\left|(P-P_N)\left(Y-\widehat{f}^{(m_{stop})}(\widetilde{\mathbf{X}})\right)^2\right|\xrightarrow{\mathbb{P}}0. \tag{18}$$

As $\widehat{f}^{(m_{stop})}$ depends on the realizations $\widetilde{\mathbf{x}}^N$ and y^N, the convergence above is not trivial. For simplicity we drop the dependency of $f,\widehat{f}^{(m_{stop})}$ on $\widetilde{\mathbf{X}}$ in the proof below.

Proof. We decompose

$$\begin{aligned}&|(P_N-P)\left(Y-\widehat{f}^{(m_{stop})}\right)^2|\\&\leq\underbrace{|(P_N-P)Y^2|}_{\text{I}}+\underbrace{2|(P_N-P)\widehat{f}^{(m_{stop})}Y|}_{\text{II}}+\underbrace{|(P_N-P)\left(\widehat{f}^{(m_{stop})}\right)^2|}_{\text{III}}.\end{aligned}$$

To prove (18) we show the convergence in probability for I − III. Term I goes to 0 in probability since Y has a finite fourth moment. For term II it holds for any $\xi>0$ that

$$\begin{aligned}&\mathbb{P}\left(|(P_N-P)Y\widehat{f}^{(m_{stop})}|\geq\xi\right)\\&=\mathbb{P}\left(\left(|(P_N-P)Y\widehat{f}^{(m_{stop})}|\geq\xi\right)\cap\left(||\widehat{f}^{(m_{stop})}||_H\in h(N)\mathcal{F}_N\right)\right)\\&\quad+\mathbb{P}\left(\left(|(P_N-P)Y\widehat{f}^{(m_{stop})}|\geq\xi\right)\cap\left(||\widehat{f}^{(m_{stop})}||_H\notin h(N)\mathcal{F}_N\right)\cap\left\{\widetilde{\mathbf{X}}^N\in\mathcal{B}_N\right\}\right)\\&\quad+\mathbb{P}\left(\left(|(P_N-P)Y\widehat{f}^{(m_{stop})}|\geq\xi\right)\cap\left(||\widehat{f}^{(m_{stop})}||_H\notin h(N)\mathcal{F}_N\right)\cap\left\{\widetilde{\mathbf{X}}^N\notin\mathcal{B}_N\right\}\right)\\&\leq\mathbb{P}\left(\left(|(P_N-P)Y\widehat{f}^{(m_{stop})}|\geq\xi\right)\cap\left(||\widehat{f}^{(m_{stop})}||_H\in h(N)\mathcal{F}_N\right)\right)\\&\quad+\mathbb{P}\left(\left(||\widehat{f}^{(m_{stop})}||_H\notin h(N)\mathcal{F}_N\right)\cap\left\{\widetilde{\mathbf{X}}^N\in\mathcal{B}_N\right\}\right)\\&\quad+\mathbb{P}\left(\left\{\widetilde{\mathbf{X}}^N\notin\mathcal{B}_N\right\}\right)\\&\leq\mathbb{P}\left(\sup_{f\in h(N)\mathcal{F}_N}|(P_N-P)Yf|\geq\xi\right)\\&\quad+\mathbb{P}\left(\left(||\widehat{f}^{(m_{stop})}||_H\notin h(N)\mathcal{F}_N\right)\cap\left\{\widetilde{\mathbf{X}}^N\in\mathcal{B}_N\right\}\right)\\&\quad+\mathbb{P}\left(\left\{\widetilde{\mathbf{X}}^N\notin\mathcal{B}_N\right\}\right).\end{aligned}$$

with probability going to 1 for $N \to \infty$. For $h(N) \in o\left(N^{1/4}\right)$, the first line on the r.h.s. goes to 0 by Corollary 1 and the second line vanishes by Lemma 5. The third line converges to 0 due to Assumption 4′. This shows the convergence in probability of term II.
Similarly, for term III it holds for any $\xi > 0$ that

$$\begin{aligned}
&\mathbb{P}\left(|(P_N - P)\left(\widehat{f}^{(m_{stop})}\right)^2| \geq \xi\right) \\
&= \mathbb{P}\left(\left(|(P_N - P)\left(\widehat{f}^{(m_{stop})}\right)^2| \geq \xi\right) \cap \left(||\widehat{f}^{(m_{stop})}||_H \in h(N)\mathcal{F}_N\right)\right) \\
&\quad + \mathbb{P}\left(\left(|(P_N - P)\left(\widehat{f}^{(m_{stop})}\right)^2| \geq \xi\right) \cap \left(||\widehat{f}^{(m_{stop})}||_H \notin h(N)\mathcal{F}_N\right) \cap \{\widetilde{\mathbf{X}}^N \in \mathcal{B}_N\}\right) \\
&\quad + \mathbb{P}\left(\left(|(P_N - P)\left(\widehat{f}^{(m_{stop})}\right)^2| \geq \xi\right) \cap \left(||\widehat{f}^{(m_{stop})}||_H \notin h(N)\mathcal{F}_N\right) \cap \{\widetilde{\mathbf{X}}^N \notin \mathcal{B}_N\}\right) \\
&\leq \mathbb{P}\left(\sup_{f \in h(N)\mathcal{F}_N} |(P_N - P)f^2| \geq \xi\right) \\
&\quad + \mathbb{P}\left(\left(||\widehat{f}^{(m_{stop})}||_H \notin h(N)\mathcal{F}_N\right) \cap \{\widetilde{\mathbf{X}}^N \in \mathcal{B}_N\}\right) \\
&\quad + \mathbb{P}\left(\{\widetilde{\mathbf{X}}^N \notin \mathcal{B}_N\}\right).
\end{aligned}$$

For $h(N) \in o\left(N^{1/4}\right)$, the first line on the r.h.s. converges to 0 due to Corollary 2 and the other terms behave as described for term II. This shows the convergence in probability for term III. Overall, this proves Condition 1. □

4.2.2 Consistency of Variance Estimation

In this paragraph we prove Condition 2. in the proof of Theorem 5. We fix again k and assume that $\varpi_{\pi^0}(k)$ has size d, that is, $\pi^0(k) = d+1$. We set $\widetilde{\mathbf{X}} = \left(\widetilde{X}_1, \ldots, \widetilde{X}_d\right) = \mathbf{X}_{\varpi_{\pi^0}(k)}$ and $Y = X_k$.

Theorem 7. *Let*

$$Y = f^0(\widetilde{\mathbf{X}}) + \varepsilon, \quad \varepsilon \sim \mathcal{N}(0, \sigma^2),$$

where f^0 lies in a RKHS H for which Assumption 4′ holds with $||f^0||_H = R$. Then, it holds with the number of boosting steps chosen as $m_{stop} = m(N) = N^{\frac{1}{4}\frac{C_u + C_d + 1/2}{C_d + 1}}$ for $N \to \infty$

$$\left|\frac{1}{N}\sum_{\ell=1}^{N}(y_\ell - \widehat{f}^{(m_{stop})}(\widetilde{\mathbf{x}}_\ell))^2 - \mathbb{E}\left[\left(Y - f^0(\widetilde{\mathbf{X}})\right)^2\right]\right| = \left|\widehat{\sigma}^2 - \sigma^2\right| \to 0$$

in probability.

Proof. We define the semi-norm

$$||g||_{2,N}^2 := \frac{1}{N}\sum_{\ell=1}^{N} g(y_\ell, \widetilde{\mathbf{x}}_\ell)^2.$$

By the triangle inequality it holds that

$$||Y - \widehat{f}^{(m_{stop})}||_{2,N} \leq ||Y - f^0||_{2,N} + ||f^0 - \widehat{f}^{(m_{stop})}||_{2,N}. \tag{19}$$

Next, we show how to asymptotically lower and upper bound $||Y - \widehat{f}^{(m_{stop})}||_{2,N}$ by σ.

Lower bound: By Remark 3 and as $||f^0||_H = R$, $||f^0||_\infty$ is bounded. Further, $f^0(\widetilde{\mathbf{x}}) = \mathbb{E}\left[Y|\widetilde{\mathbf{X}} = \widetilde{\mathbf{x}}\right]$ and the noise $Y - f^0(\widetilde{\mathbf{x}}) = \varepsilon$ is Gaussian and thus sub-Gaussian with an Orlicz norm that is uniformly bounded. Hence, by Theorem 6 and the continuous mapping theorem the l.h.s. of (19) converges and it holds

$$||Y - \widehat{f}^{(m_{stop})}||_{2,N} = \sqrt{||Y - \widehat{f}^{(m_{stop})}||_{2,N}^2} \to \sqrt{\mathbb{E}\left[||Y - \widehat{f}^{(m_{stop})}||_2^2\right]} \geq \sqrt{\mathbb{E}\left[||Y - f^0||_2^2\right]} = \sigma$$

in probability.

Upper bound: The term $||Y - f^0||_{2,N} = \sqrt{\frac{1}{N}\sum_{\ell=1}^N \varepsilon_\ell^2}$ converges to σ in probability. Thus, it remains to show that $||f^0 - \widehat{f}^{(m_{stop})}||_{2,N} \xrightarrow{\mathbb{P}} 0$ for $N \to \infty$, which follows from Lemma 1 below. □

Lemma 1. *For $Y = f^0(\widetilde{\mathbf{X}}) + \varepsilon$, $\varepsilon \sim \mathcal{N}(0, \sigma^2)$, $f^0 \in H$, and if $\widehat{f}$ is the boosting estimate with $m_{stop} = m(N) = N^{\frac{1}{4}\frac{C_u + C_d + 1/2}{C_d + 1}}$ boosting steps, then $\widehat{f}^{(m_{stop})}$ converges to f^0 in a fixed design, that is,*

$$||f^0 - \widehat{f}^{(m_{stop})}||_{2,N} = \left(\frac{1}{N}\sum_{\ell=1}^N \left(f^0(\widetilde{\mathbf{x}}_\ell) - \widehat{f}^{(m_{stop})}(\widetilde{\mathbf{x}}_\ell)\right)^2\right)^{1/2} \xrightarrow{\mathbb{P}} 0.$$

Proof. The proof is in the Appendix C. □

Remark 6. *Lemma 1 also holds for heteroscedastic noise and is stated similarly in Bühlmann and Yu [2003], Raskutti et al. [2014].*

Remark 7. *The combination of Theorem 6 and 7 is insightful. If we are unsure whether $f^0 \in H$ and the noise is independent, then limiting appropriately the number of boosting iterations leads to*

1. *a consistent estimator for f^0 if the assumptions hold, and*
2. *an estimator, so that the prediction error on the samples $\frac{1}{N}\sum_{\ell=1}^N \left(y_\ell - \widehat{f}^{(m_{stop})}(\widetilde{\mathbf{x}}_\ell)\right)^2$ is asymptotically close to the L^2-error $\mathbb{E}\left[\left(Y - \widehat{f}^{(m_{stop})}(\widetilde{\mathbf{X}})\right)^2\right]$ for yet unobserved realizations of $(\widetilde{\mathbf{X}}, Y)$. Here, $\frac{1}{N}\sum_{\ell=1}^N \left(y_\ell - \widehat{f}^{(m_{stop})}(\widetilde{\mathbf{x}}_\ell)\right)^2$ depends only on the observations used for learning $\widehat{f}^{(m_{stop})}$ and thus no hold-out set is necessary.*

We emphasize, that although the results are stated for the Gaussian kernel they can be adapted to kernels with other eigenvalue decay rates.

4.3 Boosting DAGs for Large Dimensions

Theorem 5 shows that we can asymptotically identify the true causal ordering using boosting regressions. However, there are $p!$ possible permutations on $\{1, \ldots, p\}$ that constitute the search space. Thus, beyond a very small p or without extensive prior knowledge on the topological order the computational costs are prohibitive. We address this issue through component-wise boosting in an additive noise model.

Additive Noise Model The true graph G^0 and the true structural equations $f_1^0, \ldots, f_p^0$ can be represented by a function $F^0 : \mathbb{R}^p \to \mathbb{R}^p$, where

$$F^0(\mathbf{x}) = \left(f_1^0(\mathbf{x}), \ldots, f_p^0(\mathbf{x})\right)^\top = \left(f_1^0(\mathbf{x}_{\mathbf{pa}(1)}), \ldots, f_p^0(\mathbf{x}_{\mathbf{pa}(p)})\right)^\top.$$

Thus, the graph G^0 has an edge from X_j to X_k if and only the function f_k^0 is not constant in its $j-$th component. Note that the set of functions $F : \mathbb{R}^p \to \mathbb{R}^p$ corresponding to a DAG is non-convex [Zheng et al., 2020]. We assume that the structural equations $f_k^0(\mathbf{x}_{\mathbf{pa}(k)})$ decompose additively, that is $f_k^0(\mathbf{x}_{\mathbf{pa}(k)}) = \sum_{j \in \mathbf{pa}(k)} f_{kj}^0(x_j)$.

Component-wise Boosting Instead of applying L^2-boosting to estimate f_k^0, $k = 1, \ldots, p$ one-by-one as in Theorem 5, we employ component-wise boosting to estimate F^0. This means, we define the loss function on the functions $F : \mathbb{R}^p \to \mathbb{R}^p$ as the log-likelihood function given $\mathbf{x}^N$

$$L(F, \mathbf{x}^N) = L\left((f_1, \ldots, f_p), \mathbf{x}^N\right) = \sum_{k=1}^{p} \log\left(\sum_{\ell=1}^{N} \left(\mathbf{x}_{\ell k} - f_k(\mathbf{x}_\ell)\right)^2\right)$$

and proceed as follows. Choose a step size $0 < \mu \leq 1$ and let $F^{(1)} = 0$ be the starting value. Then, for $m = 1, 2, \ldots$ we set $f_k^{(m+1)}(\mathbf{x}) = f_k^{(m)}(\mathbf{x}) + \mu \widehat{f}_{kj}(x_j)$, where $(j, k) \in \{1, \ldots, p\} \times \{1, \ldots, p\}$ is the solution of

$$\underset{(j,k) \notin N^m}{\arg\min}\, S(j, k; F^{(m)}),$$

and $S(j, k; F^{(m)})$ is the score on the edges defined by

$$S(j, k; F^{(m)}) := \log\left(\sum_{\ell=1}^{N} \left(\widehat{f}_{kj}(\mathbf{x}_{\ell j}) - \left(\mathbf{x}_{\ell k} - f_k^{(m)}(\mathbf{x}_\ell)\right)\right)^2\right). \tag{20}$$

Here, the candidate functions $\widehat{f}_{kj}$ are determined by solving the kernel ridge regression

$$\widehat{f}_{kj} = \underset{g_{kj} \in H_j}{\arg\min} \sum_{\ell=1}^{N} \left(g_{kj}(\mathbf{x}_{\ell j}) - \left(\mathbf{x}_{\ell k} - f_k^{(m)}(\mathbf{x}_\ell)\right)\right)^2 + \lambda ||g_{kj}||_{H_j}^2. \tag{21}$$

In the set N^m we track the edges (j, k) that would cause a cycle when added to $F^{(m)}$. Hence, $F^{(m)}$ corresponds to a DAG for any $m = 1, 2, \ldots$.

Note that if the edge (j, k) is chosen, then we only need to update $S(j, k; F^{(m+1)})$ for $(j, k) \notin N^{m+1}$, while this number remains unchanged for $k' \neq k$, that is, $S(j, k'; F^{(m+1)}) = S(j, k'; F^{(m)})$. This reduces the computational burden. We stop the procedure after m_{stop} steps, which is the crucial tuning parameter of (component-wise) boosting.

Choosing m_{stop} Inspired by results from boosting for regression [Tutz and Binder, 2006, Bühlmann and Hothorn, 2007] we use the Akaike Information Criterion (AIC) to select m_{stop}. For any $f_k^{(m)}$ we calculate the trace of the mapping $B_k^{(m)} : (x_{1k}, \ldots, x_{Nk}) \mapsto \left(f_k^{(m)}(\mathbf{x}_1), \ldots, f_k^{(m)}(\mathbf{x}_N)\right)$. Then we define the AIC score

$$AIC(F^{(m)}, \mathbf{x}^N) = \sum_{k=1}^{p} AIC_k(f_k^{(m)}, \mathbf{x}^N) = \sum_{k=1}^{p} \left(\sum_{\ell=1}^{N} \left(x_{\ell k} - f_k^{(m)}(\mathbf{x}_\ell)\right)^2 + tr(B_k^{(m)})\right). \tag{22}$$

We stop the procedure and set $m_{stop} = m$ if $AIC(F^{(m)}, \mathbf{x}^N)$ increases with m. We emphasize that this is merely a local minimum w.r.t to the AIC score and the global optimum is hard to find due to the non-convexity of the search space. The algorithm using the AIC is outlined in Algorithm 1 in Appendix D. It can be understood as a component-wise functional gradient descent in the space of those additive functions $\mathbb{R}^p \to \mathbb{R}^p$ which imply a DAG.

Pruning We prune the estimated graph $\widehat{G}$ by running an additive model regression of every node on its parents. Finally, we keep only those nodes as parents whose p-value is below 0.001. For more details on pruning, see Bühlmann et al. [2014].

5 Simulation Study

In this section we empirically investigate the proposed algorithms. In Section 5.1 we describe the data-generating processes. In Section 5.2 we verify Theorem 5 for data sets of small dimensions. In Section 5.3 we benchmark the algorithm of Section 4.3 on high-dimensional data sets against state-of-the-art methods. Further, we run a sensitivity analysis on the effect of the hyperparameters.

Every presented result is based on 100 randomly generated data sets unless stated otherwise. Aside from the sensitivity analysis, we set the step size $\mu = 0.3$ and the penalty parameter $\lambda = 0.01$. While it is known that boosting is commonly robust with respect to the step size (as long as it is small enough), we find in Section 5.3.2, that our method is robust also against the specific choice of λ. We therefore refrain from further tuning here.

5.1 Sampling SEMs and Data Generation

The generation of the synthetic data sets follows closely Bühlmann et al. [2014], Lachapelle et al. [2019], Zheng et al. [2020], Ng et al. [2022b], that is, they are generated as follows.

1. Generate **underlying graph** G^0 with one of the following two methods.
   (a) Generate a DAG according to the Erdös-Renyi (ER) model [Erdös and Rényi, 1959]. That means, we first generate a random DAG with the maximal number of edges and keep every edge with a constant probability. Every node has the same distribution for the number of its neighbors.
   (b) Generate a scale-free (SF) graph using the model of Barabási and Albert [1999]. There exist hubs of nodes with large degree, while other nodes have a smaller degree. This graph structure is observed in various applications [Jeong et al., 2000, Wille et al., 2004, Kertel et al., 2023].
2. Generate **structural equations** (SEs) in one of the two ways.
   (a) **Additive:** For any edge (j, k) in the graph, sample f^0_{kj} from a Gaussian process with mean 0 and covariance function $cov(f^0(x_{\ell j}), f^0(x_{\ell' j})) = \exp\left(-\frac{(x_{\ell j} - x_{\ell' j})^2}{2}\right)$ and set $f^0_k(\mathbf{x}_{\ell \mathbf{pa}_{G^0}(k)}) = \sum_{j \in \mathbf{pa}_{G^0}(k)} f^0_{kj}(x_{\ell j})$ for $\ell, \ell' = 1, \ldots, N$.

(b) **Non-additive:** For every node k consider its parents $\mathbf{pa}_{G^0}(k)$ and sample f_k^0 from a Gaussian Process with mean 0 and covariance function

$$cov(f^0(x_{\ell \mathbf{pa}_{G^0}(k)}), f^0(x_{\ell' \mathbf{pa}_{G^0}(k)})) = \exp\left(-\frac{||x_{\ell \mathbf{pa}_{G^0}(k)} - x_{\ell' \mathbf{pa}_{G^0}(k)}||_2^2}{2}\right),$$

for $\ell, \ell' = 1, \ldots, N$.

3. Sample the **standard deviation** σ_k^0 for all $k = 1, \ldots, p$ from a uniform distribution on $\left[\sqrt{2}/5, \sqrt{2}\right]$.

Finally, we sample the variables with no incoming edges from a centered Gaussian distribution with standard deviation as chosen in step 3. Following the topological order of G^0, we generate the data set recursively according to the SEM. We emphasize, that non-additive SEs conflict with Assumption 1.

5.2 Low-Dimensional Data

In this low-dimensional simulation study we generate data by setting $p = 5$ and sample from an ER graph with on average five edges.

Then, we calculate the score for any of the $p!$ permutations π as described in Section 2. For every regression we choose m_{stop} to be the minimal m, for which $AIC_k(f_k^{(m)}, \mathbf{x}^N)$ increases with m. Recall that $f_k^{(m)}$ is a function on $\mathbf{x}_{\varpi_\pi(k)}$ and we use regular (not componentwise) boosting. We generate data sets with $N = 10, 20, 50, 100, 200$ observations with either additive or non-additive SEs. To evaluate the quality of the estimated permutations, we use the transposition distance

$$d_{trans}(\pi_1, \pi_2) := \min \left|\{\text{transpositions } \sigma_1, \ldots, \sigma_J : \sigma_1 \circ \ldots \circ \sigma_J \circ \pi_1 = \pi_2\}\right|.$$

For an estimated permutation $\widehat{\pi}$ we then set

$$d_{trans}\left(\widehat{\pi}, \Pi^0\right) := \min_{\pi^0 \in \Pi^0} d_{trans}(\widehat{\pi}, \pi^0).$$

Thus, we calculate the minimal number of adjacent swaps so that the estimated permutation aligns with the underlying topological order. The results are depicted in Table 1 and as

	Additve SEs		Non-additve SEs	
N	$\overline{(d_{trans}(\widehat{\pi}, \Pi^0))}$	$\mathrm{SD}\big(d_{trans}(\widehat{\pi}, \Pi^0)\big)$	$\overline{(d_{trans}(\widehat{\pi}, \Pi^0))}$	$\mathrm{SD}\big(d_{trans}(\widehat{\pi}, \Pi^0)\big)$
10	3.29	1.90	3.26	1.95
20	1.76	1.93	2.03	2.00
50	0.49	1.12	0.85	1.55
100	0.35	0.92	0.56	1.04
200	0.03	0.22	0.23	0.61

Table 1: Mean ($\overline{(d_{trans}(\widehat{\pi}, \Pi^0))}$) and standard deviation (SD) of the transposition distances (d_{trans}) between the estimated permutation $\widehat{\pi}$ and the set of true permutations Π^0 for SEMs with additive and non-additive SEs. The underlying graphs are of ER type with on average five edges.

expected, increasing the sample size decreases the transposition distances. This supports Theorem 5. Further, the convergence seems also to hold under non-additive SEs. This indicates a robustness of the algorithm with respect to non-additve SEs.

5.3 High- Dimensional Data

5.3.1 Comparison with Existing Methods

In the following we compare the method proposed in Section 4.3 (denoted by DAGBoost) with Bühlmann et al. [2014] (denoted by CAM) and Zheng et al. [2020] (denoted by DAGSNOTEARS). For CAM we employ the default configuration and for DAGSNOTEARS we set $\lambda = 0.03$ and cutoff to 0.3. For a comparison with other methods see Bühlmann et al. [2014].

As performance measure we calculate the Structural Hamming Distance (SHD) between the true and the estimated graph for data sets containing $N = 200$ observations of $p = 100$ variables. The mean and standard deviation (SD) of the SHDs are given in Table 2.

		CAM		DAGBoost		DAGSNOTEARS	
Additive	Graph	$\overline{\text{SHD}}$	SD(SHD)	$\overline{\text{SHD}}$	SD(SHD)	$\overline{\text{SHD}}$	SD(SHD)
True	SF	30.18	9.47	30.08	9.57	176.96	23.06
True	ER	12.07	4.24	14.90	7.12	127.64	15.84
False	SF	77.63	7.57	67.47	5.73	135.32	18.20
False	ER	36.40	7.96	37.57	9.15	111.78	13.16

Table 2: Mean ($\overline{\text{SHD}}$) and standard deviation (SD) of SHDs between the true graph and the graphs estimated by the three presented algorithms.

From this table we make the following observations. DAGSNOTEARS does not provide satisfying results, while CAM and DAGBoost perform noticeably better in all simulation scenarios. Compared to DAGBoost, CAM achieves slightly better results for the easiest setting of ER graphs and additive SEs (improvement by 0.5 standard deviations). Generally, all methods suffer from an uneven edge distribution (SF graphs). However, for DAGBoost the mean of the SHDs increases less than CAM when the graph is of type SF instead of ER. Thus, in the most complex scenario of non-additve SEs and a non-even distribution of the edges among the nodes (SF graphs), DAGBoost is more than one standard deviation better than CAM. This is an important insight for real-world applications, which often follow SF graphs.

		CAM		DAGBoost		DAGSNOTEARS	
Additive	Graph	Precision	Recall	Precision	Recall	Precision	Recall
True	SF	0.868	0.817	0.967	0.712	0.165	0.198
True	ER	0.907	0.979	0.933	0.920	0.174	0.154
False	SF	0.676	0.383	0.930	0.345	0.037	0.024
False	ER	0.823	0.788	0.918	0.693	0.127	0.083

Table 3: Mean of precision and recall for the three presented algorithms. Precision is the ratio between the correctly identified edges and all identified edges. Recall is the share of the correctly identified edges among all true edges.

Table 3 furthermore summarizes the mean of the precision and the recall for the three algorithms. The precision, that is, the ratio of the correctly identified edges to all identified edges, is larger for DAGBoost. On the other hand, the recall, which is the share of the

identified edges among all true edges, is larger for CAM. Hence, the edges of DAGBoost are more reliable, while CAM misses a lower number of the underlying relationships.

Overall, DAGBoost is a strong competitor to CAM, both of which outperform DAGSNOTEARS. In particular, DAGBoost tends to estimate more reliable edges and is superior compared to CAM in complex non-additive and SF settings. Last, we make note the advantage of less tuning required for DAGBoost compared to CAM. While CAM has additional tuning parameters for the preliminary neighborhood selection [see Bühlmann et al., 2014], the tuning parameters of DAGBoost are merely the step size μ and the penalty parameter λ which we found to be rather robust (see the sensitivity analysis below), thus relatively easy to tune. Pruning the resulting graph has a positive effect on the SHD between the estimated and the true graph for DAGBoost. However, this effect is much larger for CAM as the graph before pruning contains many more edges. Thus, DAGBoost is less reliant on the hyperparameters of the pruning step compared to CAM.

5.3.2 Sensitivity Analysis

We investigate the performance of DAGBoost with respect to a variation of the step size μ and the regularization parameter λ.

When varying λ we set $\mu = 0.3$. On the other hand when varying μ we fix $\lambda = 0.01$. We conduct our analysis with ER graphs with $p = 100$ nodes and additive SEs. The mean and standard deviation of the SHD between the estimated and true graph are reported in Tables 4 and 5.

μ	mean(SHD)	SD(SHD)	mean(runtime in s)	SD(runtime in s)
0.3	14.90	7.12	58.47	14.82
0.5	14.21	7.17	43.72	10.97
0.7	13.65	7.13	37.45	7.46
0.9	13.17	7.05	20.17	2.84

Table 4: Mean and SD of SHDs between estimated and true graph for a varying step size μ. The penalty parameter is fixed at $\lambda = 0.01$. The graphs are of ER type and the SEs are additive. The runtime statistics are based on 10 experiments.

λ	mean(SHD)	SD(SHD)	mean(runtime in s)	SD(runtime in s)
0.001	15.58	7.56	43.54	10.23
0.01	12.12	6.92	64.10	19.28
0.1	14.90	7.12	204.87	63.54

Table 5: Mean and SD of SHDs between estimated and true graph for a varying penalty parameter λ. The step size is fixed at $\mu = 0.3$. The graphs are of ER type and the SEs are additive. The runtime statistics are based on 10 experiments.

One can see that the influence of the hyperparameters on the SHDs is minor. The AIC score controls the number of boosting steps m_{stop} very efficiently. It thus accounts well for the different base learners, which depend on the step size μ and the penalty parameter λ. An increase in the step size or a reduction in the penalty parameter leads to a smaller number of boosting iterations which in turn leads to a reduced runtime. At the same time

the quality of the estimated graph is not strongly effected. We thus recommend to use DAGBoost with a large step size or a low penalty parameter if the computational resources or the time are limited.

Although the impact of the hyperparameters is shown for one specific setting, based on our observations, they similarly hold in a wide range of data-generating processes.

6 Conclusion

In this work we investigated boosting for causal discovery and for the estimation of the causal order. We proposed a generic score function on the orderings that depends on a regression estimator. We presented two sufficient conditions on the regression estimator, so that the score function can consistently distinguish between aligning and incompatible orderings.

1. The regression estimator must consistently find the true regression function in a correctly specified scenario with homoscedastic noise, and

2. the mean squared prediction error on the samples must converge to the expected L^2-prediction error for yet unseen observations even in general misspecified scenarios.

Together, the conditions imply a safety net for the regression estimator, which is interesting on its own. In a misspecified setting, the fit of the regression function to the observed samples gives a good estimate for the expected squared prediction error for yet unobserved realizations. In a correctly specified setting on the other side, the regression estimator still identifies the underlying functional relationship.

We showed that boosting with appropriate early stopping provides this safety net. Thus, our analysis gives insights on the generalization ability of boosting procedures for real-world data, which most likely does not meet all model assumptions.

In order to use a score function on the orderings for the identification of the topological order, one needs to score every possible permutation. This is infeasible for large p and insufficient prior knowledge on the causal structure. Thus we proposed a greedy boosting in the space of functions of $\mathbb{R}^p \to \mathbb{R}^p$ which correspond to a DAG. The algorithm can be understood as a functional gradient descent in the space of additive SEMs, aka componentwise boosting.

A simulation study underlined that the score function on the permutations consistently prefers a correct causal order and the convergence manifests already for small N. These findings were even robust to non-additive SEs. For small p or in case of extensive prior knowledge that drastically reduces the search space of permutations, the combination of the score and a feature selection procedure can be used for deriving the causal graph.

The second part of our simulations study showed that the gradient descent is highly competitive with state-of-the-art algorithms. Particularly for complex data-generating processes, the algorithm provides a noticeable benefit. Besides, the exact choice of the hyperparameters are efficiently and automatically balanced by the number of boosting iterations and the AIC score. Thus, the procedure is easy to tune and ready to be applied to a variety of data sets.

Many parts of our analysis were generic and the RKHS regression can easily be replaced by other regression estimators as spline regression or neural networks. Thus one can further

investigate which regression estimators lead to consistent estimators for the causal order and the empirical performance and theoretical properties are to be explored. A combination of gradient-based methods as in Zheng et al. [2018] and boosting could also lead to insights.

A Proof of Proposition 1

Proof. Recall that $\varpi_\pi(k) := \{j : \pi(j) < \pi(k)\}$.

$$
\begin{aligned}
\widehat{S}(\pi) &= \sum_{k=1}^{p} \log(\widehat{\sigma}^2_{k,\widehat{f}_{k,\pi}}) \\
&\geq \min_{(f_1,\ldots,f_p)\in\vartheta(\pi)} \sum_{k=1}^{p} \log\left((P_N - P)\left(X_k - \widehat{f}_{k,\pi}(\mathbf{X}_{\varpi_\pi(k)})\right)^2 + \sigma^2_{k,p_{\theta^0},f_k,\pi}\right) \\
&\geq \min_{(f_1,\ldots,f_p)\in\vartheta(\pi)} \sum_{k=1}^{p} \log(\sigma^2_{k,p_{\theta^0},f_k,\pi}) \\
&\quad + \underbrace{\max\left\{0, \frac{-(P_N - P)\left(X_k - \widehat{f}_{k,\pi}(\mathbf{X}_{\varpi_\pi(k)})\right)^2}{\sigma^2_{k,p_{\theta^0},f_k,\pi} + (P_N - P)\left(X_k - \widehat{f}_{k,\pi}(\mathbf{X}_{\varpi_\pi(k)})\right)^2}\right\}}_{=:\Delta_{N,k}} \\
&= \min_{(f_1,\ldots,f_p)\in\vartheta(\pi)} \sum_{k=1}^{p} \log(\sigma^2_{k,p_{\theta^0},f_k,\pi}) + \sum_{k=1}^{p} \Delta_{N,k} \\
&= S(\pi) + \sum_{k=1}^{p} \Delta_{N,k} \\
&= S(\pi^0) + \sum_{k=1}^{p} \Delta_{N,k} + \underbrace{S(\pi) - S(\pi^0)}_{\xi_{\pi,\pi^0}>0} \\
&= \sum_{k=1}^{p} \log\left(\left(\sigma^0_k\right)^2 - \widehat{\sigma}^2_{k,\widehat{f}_{k,\pi^0}} + \widehat{\sigma}^2_{k,\widehat{f}_{k,\pi^0}}\right) + \Delta_{N,k} + \xi_{\pi,\pi^0} \\
&\geq \sum_{k=1}^{p} \log\left(\widehat{\sigma}^2_{k,\widehat{f}_{k,\pi^0}}\right) + \underbrace{\max\left\{0, -\frac{\left(\sigma^0_k\right)^2 - \widehat{\sigma}^2_{k,\widehat{f}_{k,\pi^0}}}{\left(\sigma^0_k\right)^2}\right\}}_{=:\gamma_{N,k}} + \Delta_{N,k} + \xi_{\pi,\pi^0} \\
&= \widehat{S}(\pi^0) + \sum_{k=1}^{p} (\gamma_{N,k} + \Delta_{N,k}) + \xi_{\pi,\pi^0}
\end{aligned}
$$

In the first inequality, we used that for any $k = 1, \ldots, p$

$$
\begin{aligned}
P\left((X_k - \widehat{f}_{k,\pi}(\mathbf{X}_{\varpi_\pi(k)})^2\right) &= \mathbb{E}_{p_{\theta^0}}\left[(X_k - \widehat{f}_{k,\pi}(\mathbf{X}_{\varpi_\pi(k)})^2\right] \\
&\geq \min_{(f_1,\ldots,f_p)\in\vartheta(\pi)} \mathbb{E}_{p_{\theta^0}}\left[(X_k - f_k(\mathbf{X}_{\varpi_\pi(k)})^2\right] \\
&= \min_{(f_1,\ldots,f_p)\in\vartheta(\pi)} \sigma^2_{k,p_{\theta^0},f_k,\pi}.
\end{aligned}
$$

In the second and last inequality Lemma 2 below was used. Further, $\xi_{\pi,\pi^0} > 0$, which does not depend on N, by the identifiability of the model. By the assumptions and the

continuous mapping theorem it holds

$$\mathbb{P}\left(|\Delta_{N,k}| \geq \frac{\xi_{\pi,\pi^0}}{2p}\right) \to 0 \text{ and } \mathbb{P}\left(|\gamma_{N,k}| \geq \frac{\xi_{\pi,\pi^0}}{2p}\right) \to 0 \text{ for all } k = 1, \ldots, p \text{ and } N \to \infty,$$

from which we derive that

$$\xi_N = |\sum_{k=1}^{p} \Delta_{N,k} + \gamma_{N,k}| \leq \sum_{k=1}^{p} |\Delta_{N,k}| + \sum_{k=1}^{p} |\gamma_{N,k}| < \xi_{\pi,\pi^0}$$

with probability going to 1 for $N \to \infty$. □

Lemma 2. *For $x > 0$ and $x + \delta > 0$ it holds that*

$$\log(x + \delta) \geq \log(x) + \max\left\{0, -\frac{\delta}{x + \delta}\right\}$$

Proof. The statement is true for $\delta \geq 0$. For $\delta < 0$ it holds that

$$\begin{aligned}\log(x) &= \log(x - |\delta| + |\delta|) \leq \log(x - |\delta|) + \frac{|\delta|}{x - |\delta|} = \log(x + \delta) - \frac{\delta}{x + \delta} \\ &= \log(x + \delta) - \max\left\{0, -\frac{\delta}{x + \delta}\right\}\end{aligned}$$

from which the result follows. □

B Proof of Theorem 6

The proof of Theorem 6 decomposes $(P_N - P)\left(Y - \widehat{f}^{(m_{stop})}\right)^2$ into $(P_N - P)\left(Y\widehat{f}^{(m_{stop})}\right)$ and $(P_N - P)\left(\widehat{f}^{(m_{stop})}\right)^2$. We show both convergences using the theory of empirical processes.

For this purpose, we show in Section B.1 that $||\widehat{f}^{(m_{stop})}||_H$ can be upper-bounded. Section B.2 then derives the convergence of $(P_N - P)\left(Y\widehat{f}^{(m_{stop})}\right)$ using *Covering Numbers*. The convergence of $(P_N - P)\left(\widehat{f}^{(m_{stop})}\right)^2$ is then shown using *Rademacher Complexities*.

B.1 Upper-Bound of the RKHS Norm of Regression Function Estimate

The main result of this section is Lemma 5, which upper-bounds the Hilbert space norm of the boosting estimate $||\widehat{f}^{(m)}||_H$ with a probability going to 1 for $N \to \infty$ for a suitably chosen number of boosting iterations m. The analysis is based on the fact that $||\widehat{f}^{(m)}||_H^2$ can be expressed as a quadratic form.

Lemma 3. *Let S be the kernel regression learner with penalty parameter λ. Assume that the Gram matrix G is invertible. Then it holds for the boosting estimate after m boosting steps that*

$$||\widehat{f}^{(m)}||_H^2 = \frac{1}{N}(y^N)^T U(I - (I - D)^m)^2 \Lambda^{-1} U^T y^N,$$

where $U, D, \Lambda \in \mathbb{R}^{N \times N}$ and D, Λ are diagonal matrices. Λ has the eigenvalues $\widehat{\mu}_1, \ldots, \widehat{\mu}_N$ of G and D has $\frac{\widehat{\mu}_1}{\widehat{\mu}_1 + \lambda}, \ldots, \frac{\widehat{\mu}_N}{\widehat{\mu}_N + \lambda}$ on the diagonal. U contains the corresponding eigenvectors of G.

Proof. It holds for the linear base learner S mapping y^N to $\widehat{f}(\mathbf{x}^N)$, that

$$S = G(G + \lambda I)^{-1}.$$

The matrix S is symmetric and has the eigenvalues $d_1 = \frac{\widehat{\mu}_1}{\widehat{\mu}_1+\lambda}, \ldots, d_N = \frac{\widehat{\mu}_N}{\widehat{\mu}_N+\lambda}$. Thus, for the orthogonal matrix U containing the eigenvectors of S and the diagonal matrix D containing the eigenvalues $d_1, \ldots, d_N$ of S it holds that

$$S = UDU^T.$$

By Equation (14), the estimate $\widehat{f}^{(m)} = B^{(m)}y^N = (I - (I - S)^m)y^N$ can be expressed by

$$\widehat{f}^{(m)}(\widetilde{\mathbf{x}}) = \frac{1}{\sqrt{N}} \sum_{k=1}^{N} \widehat{\beta}_k K(\widetilde{\mathbf{x}}, \widetilde{\mathbf{x}}_k)$$

for some $\widehat{\beta} \in \mathbb{R}^N$. Using the results of Section 3.2 we obtain

$$\begin{aligned} ||\widehat{f}^{(m)}||_H^2 &= \widehat{\beta}^\top G \widehat{\beta} \\ &= \widehat{f}^{(m)}(\widetilde{x}^N)^\top \frac{G^{-1}}{\sqrt{N}} G \frac{G^{-1}}{\sqrt{N}} \widehat{f}^{(m)}(\widetilde{x}^N) \\ &= \frac{1}{N}(y^N)^\top B^{(m)} G^{-1} B^{(m)} y^N. \end{aligned}$$

Note that G, S and $B^{(m)}$ have the same eigenvectors. Besides, Λ and $I-(I-D)^m$ are diagonal matrices, so they commute. Hence, $B^{(m)}G^{-1}B^{(m)} = U(I-(I-D)^m)U^\top U\Lambda^{-1}U^\top U(I-(I-D)^m)U^\top = U(I-(I-D)^m)^2\Lambda^{-1}U^T$. Thus,

$$||\widehat{f}^{(m)}||_H^2 = \frac{1}{N}(y^N)^\top B^{(m)} G^{-1} B^{(m)} y^N = \frac{1}{N}(y^N)^\top U(I-(I-D)^m)^2\Lambda^{-1}U^\top y^N.$$

□

The following Hanson-Wright-inequality gives probabilistic upper bounds for quadratic forms as derived in Lemma 3.

Theorem 8 (Hanson-Wright-Inequality)**.** *[Rudelson and Vershynin, 2013, Theorem 1.1] Consider a random vector $\mathbf{Z} = (Z_1, \ldots, Z_N) \in \mathbb{R}^N$ with independent components, for which $\mathbb{E}(Z_\ell) = 0, \ell = 1, \ldots, N$ and for which the Orlicz norm of $Z_1, \ldots, Z_N$ is uniformly bounded by s_{max}. For any $M \in \mathbb{R}^{N\times N}$ it holds for every $t > 0$*

$$\mathbb{P}\left(\left|||M\mathbf{Z}||^2 - \mathbb{E}\left[||M\mathbf{Z}||^2\right]\right| > t\right) \leq 2\exp\left(-c\min\left\{\frac{t^2}{s_{max}^4||M^2||_F^2}, \frac{t}{s_{max}^2||M^2||}\right\}\right).$$

Lemma 3 shows that the RKHS norm can be expressed as a quadratic form. In the next steps, we upper bound the quadratic form, that is, the RKHS norm of $\widehat{f}^{(m)}$ using Theorem 8. We see, that we can choose M in Theorem 8 as the matrix square root of $U(I-(I-D)^m)^2\Lambda^{-1}U^T$ so that $||MY||^2 = ||\widehat{f}^{(m)}||_H^2$. Thus, these bounds depend on the number of boosting steps m and D and Λ (which are functions of G), where the latter are probabilistically depending on $\widetilde{\mathbf{X}}^N$. Controlling D, this allows us to vary m with N such that the growth of the upper bound for $||\widehat{f}^{(m)}||_H^2$ can be controlled with a high probability. Observe that Theorem 8 requires centered random variables $Z_1, \ldots, Z_N$.

Lemma 4. *Decompose* $Y = \mu(\widetilde{\mathbf{X}}) + \varepsilon(\widetilde{\mathbf{X}})$, *where* $\mu(\widetilde{\mathbf{X}}) = \mathbb{E}\left[Y|\widetilde{\mathbf{X}}\right]$ *and* $\varepsilon(\widetilde{\mathbf{X}}) = Y - \mathbb{E}\left[Y|\widetilde{\mathbf{X}}\right]$. *Assume that* $||\mu||_\infty < \mu_{max}$ *and the Orlicz norm and variance of the conditional distribution of* $\varepsilon(\widetilde{\mathbf{X}})$ *given some realization* $\widetilde{\mathbf{x}}$ *of* $\widetilde{\mathbf{X}}$ *is uniformly bounded by* s_{max} *and* σ^2_{max}, *respectively. Let* $\widehat{f}^{(m)}$ *be the boosting estimate after* m *boosting steps and* $\Lambda, D, U \in \mathbb{R}^{N\times N}$ *as in Lemma 3. We can upper bound*

$$\begin{aligned}\mathbb{P}\left(||\widehat{f}^{(m)}||_H^2 > 1 + 2N(\mu^2_{max} + \sigma^2_{max})||M_m^{1/2}(\widetilde{\mathbf{x}}^N)||_F^2 | \widetilde{\mathbf{X}}^N = \widetilde{\mathbf{x}}^N\right) \\ \leq 2\exp\left(-C\min\left\{\frac{1}{4s^4_{max}||M_m||_F^2}, \frac{1}{2s^2_{max}||M_m||}\right\}\right).\end{aligned}$$

where

$$M_m(\widetilde{\mathbf{x}}^N) := M_m := \frac{1}{N}U(I-(I-D)^m)^2\Lambda^{-1}U^T.$$

Proof. By Lemma 3 it holds

$$||\widehat{f}^{(m)}||_H^2 = \frac{1}{N}(y^N)^T U(I-(I-D)^m)^2\Lambda^{-1}U^T y^N = \left(y^N\right)^\top M_m(\widetilde{\mathbf{x}}^N)y^N.$$

We emphasize that G and thus D, U and S are functions of $\widetilde{\mathbf{X}}^N$. We calculate for $\mu^N = (\mu(\widetilde{\mathbf{x}}_1), \ldots, \mu(\widetilde{\mathbf{x}}_N)) \in \mathbb{R}^N$ and $\varepsilon^N = (\varepsilon_1(\widetilde{\mathbf{x}}_1), \ldots, \varepsilon_N(\widetilde{\mathbf{x}}_N)) \in \mathbb{R}^N$:

$$\begin{aligned}&\mathbb{P}\left(||\widehat{f}^{(m)}||_H^2 - 2N(\mu^2_{max}+\sigma^2_{max})||M_m^{1/2}(\widetilde{\mathbf{X}}^N)||_F^2 > 1 | \widetilde{\mathbf{X}}^N = \widetilde{\mathbf{x}}^N\right) \\ &= \mathbb{P}\left(||M_m^{1/2}(\widetilde{\mathbf{X}}^N)Y^N||^2 - 2N(\mu^2_{max}+\sigma^2_{max})||M_m^{1/2}(\widetilde{\mathbf{X}}^N)||_F^2 > 1 | \widetilde{\mathbf{X}}^N = \widetilde{\mathbf{x}}^N\right) \\ &\leq \mathbb{P}\Big(2||M_m^{1/2}(\widetilde{\mathbf{X}}^N)\mu^N||^2 + 2||M_m^{1/2}(\widetilde{\mathbf{X}}^N)\varepsilon^N||^2 \\ &\quad - 2N(\mu^2_{max}+\sigma^2_{max})||M_m^{1/2}(\widetilde{\mathbf{X}}^N)||_F^2 > 1 | \widetilde{\mathbf{X}}^N = \widetilde{\mathbf{x}}^N\Big)\end{aligned}$$

In the third line we used the decomposition $y^N = \mu^N + \varepsilon^N$ and the inequality $||a+b||^2 \leq 2||a||^2 + 2||b||^2$. In the following we upper-bound the terms $||M_m^{1/2}(\widetilde{\mathbf{X}}^N)\mu^N||$ and $-N||M_m^{1/2}(\widetilde{\mathbf{X}}^N)||_F^2\sigma^2_{max}$. For the first term observe that

$$\begin{aligned}||M_m^{1/2}(\widetilde{\mathbf{X}}^N)\mu^N||^2 &= tr\left(\left(\left(\mu^N\right)^\top M_m(\widetilde{\mathbf{X}}^N)\mu^N\right)\right) \\ &= tr\left(M_m(\widetilde{\mathbf{X}}^N)\mu^N\left(\mu^N\right)^\top\right) \\ &\leq tr\left(M_m(\widetilde{\mathbf{X}}^N)\right) tr\left(\mu^N\left(\mu^N\right)^\top\right) \\ &\leq N||M_m^{1/2}\left(\widetilde{\mathbf{X}}^N\right)||_F^2\mu^2_{max}.\end{aligned}$$

For the latter term using that $\mathbf{E}\left[\varepsilon_k^2|\widetilde{\mathbf{X}}^N\right] \leq \sigma^2_{max}, k = 1, \ldots, N$ it holds analogously

$$\begin{aligned}\mathbb{E}\left[||M_m^{1/2}(\widetilde{\mathbf{X}}^N)\varepsilon^N||^2|\widetilde{\mathbf{X}}^N\right] &= \mathbb{E}\left[tr\left(\left(\varepsilon^N\right)^\top M_m(\widetilde{\mathbf{X}}^N)\varepsilon^N\right)|\widetilde{\mathbf{X}}^N\right] \\ &\leq \mathbb{E}\left[tr\left(M_m(\widetilde{\mathbf{X}}^N)\right) tr\left(\varepsilon^N\left(\varepsilon^N\right)^\top\right)|\widetilde{\mathbf{X}}^N\right] \\ &\leq tr\left(M_m(\widetilde{\mathbf{X}}^N)\right)\mathbb{E}\left[\left(\varepsilon^N\right)^\top\varepsilon^N|\widetilde{\mathbf{X}}^N\right] \\ &\leq N||M_m^{1/2}(\widetilde{\mathbf{X}}^N)||_F^2\sigma^2_{max}\end{aligned}$$

and hence $-N||M_m^{1/2}(\widetilde{\mathbf{X}}^N)||_F^2\sigma_{max}^2 \leq -\mathbb{E}\left[||M_m^{1/2}(\widetilde{\mathbf{X}}^N)\varepsilon^N||^2|\widetilde{\mathbf{X}}^N\right]$. Using these results and plugging in the condition $\widetilde{\mathbf{X}}^N = \widetilde{\mathbf{x}}^N$ we can upper-bound

$$\begin{aligned}
&\mathbb{P}\left(2||M_m^{1/2}\mu^N||^2 + 2||M_m^{1/2}\varepsilon^N||^2 - 2N(\mu_{max}^2+\sigma_{max}^2)||M_m^{1/2}(\widetilde{\mathbf{X}}^N)||_F^2 > 1|\widetilde{\mathbf{X}}^N = \widetilde{\mathbf{x}}^N\right)\\
&\leq \mathbb{P}\left(||M_m^{1/2}\varepsilon^N||^2 - \mathbb{E}\left(||M_m^{1/2}(\widetilde{\mathbf{X}}^N)\varepsilon^N||^2|\widetilde{\mathbf{X}}^N\right) > \frac{1}{2}|\widetilde{\mathbf{X}}^N = \widetilde{\mathbf{x}}^N\right)\\
&= \mathbb{P}\left(||M_m^{1/2}\varepsilon^N||^2 - \mathbb{E}\left(||M_m^{1/2}\varepsilon^N||^2\right) > \frac{1}{2}|\widetilde{\mathbf{X}}^N = \widetilde{\mathbf{x}}^N\right)\\
&\leq 2\exp\left(-C\min\left\{\frac{1}{4s_{max}^4||M_m||_F^2}, \frac{1}{2s_{max}^2||M_m||}\right\}\right)
\end{aligned}$$

by Theorem 8 (Hanson-Wright-inequality) and the fact that $\varepsilon^N|\widetilde{\mathbf{X}}^N = \widetilde{\mathbf{x}}^N$ is a centered, sub-Gaussian random vector with independent components with Orlicz norm bounded by s_{max}. □

We now show that if we choose $m = m(N) = N^{\frac{1}{4}\frac{C_u+C_d+1/2}{C_d+1}}$ then the growth of $||\widehat{f}^{(m)}||_H$ with N is of lower order as $N^{1/4}$ with a probability going to 1 if $\widetilde{\mathbf{x}}^N \in \mathcal{B}_N$.

Lemma 5 (Upper bound $||\widehat{f}^{(m_{stop})}||_H$)**.** *Under Assumption 4′ and for* $m_{stop} = m(N) = N^{\frac{1}{4}\frac{C_u+C_d+1/2}{C_d+1}}$ *there exists a* $\delta > 0$ *and a* $h(N) \in o(N^{1/4-\delta})$ *so that*

$$\mathbb{P}\left(\widehat{f}^{(m_{stop})} \notin h(N)\mathcal{F}_N \cap \{\widetilde{\mathbf{X}}^N \in \mathcal{B}_N\}\right) \to 0 \tag{23}$$

for $N \to \infty$.

Proof. As outlined in relation (14), by the representation theorem it holds that $\widehat{f}^{(m)} \in h(N)\mathcal{F}_N$ for some $h(N) \in \mathbb{R}$. Thus we need to show that $||\widehat{f}^{(m_{stop})}||_H \leq h(N)$. We prove the statement by showing the convergence

$$\mathbb{P}\left(||\widehat{f}^{(m_{stop})}||_H^2 > h(N)^2|\widetilde{\mathbf{X}}^N = \widetilde{\mathbf{x}}^N\right) \to 0$$

uniformly for any $\widetilde{\mathbf{x}}^N \in \mathcal{B}_N$ and some $h(N) \in o(N^{1/4-\delta})$. The statement then follows by integrating out with respect to $\widetilde{\mathbf{x}}^N$. Applying Lemma 4, we need to prove the following two statements.

1. $N||M_m^{1/2}||_F^2 \in o\left(N^{1/2-2\delta}\right)$. This implies, $||M_m||_F^2 \leq ||M_m^{1/2}||_F^4 \in o(1)$.
2. $||M_m|| \in o(1)$.

It is emphasized that M_m is a function of the random sample $\widetilde{\mathbf{x}}^N$.
The statement is proven in Lemma 6. □

Lemma 6. *Under Assumption 4′ it holds for*

$$M_m(\widetilde{\mathbf{x}}^N) := M_m := \frac{1}{N}U(I-(I-D)^m)^2\Lambda^{-1}U^T$$

that there exists a $\delta > 0$ *so that the following statements hold for* $m = m(N) = N^{\frac{1}{4}\frac{C_u+C_d+1/2}{C_d+1}}$ *uniformly in* $\widetilde{\mathbf{x}}^N \in \mathcal{B}_N$*, where* $\mathcal{B}_N$ *is defined in Assumption 4′.*

1. $N^{1/2+2\delta}||M_m^{1/2}||_F^2 = \frac{1}{N^{1/2-2\delta}} tr\left((I-(I-D)^m)^2\Lambda^{-1}\right) \to 0$,

2. $||M_m||_F^2 \to 0$, *and*

3. $||M_m|| \to 0$.

Proof. **1.:** Uniformly in $\widetilde{\mathbf{x}}^N \in \mathcal{B}_N$ it holds that:

$$\begin{aligned}
N^{1/2+2\delta}||M_m^{1/2}||_F^2 &= N^{1/2+2\delta} tr(M_m) \\
&= \frac{1}{N^{1/2-2\delta}} \sum_{k=1}^{N} (1-(1-d_k)^m)^2 \widehat{\mu_k}^{-1} \\
&= \frac{1}{N^{1/2-2\delta}} \sum_{k=1}^{N} \left(1-\left(1-\frac{\widehat{\mu_k}}{\widehat{\mu_k}+\lambda}\right)^m\right)^2 \widehat{\mu_k}^{-1} \\
&\overset{(i)}{\leq} \frac{1}{N^{1/2-2\delta}} \sum_{k=1}^{N} \min\left\{1, m^2\left(\frac{\widehat{\mu_k}}{\widehat{\mu_k}+\lambda}\right)^2\right\} \widehat{\mu_k}^{-1} \\
&\leq \frac{1}{\lambda^2 N^{1/2-2\delta}} \sum_{k=1}^{N} \min\left\{\lambda^2 \widehat{\mu_k}^{-1}, m^2 \widehat{\mu_k}\right\} \\
&\leq \frac{1}{\lambda^2 N^{1/2-2\delta}} \left(\sum_{k=1}^{\lfloor K_0 \rfloor} \min\left\{\lambda^2 \widehat{\mu_k}^{-1}, m^2 \widehat{\mu_k}\right\} + \sum_{k=\lceil K_0 \rceil}^{N} \min\left\{\lambda^2 \widehat{\mu_k}^{-1}, m^2 \widehat{\mu_k}\right\}\right) \\
&\leq \frac{1}{\lambda^2 N^{1/2-2\delta}} \left(K_0 \widehat{\mu_{K_0}}^{-1} \lambda^2 + \sum_{k=\lceil K_0 \rceil}^{N} m^2 \widehat{\mu_k}\right),
\end{aligned}$$

which holds for any $K_0 \in \mathbb{N}$. The inequality (i) is shown in Lemma 8. The last inequality is due to the fact that $\widehat{\mu}_k^{-1}$ is monotonically increasing. We choose $K_0 = K_0(N) = \lfloor \frac{1}{2C_d+1} \ln(N) \rfloor$. Uniformly in $\widetilde{\mathbf{x}}^N \in \mathcal{B}_N$ it holds by Assumption 4′ for a small $\delta > 0$

$$\frac{K_0 \widehat{\mu_{K_0}}^{-1}}{N^{1/2-2\delta}} \leq \frac{K_0 \exp(C_d K_0)}{N^{1/2-2\delta}} \to 0.$$

For the latter part observe that it holds for any $\widetilde{\mathbf{x}}^N \in \mathcal{B}_N$ and by Assumption $4'$

$$\begin{aligned}
\frac{1}{N^{1/2-2\delta}} \sum_{k=\lceil K_0 \rceil}^{N} m^2 \widehat{\mu_k} &\leq \frac{m^2}{N^{1/2-2\delta}} \sum_{k=\lceil K_0 \rceil}^{N} \exp(-C_u k) \\
&\leq \frac{m^2}{N^{1/2-2\delta}} \int_{K_0}^{\infty} \exp(-C_u(z-1))dz \\
&= \frac{m^2 \exp(C_u)}{N^{1/2-2\delta} C_u} \exp(-C_u K_0) \\
&= \frac{m^2 \exp(C_u)}{N^{1/2-2\delta} C_u} \exp(-C_u(\underbrace{K_0+1}_{\geq \frac{1}{2C_d+1}\ln(N)} -1)) \\
&\leq \frac{m^2 \exp(2C_u)}{N^{1/2-2\delta} C_u} N^{-\frac{C_u}{2C_d+1}} \\
&= \frac{m^2 \exp(2C_u)}{C_u} N^{-\frac{C_u+C_d+1/2}{2C_d+1}} N^{2\delta}.
\end{aligned}$$

Observe that for $m = m(N) = N^{\frac{1}{4}\frac{C_u+C_d+1/2}{C_d+1}}$ where the constants are chosen independently of $\widetilde{\mathbf{x}}^N$, it holds

$$m^2 N^{-\frac{C_u+C_d+1/2}{2C_d+1}} = N^{\frac{C_u+C_d+1/2}{2C_d+2} - \frac{C_u+C_d+1/2}{2C_d+1}} = N^{-\xi},$$

where $\xi := \frac{C_u+C_d+1/2}{2C_d+1} - \frac{C_u+C_d+1/2}{2C_d+2} > 0$. For $\delta < \frac{\xi}{2}$ it holds that $\frac{1}{N^{1/2-2\delta}} \sum_{k=\lceil K_0 \rceil}^{N} m^2 \widehat{\mu_k} \to 0$.

2. follows by $||M_m||_F^2 \leq ||M_m^{1/2}||_F^4 \to 0$.

3. follows by

$$\begin{aligned}
||M_m|| &= \frac{1}{N} \max_{k=1,\ldots,N} \left(1 - \left(1 - \frac{\widehat{\mu}_k}{\widehat{\mu}_k + \lambda}\right)^m\right)^2 \widehat{\mu}_k^{-1} \\
&\overset{(i)}{\leq} \frac{1}{N} \max_{k=1,\ldots,N} \min\left\{1, m^2 \left(\frac{\widehat{\mu_k}}{\widehat{\mu_k} + \lambda}\right)^2\right\} \widehat{\mu_k}^{-1} \\
&\leq \frac{1}{\lambda^2 N} \max_{k=1,\ldots,N} \min\left\{\widehat{\mu_k}^{-1}, m^2 \widehat{\mu_k}\right\} \leq \frac{m^2 \widehat{\mu}_1}{\lambda^2 N} \to 0,
\end{aligned}$$

where inequality (i) follows again from Lemma 8 as $\widehat{\mu}_1 \leq 1$. □

The following Lemma immediately follows from the proof of Lemma 6.

Lemma 7. *Under Assumption $4'$ it holds for $m = m(N) = N^{\frac{1}{4}\frac{C_u+C_d+1/2}{C_d+1}}$ on $\mathcal{B}_N$, that*

$$\frac{1}{\sqrt{N}} \sum_{\ell=1}^{N} \left(1 - \left(1 - \frac{\widehat{\mu}_\ell}{\widehat{\mu}_\ell + \lambda}\right)^m\right)^2 \to 0.$$

Lemma 8. *It holds for $0 \leq \widehat{\mu_k} \leq 1$, that*

$$1 - (1 - \widehat{\mu_k})^m \leq 1 - \max\{0, 1 - m\widehat{\mu_k}\} = \min\{1, m\widehat{\mu_k}\}$$

Proof. It is equivalent to show that

$$(1-\widehat{\mu_k})^m \geq \max\{0, 1-m\widehat{\mu_k}\}.$$

The l.h.s. and the r.h.s. are equal for $\widehat{\mu_k} = 0$. On the interval $[0, \frac{1}{m})$ it holds that

$$\frac{\partial(1-\widehat{\mu_k})^m}{\partial\widehat{\mu_k}} = -m(1-\widehat{\mu_k})^{m-1} \geq -m = \frac{\partial(\max\{0, 1-m\widehat{\mu_k}\}}{\partial\widehat{\mu_k}}.$$

Hence,

$$(1-\widehat{\mu_k})^m \geq \max\{0, 1-m\widehat{\mu_k}\} \text{ for } \widehat{\mu_k} \in [0, \frac{1}{m}).$$

Clearly, for $\mu_k \in [\frac{1}{m}, 1]$

$$(1-\widehat{\mu_k})^m \geq \max\{0, 1-m\widehat{\mu_k}\}.$$

□

B.2 Results on Covering Numbers

Lemma 5 has shown that for $m_{stop} = N^{\frac{1}{4}\frac{C_u+C_d+1/2}{C_d+1}}$, it holds $\widehat{f}^{(m_{stop})} \in h(N)\mathcal{F}_N$ for some $h(N) \in o(N^{1/4})$ with probability going to 1 for $N \to \infty$. In this section we use the covering numbers from empirical process theory to show the convergence of the inner product

$$|(P - P_N)Y\widehat{f}^{(m_{stop})}|. \tag{24}$$

The covering numbers measure the complexity of a function class.

Definition 4. *For a function class $\mathcal{F}$ and a semi-metric d on $\mathcal{F}$ the covering number $\mathcal{N}(\varepsilon, \mathcal{F}, d)$ is the minimal size of a subset $S \subset \mathcal{F}$, such that for every $f \in \mathcal{F}$ there is an $s \in S$ so that $d(f, s) < \varepsilon$. More precisely,*

$$\mathcal{N}(\varepsilon, \mathcal{F}, d) = \min_{\{S \subset \mathcal{F} | \forall f \in \mathcal{F}\, \exists s \in S : d(f,s) < \varepsilon\}} |S|.$$

In this work, we choose $d(f, g) = ||f - g||_\infty$ and thus write $\mathcal{N}(u, \mathcal{F}, ||\cdot||_\infty) = \mathcal{N}(u, \mathcal{F})$. For a suitable C_0 [chosen as in Dudley's Theorem, see Theorem 8.4 of van de Geer, 2014] not depending on $\mathcal{F}$ we define the covering number entropy integral by

$$\mathcal{J}(z, \mathcal{F}) := C_0 z \int_0^1 \sqrt{\log \mathcal{N}\left(\frac{uz}{2}, \mathcal{F}\right)} du.$$

For a constant $C > 0$, let $\mathcal{H} = \{Ch | h \in \mathcal{G}\}$ be a scaled version of some function class $\mathcal{G}$. The following remark shows that the entropy integral $\mathcal{J}(z, \mathcal{H})$ of $\mathcal{H}$ can be upper bounded by an expression depending on C and the entropy integral $\mathcal{J}(z, \mathcal{G})$ of $\mathcal{G}$.

Remark 8. *For $C > 0$ the identity*

$$\mathcal{N}(Cz, C\mathcal{G}, ||\cdot||) \leq \mathcal{N}(z, \mathcal{G}, ||\cdot||)$$

holds and thus $\mathcal{J}(Cz, C\mathcal{G}) \leq C\mathcal{J}(z, \mathcal{G})$, where $C\mathcal{G} = \{Cg | g \in \mathcal{G}\}$. This upper bound is not optimal [see Cucker and Smale, 2002] but sufficient for our purposes.

Eventually, we will show in Corollary 1 that if $\widehat{f}^{(m_{stop})}$ is in the ball of radius $h(N) \in o(N^{1/4})$, then this growth rate $h(N)$ is slow enough to ensure the convergence of (24). We rely on the following theorem.

Theorem 9 (van de Geer [2014, Theorem 3.2])**.** *Let $K = \sup_{f\in\mathcal{F}} ||f||_\infty$ and assume that Y is sub-Gaussian with Orlicz norm smaller than s. Then for t, N such that* [1] $\sqrt{\frac{2t}{N}} + \frac{t}{N} \leq 1$ *it holds with probability* $1 - 8\exp(-t)$

$$\sup_{f\in\mathcal{F}} |(P_N - P)Yf|/C \leq \frac{2\mathcal{J}(Ks, \mathcal{F}) + Ks\sqrt{t}}{\sqrt{N}}. \tag{25}$$

Corollary 1. *Let $f_1, f_2, \ldots$ be a sequence in H such that*

$$f_N \in h(N)\mathcal{F}_N,$$

where $h(N) \in o(N^{1/4})$. If $\mathcal{J}(z, \mathcal{F}_N) \leq \mathcal{J}(z, B_1) = C_0 z \int_0^1 \sqrt{\log\left(\mathcal{N}(\frac{uz}{2}, B_1)\right)} < \infty$ for all $z > 0$ as in Assumption 3′, then

$$|(P_N - P)Yf_N| \leq \xi.$$

converges to 0 *in probability.*

Proof. For $h(N)\mathcal{F}_N$ it holds that

$$K := \sup_{f\in h(N)\mathcal{F}_N} ||f||_\infty \leq \sup_{f\in h(N)B_1} ||f||_\infty \leq Bh(N).$$

Recalling the definition of $\mathcal{J}(u, \mathcal{F})$ and applying Remark 8, we obtain

$$\mathcal{J}(Ku, h(N)\mathcal{F}_N) \leq \mathcal{J}(Bh(N)u, h(N)B_1) \leq h(N)\mathcal{J}(Bu, B_1)\,\forall u \in \mathbb{R}_+.$$

As the Orlicz norm fulfills the triangle inequality and as the Orlicz norm of the bounded random variable $\mu(\widetilde{\mathbf{X}})$ is finite, the Orlicz norm of $Y = \mu(\widetilde{\mathbf{X}}) + \varepsilon$, denoted by s, is bounded and Y is thus sub-Gaussian. We now apply Theorem 9 and set $t = N^{1/2}$ (the condition $\sqrt{\frac{2t}{N}} + \frac{t}{N} \leq 1$ is then fulfilled for $N > \frac{1}{2}\left(7 + 3\sqrt{5}\right)$).

It holds with probability $1 - 8\exp(-N^{1/2})$

$$\begin{aligned}
|(P_N - P)Yf_N| &\leq \sup_{f\in h(N)\mathcal{F}_N} |(P_N - P)Yf| \\
&\leq \frac{\mathcal{J}(Bh(N)s, h(N)\mathcal{F}_N) + Bh(N)s_{max}N^{1/4}}{\sqrt{N}} \\
&\leq \frac{h(N)\mathcal{J}(Bs, \mathcal{F}_N) + Bh(N)s_{max}N^{1/4}}{\sqrt{N}} \\
&\leq \frac{h(N)\mathcal{J}(Bs, B_1) + Bh(N)s_{max}N^{1/4}}{\sqrt{N}}.
\end{aligned}$$

[1] There is a typo in van de Geer [2014], where it says $J_0(K\sigma, \mathcal{F})$ instead of $J_\infty(K\sigma, \mathcal{F})$.

$\mathcal{J}(Bs, B_1)$ is finite and constant in N by Assumption 3′. Let $\xi > 0$ be arbitrary. As $h(N) \in o(N^{1/4})$, there exists N^0 so that

$$\frac{h(N)\mathcal{J}(Bs, B_1)}{\sqrt{N}} \leq \frac{\xi}{2} \forall N > N^0.$$

For the second term observe that as $h(N) \in o(N^{1/4})$

$$\frac{Bh(N)s_{max}N^{1/4}}{\sqrt{N}} = \frac{Bh(N)s_{max}}{N^{1/4}} \leq \frac{\xi}{2}$$

for all $N > N^1$ for some $N^1 \in \mathbb{N}$. Thus for $N > \max\{N^0, N^1\}$

$$|(P_N - P)Yf_N| \leq \xi$$

with probability $1 - 8\exp(-8N^{1/2}) \to 1$. This proves the convergence in probability. □

B.3 Results on Rademacher Complexities

In this section we use concept of the Rademacher complexity to show the convergence of

$$\left|(P - P_N)\left(\widehat{f}^{(m_{stop})}\right)^2\right|.$$

It is again based on Lemma 5, which ensures that for $m_{stop} = N^{\frac{1}{4}\frac{C_u+C_d+1/2}{C_d+1}}$, it holds $\widehat{f}^{(m_{stop})} \in h(N)\mathcal{F}_N$ for some $h(N) \in o(N^{1/4-\delta})$ for some $\delta > 0$ with probability going to 1 for $N \to \infty$.

Definition 5 (Rademacher complexity)**.** *Let $\mathcal{F}$ be a function class on $\mathbf{X}$. Let $\sigma_1, \ldots, \sigma_N$ be i.i.d. realizations of Rademacher random variables, which are independent of $\mathbf{X}$. Further let $\mathbf{x}_1, \ldots, \mathbf{x}_N$ be i.i.d realizations of $\mathbf{X}$. We define the Rademacher complexity $R_N(\mathcal{F})$ by*

$$R_N(\mathcal{F}) = \frac{1}{N}\mathbb{E}_{\mathbf{X},\sigma}\left[\sup_{f\in\mathcal{F}}\left|\sum_{\ell=1}^{N}\sigma_\ell f(\mathbf{x}_\ell)\right|\right].$$

Note that $R_N(\mathcal{F})$ is deterministic. Given fixed observations $\mathbf{x}_1, \ldots, \mathbf{x}_N$, the empirical Rademacher complexity is given by

$$\widehat{R}_N(\mathcal{F}|\mathbf{x}_1, \ldots, \mathbf{x}_N) = \widehat{R}_N(\mathcal{F}) = \frac{1}{N}\mathbb{E}_{\sigma}\left[\sup_{f\in\mathcal{F}}\left|\sum_{\ell=1}^{N}\sigma_\ell f(\mathbf{x}_\ell)\right|\right].$$

The Rademacher complexity is a tool to upper bound $\sup_{f\in\mathcal{F}}|(P_N - P)f|$.

Theorem 10. *[Wainwright, 2019, Theorem 4.10] Let $\mathcal{F}$ be uniformly bounded with constant B. Then it holds for any $N \in \mathbb{N}$ and with probability $1 - \exp(-\frac{\varepsilon^2 N}{2B^2})$, that*

$$\sup_{f\in\mathcal{F}}|(P_N - P)f| \leq 2R_N(\mathcal{F}) + \varepsilon.$$

The Rademacher complexity is linked to its empirical counterpart with high probability.

Theorem 11 (Bartlett and Mendelson [2002, Theorem 11]). *Let $\mathcal{F}$ be uniformly bounded by B. Then it holds with probability $1 - 2\exp\left(-\frac{\varepsilon^2 N}{B^2}\right)$*

$$\left|\widehat{R}_N(\mathcal{F}) - R_N(\mathcal{F})\right| < \varepsilon.$$

We collect some helpful relationships for the Rademacher complexity.

Theorem 12 (Bartlett and Mendelson [2002, Theorem 12]). *Let $\mathcal{F}, \mathcal{F}_1, \mathcal{F}_2, \ldots, \mathcal{F}_p$ be function classes and let $\mathcal{F}$ be uniformly bounded by B. Then*

1. *for $c \in \mathbb{R}$: $R_N(c\mathcal{F}) = |c| R_N(\mathcal{F})$,*
2. *$R_N\left(\sum_{j=1}^p \mathcal{F}_j\right) \leq \sum_{j=1}^p R_N\left(\mathcal{F}_j\right)$,*
3. *for $\mathcal{F}^2 := \{|f|^2 : f \in \mathcal{F}\}$, it follows $R_N(\mathcal{F}^2) \leq 4B R_N(\mathcal{F})$.*

The Rademacher complexity can be upper bounded for kernel functions.

Theorem 13 (Bartlett and Mendelson [2002, Lemma 22]). *The empirical Rademacher complexity of $\mathcal{F}_N$ is upper bounded by*

$$\widehat{R}_N(\mathcal{F}_N) \leq \left(\frac{2}{N}\sum_{k=1}^N \widehat{\mu_k}\right)^{\frac{1}{2}},$$

while the population Rademacher complexity can be upper bounded by

$$R_N(\mathcal{F}_N) \leq \left(\frac{2}{N}\sum_{k=1}^N \mu_k\right)^{\frac{1}{2}}.$$

Note that if the sequence $\mu_1, \mu_2, \ldots$ is summable, then $R_N(\mathcal{F}_N) \in O(N^{-1/2})$. For this case we connect the results above in a corollary.

Corollary 2. *For any sequence $f_1, f_2, \ldots$ in H for which*

$$f_N \in h(N)\mathcal{F}_N,$$

where $h(N) \in o(N^{1/4-\delta})$ for some $\delta > 0$ and $\widehat{R}_N(\mathcal{F}_N) \in O(N^{-1/2})$, it holds

$$|(P_N - P) f_N^2| \xrightarrow{\mathbb{P}} 0 \text{ for } N \to \infty.$$

Proof. Let $\xi > 0$ be arbitrary and fixed. By Theorem 12 it holds that

$$R_N(h(N)\mathcal{F}_N) = h(N) R_N(\mathcal{F}_N),$$

and as $\mathcal{F}_N$ is uniformly bounded by $Bh(N)$ it holds by Theorem 12

$$R_N\left((h(N)\mathcal{F}_N)^2\right) = 4Bh(N) R_N(\mathcal{F}_N).$$

From Theorem 10 we obtain that for any $\xi > 0$

$$|(P_N - P)f_N^2| \leq \sup_{f \in h(N)\mathcal{F}_N} \left|(P_N - P)f^2\right| \leq 2R_N\left((h(N)\mathcal{F}_N)^2\right) + \frac{\xi}{2} = 8Bh(N)^2 R_N(\mathcal{F}_N) + \frac{\xi}{3}$$

with probability $1 - 2\exp\left(\frac{\xi^2 N}{4(Bh(N))^2}\right)$, which converges to 1 as $h(N) \in o(N^{1/4})$ for $N \to \infty$. Similarly, as $\mathcal{F}_N$ is uniformly bounded by B and by setting $\varepsilon = N^{-1/2+2\delta}$ in Theorem 11, we observe that

$$|\widehat{R}_N(\mathcal{F}_N) - R_N(\mathcal{F}_N)| \leq N^{-1/2+2\delta}$$

with probability going to 1 for any $\delta > 0$. We conclude that

$$\begin{aligned}
|(P_N - P)f_N^2| &\leq \sup_{f \in h(N)\mathcal{F}_N} \left|(P_N - P)f^2\right| \\
&\leq 8Bh(N)^2 R_N(\mathcal{F}_N) + \frac{\xi}{3} \\
&\leq 8Bh(N)^2 |R_N(\mathcal{F}_N) - \widehat{R}_N(\mathcal{F}_N)| + 8Bh(N)^2 \widehat{R}_N(\mathcal{F}_N) + \frac{\xi}{3} \\
&\leq 8Bh(N)^2 N^{-1/2+2\delta} + 8Bh(N)^2 \widehat{R}_N(\mathcal{F}_N) + \frac{\xi}{3}
\end{aligned}$$

with probability going to 1 for $N \to \infty$. As $h(N) \in o(N^{1/4})$, it holds $h(N)^2 \widehat{R}_N(\mathcal{F}_N) < \frac{\xi}{3}$ by Assumption 4′ and Theorem 13 for N chosen large enough. Similarly, $h(N)^2 N^{-1/2+2\delta} < \frac{\xi}{3}$ for N chosen large enough, as $h(N) \in o\left(N^{1/4-\delta}\right)$. This proves the statement. □

C Proof of Theorem 7

Proof of Lemma 1. Using

$$\begin{aligned}
||f^0 - \widehat{f}^{(m)}||_{2,N}^2 &= \frac{1}{N}||f^0(\widetilde{\mathbf{x}}^N) - B^{(m)} y^N||_2^2 \\
&= \frac{1}{N}||f^0(\widetilde{\mathbf{x}}^N) - U\left(I - (I-D)^m\right) U^\top y^N||_2^2 \\
&= \frac{1}{N}||f^0(\widetilde{\mathbf{x}}^N) - U\left(I - (I-D)^m\right) U^\top \left(f^0(\widetilde{\mathbf{x}}^N) + \varepsilon^N\right)||_2^2 \\
&\leq \frac{2}{N}||f^0(\widetilde{\mathbf{x}}^N) - U\left(I - (I-D)^m\right) U^\top f^0(\widetilde{\mathbf{x}}^N)||_2^2 \\
&\quad + \frac{2}{N}||U\left(I - (I-D)^m\right) U^\top \varepsilon^N)||_2^2 \\
&\leq \underbrace{\frac{2}{N}||U\left(I-D\right)^m U^\top f^0(\widetilde{\mathbf{x}}^N)||_2^2}_{\text{I}} + \underbrace{\frac{2}{N}||U\left(I - (I-D)^m\right) U^\top \varepsilon^N||_2^2}_{\text{II}},
\end{aligned}$$

where the first inequality holds due to $||a+b||^2 \leq 2||a||^2 + ||b||^2$, we show the convergence of I and II to 0 in probability for $N \to \infty$. Recall that $S = UDU^T$ with U containing the orthonormal eigenvalues of S and D being a diagonal matrix with diagonal entries $D_{\ell\ell} = d_\ell = \frac{\widehat{\mu}_\ell}{\widehat{\mu}_\ell + \lambda}$, where $\widehat{\mu}_\ell, \ell = 1, \ldots, N$ are the eigenvalues of G.

Convergence of I:

$$\frac{2}{N}||U\,(I-D)^m\,U^\top f^0(\widetilde{\mathbf{x}}^N)||_2^2 = \frac{2}{N} f^0(\widetilde{\mathbf{x}}^N)^\top U\,(I-D)^m\,U^\top U\,(I-D)^m\,U^\top f^0(\widetilde{\mathbf{x}}^N)$$
$$= \frac{2}{N}\sum_{\ell=1}^N (1-d_\ell)^{2m}\left(U^\top f^0(\widetilde{\mathbf{x}}^N)\right)_\ell^2$$

As G has full rank, there exists a $\beta \in \mathbb{R}^N$ such that $f^0(\widetilde{\mathbf{x}}^{\mathbf{N}}) = \sqrt{N}G\beta$. Define $\widetilde{f} := \frac{1}{\sqrt{N}}\sum_{\ell=1}^N \beta_\ell K(\cdot, \widetilde{\mathbf{x}}_\ell)$ for which holds $\widetilde{f}(\widetilde{\mathbf{x}}^N) = f^0(\widetilde{\mathbf{x}}^N)$. By the representation theorem it further holds $\beta^\top G\beta = ||\widetilde{f}||_H^2 \leq ||f^0||_H^2 = R^2$. Let $D_{\widehat{\mu}} \in \mathbb{R}^{N\times N}$ be the diagonal matrix with diagonal entries $\widehat{\mu}_1, \ldots, \widehat{\mu}_N$. Then,

$$\begin{aligned}
\frac{2}{N}\sum_{\ell=1}^N (1-d_\ell)^{2m}\left(U^\top f^0(\widetilde{\mathbf{x}}^N)\right)_\ell^2 &\leq \frac{2}{N}\sum_{\ell=1}^N \frac{\left(U^\top f^0(\widetilde{\mathbf{x}}^N)\right)_\ell^2}{2e\,m\,d_\ell} \\
&= \frac{1}{N}\sum_{\ell=1}^N \frac{\left(U^\top\sqrt{N}G\beta\right)_\ell^2}{e\,m\,d_\ell} \\
&= \sum_{\ell=1}^N \frac{\left(U^\top G\beta\right)_\ell^2}{e\,m\,d_\ell} \\
&\leq (1+\lambda)\sum_{\ell=1}^N \frac{\left(U^\top G\beta\right)_\ell^2}{e\,m\,\widehat{\mu}_\ell} \\
&= \frac{1+\lambda}{e\,m} tr\left(D_{\widehat{\mu}}^{-1}U^\top G\beta\beta^\top GU\right) \\
&= \frac{1+\lambda}{e\,m} tr\left(\underbrace{UD_{\widehat{\mu}}^{-1}U^\top}_{G^{-1}} G\beta\beta^\top G\right) \\
&= \frac{1+\lambda}{e\,m}\beta^\top G\beta = \frac{1+\lambda}{e\,m}||\widetilde{f}||_H^2 \leq \frac{1+\lambda}{e\,m}R^2,
\end{aligned}$$

which goes to 0 as $m(N) \to \infty$ for $N \to \infty$. In the first inequality we have used the fact that $(1-x)^{2m} \leq \exp(-x)^{2m} = \exp(-2mx) \leq \frac{1}{2e\,m\,x}$ for all $x \in \mathbb{R}$, where e is Euler's number. In the second inequality we have used $\frac{1}{d_\ell} = \frac{\widehat{\mu}_\ell+\lambda}{\widehat{\mu}_\ell} \leq \frac{1+\lambda}{\widehat{\mu}_\ell}$, as $0 < \widehat{\mu}_\ell \leq 1$ for all $l = 1, 2, \ldots, N$.

Convergence of II:

$$\mathbb{P}\left(\frac{1}{N}||U\,(I-(I-D)^m)\,U^\top\varepsilon^N||_2^2 > \xi\right)$$
$$\leq \mathbb{P}\left(\left(\frac{1}{N}||U\,(I-(I-D)^m)\,U^\top\varepsilon^N||_2^2 > \xi\right) \cap \left\{\widetilde{\mathbf{X}}^N \in \mathcal{B}_N\right\}\right) + \mathbb{P}\left(\left\{\widetilde{\mathbf{X}}^N \notin \mathcal{B}_N\right\}\right)$$

The latter term goes again to 0 by Assumption 4′. For any $\widetilde{\mathbf{x}}^N \in \mathcal{B}_N$ we show that

$$\mathbb{P}\left(\frac{1}{N}||U\,(I-(I-D)^m)\,U^\top\varepsilon^N||_2^2 > \xi \mid \widetilde{\mathbf{X}}^N = \widetilde{\mathbf{x}}^N\right) \to 0$$

for any $\xi > 0$ and uniformly in $\widetilde{\mathbf{x}}^N$. Recall that $\varepsilon^N \mid \widetilde{\mathbf{X}}^N = \widetilde{\mathbf{x}}^N$ is a sub-Gaussian vector with mean 0 and independent components. Thus, we can apply the Hanson-Wright inequality

of Theorem 8. Remember that D is a function of $\widetilde{\mathbf{x}}^N$. We need to show for any $\widetilde{\mathbf{x}}^N \in \mathcal{B}_N$, $A_m = \frac{1}{\sqrt{N}} U \left(I - (I-D)^m\right) U^\top$ that $\mathbb{E}\left[\left|\left|A_m \varepsilon^N\right|\right|_2^2 \mid \widetilde{\mathbf{X}}^{\mathbf{N}} = \widetilde{\mathbf{x}}^N\right] \to 0$, $||A_m^2||_F^2 \to 0$ and $||A_m^2|| \to 0$. It holds by Lemma 8 uniformly for $\widetilde{\mathbf{x}}^N \in \mathcal{B}_N$

$$\begin{aligned}
\mathbb{E}\left[\left|\left|A_m \varepsilon^N\right|\right|_2^2 | \widetilde{\mathbf{X}} = \widetilde{\mathbf{x}}^N\right] &= \mathbb{E}_{Y^N | \widetilde{\mathbf{X}}^N = \widetilde{\mathbf{x}}^N} \left[\left|\left| \frac{1}{\sqrt{N}} U \left(I - (I-D)^m\right) U^\top \varepsilon^N \right|\right|_2^2\right] \\
&= \mathbb{E}_{Y^N | \widetilde{\mathbf{X}}^N = \widetilde{\mathbf{x}}^N} \left[\frac{1}{N} tr\left(\left(\varepsilon^N\right)^\top U \left(I - (I-D)^m\right)^2 U^\top \varepsilon^N\right)\right] \\
&\leq \mathbb{E}_{Y^N | \widetilde{\mathbf{X}}^N = \widetilde{\mathbf{x}}^N} \left[\frac{1}{N} tr\left(\varepsilon^N \left(\varepsilon^N\right)^\top\right) tr\left(U \left(I - (I-D)^m\right)^2 U^\top\right)\right] \\
&\leq \frac{\sigma^2}{N} tr\left(\left(I - (I-D)^m\right)^2\right) \\
&= \frac{\sigma^2}{N} \sum_{\ell=1}^N \left(1 - (1-d_\ell)^m\right)^2 \\
&= \frac{\sigma^2}{N} \sum_{\ell=1}^N \left(1 - \left(1 - \frac{\widehat{\mu}_\ell}{\lambda + \widehat{\mu}_\ell}\right)^m\right)^2 \to 0
\end{aligned}$$

for $N \to \infty$. Further,

$$||A_m^2||_F^2 = \frac{1}{N^2} tr\left(\left(I - (I-D)^m\right)^4\right) \leq \left(\frac{1}{N} tr\left(\left(I - (I-D)^m\right)^2\right)\right)^2,$$

which goes to 0 with the same calculation as above. Finally,

$$||A_m^2|| \leq \frac{1}{N} \to 0.$$

The statement follows by integrating out $\mathcal{B}_N$ with respect to $\widetilde{\mathbf{x}}^N$.

□

D Algorithm

Data: $\mathbf{x}^N = (\mathbf{x}_1, \ldots, \mathbf{x}_N)^\top \in \mathbb{R}^{N \times p}$
Input: Kernels $K_1, \ldots, K_p$ implying RKHSs $H_1, \ldots, H_p$, penalty λ, step size μ
Output: $\widehat{G}, \widehat{F}$
$F^{(1)} \leftarrow 0$
$\widehat{G} \leftarrow \emptyset$
$N \leftarrow \emptyset$
$\widehat{f}_{kj} = \arg\min_{g_{kj} \in H_j} \sum_{\ell=1}^{N} \left(g_{kj}(\mathbf{x}_{\ell j}) - \mathbf{x}_{\ell k}\right)^2 + \lambda ||g_{kj}||^2_{H_j}, j, k = 1, \ldots, p, j \neq k$
$S(j,k) \leftarrow \log\left(\sum_{\ell=1}^{N} \left(\widehat{f}_{kj}(\mathbf{x}_{\ell j}) - \mathbf{x}_{\ell k}\right)^2\right)$
for $m \leftarrow 2$ **to** m_{stop} **do**
  // Find the next edge and update graph
  $(j^0, k^0) \leftarrow \arg\min_{(j,k) \notin N} S(j, k, F^{(m-1)})$; $\widehat{G} \leftarrow \widehat{G} + (j^0, k^0)$
  $f_{k^0}^{(m)} \leftarrow f_{k^0}^{(m-1)} + \mu \widehat{f}_{k^0 j^0}$
  $f_k^{(m)} \leftarrow f_k^{(m-1)}, k = 1, \ldots, k^0 - 1, k^0 + 1, \ldots, p$
  $F^{(m)} \leftarrow \left(f_1^{(m)}, \ldots, f_p^{(m)}\right)$
  // Check if AIC score has increased
  if $AIC(F^{(m)}, \mathbf{x}^N) > AIC(F^{(m-1)}, \mathbf{x}^N)$ **then** break
  // Update edges that cause cycle and ensure they are not chosen anymore
  $N \leftarrow getForbiddenEdges(\widehat{G})$
  // Update S
  $\widehat{f}_{k^0 j} \leftarrow$ Equation (21), $j = 1, \ldots, p$
  $S(j, k^0, F^{(m)}) \leftarrow$ Equation 20, $j = 1, \ldots, p$
end
return $\widehat{G}, F^{(m)}$

References

Trent Kyono, Yao Zhang, and Mihaela van der Schaar. CASTLE: Regularization via auxiliary causal graph discovery. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 1501–1512. Curran Associates, Inc., 2020.

Sara Aibar, Carmen Bravo González-Blas, Thomas Moerman, Vân Anh Huynh-Thu, Hana Imrichova, Gert Hulselmans, Florian Rambow, Jean-Christophe Marine, Pierre Geurts, Jan Aerts, et al. Scenic: single-cell regulatory network inference and clustering. *Nature methods*, 14(11):1083–1086, 2017.

Maximilian Kertel, Stefan Harmeling, Markus Pauly, and Nadja Klein. Learning causal graphs in manufacturing domains using structural equation models. *International Journal of Semantic Computing*, 17(04):511–528, 2023.

Rohit Bhattacharya, Tushar Nagarajan, Daniel Malinsky, and Ilya Shpitser. Differentiable causal discovery under unmeasured confounding. In Arindam Banerjee and Kenji Fuku-

mizu, editors, *International Conference on Artificial Intelligence and Statistics*, volume 130, pages 2314–2322. The Proceedings of Machine Learning Research, 13–15 Apr 2021.

Kun Zhang, Jonas Peters, Dominik Janzing, and Bernhard Schölkopf. Kernel-based conditional independence test and application in causal discovery. In *Conference on Uncertainty in Artificial Intelligence*, pages 804–813, Arlington, Virginia, USA, 2011. AUAI Press. ISBN 9780974903972.

Rajen Shah and Jonas Peters. The hardness of conditional independence testing and the generalised covariance measure. *Annals of Statistics*, 48(3):1514–1538, 2018.

Jonas Peters, Dominik Janzing, and Bernhard Schölkopf. *Elements of Causal Inference: Foundations and Learning Algorithms*. MIT Press, Cambridge, MA, USA, 2017.

Peter Spirtes, Clark Glymour, and Richard Scheines. *Causation, Prediction, and Search*, volume 81. MIT Press, 01 1993. ISBN 978-1-4612-7650-0. doi: https://doi.org/10.7551/mitpress/1754.001.0001.

Markus Kalisch and Peter Bühlman. Estimating high-dimensional directed acyclic graphs with the PC-algorithm. *The Journal of Machine Learning Research*, 8(3):613–636, 2007.

Jonas Peters, Joris M. Mooij, Dominik Janzing, and Bernhard Schölkopf. Causal discovery with continuous additive noise models. *The Journal of Machine Learning Research*, 15 (1):2009–2053, 01 2014. ISSN 1532–4435.

Matthew J. Vowels, Necati Cihan Camgoz, and Richard Bowden. D'ya like DAGs? a survey on structure learning and causal discovery. *ACM Computing Surveys*, 55(4), 11 2022.

Sébastien Lachapelle, Philippe Brouillard, Tristan Deleu, and Simon Lacoste-Julien. Gradient-based neural DAG learning. In *International Conference on Learning Representations*, 2019.

Yue Yu, Jie Chen, Tian Gao, and Mo Yu. DAG-GNN: DAG structure learning with graph neural networks. In Kamalika Chaudhuri and Ruslan Salakhutdinov, editors, *International Conference on Machine Learning*, volume 97 of *The Proceedings of Machine Learning Research*, pages 7154–7163. PMLR, 09–15 Jun 2019.

Xun Zheng, Chen Dan, Bryon Aragam, Pradeep Ravikumar, and Eric Xing. Learning sparse nonparametric dags. In *International Conference on Artificial Intelligence and Statistics*, pages 3414–3425. The Proceedings of Machine Learning Research, 2020.

Diviyan Kalainathan, Olivier Goudet, Isabelle Guyon, David Lopez-Paz, and Michale Sebag. Structural agnostic modeling: Adversarial learning of causal graphs. *The Journal of Machine Learning Research*, 23(219):1–62, 2022.

Ignavier Ng, Sébastien Lachapelle, Nan Rosemary Ke, Simon Lacoste-Julien, and Kun Zhang. On the convergence of continuous constrained optimization for structure learning. In *International Conference on Artificial Intelligence and Statistics*, pages 8176–8198. The Proceedings of Machine Learning Research, 2022a.

Ignavier Ng, Shengyu Zhu, Zhuangyan Fang, Haoyang Li, Zhitang Chen, and Jun Wang. Masked gradient-based causal structure learning. In *International Conference on Data Mining (SDM)*, pages 424–432. SIAM, 2022b.

Shohei Shimizu, Takanori Inazumi, Yasuhiro Sogawa, Aapo Hyvärinen, Yoshinobu Kawahara, Takashi Washio, Patrik O Hoyer, and Kenneth Bollen. DirectLiNGAM: A direct method for learning a linear non-Gaussian structural equation model. *The Journal of Machine Learning Research*, 12:1225–1248, 2011.

Peter Bühlmann, Jonas Peters, and Jan Ernest. CAM: Causal additive models, high-dimensional order search and penalized regression. *The Annals of Statistics*, 42(6):2526–2556, 2014. doi: 10.1214/14-AOS1260.

Marcus Kaiser and Maksim Sipos. Unsuitability of NOTEARS for causal graph discovery when dealing with dimensional quantities. *Neural Processing Letters*, 54(3):1587–1595, 2022.

Marc Teyssier and Daphne Koller. Ordering-based search: A simple and effective algorithm for learning bayesian networks. In *Conference on Uncertainty in Artificial Intelligence*, pages 584–590, Arlington, Virginia, USA, 2005. AUAI Press. ISBN 0974903914.

Ali Shojaie and George Michailidis. Penalized likelihood methods for estimation of sparse high-dimensional directed acyclic graphs. *Biometrika*, 97(3):519–538, 07 2010.

Peter Bühlmann and Bin Yu. Boosting with the l2 loss. *Journal of the American Statistical Association*, 98(462):324–339, 2003. doi: 10.1198/016214503000125. URL `https://doi.org/10.1198/016214503000125`.

Garvesh Raskutti, Martin J. Wainwright, and Bin Yu. Early stopping and non-parametric regression: an optimal data-dependent stopping rule. *The Journal of Machine Learning Research*, 15(1):335–366, 2014.

Robert E. Schapire and Yoav Freund. *Boosting: Foundations and Algorithms*. The MIT Press, 2012. ISBN 0262017180.

Grace Wahba. *Spline Models for Observational Data*. Society for Industrial and Applied Mathematics, 1990. doi: 10.1137/1.9781611970128.

Bernhard Schölkopf and Alexander J. Smola. *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. MIT Press, Cambridge, MA, USA, 2001. ISBN 0262194759.

Martin J. Wainwright. *High-Dimensional Statistics: A Non-Asymptotic Viewpoint*. Cambridge Series in Statistical and Probabilistic Mathematics. Cambridge, 2019.

Kirthevasan Kandasamy and Yaoliang Yu. Additive approximations in high dimensional nonparametric regression via the salsa. In *International Conference on Machine Learning*, pages 69–78. The Proceedings of Machine Learning Research, 2016.

Hongwei Sun. Mercer theorem for RKHS on noncompact sets. *Journal of Complexity*, 21 (3):337–349, 2005. ISSN 0885-064X.

Francis R. Bach and Michael I. Jordan. Kernel independent component analysis. *The Journal of Machine Learning Research*, 3(Jul):1–48, 2002.

Felipe Cucker and Steve Smale. On the mathematical foundations of learning. *Bulletin of the American Mathematical Society*, 39(1):1–49, 2002.

Bruno Peter Zwahlen. Über die Eigenwerte der Summe zweier selbstadjungierter Operatoren. *Commentarii Mathematici Helvetici*, 40:81–116, 1965/66.

Ha Quang Minh. Some properties of Gaussian reproducing kernel Hilbert spaces and their implications for function approximation and learning theory. *Constructive Approximation*, 32:307–338, 2010.

Gerhard Tutz and Harald Binder. Generalized additive modeling with implicit variable selection by likelihood-based boosting. *Biometrics*, 62(4):961–971, 2006.

Peter Bühlmann and Torsten Hothorn. Boosting algorithms: Regularization, prediction and model fitting. *Statistical Science*, 22(4):477–505, 2007. doi: 10.1214/07-STS242.

Paul Erdös and Alfréd Rényi. On random graphs i. *Publicationes Mathematicae Debrecen*, 6:290, 1959.

Albert-László Barabási and Réka Albert. Emergence of scaling in random networks. *Science*, 286(5439):509–512, 1999. doi: 10.1126/science.286.5439.509.

Hawoong Jeong, Bálint Tombor, Réka Albert, Zoltan N Oltvai, and A-L Barabási. The large-scale organization of metabolic networks. *Nature*, 407(6804):651–654, 2000.

Anja Wille, Philip Zimmermann, Eva Vranová, Andreas Fürholz, Oliver Laule, Stefan Bleuler, Lars Hennig, Amela Prelić, Peter von Rohr, Lothar Thiele, et al. Sparse graphical gaussian modeling of the isoprenoid gene network in arabidopsis thaliana. *Genome biology*, 5(11):1–13, 2004.

Xun Zheng, Bryon Aragam, Pradeep Ravikumar, and Eric P Xing. DAGs with NO TEARS: Continuous optimization for structure learning. In *Advances in Neural Information Processing Systems*, volume 32, 2018.

Mark Rudelson and Roman Vershynin. Hanson-Wright inequality and sub-Gaussian concentration. *Electronic Communications in Probability*, 18:1–9, 2013. doi: 10.1214/ECP.v18-2865.

Sara van de Geer. On the uniform convergence of empirical norms and inner products, with application to causal inference. *Electronic Journal of Statistics*, 8(1):543–574, 2014.

Peter L. Bartlett and Shahar Mendelson. Rademacher and Gaussian complexities: Risk bounds and structural results. *The Journal of Machine Learning Research*, 3(Nov):463–482, 2002.

Additional Article

Wehner et al. (2023)

Interactive and Intelligent Root Cause Analysis in Manufacturing with Causal Bayesian Networks and Knowledge Graphs

1st Christoph Wehner
Cognitive Systems Group
University of Bamberg
Bamberg, Germany
christoph.wehner@uni-bamberg.de

2nd Maximilian Kertel
Technology Development Battery Cell
BMW Group
Munich, Germany
maximilian.kertel@bmw.de

3rd Judith Wewerka
Digitalization and Innovation
BMW Group
Munich, Germany
judith.wewerka@bmw.de

***Abstract*—Root Cause Analysis (RCA) in the manufacturing of electric vehicles is the process of identifying fault causes. Traditionally, the *RCA* is conducted manually, relying on process expert knowledge. Meanwhile, sensor networks collect significant amounts of data in the manufacturing process, and using the data for *RCA* makes it more efficient. However, purely data-driven methods like *Causal Bayesian Networks* have problems scaling to large-scale, real-world manufacturing processes due to the vast amount of potential cause-effect relationships (*CER's*). Furthermore, purely data-driven methods have the potential to leave out already known *CER's* or to learn spurious *CER's*. The paper contributes by proposing an interactive and intelligent *RCA* tool that combines expert knowledge of electric vehicle manufacturing processes and a data-driven machine learning method. It uses reasoning over a large-scale *Knowledge Graph* of the manufacturing process while learning a *Causal Bayesian Network*. In addition, an *Interactive User Interface* enables a process expert to give feedback to the root cause graph by adding and removing information to the *Knowledge Graph*. The interactive and intelligent *RCA* tool reduces the learning time of the *Causal Bayesian Network* while decreasing the number of spurious *CER's*. Thus, the interactive and intelligent RCA tool closes the feedback loop between expert and machine learning method.**

***Index Terms*—Root Cause Analysis, Sensor Networks, Electric Vehicles, Interpretable Machine Learning, Interactive Learning, Bayesian Network, Knowledge Graph**

I. Introduction

Machine learning is heavily used in driver assistant systems of electric vehicles [1]. The electric vehicles benefit from machine learning not only on the road, but also during the manufacturing process as the latter has become more intelligent and data-driven. It is monitored via sensor networks, resulting in significant amounts of data hour by hour. The data includes faults diagnosed by quality control process steps. However, detecting a fault in a complex manufacturing process does not necessarily uncover in which *Process Steps* the fault was induced. Finding the cause-effect relationships *(CER's)* leading to the diagnosed fault is called *Root Cause Analysis (RCA)* [2].

Traditionally, *RCA* is a manual, labor-intense, and thus expensive process [3, 4, 5], involving design of experiments to study manipulations and resulting effects. *RCA* requires high amounts of process expert knowledge. Such that only process experts can conduct a *RCA* in a reasonable timeframe.

Digital support tools promise to make *RCA* more efficient, data-driven, and less dependent on individual process knowledge [2], to reduce costs and improve the performance of the manufacturing process.

Previous work identified *Causal Bayesian Networks* as a machine learning technique to automate *RCA* [6] by learning *CER's* between *Sensor Variables*. The *Causal Bayesian Networks* constructs a root cause graph over the manufacturing process, showing potential root causes [7]. However, learning *Causal Bayesian Networks* for large manufacturing processes becomes prohibitively expensive as the search space for potential cause-effect relations explodes. This problem can be mitigated by including process expert knowledge in the learning process of the *Causal Bayesian Networks* [8]. In addition, expert knowledge can improve the learned root cause graph by identifying spurious *CER's*.

The remaining challenges for scaling *Causal Bayesian Networks* to large-scale manufacturing processes are to model process expert knowledge in detail and to provide an accessible way for the process expert to interact with and improve the root cause graph. The paper investigates those challenges by combining a large-scale *Knowledge Graph* of the manufacturing process with a *Causal Bayesian Network*. The *Knowledge Graph* allows detailed modeling of large-scale manufacturing processes and enables the process expert to improve the root cause graph.

In this work, a support tool for interactive and intelligent *RCA* in the manufacturing process of electric vehicles is proposed. The *RCA* is conducted by finding *CER's* between *Sensor Variables* of the manufacturing process.

In detail, the paper contributes by:

- Detailed modelling of process expert knowledge in a large-scale *Knowledge Graph* for a real-world electric vehicle manufacturing process.
- Automatically considering manufacturing knowledge from the *Knowledge Graph* to drastically prune the search

space while learning the *Causal Bayesian Network*.
- Allowing the process expert to interact with and improve the root cause graph by explaining the *Causal Bayesian Network* where and how to do better.

II. Related Work

RCA in manufacturing is dominated by methods defined in the ISO/IEC 31010 [9], including versions of the *Five Why's* [3], the *Failure Mode and Effect Analysis (FMEA)* [4], or the *Fault Tree Analysis* [5]. The methods structure expert knowledge, such that a process expert can use it as a manual for *RCA*. However, the decision-support tools do not take advantage of today's sensor networks and data.

Thus, machine learning methods were introduced in *RCA* to enable automated and intelligent decision-support tools, e.g. [10, 2]. In particular, *Causal Bayesian Networks* were identified as beneficial for *RCA* by learning *CER's* between measurements [11, 12, 13, 14].

For large and complex manufacturing processes, the number of possible *Causal Bayesian Network* grows super-exponentially and its derivation becomes challenging [15]. Additionally, one is not only interested in the existence of *CER's*, but also in their strength to prioritize potential root causes. The derivation of the *CER's* is called *structure learning*, while the identification of the strengths is called *parameter learning*. [7] proposes an *FMEA*-based approach, while [6] proposes an ontology of the manufacturing process to determine the structure of the *Causal Bayesian Network*. However, the ontology only considers classes of entities and does not allow the search space pruning for individual measurements. Furthermore, in complex manufacturing processes many *CER's* are unknown to the experts and both methods are unable to uncover them. On the contrary, [15] combines expert knowledge and data to derive the *Causal Bayesian Network*. This idea improves the quality of the root cause graph, as shown by [16]. While *Sensor Variables* can be continuous, most of the aforementioned approaches consider a discretization to decrease the complexity of learning the *Causal Bayesian Network* [13]. However, a loss of information is accepted, such that the data-driven identification of the *CER's* may become infeasible [17]. Recent approaches [18, 19] for learning complex *Causal Bayesian Networks* based on *Structural Equation Models* (*SEM's*) were applied on manufacturing processes [8, 20] and the benefits of the inclusion of expert knowledge were demonstrated [8].
In this work, we present a large-scale modelling of domain expertise into a *Knowledge Graph* that leverages the sensor network for deriving the *Causal Bayesian Network* of a real-world manufacturing process of electric vehicles. Furthermore, we show how the *Knowledge Graphs* contributes to an interactive *RCA* and closes the feedback loop for the *Causal Bayesian Network* learning algorithm of [8].

III. System Overview

The system for interactive and intelligent *RCA* consists of five components (cf. Figure 1), where each is a microservice on its own.

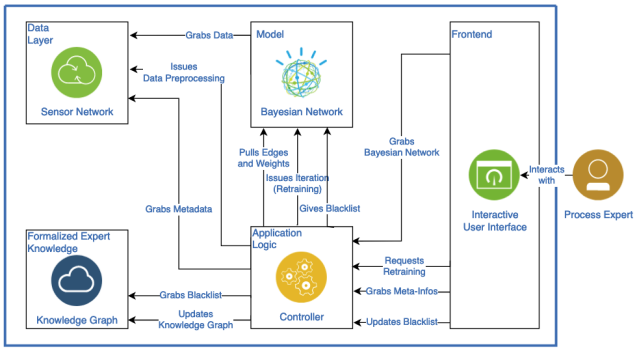

Fig. 1. The system architecture of the interactive and intelligent root cause analysis tool.

The *Controller* and the *Data Layer* enable the core components. The *Knowledge Graph*, *Causal Bayesian Network*, and *Interactive User Interface* are the core components of the interactive and intelligent root cause analysis tool.

The following describes the interaction between components and its high-level functionality. Later on, the three core components are described in detail.

The heart of the system is a *Controller* that handles the communication between all other components. It receives requests and triggers processes.

The *Data Layer* accesses data from a vehicle manufacturing-sensor network. Next, the data is preprocessed according to the input format of the *Causal Bayesian Network* and stored in the *Data Layer*. The preprocessing shall be elaborated in Subsection IV-A.

Another primary source of data is the *Knowledge Graph* (cf. Subsection III-A). The *Knowledge Graph* holds formalized expert knowledge about the manufacturing process of electric vehicles, like the temporal-spatial relations of *Stations*, *Process Steps*, and *Sensor Variables*.

The *Causal Bayesian Network* takes the preprocessed sensor data from the *Data Layer* for learning *CER's* between the *Sensor Variables* (cf. Subsection III-B). The information from the *Knowledge Graph* is used while learning to reduce the search space drastically. The resulting *Causal Bayesian Network* shows, which *Sensor Variables* may induce faults in other *Sensor Variables*. Thus, we call the graph learned by the *Causal Bayesian Network* a root cause graph.

The root cause graph is visualized in an *Interactive User Interface* (cf. Subsection III-C). The visualization enables experts to explore root causes in the manufacturing process of electric vehicles. Additionally, it allows for feedback of the process expert on the root cause graph. For example, the process expert may add an edge between two *Sensor Variables* to the *Knowledge Graph* or define a *Sensor Variable* as a *Root Variable*.

A. Knowledge Graph

A *Knowledge Graph* [21] is a directed labelled graph [22, 23] and, therefore, optimal for storing highly relational

data while preserving its semantics and allowing for deductive reasoning [21].

The *Knowledge Graph* of the *RCA* tool formalizes expert knowledge of the manufacturing process of electric vehicles. This allows the automatic, efficient, and large-scale mining of expert knowledge, which is then used while learning the *Causal Bayesian Network*.

The expert knowledge of the manufacturing process is modeled as depicted in Figure 2.

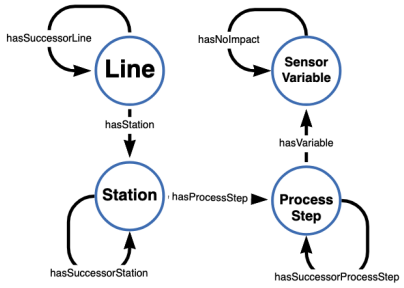

Fig. 2. Schema of the manufacturing *Knowledge Graph*.

The *Knowledge Graph* models the manufacturing process of electric vehicles as a sequence of *Lines*. A *Line* has various *Stations*. The *Stations* are also modeled as a sequence. Each *Station* implements multiple *Process Steps*. Again, the *Knowledge Graph* models the sequence in which *Process Steps* are executed. Each *Process Step* measures *Sensor Variables*. The *Knowledge Graph* models explicitly a *"hasNoImpact"* relation between *Sensor Variables* that are known to have no causal effect on each other.

In addition, a *Sensor Variable* can be a member of one subclass, which are *Root*, *Leaf*, and *Irrelevant Variable*.

Formalizing expert knowledge of the manufacturing process in such a way allows automatic reasoning over the manufacturing process. The intelligent and interactive *RCA* tool employs reasoning from the two following dimensions.

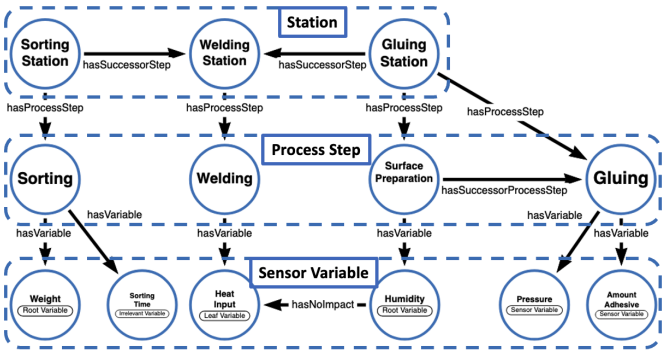

Fig. 3. Example of the manufacturing *Knowledge Graph*.

a) Temporal-spatial relations between Variables: The *Line* is modeled as a sequence of *Stations*, each with a sequence of *Process Steps*. At the bottom of this hierarchy are sensors measuring the *Sensor Variables*. Thus, the *Knowledge Graph* constructs a topology of the sensor network. It enables automatic reasoning to create a partial ordering over the *Sensor Variables*. For every *Sensor Variable*, it is known which *Sensor Variable* is measured before and after and a *Sensor Variable* measured in an earlier *Process Step* may influence a *Sensor Variable* measured in a subsequent *Process Step*, but not vice versa.

Take the example *Knowledge Graph* from Figure 3: The topology of the sensor network allows inferring that the *Sensor Variables "Weight"* and *"Sorting Time"* may causally impact the *Sensor Variable "Heat Input"*, but not vice versa and not the *Sensor Variables "Humidity"*, *"Amount Adhesive"*, and *"Pressure"*.

The example shows the reasoning behind how a significant amount of possible *CER's* between *Sensor Variables* are excluded a priori and do not have to be considered while training the *Causal Bayesian Network*. This reduces the search space of the *Causal Bayesian Network* learning algorithm drastically, compared to a naive approach of learning the *Causal Bayesian Network* solely on tabular data (cf. Figures 4 and 5).

b) Properties of Variable subclasses: Additionally, the *Knowledge Graph* incorporates expert knowledge over the *Sensor Variables* in the form of their subclass membership, which is introduced to the *Knowledge Graph* by feedback of a process expert and holds vital information to reduce the *Causal Bayesian Network*'s learning algorithm's search space.

Root Variables cannot be impacted by any *Sensor Variables*. However, they may impact other *Sensor Variables* temporarily after them. For example, the *Sensor Variable "Humidity"* in Figure 3 cannot be affected by *Sensor Variables*, as the air humidity is external to the manufacturing process. However, *"Humidity"* may affect other *Sensor Variables* later in the manufacturing process.

Leaf Variables are the inverse of *Root Variables*. They may be impacted by *Sensor Variables* measured before them, but cannot impact other *Sensor Variables*. The *"Heat Input" Sensor Variable* in Figure 3 is an example of a *Leaf Variable*, as it is the only *Sensor Variable* of the final *Process Step*.

Finally, there are *Irrelevant Variables*. Any manufacturing process measures a minor amount of *Sensor Variables* that do not impact any other *Sensor Variables* in the process. *Irrelevant Variables* are artifacts of unclear requirements, highly specific use cases, or legislation. The *"Sorting Time" Sensor Variable* in Figure 3 serves as an example, as the sorting time of the *Products* RCAs not have any impact on other *Sensor Variables*.

Figures 4 and 5 show the impact of reasoning over the temporal-spatial relations between *Sensor Variables* and their subclasses on the search space of the *Causal Bayesian Network* learning algorithm. In Figure 4, every *Sensor Variable*'s fault is potentially caused by any other *Sensor Variable*, as the sensor network's topology modeled in the *Knowledge Graph* is not considered. This amounts to a total of 30 possible *CER's* between the six *Sensor Variables*. However, considering the sensor network topology, the search space of the *Causal Bayesian Network* learning algorithm is reduced to seven

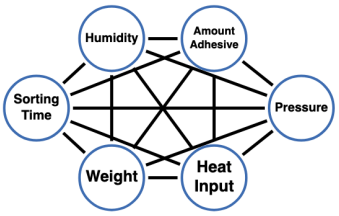

Fig. 4. The potential *CER's* between the *Sensor Variables* from the manufacturing *Knowledge Graph* (cf. Figure 3) without considering the topology of the sensor network.

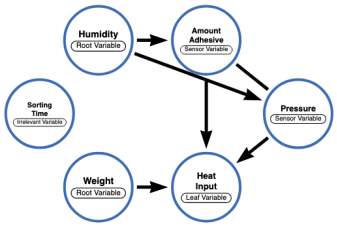

Fig. 5. The potential *CER's* between the *Sensor Variables* from the manufacturing *Knowledge Graph* (cf. Figure 3) while considering the topology of the sensor network.

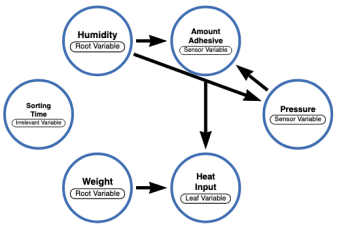

Fig. 6. The assumed true causal graph between the *Sensor Variables*. Its edges are contained in the relations of the potential *CER's* graph of Figure 5.

potential *CER's* between the *Sensor Variables* (cf. Figure 5). This example shows that automated reasoning over the expert knowledge formalized in the *Knowledge Graph* prunes the search space of the *Causal Bayesian Network* learning algorithm significantly.

B. Causal Bayesian Networks

This section introduces important concepts of *Causal Bayesian Networks*, which represent *CER's* behind data sets, and how it incorporates the *Knowledge Graph* into its training. *CER's* are dependencies, where the manipulation of one *Sensor Variable* changes the value of the other even if all remaining *Sensor Variables* are fixed. Assume the true causal graph is depicted in Figure 6. Then a change of *"Pressure"* influences *"Heat Input"* through the path *"Pressure"* $\rightarrow$ *"Amount Adhesive"* $\rightarrow$ *"Heat Input"*. For example, a higher *"Pressure"* might lead to a higher *"Amount Adhesive"*, which in turn results in a larger *"Heat Input"*. Now imagine that after an increase of *"Amount Adhesive"* we manually change all *Sensor Variables*, but *"Heat Input"* back to their original level. Then, the manipulation of *"Pressure"* does not impact *"Heat Input"* anymore, as it is only directly influenced by *"Weight"* and *"Amount Adhesive"*, which are as before. Therefore, the influential path *"Pressure"* $\rightarrow$ *"Amount Adhesive"* $\rightarrow$ *"Heat Input"* is blocked. These direct impacts we understand as the *CER's*. They are pivotal for root-cause analysis, process control, and process understanding. *CER's* imply an order over the *Sensor Variables*, which we call the *causal order*. In our running example (cf. Figure 6), one possible causal order of the *Sensor Variables* is (*"Sorting Time"*, *"Weight"* *"Humidity"*, *"Pressure"*, *"Amount Adhesive"*, *"Heat Input"*).

This paper's scope is to derive the unknown causal graph. Instead of labor and cost-intensive design of experiments, we would like to leverage our sensor network along the manufacturing process to derive the causal graph in a data-driven manner. However, if we rely exclusively on data, this is challenging due to the following reasons:

1) **Confounding:** Even if we have identified a correlation, say between *"Amount Adhesive"* and *"Heat Input"* it might be, that there is third *Sensor Variable*, say machine operator, impacting both and there is no direct impact between the two. We call this third *Sensor Variable* a confounder. Besides pathological cases, confounders lead to extra spurious relationships.
2) **Causal Order:** Even if we can exclude confounding it remains still unclear, whether *"Amount Adhesive"* is the cause and *"Heat Input"* the effect or it is vice versa.
3) **Complexity:** The number of potential graphs grows super-exponentially. Beyond a very small number of *Sensor Variables* it is thus impossible to evaluate all of them. This is especially worrisome if there are many cause-effect pairs.

Expert Knowledge to the Rescue: In the following, we present how expert knowledge contributes to derive the causal graph. As Section III-A mentions, the number of potential causal graphs can be drastically reduced by the sensor network's topology and expert knowledge on the manufacturing process:

1) sensor network topology provides a **partial ordering** of the *Sensor Variables* and
2) expert knowledge helps to exclude certain relationships and thus **avoids spurious relationships**.

Additionally, even a partial ordering drastically reduces the number of potential graphs. The classification of *Sensor Variables* into roots and leafs reduces not only the number of potential edges but also the number of relevant orderings. For example, as *"Weight"* and *"Humidity"* are defined as *Root Variables*, the relevant orderings are those that assign them to any of the first two positions. This reduces the number of permutations by a factor of 30. The effect of *Leaf Variables* is analogous. In the following, we present how we leverage the background information for causal graph identification.

Causal Additive Models: The derivation of the *CER's* from observed data is of central interest in many domains [24, 25]. For this task, score-based methods relying on *Structural Equation Models (SEM's)* [26] have recently become increasingly popular due to their ability to incorporate machine learning methods [18, 19, 27, 28]. The underlying assumption is that all *Sensor Variables* can be expressed as a function of inputs and a noise term, that captures unknown influences. In the example of Figure 5, *"Heat Input"* can be described by

$$HeatInput = f(Weight, AmountAdhesive, Noise). \quad (1)$$

We emphasize that this follows an intuitive understanding of a manufacturing process. *Process Steps* transform input material to an output, while they are impacted by machine settings and the environment. We assume that the data follows a *Causal Additive Model*, where Equation (1) is replaced by:

$$HeatInput = f_1(Humidity) + f_2(AmountAdhesive) + Noise, \quad (2)$$

The *Noise* is normally distributed and f_1 and f_2 are non-linear. Under mild assumptions [26] on f_1, f_2 and the noise, one can derive the true causal graph from observed data. [19] proposes an approach, that identifies the graph using the following steps:

1) Find the causal ordering of the *Sensor Variables*
2) Identify the *CER's*.

If the *Knowledge Graph* provides a complete ordering, then we skip the first step, and the graph identification is straightforward using regression techniques [29]. Unfortunately, the ordering is usually partial, as *Sensor Variables* measured at the same station cannot be ordered. We employ the algorithm of [8], which proposes an efficient adaption in case of expert knowledge. It limits step (1) to finding the causal ordering of the *Sensor Variables* within the *Process Steps* while mining the rest of the ordering from the *Knowledge Graph*

C. Interactive User Interface

The *Interactive User Interface* visualizes the learned *Causal Bayesian Network* (cf. Figure 7). Thereby, the *Interactive User Interface* enables no-code usage and adaption of the root cause graph by a process expert.

a) Root Cause Analysis: The workflow supported by the *Interactive User Interface* is as follows (cf. Figure 7):

The process expert searches for the *Sensor Variable* with the diagnosed fault. All possible root cause paths for the faulty *Sensor Variable* are displayed to the process expert. The process expert now has the information on how much the faulty *Sensor Variable* depends on other *Sensor Variables*. That way, without further labor-intense analysis, the process expert can move to the physical manufacturing process and check all *Process Steps* related to the identified *Sensor Variables*. The process expert benefit from a highly directed root cause search in the physical manufacturing process. Thus, the interactive and intelligent root cause analysis tool minimizes and sometimes even prevents the downtime of the manufacturing process and maximizes the manufacturing process's output.

b) Expert Feedback: The *RCA* is supported by various means of interaction with the root cause graph (cf. Figure 7).

The process expert can choose the *Product* to be considered in the *RCA*. This results in accurate root cause graphs, as *Sensor Variable* measurements significantly depend on the *Product* measured. In addition, the process expert can select the *RCA* timeframe via the *Interactive User Interface*. Selecting a timeframe enables the process expert to look precisely at the period when the faults were detected. It thus customizes the *RCA* to the individual fault at hand.

However, the root cause graph might include spurious *CER's* and need to be corrected. In this case, the expert is able to explain the *Causal Bayesian Network* where and how to do better, via adding or removing information to the *Knowledge Graph*. The *Causal Bayesian Network* may learn a *CER* between two *Sensor Variables*, which has to be rejected based on the knowledge of the process expert. The process expert can select such an edge and check "Add to Blacklist", which adds an *"hasNoImpact"* relation between the two *Sensor Variables* to the *Knowledge Graph* (cf. Figure 7 (1)) . The expert feedback is considered at the next iteration of the *Causal Bayesian Network* and thus in the following *RCA*.

In addition, spurious *CER's* might be the cause of incomplete information over subclass membership of *Sensor Variables* in the *Knowledge Graph*. Thus, the process expert can add and remove *Root*, *Leaf*, and *Irrelevant Variable* subclass membership from *Sensor Variables* (cf. Figure 7 (2)). This feedback is also considered in the next iteration of the *Causal Bayesian Network*. Assuming that the expert feedback is correct, the root cause graph converges each iteration closer to the true root cause graph.

IV. Evaluation

The evaluation is conducted on a real-world electric vehicle manufacturing process. One *Line* is taken as an example to show the *RCA* with the interactive and intelligent tool.

A. Data

The data is two-folded. There is the *Knowledge Graph* and the *Data Layer* as a source of data.

The *Knowledge Graph* is implemented in *Neo4J* [30]. This allows querying all knowledge via the Cypher API—a deductive reasoning query language [30]. The real-world *Knowledge Graph* holds a total of 100,015 nodes and 417,944 relationships. This amount of expert knowledge constitutes a large-scale knowledge graph. For the *Line* of this evaluation, there are 53 *Stations* with 96 *Process Steps* and 1683 *Sensor Variables* and a total of 2143 relations modeled.

To learn the *Causal Bayesian Network* we prepare the data by assigning the *Sensor Variables* to individual output products. Then, we preprocess the data by removing columns with only one value. Further, we remove one part of a column pair, if they have a correlation above 0.95 to avoid collinearity. Afterwards, we remove columns and rows, where more than 50% of the values are missing. As the rate of missing values is low, we simply impute the column's mean for missing values. The presented approach is generic and can be applied dynamically to different products and time windows. For this section, we consider data of one day and for one specific product type. The resulting data set has 458 *Sensor Variables*.

B. Experiments

The contribution of the paper is to give a real-world example of how to interactively improve the performance of *Causal Bayesian Networks* for *RCA* with large-scale modelled expert knowledge in the form of a *Knowledge Graph*. The contribution is shown by three experiments.

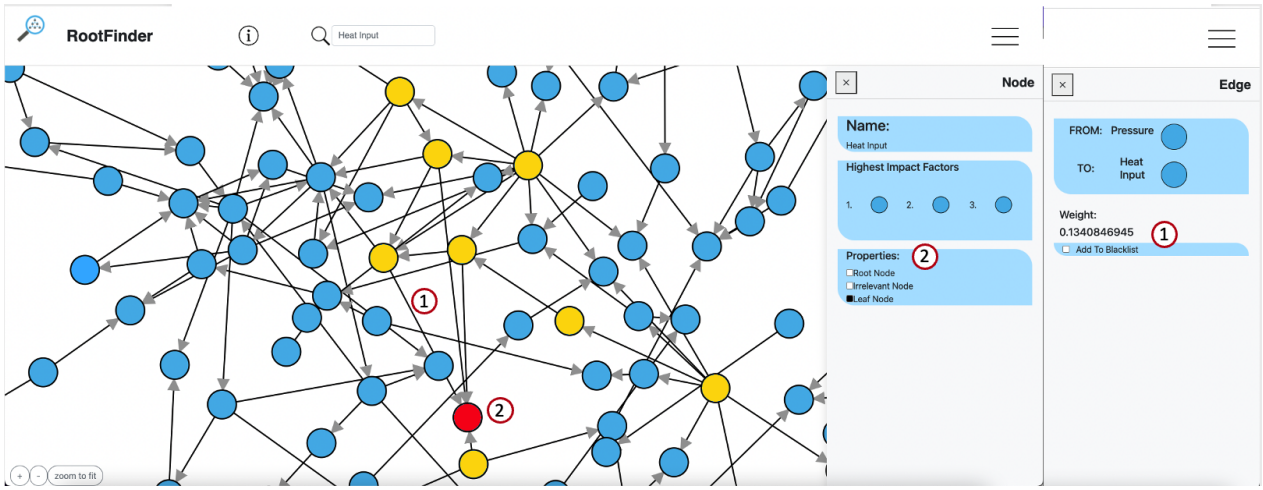

Fig. 7. The user interface of the interactive and intelligent root cause analysis tool. The fault is marked red. *Sensor Variables* part of the learned root cause path are marked yellow. The numbers in the red circle highlight the interactions.

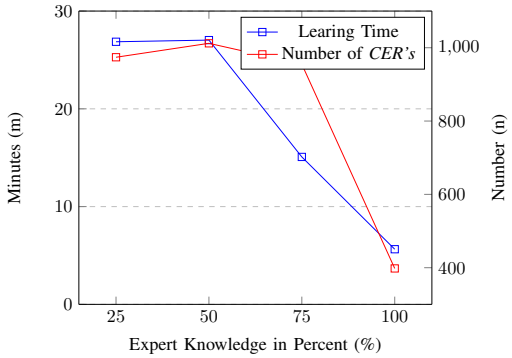

Fig. 8. Decrease of learning time and *CER's* with an increase of data from the *Knowledge Graph*.

1) Decrease in learning time of the Causal Bayesian Network by including the Knowledge Graph: As argued above, one major motivation for including expert knowledge is a decrease in the learning time of the *Causal Bayesian Network* algorithm. Figure 8 shows the average decrease in learning time with an increase in expert knowledge. The experiment was conducted five times. For each run-through, randomly 25%, 50%, 75%, and 100% of data from the *Knowledge Graph* were selected for training the *Causal Bayesian Network*. A decline by 79.0% in training time can be observed, with an increase in data from the Knowledge Graph. Thus, expert knowledge optimizes and enables the training of *Causal Bayesian Network* in large-scale manufacturing processes.

2) Decline of spurious CER's in the Causal Bayesian Network by including the Knowledge Graph: As argued above, *Causal Bayesian Networks* for manufacturing processes tend to include spurious *CER's*. Figure 8 shows the average decline in spurious CER's with an increase in expert knowledge. The experiment was conducted as described in IV-B1. A steep decline in the amount of learned *CER's* is observable. The result indicates that data from the *Knowledge Graph* prunes large numbers 60.6% of spurious *CER's*. This leads to a more accurate root cause graph and thus increases the performance of the root cause analysis tool. The following experiment shall underline the latter claim.

3) Expert feedback improves the Causal Bayesian Network: To evaluate the usefulness of the interactive component, we extend the evaluation proposed by [8]. A learned *Causal Bayesian Network* is compared to a partially known root cause graph, using an *adapted Structural Hamming Distance (aSHD)*, which describes the deviation of the learned *Causal Bayesian Network* from the partially known root cause graph [8]. A lower number indicates a better result. The learning algorithm is applied on 100 random samples of eight *Sensor Variables*, using the *temporal-spatial ordering*. For each learned *Causal Bayesian Network*, the *aSHD* is calculated. The mean and standard deviation of the *aSHD* are 0.47 and 0.74, respectively. If an expert now adds one *hasNoImpact* relation to the *Knowledge Graph*, the mean and standard deviation of the *aSHD* go down to 0.44 and 0.70. This corresponds to an improvement of the *aSHD* by almost 6% and the learning algorithm is stabilized, as the standard deviation is lower. The results illustrate the benefits of including expert knowledge and exemplify one iteration step of the interactive and intelligent root cause analysis tool.

V. Conclusion

This work proposes an interactive and intelligent root cause analysis tool for the manufacturing process of electric vehicles. It shows how to model detailed expert knowledge of a real-world manufacturing process in a large-scale *Knowledge Graph*, which is used for automatic reasoning over potential root causes. It is described how the reasoning is used to prune the search space of a *Causal Bayesian Network* learning algorithm. In addition, this work shows how to include expert feedback in the learning of the *Causal Bayesian Network* via

the *Interactive User Interface* and the *Knowledge Graph* to identify and exclude spurious *CER's*. The evaluation of the interactive and intelligent root cause analysis tool on a real-world manufacturing process of electric vehicles proves the tool's feasibility.

Machine learning makes *RCA* more efficient. However, it will only succeed if we rely on data and expert knowledge. Let us put the human back in the loop.

ACKNOWLEDGMENT

We thank the *BMW Group* for supporting the research. In particular, we thank Joscha Eirich for enabling the research and for his feedback on the intelligent and interactive root cause analysis tool.

REFERENCES

[1] C. Wehner, F. Powlesland, B. Altakrouri, and U. Schmid, "Explainable online lane change predictions on a digital twin with a layer normalized lstm and layer-wise relevance propagation," in *Advances and Trends in Artificial Intelligence. Theory and Practices in Artificial Intelligence*, H. Fujita, P. Fournier-Viger, M. Ali, and Y. Wang, Eds. Cham: Springer International Publishing, 2022, pp. 621–632.

[2] E. e Oliveira, V. L. Miguéis, and J. L. Borges, "Automatic root cause analysis in manufacturing: an overview and conceptualization," *Journal of Intelligent Manufacturing*, 2022. [Online]. Available: https://doi.org/10.1007/s10845-022-01914-3

[3] O. Serrat, *The Five Whys Technique*. Singapore: Springer Singapore, 2017, pp. 307–310.

[4] K. M. Tay and C. P. Lim, "On the use of fuzzy inference techniques in assessment models: part ii: industrial applications," *Fuzzy Optimization and Decision Making*, vol. 7, pp. 283–302, 2008. [Online]. Available: https://doi.org/10.1007/s10700-008-9037-y

[5] E. Ruijters and M. Stoelinga, "Fault tree analysis: A survey of the state-of-the-art in modeling, analysis and tools," *Computer Science Review*, vol. 15-16, pp. 29–62, 2015. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1574013715000027

[6] S. Pradhan, R. Singh, K. Kachru, and S. Narasimhamurthy, "A bayesian network based approach for root-cause-analysis in manufacturing process," in *2007 International Conference on Computational Intelligence and Security (CIS 2007)*, 2007, pp. 10–14.

[7] M. Kirchhof, K. Haas, T. Kornas, S. Thiede, M. Hirz, and C. Hermann, "Root cause analysis in lithium-ion battery production with fmea-based large-scale bayesian network," 06 2020.

[8] M. Kertel, S. Harmeling, and M. Pauly, "Learning causal graphs in manufacturing domains using structural equation models," 2022. [Online]. Available: https://arxiv.org/abs/2210.14573

[9] ISO/IEC, "Risk management — Risk assessment techniques," International Organization for Standardization, Geneva, CH, Standard, Jun. 2019.

[10] Q. Ma, H. Li, and A. Thorstenson, "A big data-driven root cause analysis system: Application of machine learning in quality problem solving," *Computers and Industrial Engineering*, vol. 160, p. 107580, 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0360835221004848

[11] A. Alaeddini and I. Dogan, "Using bayesian networks for root cause analysis in statistical process control," *Expert Systems with Applications*, vol. 38, no. 9, pp. 11 230–11 243, 2011. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0957417411003952

[12] A. Lokrantz, E. Gustavsson, and M. Jirstrand, "Root cause analysis of failures and quality deviations in manufacturing using machine learning," *Procedia CIRP*, vol. 72, pp. 1057–1062, 2018, 51st CIRP Conference on Manufacturing Systems. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2212827118303895

[13] K. Marazopoulou, R. Ghosh, P. Lade, and D. Jensen, "Causal discovery for manufacturing domains," 05 2016.

[14] J. Li and J. Shi, "Knowledge discovery from observational data for process control using causal bayesian networks," *IIE Transactions*, vol. 39, no. 6, pp. 681–690, 2007. [Online]. Available: https://doi.org/10.1080/07408170600899532

[15] L. Abele, M. Anic, T. Gutmann, J. Folmer, M. Kleinsteuber, and B. Vogel-Heuser, "Combining knowledge modeling and machine learning for alarm root cause analysis," *IFAC Proceedings Volumes*, vol. 46, no. 9, pp. 1843–1848, 2013, 7th IFAC Conference on Manufacturing Modelling, Management, and Control. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1474667016345633

[16] G. Weidl, A. Madsen, and S. Israelson, "Applications of object-oriented bayesian networks for condition monitoring, root cause analysis and decision support on operation of complex continuous processes," *Computers and Chemical Engineering*, vol. 29, no. 9, pp. 1996–2009, 2005. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S009813540500133X

[17] D. Zhang, L. Wang, Q. Hong, and K. Zhang, "Research on fault diagnosis of steam turbine based on bayesian network," *Journal of Physics: Conference Series*, vol. 1754, p. 012136, 02 2021.

[18] X. Zheng, C. Dan, B. Aragam, P. Ravikumar, and E. Xing, "Learning sparse nonparametric dags," in *International Conference on Artificial Intelligence and Statistics*. PMLR, 2020, pp. 3414–3425.

[19] P. Bühlmann, J. Peters, and J. Ernest, "CAM: Causal additive models, high-dimensional order search and penalized regression," *The Annals of Statistics*, vol. 42, no. 6, pp. 2526 – 2556, 2014. [Online]. Available: https://doi.org/10.1214/14-AOS1260

[20] G. Menegozzo, D. Dall'Alba, and P. Fiorini, "Cipcad-bench: Continuous industrial process datasets for benchmarking causal discovery methods," in *2022 IEEE 18th International Conference on Automation Science and Engineering (CASE)*. IEEE, 2022, pp. 2124–2131.

[21] A. Hogan, E. Blomqvist, M. Cochez, C. D'amato, G. Melo, C. Gutierrez, S. Kirrane, J. E. L. Gayo, R. Navigli, S. Neumaier, A.-C. N. Ngomo, A. Polleres, S. M. Rashid, A. Rula, L. Schmelzeisen, J. Sequeda, S. Staab, and A. Zimmermann, "Knowledge graphs," *ACM Computing Surveys*, vol. 54, pp. 1–37, 2021.

[22] F. Harary, R. Z. R. Z. Norman, and D. Cartwright, *Structural models: an introduction to the theory of directed graphs*. Wiley, 1965.

[23] J. Gallian, "A dynamic survey of graph labeling," *Electron. J. Combin., Dynamic Surveys*, vol. 19, 11 2000.

[24] A. Wille, P. Zimmermann, E. Vranová, A. Fürholz, O. Laule, S. Bleuler, L. Hennig, A. Prelić, P. von Rohr, L. Thiele *et al.*, "Sparse graphical gaussian modeling of the isoprenoid gene network in arabidopsis thaliana," *Genome biology*, vol. 5, no. 11, pp. 1–13, 2004.

[25] G. N. Saxe, S. Ma, L. J. Morales, I. R. Galatzer-Levy, C. Aliferis, and C. R. Marmar, "Computational causal discovery for post-traumatic stress in police officers," *Translational psychiatry*, vol. 10, no. 1, pp. 1–12, 2020.

[26] J. Peters, D. Janzing, and B. Schölkopf, *Elements of Causal Inference: Foundations and Learning Algorithms*. Cambridge, MA, USA: MIT Press, 2017.

[27] S. Shimizu, T. Inazumi, Y. Sogawa, A. Hyvärinen, Y. Kawahara, T. Washio, P. O. Hoyer, and K. Bollen, "Directlingam: A direct method for learning a linear non-gaussian structural equation model," *J. Mach. Learn. Res.*, vol. 12, no. null, p. 1225–1248, jul 2011.

[28] Y. Yu, J. Chen, T. Gao, and M. Yu, "DAG-GNN: DAG structure learning with graph neural networks," in *Proceedings of the 36th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, K. Chaudhuri and R. Salakhutdinov, Eds., vol. 97. PMLR, 09–15 Jun 2019, pp. 7154–7163. [Online]. Available: https://proceedings.mlr.press/v97/yu19a.html

[29] A. Shojaie and G. Michailidis, "Penalized likelihood methods for estimation of sparse high-dimensional directed acyclic graphs," *Biometrika*, vol. 97, no. 3, pp. 519–538, 07 2010. [Online]. Available: https://doi.org/10.1093/biomet/asq038

[30] M. Lal, *Neo4j Graph Data Modeling*. Packt Publishing, 2015.

Name: Maximilian Kertel Matrikelnummer: 243313

Erklärung

Hiermit erkläre ich, dass ich die vorliegende Dissertation mit dem Titel

"Probabilistic Graphical Models in the Manufacturing of Electric Vehicles"

selbständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel verwendet sowie die wörtlich oder inhaltlich übernommenen Stellen als solche kenntlich gemacht habe und die Satzung der Technischen Universität Dortmund zur Sicherung guter wissenschaftlicher Praxis in der jeweils gültigen Fassung beachtet habe. Ich versichere außerdem, dass ich die beigefügte Dissertation nur in diesem und keinem anderen Promotionsverfahren eingereicht habe und dass diesem Promotionsverfahren keine endgültig gescheiterten Promotionsverfahren vorausgegangen sind. Ferner erkläre ich, dass keine Aberkennung eines bereits erworbenen Doktorgrades vorliegt. Ich versichere an Eides statt, dass ich nach bestem Wissen die reine Wahrheit erklärt und nichts verschwiegen habe.

München, den .....

Maximilian Kertel