
# Causal discovery from observational tabular data with generative adversarial learning

PhD Thesis

Hristo Petkov
Computer and Information Sciences
University of Strathclyde, Glasgow

November 20, 2025

This thesis is the result of the author's original research. It has been composed by the author and has not been previously submitted for examination which has led to the award of a degree.



Signed: Hristo Petkov

Date: November 20, 2025

# Abstract


### Background

Causal knowledge is essential for understanding complex systems and revealing relationships between variables. It enables researchers to transition beyond correlations, reason about cause and effect, and derive scientific insights. Although Randomized Controlled Trials (RCT) remain the gold standard for causal inference, they are often infeasible due to ethical, logistical, or financial constraints and may lack real-world applicability. In contrast, observational data offer abundant, diverse samples, making them well-suited for large-scale analysis. Despite susceptibility to confounding, advances in structure learning from observations allow researchers to identify causal relationships without relying on randomized experiments.

### Research objectives

This thesis challenges conventional maximum likelihood estimation (MLE)-based methods by exploring adversarial causal discovery approaches. It leverages the Wasserstein Generative Adversarial Network with Gradient Penalty (WGAN-GP) framework to address key limitations: (1) model overfitting from simplistic loss functions; (2) dependence on single parametric assumptions that hinder accurate causal graph recovery reflective of true data relationships; (3) high computational cost from Augmented Lagrangian optimization in the NOTEARS framework; and (4) inability to perform causal discovery and tabular data synthesis simultaneously under a single framework.

## Methods

Three models were developed using the WGAN-GP framework. The first, DAG-WGAN integrates WGAN-GP with variational inference, leveraging hybrid losses for improved causal modeling. The second, DAG-WGAN+ enhances continuous optimization with efficient structure learning techniques. The third, DAGAF captures variable interdependencies under various causal assumptions to generate synthetic data preserving causal relations.

## Results

All models target multivariate causal discovery and were rigorously evaluated using Structural Hamming Distance (SHD). Results show they outperform leading methods in causal discovery across 97.47% of all test cases. In real-world experiments, the proposed models achieve superior accuracy (SHD = 8 vs. $>$ 10 for state-of-the-art models). Findings further reveal that precise causal modeling enhances synthetic data quality by preserving underlying causal mechanisms.

# Contents

# List of Figures

List of Figures

# List of Tables

# Preface/Acknowledgements

I would like to acknowledge the Department of Science at the University of Strathclyde for accommodating me during my Ph.D. studies. I also acknowledge my first supervisor Prof. Feng Dong and my second supervisor Prof. Roma Maguire for guiding me through the process of conducting research and writing up my thesis.

I also greatly appreciate my parents for their moral and financial support during this journey. Additionally, I recognize my cousin Slav Ivanov and my friend Colin Hanley as sources of inspiration and my colleague Calum MacLellan for their data analysis notebook, without them some of the experimental results described in my thesis would not be possible.

Last but not least, I would like to thank our lord and savior Jesus Christ for giving me the strength to turn my dreams into reality.

# List of Abbreviations & Notation

This section contains a list of abbreviations alongside the mathematical notation used throughout this work. The glossary style is adapted from [1].

**List of Abbreviations**

| | |
|---|---|
| ACD | Amortized Causal Discovery |
| AE | Auto-Encoder(s) |
| AI | Artificial Intelligence |
| ANM | Additive Noise Model |
| ANN | Artificial Neural Network(s) |
| AOC | Area Over Curve |
| AUC | Area Under Curve |
| BDe(u) | Bayesian Dirichlet equivalence |
| BGe | Bayesian Gaussian equivalent |
| BIC | Bayesian Information Criterion |
| BN | Bayesian Network(s) |
| CAN | Causal Adversarial Network(s) |

| | |
|---|---|
| CausalVAE | Causal Variational AutoEncoder(s) |
| CBM | Constrained-based Method(s) |
| CEL | Cross-Entropy Loss |
| CMGAN | Generative Adversarial Network(s) embedded with Causal Matrix |
| CO | Continuous Optimization |
| CPDAG | Completed Partially Directed Acyclic Graph(s) |
| DAG | Directed Acyclic Graph(s) |
| DAGAF | Directed Acyclic Generative Adversarial Framework |
| DEAR | Disentangled gEnerative cAusal Representation learning |
| DGM | Deep Generative Models |
| DRL | Disentangled Representation Learning |
| ELBO | Evidence Lower BOund |
| EMD | Earth Mover Distance |
| ER | Erdos-Renyi |
| ESL | Efficient Structure Learning |
| FDR | False Discovery Rate |
| FPR | False Positive Rate |
| GAN | Generative Adversarial Network(s) |
| GES | Greedy Equivalence Search |
| GLM | Generalized Linear Model(s) |

| | |
|---|---|
| i.f.f | if and only if |
| i.i.d. | independent and identically distributed |
| ICL | Imputated Causal Discovery |
| JSD | Jensen-Shanon Divergence |
| KLD | Kullback-Lieber Divergence |
| LiNGAM | Linear Non-Gaussian Model |
| MDL | Minimum Description Length |
| MEC | Markov Equivalence Class(es) |
| MLE | Maximum Likelihood Estimation/Estimates |
| MLP | Multi-Layer Perceptron |
| MMD | Maximum Mean Discrepancy |
| MMPC | Min-Max Parent and Children |
| MSE | Mean Squared Error |
| NLL | Negative Log-Likelihood |
| NOTEARS | Non-combinatorial Optimization via Trace Exponential and Augmented lagRangian for Structure learning |
| NP | Non-polynomial |
| PNL | Post-Nonlinear Model |
| RCT | Randomized Control Trial(s) |
| SAM | Structural Agnostic Modelling |
| SBM | Score-based Method(s) |
| SCM | Structural Causal Model |

| | |
|---|---|
| SEM | Structural Equation Model(s) |
| SGD | Stochastic Gradient Decent |
| SHD | Structural Hamming Distance |
| SID | Structural Interventional Distance |
| STD | STandard Deviation |
| TPR | True Positive Rate |
| VAE | Variational AutoEncoder(s) |
| WACD | Wasserstein Adversarial Causal Discovery |
| WD | Wasserstein Distance |
| WGAN | Wasserstein Generative Adversarial Network |
| WGAN-GP | Wasserstein Generative Adversarial Network(s) with Gradient Penalty |

**Mathematical Notation**

| | |
|---|---|
| $(i \rightarrow j) \in E$ | A directed edge |
| $\chi$ | A tabular dataset |
| $\hat{\mathbf{X}}$ | Output of the WGAN-GP architecture |
| $\mathbb{D}$ | Directed acyclic graph search space |
| $\mathbf{N}$ | Number of objects in a set |
| $\mathbf{X} = \{\mathbf{X}_1, ..., \mathbf{X}_n\}$ | A set of input samples |
| $\mathbf{X}_i$ | The $i^{th}$ input sample |

| | |
|---|---|
| $\mathcal{A} = \{\mathcal{A}_1, ..., \mathcal{A}_\mathbf{N}\}$ | A set of assumptions |
| $\mathcal{F} = \{f_1, ..., f_d\}$ | A set of causal mechanisms |
| $\mathcal{G} = \{g_1, ..., g_d\}$ | A set of post-nonlinear functions |
| $\mathcal{M}$ or $\mathcal{M}\langle X, \mathcal{Z}, \mathcal{F}\rangle$ | A Structural Causal Model consisting of a set of input variables $X = \{X_1, ..., X_d\}$, external variables $\mathcal{Z} = \{\mathcal{Z}_1, ..., \mathcal{Z}_d\}$ and a set of causal mechanisms $\mathcal{F} = \{f_1, ..., f_d\}$ |
| $\mathcal{O}(.)$ | Big-O notation used for computational complexity analysis. |
| $\mathcal{Z} = \{\mathcal{Z}_1, ..., \mathcal{Z}_d\}$ | A set of external variables |
| $\mathcal{Z}_j$ | The $j^{th}$ external variable |
| $\mathbf{A}$ or $\mathbf{A} \in \mathbb{R}^{v \times v}$ | A (weighted) adjacency matrix representing the structure of the underlying ground truth graph |
| $\mathbf{G_A^0}$ | The underlying ground truth graph |
| $\mathbf{G_A}$ | A causal graph |
| $\mathbf{G}$ or $\mathbf{G}\langle V, E\rangle$ | A directed acyclic graph consisting of $V$ vertices and $E$ edges |
| $\mathbf{Z} = \{\mathbf{Z}_1, ..., \mathbf{Z}_n\}$ | A set of noise vectors |
| $\mathbf{Z}_i$ | The $i^{th}$ noise vector |
| $\mu$ | Mean of a probability distribution |
| $\sigma$ | Standard deviation of a probability distribution |
| $\tilde{\chi}$ | A simulated tabular dataset |
| $\tilde{\mathbf{X}}$ | Reconstructed data |
| $\{\mathcal{G}_{\mathbf{A}_{init}}\}$ | Equivalent DAG search space |

| | |
|---|---|
| $BN = (\mathbf{G}, \varphi)$ | A Bayesian Network consisting of a directed acyclic graph $\mathbf{G}$ and parameters $\varphi$ |
| $c$ | Number of categories present in the input data |
| $D$ | The discriminator from the GAN framework |
| $d$ | Number of variables in the input data |
| $E = \{(i, j) \in \mathbb{R}^{v \times v}\}$ | A set of edges |
| $f_j$ | The $j^{th}$ causal mechanism |
| $G$ | The generator model from the GAN framework |
| $g_j$ | The $j^{th}$ post-nonlinear function |
| $k$ | Number of iterations |
| $MEC = \{\mathbf{G}_1, ..., \mathbf{G_N}\}$ | A Markov Equivalence Class |
| $n$ | Size of a input data set |
| $P(\hat{\mathbf{X}})$ | The implicitly defined probability distribution from the WGAN-GP architecture |
| $P(\mathbf{X})$ | Observational data distribution |
| $P(\mathcal{Z})$ | Probability distribution of external variables |
| $P(\mathbf{Z})$ | Probability distribution of noise vectors |
| $P(\tilde{\mathbf{X}})$ | Reconstructed data distribution |
| $P(Z)$ | Prior distribution of the latent variable $Z = \{Z_1, ..., Z_{\mathbf{N}}\}$ |
| $P_{\mathbf{G}^0_\mathbf{A}}(\mathbf{X})$ | An alternative representation of $P(\mathbf{X})$ involving causal information |
| $P_{\mathbf{G_A}}(\tilde{\mathbf{X}})$ | An alternative representation of $P(\tilde{\mathbf{X}})$ involving causal information |

| | |
|---|---|
| $v$ | Number of vertices |
| $V = \{V_1, ..., V_{\mathbf{N}}\}$ | A set of vertices |
| $X = \{X_1, ..., X_d\}$ | A set of data variables |
| $X_j$ | The $j^{th}$ input variable |
| $Z = \{Z_1, ..., Z_{\mathbf{N}}\}$ | A set of latent variables |
| $Z_i$ | The $i^{th}$ latent variable |
| $Pa_j$ | Parents of an input variable |

# Chapter 1

# Introduction

Causality is a fundamental property that shapes our view of reality. It is based on the idea of cause and effect, which was first suggested by the ancient Greek philosopher Aristotle in their works; *"Metaphysics"* [2] and *"Posterior Analytics"* [3]. Since then, causality has been closely linked to science, allowing people to gain insight into complex concepts such as the Universe and Life itself.

Throughout the centuries, people have developed their own interpretations of causality to explain its effects in different fields. For instance, the work of Galileo Galilei in physics highlighted the distinction between observing the causal relationships within a system and studying them by manipulating the parameters that influence its behavior. This concept is referred to as intervention, and its introduction to causation enabled Galileo to change the way people view the Universe.

In the field of medicine, Sir Austin Bradford Hill recognized that exploring the influence of causal connections could provide valuable insight into diseases, their treatments, and their outcomes. In his paper *"The Environment and Disease: Association or Causation?"* [4], Hill proposed a set of criteria (Plausibility, Consistency, Temporality, Strength, and Specificity) as a guide to examine causality in epidemiological studies. By utilizing these criteria in randomized control trials (RCT) [5], Hill concluded that cigarette smoking was one of the leading causes of lung cancer and investigated the effects of streptomycin as a form of treatment for tuberculosis. Although his criteria are somewhat outdated nowadays, they are still considered essential for investigating the

effect of causation in various research fields, including Criminology [6], Economics [7], Psychology [8] and Marketing [9].

In recent decades, the increasing use of computers and their capacity to store data have shifted the study and application of causality toward a more digital direction. This is mainly due to the accumulation of large datasets and their complexity, making it difficult for people to comprehend the connections between the data. As a result, a new scientific field known as Causal Structure Learning was established, which seeks to learn the causal relationships within data through interventions or observations. Pioneers such as Judea Pearl [10], Peter Spirtes [11] and Xun Zheng [12] have enabled us to transition from manually searching for causal relationships between variables or using rule-based algorithms, to discovering them by applying advanced machine learning techniques for modeling the dependencies in data. At present, causal structure learning is still an active area of research, with people studying the effects of causation in fields such as Computer Science, Data Analytics, Medicine, and Physics, further emphasizing its importance.

## 1.1 Why Causation?

Humans have a natural inclination to discover how things relate to each other. Our inherent curiosity drives us to ask the essential question: "*How everything works?*". A crucial stepping stone on our collective path to answering this question is gaining the knowledge of how all is connected. Our desire to understand the world around us emphasizes the importance of causality (determined by observation, reasoning, and experimentation) in our thought process.

We can use our capacity for reasoning to recognize causal relationships by simply observing our surroundings or by actively engaging with them. Examples of this are: 1) Observing the influence of the rotation of Earth on the day/night cycle. 2) Experiencing the pain and discomfort from touching a hot stove. These are accepted as being accurate based on observation and experimentation. However, not all causal statements are necessarily accurate. It is possible to make connections that are partially wrong or

completely unjustified. This is usually due to our limited rationality combined with our ability to correlate observations.

Correlations are a way of connecting different events or activities. They are seen as high-level patterns in data or decisions made based on past experiences. Although statistical dependencies can be used to make inferences about cause and effect, it is important to remember that correlations do not necessarily imply causation [13], [14]. Nevertheless, people often mistake them for causal relationships because of their inability to distinguish between the two.

Every causal statement consists of two components: confounding and causal association. The former is a shared cause between two or more variables, creating an indirect relationship, while the latter establishes a direct link between two variables. Correlations are unable to express 'confounding', but they do contribute to causal associations, which is why they are often mistaken for causal statements. This issue is even more noticeable when people are presented with a large amount of data.

The misidentification of statistical patterns as connections between variables can lead to a false understanding of the causality in a dataset. This discrepancy between the actual causal structure of the data and the one suggested by its statistical dependencies manifests itself in the form of contradictions, such as *Simpson's paradox* [15], [16]. This paradox occurs when confounders are present, making it impossible to determine causal relationships from correlations. The presence of such paradoxes in datasets creates problems that can only be solved by studying and understanding the causation between data variables. As the number of such datasets increases, it becomes increasingly important to analyze the data from a causal perspective.

Causal inference not only plays an essential role in studying datasets but also has significant implications for our daily lives. It enables people to gain key insight into the connections between aspects within specific scenarios. For that reason, whether we are engaged in a task, problem-solving or decision-making, people prefer to substantiate their reasoning with logical arguments based on causal relationships. To elaborate further on the intuition behind causation, the following examples are presented: 3) a data analyst studying how the values of different columns affect the rest of the data

in a dataset; 4) a medical professional assessing the effects of various treatments on a disease.

These examples demonstrate the ability of causal relationships to express different ideas. In the third example, causality provides an *explanation* of how the data is connected. In the fourth scenario, causation allows the physician to *objectively* determine which treatment will result in the most favorable outcome. Although the way in which causal inference is expressed may vary, its influence remains the same. It enables people to infer the effect of various factors (e.g. treatment, policy, intervention, action, or decision) on a potential outcome by examining how they affect it.

Unfortunately, performing causal inference is challenging because of our inability to generalize the search space that contains the relationships between variables. Hence, people always rely on assumptions when conducting causal studies. They enable us to control the number of possible connections by specifying a scenario with a corresponding set of circumstances, thus significantly limiting the formulation of causal statements. Moreover, assumptions are essential for causal inference, as they facilitate the discovery of path diagrams through observations or interventions. Despite the two aforementioned approaches using different sets of assumptions, they both attempt to retrieve the relationships exhibited within data through a procedure called causal structure learning.

## 1.2 Causal Structure Learning

Research in the field of causal structure learning focuses on discovering the causal mechanisms within a dataset. Most contributions to the field involve computing a graphical representation through interventions or observations that best describes the causal relationships in the data. Interventions are considered the gold standard for causal structure learning, with most research in this direction conducted by actively manipulating one or multiple variables in randomized control trials [17]. However, such experiments can be difficult to set up due to ethical, feasibility, and cost issues. Therefore, algorithms have been developed to directly retrieve causal relationships from

data, making observational studies more significant [18], [19]. This section provides a brief overview of the history of causal structure learning from observational data; for a more detailed discussion, see Section 2.2.

Studying the causal connections between different data variables within a dataset can benefit multiple research domains and contribute to scientific knowledge. Particularly in structure learning, Bayesian Networks (BN) are a useful tool for discovering causal relationships from observational data. Represented as Directed Acyclic Graphs (DAG), they can be used to infer causality in complex systems. Their unique structure allows people to describe how the contents of a dataset are related, leading to a deeper understanding of diverse fields such as medicine, justice or physics. More importantly, BN have many applications in machine learning, as they can model the conditional dependencies between variables while being easily interpretable and computationally tractable. Noteworthy examples include "*Finding Optimal Models for Small Gene Networks*" [20] and *Causal Protein-Signaling Networks Derived from Multiparameter Single-Cell Data* [21].

The challenge of causal structure learning lies in the vast number of potential DAG that form the search space. As the number of variables increases, the DAG search space expands exponentially, rendering attempts of combinatorial nature computationally intractable. To address this NP-hard problem [22], various approaches have been developed, such as traditional score-based, constraint-based and hybrid methods, as well as machine learning techniques like continuous optimization and efficient structure learning.

Traditional methods for learning DAG structures rely on independence tests to limit the search space [23], [11], [24], or perform discrete score-based searches to identify the DAG that best fits the input data [25], [26], [27]. Both of these techniques have their drawbacks. Constraint-based methods (CBM), such as the PC [28] and FCI [29] algorithms, produce graphs that satisfy a set of conditional independencies, not a learned causal structure, and thus the output of these methods is often incomplete. Furthermore, these models are not robust to significant changes in the size of the data variables [30]. Score-based methods (SBM), such as LiNGAM [31], PNL [32] and

CSM [33], attempt to discretely optimize a score function to find the best DAG, but additional structure assumptions and approximate searches are often necessary due to the complexity of the search space, which remains super-exponential.

Researchers have attempted to address the shortcomings of CBM and SBM by combining them into hybrid approaches. These methods have shown great promise as they simultaneously reduce the graph search space and optimize a score function, thus learning a DAG. A well-known example of this is the MMHC algorithm by [34], which uses Min-Max Parent and Children (MMPC) to limit the graph search space and optimizes the Hill-Climbing score function to compute a DAG.

Traditional causal discovery methods, though effective in earlier decades, have become increasingly impractical, as their discrete search over DAG structures leads to combinatorial intractability as the number of variables grows. To address this challenge, the problem has been reformulated as a continuous optimization task, representing graphs as weighted adjacency matrices, and acyclicity being enforced via differentiable constraints that enable gradient-based optimization. Black-box models, particularly neural networks, facilitate this approach by efficiently capturing complex, non-linear relationships in large, high-dimensional datasets, while their differentiability allows for end-to-end optimization of both structure and acyclicity constraints. This paradigm shift effectively addresses the computational and scalability limitations inherent in traditional methods.

At present, the process of causal structure learning is performed using machine learning models, made possible by the contributions of [12]. Their DAG-NOTEARS framework has revolutionized the way causal structures are discovered by transforming the problem from a combinatorial one to a continuous optimization approach that can be solved with black-box models. This has enabled the development of multiple machine learning models that can handle non-linear, continuous and discrete data [35], [36], [37], [38], [39]. For linear cases, GOLEM [40] outperforms NOTEARS, while models such as AbPNL [41] and Deep PNL [42] assume post-nonlinear models. Meanwhile, RL-BIC [43] uses reinforcement learning to learn causal graphs from data. An alternative approach is the continuous optimization framework developed by [44],

which is a bi-level optimization algorithm that discovers causal relationships by optimizing the Permutahedron of permutation vectors to learn the order of the nodes in a graph. CASPER [45] is another recently developed continuous optimization framework that addresses the shortcomings of DAG-ness independent score-based methods by introducing a new dynamic search space solved through a novel score function with integrated graph structures, leading to the discovery of optimal DAG. The latest work in the causal structure learning field is called REX [46]. The developers of this model proposed a causal discovery method that integrates machine learning (ML) models with explainability methods (based on Shapley values) to identify and interpret significant causal relationships among variables.

Most of these machine learning models are successful in optimizing a score function and imposing an acyclicity constraint. To achieve this, an augmented Lagrangian [47] is used for continuous optimization, which enables the simultaneous optimization of parameters and causal structure computation. Unfortunately, this continuous optimization process is very time-consuming, making it inefficient.

Recently, researchers have explored novel ways to learn causality more efficiently. One such method is DAG-NoCurl by [48], which is one of the first frameworks to do so effectively. Its improved time-wise performance is achieved without the need for an augmented Lagrangian or any explicit DAG constraints, thus eliminating the need for explicit parameter optimization as causal structures are learned implicitly directly from the DAG search space. Instead, constant hyper-parameters are used throughout the learning process. A hyper-parameter study was also conducted to determine a baseline of values that produce good results.

The success of DAG-NoCurl led to the development of more efficient frameworks based on novel mathematical formulations. An example of this is VI-DP-DAG [49], which uses a DAG sampling technique based on posterior distributions over edges and node permutations. Variational inference is used to minimize the gap between the prior distributions of the observational data samples and the posterior distributions, resulting in a quick and precise computation of the causal structure. Most recently, an advancement of the DAG-NoCurl framework, referred to as DAG-NCMLP [50], has

been introduced to address the constraint of its predecessor to linear causal models. DAG-NCMLP achieves this by applying a non-linear projection to an initial cyclic graph estimate, effectively mapping it into the equivalent DAG search space characterized by the original DAG-NoCurl formulation. Despite the advances made by DAG-NoCurl, DAG-NCMLP and VI-DP-DAG, efficient structure learning is still an ongoing area of research.

## 1.3 Motivations

In the last decade, there has been a steady increase in the influence of machine learning on businesses and the industry in general. A prime example of this is the development of the so-called "generative models", including Flow-based generative models [51], Variational Autoencoders (VAE) [52], Generative Adversarial Neural Network (GAN) [53] and most recently diffusion models [54], which have revolutionized the way models learn features and produce new data samples.

Today, many machine learning applications (e.g. data generation, image classification, and stock forecasting) are predicting outcomes based on features and statistical correlations rather than using causality to develop a deeper understanding, enabling them to deduce the correct result. This leads to issues like over-fitting, lack of explainability, and inability to generalize, all of which prevent us from trusting in the decision-making capabilities of machine learning applications.

Developing algorithms that can learn from both statistical patterns and causal relationships will result in better generalization and faster convergence, making them more efficient. Practically speaking, such approaches can help to resolve the trust issues between humans and artificial intelligence by providing an explanation for the produced output, which will be beneficial in areas such as healthcare, medicine and law, where making the right decision is often not so clear-cut for humans. Furthermore, these models could help us to discover new solutions to existing problems by introducing novel ideas or ways of thinking that we have not yet considered.

From a generative model standpoint, the decision to explore causal structure learning through generative models was based on the development of Variational Autoen-

coder (VAE) [52] architectures to recover causal relationships from data. Examples of such models include DAG-GNN [35], GAE [39], and DAG-GNN + NoCurl [48], which was created by the authors of the DAG-NoCurl paper. All of these models use maximum likelihood estimation (MLE) [55] to retrieve the causality present in the data samples. However, these algorithms are prone to mean-seeking mode [56], which occurs when models capture all data modes by approximating the likelihood of samples across the entire training space. This results in an overly precise reconstruction process that produces average samples and filters out most outliers. Consequently, the causal mechanisms responsible for those outliers are lost, leading to inaccuracies in causal structure learning and resulting in overfitting to the input data.

To address the limitations of generative models based on Maximum Likelihood Estimation (MLE), researchers have proposed the use of additional loss terms, including the Wasserstein distance (WD) [57], the Kullback-Leibler divergence (KLD) [58], and the Jensen Shannon divergence (JSD) [59], to penalize such approaches for learning unreasonable causal structures and synthesizing unrealistic data samples [60]. KLD and JSD are often combined with an MLE-based reconstruction loss to form an Evidence Lower Bound (ELBO) [61], which is a popular way to train Variational Autoencoders (VAE). However, there is very little research conducted regarding the influence of the Wasserstein distance on Variational Autoencoders. In particular, VAE-GAN [62] has investigated the effects of combining ELBO with adversarial loss. Despite the fact that the model produces good results, the impact of adversarial loss on the VAE architecture remains largely unexplored. The same can be said for the application of the Wasserstein distance in the area of causal structure learning.

The effects of adversarial loss in the context of causal learning can be studied from two perspectives. Firstly, including the Wasserstein distance as an additional loss term to the ELBO of VAE architectures can help to improve their accuracy in learning causal structures. Secondly, the application of Wasserstein-1 in the context of causal structure learning can be used to generate synthetic samples from learned causal structures. Currently, there are only a few models that can discover causality from tabular data using a generative adversarial network framework, such as DAG-

GAN [37], SAM [63] and MCS [64]. Hence, putting more emphasis on studying this relatively new approach will stimulate not only the development of new models but also the writing and proliferation of scientific literature. This thesis explores both perspectives and discusses their related findings.

Furthermore, from a causal perspective, most of the models developed for causal discovery assume that the underlying structural causal model (SCM) used to generate the input data is the additive noise model (ANM). This is a reasonable assumption because ANM are identifiable and there exist many approaches satisfying it, which enables method comparison. However, causal discovery is not limited to the ANM. Another identifiable model, which is still largely unexplored compared to ANM, is the Post-Nonlinear Model (PNL) [32]. In causal structure learning, there are very few models working with PNL. Among them are AbPNL [41], Deep PNL [42], [65] and [66]. None of these models applies adversarial learning in the context of causal discovery from PNL. Therefore, for a more complete study of the effect of adversarial training in causal discovery, every model developed in support of this thesis is applied to both the ANM and the PNL.

Last but not least, from a human-centric and knowledge-based standpoint, tabular data remains one of the most prevalent and versatile formats for organizing information, serving as a cornerstone for analysis and decision-making in fields such as medicine, finance, and business. Nevertheless, challenges such as incompleteness and poor data quality often undermine the reliability of insights drawn from it. To mitigate these issues, Deep Generative Models (DGM) have emerged as powerful tools for data synthesis and imputation, aiming to capture the underlying statistical distributions of real data to enhance fidelity and diversity. However, while traditional DGM excel at modeling correlations, they often lack interpretability and transparency, which are considered key qualities for trustworthy data-driven decision-making. Causally aware generative approaches have sought to overcome this by modeling the underlying cause-and-effect relationships within data, producing more realistic and explainable synthetic samples. Unfortunately, these methods still face significant challenges, including oversimplified latent representations, dependency on prior causal knowledge, and high computational

demands, which limit their practical applicability. These shortcomings emphasize the necessity for a unified framework that integrates causal discovery with tabular data synthesis. Such an approach holds the potential to advance the generative capabilities of DGM to produce realistic, diverse, and explainable synthetic data, thus bridging the trust-issue gap between machine learning and human beings.

## 1.4 Thesis Statement

This thesis studies the potential of Generative Adversarial Networks (GAN) in the context of Causal Structure Learning. To this end, several causal discovery frameworks have been developed under the WGAN-GP setting, resulting in multiple publications - see Section 1.7. The models, namely DAG-WGAN, DAG-WGAN+ and DAGAF, were evaluated against the current state-of-the-art, and were found to outperform them in multiple cases.

The objective of the research conducted by the author is to investigate whether the application of the Wasserstein distance-based adversarial loss can contribute to the solution of some of the most critical challenges in modern causal structure learning. Specifically, DAG-WGAN was developed to mitigate the weaknesses of conventional MLE-based loss functions to induce model overfitting, which reduces the generality of the causal structure learning process. DAGAF was developed as a proof-of-concept algorithm capable of limiting the reliance on single parametric assumptions that restrict the capacity to recover causal graphs that faithfully represent the true data-generating process. This algorithm also provided a solution to the inability of existing methods to simultaneously perform causal discovery and synthesize tabular data within a unified framework.

Experiments have been conducted in both parametric and non-parametric settings, and the impact of adversarial training and kernel-based disentangled representation learning with Maximum Mean Discrepancy (MMD) [67] during MLE-based parameter optimization has been thoroughly analyzed. Additionally, approaches for efficient structure learning have been explored to address the slow computation of outputs due to the Augmented Lagrangian and kernels having poor time complexities, facilitated

by the DAG-WGAN+ model. All models discussed in this work are implemented using Pytorch [68].

## 1.5 Research Methodology

This study employs a generative modeling framework to infer causal structures represented as Directed Acyclic Graphs (DAG) from observational data. Three progressively enhanced models are developed: DAG-WGAN, DAG-WGAN+, and DAGAF. Each model extends the capabilities of its predecessor by improving training stability, causal structure learning efficiency, and modeling flexibility. The overall methodological design integrates principles from causal discovery, generative adversarial networks (GAN), and probabilistic modeling to achieve accurate and interpretable causal graph estimation.

To systematically investigate and evaluate these proposed models, the author provides a research methodology section structured around three key components: (i) the rationale behind the chosen methods, which explains the theoretical and empirical basis for the selected architectures; (ii) the overall experimental framework, which outlines the datasets and evaluation metrics employed; and (iii) the analytical framework, which describes the comparative evaluation strategy, ablation analyses, and validation of results. Together, these elements form a coherent and rigorous approach to assessing the effectiveness, robustness, and reliability of the proposed causal discovery frameworks.

### 1.5.1 Rationale behind the implemented methods

The development of the DAG-WGAN model stems from an effort to push beyond the limitations of existing VAE-based methods for nonlinear causal discovery. DAG-GNN, one of the earliest and most influential models in this domain, demonstrated that combining machine learning with variational inference could effectively discover causal structures, establishing the Variational Autoencoder (VAE) as a leading framework for such tasks. However, its reliance on VAE formulations raised questions about whether challenges such as independent data-point optimization, latent collapse, and

scalability might induce deeper representational constraints. To mitigate this possibility, DAG-WGAN introduces a hybrid VAE–GAN architecture that incorporates adversarial training through the Wasserstein distance, which is a more stable and expressive measure of distributional difference. This integration is intended to enhance generative quality, stabilize training, and enable richer representations of complex data, potentially opening new directions for more flexible and robust causal inference.

The DAG-WGAN+ model builds upon the foundation established by DAG-NoCurl, which successfully integrated the DAG-GNN architecture with a curl-free constraint to enhance structural accuracy and convergence, resulting in a variant known as DAG-GNN with NoCurl. Inspired by this demonstrated synergy, the present study investigates the potential benefits of combining the efficient, acyclicity-preserving equivalence DAG formulation introduced in DAG-NoCurl with the hybrid VAE–GAN framework of DAG-WGAN. This adaptation is intended to evaluate how embedding the DAG-NoCurl framework within an adversarial causal discovery model influences both structural fidelity, training robustness and efficiency. Furthermore, by incorporating principles of disentangled representation learning, DAG-WGAN+ is designed to separate independent sources of variation within the latent space, aligning individual latent dimensions with distinct causal factors. This disentanglement is expected to enhance the interpretability and refinement of the inferred causal graph, facilitating more reliable identification of genuine causal relationships while mitigating the effects of spurious correlations.

Last but not least, the DAGAF model extends the DAG-Notears-MLP framework with the aim of exploring how its non-parametric architecture and demonstrated capacity to capture multiple identifiable causal models, such as Additive Noise Models (ANM), Post-Nonlinear (PNL) models, and Linear Non-Gaussian Acyclic Models (LiNGAM), can be further enhanced through adversarial and transfer learning techniques. Building on these strengths, the model investigates whether combining explicit likelihood estimation with distributional modeling and causally aware data generation can bridge the gap between interpretability and expressive power. To this end, DAGAF integrates the interpretability and identifiability of DAG-Notears-MLP with the genera-

tive flexibility of adversarial models. The author speculates that this fusion can improve scalability to nonlinear, high-dimensional, and non-Gaussian data modes. Moreover, the framework enables simultaneous causal discovery and tabular data synthesis within a unified structure. By incorporating a separate instance of DAG-Notears-MLP as a generator to produce realistic synthetic datasets consistent with inferred causal structures, DAGAF seeks to examine how aligning structural learning with data generation can be achieved under a single training algorithm. Through this integration, DAGAF is positioned as an exploratory step toward interpretable and data-faithful generative causal inference, with the potential to advance high-quality, diverse tabular data synthesis.

### 1.5.2 Overall experimental framework

The experimental framework described in this study follows the general principles established in recent DAG-based generative adversarial causal discovery methods, namely DAG-WGAN, DAG-WGAN+, and DAGAF. These approaches share a common experimental philosophy: evaluating both the accuracy of the learned causal structure and the generative fidelity of the corresponding data model under controlled and real-world conditions.

Experiments are conducted using a combination of synthetic and real-world datasets. Synthetic data allow for quantitative evaluation since the true causal graph is known. For these experiments, directed acyclic graphs (DAG) of varying sizes and densities are generated, and data are simulated from diverse functional mechanisms, ranging from linear to nonlinear and post-nonlinear relationships, to test the ability of each model to recover causal dependencies under different structural complexities. Essentially, this results in tabular datasets, where each column represents a data variable and each row is generated data with each cell being a manifestation of a causal mechanism and a noise vector. Furthermore, real-world and benchmark datasets, commonly used in causal discovery research, are also employed to assess practical applicability and generalization performance.

It is also important to note that unlike traditional machine learning, where data

is often split into training and validation sets, this practice is less common in causal structure learning. Although train-test splitting or cross-validation is standard in predictive modeling, causal structure identification prioritizes structural constraints and conditional independencies over predictive accuracy. Since causal relationships are inherently structural and assumed to hold across the entire dataset, partitioning the data typically offers little added value in discovering the underlying structure.

Each model is evaluated against established causal structure learning baselines (i.e., state-of-the-art methods including DAG-Notears, DAG-GNN, GraN-DAG, DAG-Notears-MLP, GAE, etc.), ensuring that comparisons are both fair and comprehensive. That being said, the frameworks differ in how they enforce the acyclicity constraint and how adversarial training is used to align the generated and observed data distributions. DAG-WGAN employs a Wasserstein-based adversarial training strategy coupled with a differentiable acyclicity regularizer; DAG-WGAN+ introduces the DAG-NoCurl formulation to improve computational efficiency and stability; and DAGAF extends the adversarial framework to jointly learn causal structures and synthesize realistic tabular data under multiple functional assumptions.

Nevertheless, commonalities between all three approaches include the two criteria used to assess model performance. The first focuses on causal accuracy, quantified by how closely the learned graph approximates the true causal structure. The metric chosen for this evaluation was the Structural Hamming Distance (SHD) because it integrates several important measures, such as True Positive Rate (TPR), False Discovery Rate (FDR), and False Positive Rate (FPR). The second concerns data fidelity, measured through the similarity between the original and generated data distributions. It is common practice to evaluate generated sample fidelity and diversity using multiple different components of data and distribution analysis including: 1) heat maps to visualize the learned causal structure; 2) box plots to assess feature importance quality for regression or classification tasks; 3) correlation matrices to compare the learned correlations to the original ones; and 4) distribution visualization to determine the diversity of the generated samples by investigating how well the generated and the original distributions overlap. The author utilizes all of the above in their experiments.

Together, these evaluations provide a balanced investigation of both causal discovery and generative capability.

All synthetic data experiments are repeated across multiple random initializations to ensure robustness, and performance is reported in aggregate to mitigate stochastic variability. This framework enables systematic comparison across model variants while maintaining consistency in data generation validation and causal structure learning evaluation. The results are available in Sections 3.2.3, 4.2 and 5.2.

### 1.5.3 Analytical framework

This study adopts an analytical framework that systematically examines the efficacy of adversarial generative models in learning causal structures and synthesizing realistic tabular data. By leveraging directed acyclic graph (DAG)-based formulations within adversarial learning paradigms, the framework integrates causal discovery and data generation into a unified evaluation process. It focuses on assessing how different optimization strategies for acyclicity enforcement, adversarial objectives, and architectural refinements influence both the interpretability and performance of learned models. In particular, the framework emphasizes a comparative analysis of recent DAG-based adversarial methods, including DAG-WGAN, DAG-WGAN+, and DAGAF, centering around comparing and validating the capacity of these models to accurately infer causal structure, synthesize high-fidelity data, and maintain computational efficiency.

The comparative evaluation strategy proceeds along three primary dimensions: structural accuracy, generative fidelity, and computational efficiency. Structural accuracy is quantified using the metrics described in the above Section 1.5.2, which collectively assess the correctness of inferred edges and their orientations. Generative fidelity is evaluated through distributional similarity metrics, including Maximum Mean Discrepancy (MMD) and Wasserstein distance (WD) applied to tabular data, alongside predictive utility tests (e.g., feature importance) on downstream tasks to determine whether synthetic samples preserve functional dependencies observed in real data. Computational efficiency is examined by recording training time, convergence behavior, and scalability with respect to the number of nodes and samples. Collectively,

these evaluation dimensions offer a comprehensive perspective on how each approach balances between causal interpretability, generative authenticity, and computational performance.

To provide a more comprehensive comparative analysis, a series of ablation experiments are performed to isolate and quantify the influence of key model components. Three main ablation paths are explored: 1) modifications to model architecture (i.e., removing components like GAN or VAE in DAG-WGAN+) to assess its role in ensuring valid causal graphs; 2) substitution or complementarity of the Wasserstein loss with alternative divergence measures (in both DAG-WGAN+ and DAGAF), such as reconstruction losses (i.e., MSE and NLL), to test sensitivity to adversarial distance formulations; and 3) the addition of regularization loss terms (including MMD and KLD functions in both DAG-WGAN+ and DAGAF) to analyze their effect on graph density and overfitting. For each ablation configuration, the same datasets and evaluation metrics are maintained to ensure direct comparability. Changes in SHD are systematically measured, revealing the contribution of each component to the overall performance of every model.

Validation of the results follows a multilayered approach encompassing internal, external, and statistical validation (the last two are applied only within the context of DAGAF). Internal validation assesses the stability and reproducibility of the training process by conducting multiple runs with different random seeds and evaluating performance across independent data partitions. External validation examines the generality of the learned causal structures and generators when weighted adjacency matrices are transferred across different model instances with varying statistical properties or noise levels. Statistical validation confirms the significance of observed performance differences using non-parametric hypothesis testing methods such as the Mann-Whitney tests, and incorporating confidence interval estimation to provide a clearer indication of the practical significance of the results. Qualitative validation is also conducted by inspecting the interpretability and plausibility of learned causal graphs in domains where partial ground truth or complete causal knowledge is available. For tabular data synthesis, the preservation of marginal distributions, pairwise correlations, and

downstream predictive utility further validates the fidelity and diversity of generated samples. Runtime profiling and memory consumption analyses substantiate claims of efficiency, while consistent computational environments and standardized codebases ensure reproducibility.

Essentially, this analytical framework combines comparative experimentation, ablation study dissection, and rigorous validation to comprehensively assess the learning dynamics and performance of DAG-WGAN, DAG-WGAN+, and DAGAF. By gathering evidence from structural, statistical, and computational perspectives, the framework provides a robust basis for evaluating adversarial DAG-learning models and contributes to a deeper understanding of how generative-adversarial mechanisms can be effectively harnessed for causal discovery and tabular data synthesis.

## 1.6 Contributions

This thesis primarily investigates the effects of the Wasserstein loss in a causal structure learning context. Its influence has been measured by incorporating the loss term into the training algorithm of existing models and applying it as means of parameter optimization for simultaneous causal discovery and tabular data synthesis under a single machine learning framework. In addition, research topics such as disentangled representation and efficient structure learning have also been explored to improve accuracy and reduce time complexity. The rest of this section briefly presents the research areas relevant to the work discussed in this thesis, while stating the contributions of the author.

### 1.6.1 Impact of adversarial training on variational inference in Causal Discovery

The author contributes to causal discovery by studying the impact of the Wasserstein loss with gradient penalty (WGAN-GP) on modeling the relationships between features in observational data. Although the application of adversarial training in various fields is well established, the use of GAN-based architectures in causal structure learning is

relatively uncommon. Notable approaches in the domain utilizing the generative adversarial network architecture include Structural Agnostic Modelling (SAM) [63], MCS [64] and DAG-GAN [37]. These methods have demonstrated an ability to learn reasonable causal relationships, but they suffer from scalability issues and do not assume multiple data types. Moreover, only SAM and MCS utilize WGAN-GP, leaving the influence of this architecture and its adversarial loss on causality learning largely unexplored.

The research carried out in support of this work leads to the development of a parametric algorithm based on the VAE-GAN [62] architecture called DAG-WGAN. The approach is an extension of DAG-GNN [35] and improves on the model by introducing a discriminator and an additional adversarial loss term during training. The causal discovery method has been thoroughly tested against other popular models in the field, and there is empirical evidence to suggest that DAG-WGAN can be used to recover accurate structures from continuous and ordinal data. Interestingly, the experiments also indicate that the Wasserstein loss with gradient penalty is most impactful when working with high-dimensional data. Further details regarding this novelty can be found in Chapter 3.

### 1.6.2 Generative Adversarial Causal Structure Learning

To facilitate the integration of GAN into the field of Causal Structure Learning, the author has developed a non-parametric generative adversarial framework called DAGAF. This model is structured as a standalone WGAN-GP extension of DAG-Notears-MLP capable of handling multiple data types. As such, it can be used to recover causal relationships from continuous and categorical datasets under various structural causal model assumptions.

The main contribution of this approach is the implicit definition of a new probability distribution with an embedded causal structure, which allows for the sampling of realistic data points that maintain the causality exhibited in the input. The framework also involves transfer learning to establish a two-step training pipeline. Initially, DAGAF learns the causal relationships among the input features, and then employs the acquired causal knowledge in a conditional generator to produce synthetic samples.

The adversarial training is further strengthened by the addition of a reconstruction loss term. The MLE-based loss term has the most significant effect on the causal discovery process, however, a theoretical analysis has been conducted to show the contribution of the adversarial loss function to causal structure learning. Further details regarding this novelty can be found in Chapter 5.

### 1.6.3 Adversarial causal discovery with the post-nonlinear model

Most machine learning algorithms used for discovering causal structures from observational data assume only Additive Noise Models (ANM) as their Structural Causal Model (SCM). This is a limitation of their design, as it implies that datasets can only be generated using that particular SCM. Meanwhile, there is very limited research conducted on causality learning using the post-nonlinear model (PNL), which is another identifiable SCM, in most cases. For settings where PNL is not identifiable, see [32]. To the best knowledge of the author, there are mostly discrete score-based methods that can perform causal discovery using the post-nonlinear model, with notable examples including [69], [70], [71], [72]. Prior to this work, there were very few machine learning models (i.e Deep PNL, AbPNL, MC-PNL and CAF-PoNo) that could recover relationships between variables under the assumption of PNL.

The author has developed three methods for adversarial causal discovery that use the post-nonlinear model either in their architectures or the input data. These models have been tested against approaches using ANM and LiNGAM (Linear Non-Gaussian Acyclic Model), which are considered to be subsets of PNL. Experiments have shown that adversarial training can be used to recover high-quality causal structures when the post-nonlinear model is assumed. These promising initial findings may stimulate the proliferation of scientific literature and further investigations into this emerging frontier of causal discovery. Additional details regarding this novelty can be found in Chapters 3, 4, 5.

### 1.6.4 Efficient Structure Learning

DAG-WGAN and DAGAF both employ an augmented Lagrangian as part of their training procedure. Although these methods produce high-quality causal structures, they are very slow due to the cubic computational complexity of the Lagrangian. To address this issue, the author shifted their focus to efficient structure learning. This field of research is devoted to creating frameworks that can identify causality from data without the use of the augmented Lagrangian. Notable approaches include DAG-NoCurl [48] and DP-DAG [49], which require significantly less time to discover causal structures.

The author contributes to the field of efficient structure learning by combining DAG-WGAN and DAG-NoCurl to develop a new model called DAG-WGAN+. Experiments were conducted to assess the accuracy and the time it took for the model to run. The outcome was a slight increase in accuracy and a considerable decrease in training time. Furthermore, an analysis of the algorithm was conducted to determine the computational complexity of DAG-WGAN+. The results indicate that the new model is significantly more efficient than DAG-WGAN, reducing its computational complexity from cubic to quadratic. Additional details regarding this novelty can be found in Chapter 4.

### 1.6.5 Disentangled Representations in Causal Structure Learning

An additional study was conducted to assess whether Disentangled Representation Learning (DRL) could improve data generation and causal discovery. This was done by adding a kernel-based Maximum Mean Discrepancy (MMD) [67] loss term to the models discussed in Chapters 4 and 5. Previously, MMD had been used in the context of causal learning with a few models, such as DAG-GAN [37], CGNN [73] and MMD-LCS [74], to produce good results. To further investigate the effects of MMD on causal discovery, an ablation study was conducted comparing versions of DAG-WGAN+ and DAGAF with and without MMD. The results favored the instances with MMD, providing evidence to support the use of DRL in this context. Additionally, a theoretical analysis was conducted to prove the contributions of MMD to causal structure learning. Further

details regarding this novelty can be found in Chapters 4 and 5.

## 1.7 Publications

Every author contribution described in this thesis has been published in the form of a research paper. The list of peer reviewed publications related to this dissertation is presented below:

- DAG-WGAN: Causal Structure Learning with Wasserstein Generative Adversarial Networks [75]
- Causality Learning with Wasserstein Generative Adversarial Networks [76]
- Efficient Generative Adversarial DAG Learning with No-Curl [77]
- AI-Powered Clinical Trials: Emulating Real-World GLP-1 Efficacy with Synthetic Patient Populations Using Causal Effect Learning [78]
- DAGAF: A directed acyclic generative adversarial framework for joint structure learning and tabular data synthesis [79]
- Emulating Real-World GLP-1 Efficacy in Type 2 Diabetes through Causal Learning and Virtual Patients [80]

## 1.8 Thesis Structure

The author formulates the rest of their thesis as follows:

- Chapter 2 offers a detailed overview of the fundamental components of causal structure learning, exploring different approaches for conducting it and summarizing the relevant literature.
- Chapter 3 discusses the theory and implementation details behind the development of a hybrid model, combining adversarial training and variational inference in the context of causality learning.

- Chapter 4 explores efficient structure learning techniques and their relevance to causal discovery. This leads to the design of an enhanced version of the algorithm introduced in the previous chapter, demonstrating better accuracy and computational complexity.

- Chapter 5 introduces causality learning under multiple causal structural model assumptions and its connection to tabular data synthesis, motivating the creation of a novel framework for simultaneous causal discovery and generation of tabular datasets.

- Chapter 6 concludes the thesis by briefly stating the impact of the research conducted by the author as well as their opinion regrading the direction of future research efforts in the area of causal structure learning. Additionally, the author shares their closing thoughts on all the work done during their Ph.D. studies.

- Appendix A provides all the mathematical proofs for the lemmas and the propositions defined throughout this thesis.

- Appendix B presents the code used to conduct the data quality experiments discussed in this work.

# Chapter 2

# Literature Review

This chapter offers an overview of the research field to which the author contributes. It begins with a brief description of the components necessary for causal structure learning and then provides a comprehensive history of approaches and techniques for discovering causation. The chapter then moves on to a critical analysis of prior works, highlighting their strengths and limitations while explaining the rationale behind the methodological choices of the author. Additionally, details of the implementation behind some models are also discussed, as they are closely related to the research conducted in this work - for further information, refer to Section 2.4.

## 2.1 Prerequisites

In this section, the author describes the concepts essential for discovering causal structures along with practical frameworks and a set of assumptions used for the development of algorithms supporting the contributions described in their thesis.

### 2.1.1 Directed Acyclic Graphs

Directed Acyclic Graphs (DAG) [81], [82] are visual constructs describing complex mathematical problems, defining sequences of processes or studying how different variables within a particular setting relate to one another. They are a special subset of graphs, which do not contain cycles between vertices $V = \{V_1, ..., V_{\mathbf{N}}\}$ and have di-

rected edges $E = \{(i, j) \in \mathbb{R}^{v \times v}\}$ connecting them. All graphs that do not contain cycles live in their own DAG space, denoted by $\mathbb{D}$. Moreover, each edge of a DAG $\mathbf{G}\langle V, E\rangle$ is defined as $(i \rightarrow j) \in E$, where according to the direction of the relationship $i$ is the ancestor of $j$ and $j$ is the descendant of $i$. DAG are used in various fields of computer science and other research areas due to their ability to be computed and defined using an adjacency matrix $\mathbf{A} \in \mathbb{R}^{v \times v}$. In this alternative representation, each element $\mathbf{A}_{ij}$ is either 0 or 1 depending on whether there is a directed edge between $i$ and $j$. Alternatively, there exist other types of adjacency matrix such as the Siedel adjacency matrix [83], where the permitted values are -1, 0 or 1 and the weighted adjacency matrix which stores the weight values assigned to each edge. This work describes models utilizing only weighted adjacency matrices.

The research presented in this thesis explores the application of DAG from a causal perspective. Particularly, the acyclicity between nodes combined with the directionality of edges facilitates the encoding of "*cause and effect*" between parents and children, allowing people to easily interpret the causality visualized in a graph [84]. Meanwhile, estimating the causal effect of individual relationships present in the structure of a DAG (causal inference) has also become a popular research topic, leading to various scientific breakthroughs [85], [86], [87], [88], [89]. This study is limited to the application of DAG for causal structure learning.

### 2.1.2 Bayesian Networks

Bayesian Networks (BN) [90] are a type of Probabilistic Graphical Model used to calculate probabilities (i.e uncertainties) by modeling the conditional dependencies between variables in a joint distribution. They enable the visualization of causal relationships through Directed Acyclic Graphs (DAG), where nodes represent variables, while edges express direct connections between them. The probability distributions and their corresponding structures described in Bayesian Networks have to satisfy the *Local Markov Assumption* [91]. This property ensures that given a set of variables $X = \{X_1, ..., X_d\}$ that form a DAG, a node $X_j$ depends solely on its immediate parents $Pa_j$, enabling us to define the Bayesian Network factorization of a joint distribution as:

$$P(X_1, ..., X_d) = \prod_j P(X_j|Pa_j). \tag{2.1}$$

Furthermore, BN play a crucial role in causal discovery because their structure can be obtained directly from observational data [10].

The concept of using machine learning techniques to retrieve a Directed Acyclic Graph (DAG) that best reflects the connections between variables concealed in data sets was first suggested by George Rebane [92]. Since then, a variety of methods have been developed to learn its structure, sparking a proliferation of literature. To recover an accurate graphical representation of the causality modeled by a Bayesian Network $BN = (\mathbf{G}, \varphi)$, one has to discover a set of its components that best describes the input: 1) a DAG $\mathbf{G}$ representing how variables within the data are connected, and 2) a set of parameters $\varphi$ that produce the probability distributions defined by the content of the graph.

Learning the parameters that best correspond to the data is straightforward using machine learning or rule-based techniques. However, learning the DAG which describes the connections between the data is very challenging. The difficulty is related to the combinatorial nature of the DAG search space [22]. Nevertheless, several techniques have been developed for learning causal structures, including score-based, constraint-based, hybrid approaches (also known as traditional methods), and most recently, continuous optimization and efficient structure learning. A brief explanation for each of them is provided in Section 2.2.

### 2.1.3 Structural Causal Models

The importance of causal structure learning is significant in fields where it is necessary to distinguish between correlations and causal relationships. Machine learning algorithms can easily detect correlations in data, but discovering causal connections is a more complex process that requires a detailed investigation of properties such as directionality, temporal sequencing, interventions and confounding. These features are not expressed through associative relationships, hence causality cannot be defined in the same way as correlations. Therefore, it is imperative to formulate a strong no-

tion for representing and validating cause-and-effect relationships expressed in a DAG. Such a concept, known as the Structural Causal Model (SCM), was first proposed by Sewell Wright [93], but it was Judea Pearl [94] who refined and developed it into a mathematical object.

A Structural Causal Model $\mathcal{M}\langle X, \mathcal{Z}, \mathcal{F}\rangle$ is composed of three sets: a set of data variables $X = \{X_1, ..., X_d\}$, a set of noise vectors $\mathcal{Z} = \{\mathcal{Z}_1, ..., \mathcal{Z}_d\}$ sampled from an external distribution $P(\mathcal{Z})$, and a set of functions $\mathcal{F} = \{f_1, ..., f_d\}$ that define the causal mechanisms between the variables in $X$. This combination of sets allows for the synthesis of $X_j$ from other variables in $X$ and noise in $\mathcal{Z}$. Assuming that there are no hidden confounders influencing the variables in $X$, the SCM takes the general form

$$X_j := f_j(Pa_j, \mathcal{Z}_j), \tag{2.2}$$

where $X_j$ is the generated variable, $Pa_j$ are its parents, $\mathcal{Z}_j$ is the noise and $f_j$ is the equation used to produce it. All of the causal mechanisms described in the SCM are visualized through a DAG $\mathbf{G}$, defining the underlying structure of the data as a causal graph, where the variables of the data are nodes and the edges are functions (structural equations).

### 2.1.4 Assumptions for Causal Discovery

This section provides a list of assumptions essential for causal structure learning. All of them are satisfied in the experiments related to this thesis.

- **Acyclicity**: The graph describing a structural causal model must not contain any cycles.
- **Local Markov assumption**: Given its parents in the DAG, a node is independent of all its non-descendants. This assumption implies that variables in the graph are solely dependent on their immediate parents.
- **Strict Causal Edges**: In a directed graph, every parent is a direct cause of all its children. This assumption enables the visualization of dependencies between variables from the probability distribution.

- **Minimality**: This assumption consist of two components. The first part is the Local Markov assumption and the second is Strict Causal Edges. It suggests that conditional independencies in a probability distribution are expressed with a minimal number of edges.

- **Causal sufficiency assumption**: There exist no unobserved common causes (i.e hidden confounders) between any of the variables in the graph.

- **Faithfulness**: This assumption enables causal structure learning from observational data. It states that a probability distribution and the DAG describing it are faithful only when the conditional independencies of the distribution are expressed in the graph [95], [96]. Under the causal sufficiency assumption, the faithfulness condition implies that if there exists a statistical dependency between two variables, then there is an underlying causal relationship between them.

- **Semi-parametric assumptions**: A group of assumptions influencing the formulation of the structural equations making up SCM. Throughout this research a variety of models are assumed (e.g. additive noise models, linear non-gaussian acyclic models and post non-linear models) based on their causal identifiability.

### 2.1.5 Structure Identifiability

Estimating causal effects from datasets requires information regarding the underlying structure of their contents. Interventional studies offer the most rigorous method for establishing causality through data manipulations. However, setting up such experiments is difficult, infeasible, or sometimes even impossible. On the other hand, observational studies are far more practical but only provide data without prior knowledge of its causal structure. Additionally, they are easier to conduct and thus preferred to interventional ones. This has led people to ask the question "*Can causal relationships be obtained from observational data?*".

As stated previously, it is impossible to recover causal structures from observational data without satisfying a specific set of assumptions. Furthermore, performing causal discovery multiple times under the same setting (i.e., no changes in observational

samples, model used, or assumptions made) can produce different results. This is a fundamental problem referred to as structure identifiability, which significantly limits our ability to learn the causality expressed in observational data.

**Definition 1.** *Structure Identifiability:* Given a set of assumptions $\mathcal{A} = \{\mathcal{A}_1, ..., \mathcal{A}_\mathbf{N}\}$ and samples $\mathbf{X}$ belonging to a probability distribution $P(\mathbf{X})$, a causal structure learning model $\mathcal{M}$ will recover an identifiable DAG $\mathbf{G}$ if and only if its implicitly defined distribution $P(\tilde{\mathbf{X}})$ cannot be generated using any other $\mathbf{G}' \in \mathbb{D}$.

In terms of graph identification, under the local Markov assumption, we can only identify the Markov Equivalence Class (MEC) to which a DAG belongs [97]. However, in some specific cases, depending on additional assumptions and the probability distribution from which the data originates, an MEC can be identified down to a DAG. For example, when using different SCM (e.g. linear non-Gaussian noise model $\tilde{X} := f(X) + \mathcal{Z}$, non-linear additive noise model $X_j := f_j(Pa_j) + \mathcal{Z}_j$ and post-nonlinear model $X_j := g_j(f_j(Pa_j) + \mathcal{Z}_j)$), with the inclusion of additional assumptions such as faithfulness and minimality, a DAG can be identified [31], [98], [32]. On the other hand, if the distribution is multinomial or linear Gaussian, despite the additional assumptions, no further identification can be made from the MEC [99], [100].

### 2.1.6 Markov Equivalence and CPDAG

The local Markov assumption implies that only partial identification of a Directed Acyclic Graph (DAG) $\mathbf{G}$ is possible based on the conditional independencies in the distribution it describes. The inability to recover the dependencies in the data through this assumption does not allow the identification of a DAG, but rather its Markov Equivalence Class (MEC) instead.

**Definition 2.** *Markov Equivalence Class:* The set $MEC = \{\mathbf{G}_1, ..., \mathbf{G}_\mathbf{N}\}$ of all graphs containing the same conditional independencies (i.e an identical skeleton and immoralities [101]).

This class of graphs is described by a Completed Partially Directed Acyclic Graph (CPDAG) [102], [103]. A CPDAG is similar to a DAG, but with a distinction in the

types of edges present in the graphs. The former includes a combination of directed and undirected edges, whereas the latter only has directed edges. In the context of independence-based causal discovery, a CPDAG incorporates both directed and undirected edges to represent immoralities and the skeleton of the graph, respectively.

### 2.1.7 Evaluation Metrics

This section contains metrics used to assess the quality of the results produced by the models described in this thesis.

- **True Positive Rate (TPR)** [104] - measures how many edges belonging to the ground truth graph $\mathbf{G_A^0}$ have been recovered. For example, if $\mathbf{G_A^0} = A \rightarrow B \rightarrow C$ and the recovered causal graph $\mathbf{G_A} = A \rightarrow B$, then the TPR is 66%. Conversely, if $\mathbf{G_A^0} = \mathbf{G_A}$ the TPR is 100%. The metric is commonly used to calculate how many edges of the ground truth graph are missing from the recovered graph.

- **False Discovery Rate (FDR)** [104] - measures how many additional edges not present in the ground truth graph $\mathbf{G_A^0}$ have been discovered. For example, if $\mathbf{G_A^0} = A \rightarrow B \rightarrow C$ and the recovered causal graph $\mathbf{G_A} = A \rightarrow B \rightarrow C \rightarrow D$, then FDR is 25%. If $\mathbf{G_A^0} = \mathbf{G_A}$ the FDR is 0%. The metric is used to represent the number of extra edges in the recovered graph.

- **False Positive Rate (FPR)** [104] - measures how many edges of the ground truth graph $\mathbf{G_A^0}$ have been recovered with incorrect directionality. For example, if $\mathbf{G_A^0} = A \rightarrow B \rightarrow C$ and the recovered causal graph $\mathbf{G_A} = A \rightarrow B \leftarrow C$, then the FPR is 33%. If $\mathbf{G_A^0} = \mathbf{G_A}$ the FPR is 0%. The metric is used to establish the number of reversed edges present in the recovered graph.

- **Structural Hamming Distance (SHD)** [105] - This distance encompasses all of the metrics described above. It measures how many adjustments are necessary to guarantee that the recovered causal graph $\mathbf{G_A}$ matches the ground truth graph $\mathbf{G_A^0}$. For example, if $\mathbf{G_A^0} = A \rightarrow B \rightarrow C$ and $\mathbf{G_A} = A \leftarrow B \rightarrow D$, then SHD is 3, taking into account extra ($D$), reversed ($A \leftarrow B$) and missing edges ($C$).

Other metrics include Area Over Curve (AOC) [104], Area Under Curve (AUC) [104] and Structural Interventional Distance (SID) [106]. In this work, evaluation is limited to the application of the metrics in the list above.

### 2.1.8 Generative Models

Generative Models, as their name suggests, are a set of frameworks capable of producing new data points that resemble some input data. Models falling into this category are widely used in unsupervised machine learning because of their ability to implicitly learn a probability distribution that closely matches the original data distribution.

A non-exhaustive list of generative models includes:

- (Gaussian) mixture model [107]
- Hidden Markov model [108]
- Variational autoencoder [52]
- Generative adversarial network [53]
- Flow-based generative model [51]
- Energy based model [109]
- Diffusion model [54]

In recent years, Variational Autoencoders (VAE) and Generative Adversarial Networks (GAN) have gained popularity due to their contributions to various research areas. Both frameworks rely on multiple Artificial Neural Networks (ANN) [110], and their implementation details are discussed in this section.

#### Variational Autoencoder

Variational Autoencoders utilize a pair of neural networks, namely an encoder and a decoder, to learn a given training data distribution and sample new data points from it. Specifically, the encoder produces a latent representation that captures the features

of the real data distribution, while the decoder uses this latent variable to generate new data samples that resemble those belonging to the original distribution. Models based on this framework are typically trained by minimizing a Maximum Likelihood Estimation (MLE) [55] objective function known as Evidence Lower BOund (ELBO) [61]. The ELBO consists of two components: 1) a reconstruction loss term (i.e., negative log-likelihood) and 2) a regularization term (i.e., Kullback-Lieber divergence).

$$ELBO = -\mathbb{E}_{Z\sim Q_\phi(Z|\mathbf{X})}[\log P_\theta(\mathbf{X}|Z)] + D_{KL}(Q_\phi(Z|\mathbf{X})||P(Z)) \tag{2.3}$$

In (2.3), $P(Z)$ denotes the prior distribution of the latent variable $Z$, while $Q_\phi(Z|\mathbf{X})$ and $P_\theta(\mathbf{X}|Z)$ represent the encoder and decoder networks respectively. As can be seen from the formulation of the ELBO, the optimization of the decoder parameters $\theta$ is dependent on the encoder parameters $\phi$, hence the objective function is used to simultaneously learn both the generative model and the inference model. Furthermore, the latent variable $Z$ used in the generator model is sampled from a probability distribution $Q_\phi(Z|\mathbf{X})$, making the network non-differentiable due to the randomness of $Z$. To fix this issue, the reparameterization trick in (2.4) is used to enable backpropagation through the decoder during training.

$$Z = \mu + \sigma \odot \epsilon \tag{2.4}$$

**Generative Adversarial Network**

Similarly to VAE, Generative Adversarial Networks (GAN) are a type of generative model consisting of two networks. The main idea behind GAN is to define a new implicit probability distribution that closely matches the original data distribution, enabling the generation of realistic data samples. This is achieved by forcing two networks to compete against each other. On one side, a generator network $G$, takes noise $\mathbf{Z}$ as input and generates new data samples that closely resemble the real data. On the other hand, a discriminator network $D$, receives these generated samples, as well as real data samples and tries to determine whether they belong to the real data distribution or not. The optimization of this framework is achieved through adversarial

training, where both networks engage in a Min-Max game and the objective is defined by the following loss function:

$$\min_G \max_D V(D,G) = \mathbb{E}_{\mathbf{X}\sim P(\mathbf{X})}[\log D(\mathbf{X})] + \mathbb{E}_{\mathbf{Z}\sim P(\mathbf{Z})}[\log(1 - D(G(\mathbf{Z})))] \tag{2.5}$$

The original GAN proposed by [53] is difficult to train and can only tell us if a sample is real or fake, hence over the years several improvements to the objective function of GAN have been proposed [111], [112], [113]. Amongst them, most notable is the inclusion of the Earth-Mover's Distance (EMD) [114].

The reformulation of the loss function with Wasserstein-1 results in a new type of GAN named Wasserstein Generative Adversarial Network (WGAN) [115]. The main advantage that WGAN have over regular GAN is the ability to measure the distance between the real data distribution $P(\mathbf{X})$ and the implicitly defined distribution $P(\tilde{\mathbf{X}})$ modeled by the generator. This reformulates the problem from detecting whether a sample is real or fake to measuring how real or fake a given sample is.

$$\min_G \max_{D\in\mathcal{W}} V(D,G) = \mathbb{E}_{\mathbf{X}\sim P(\mathbf{X})}[D(\mathbf{X})] - \mathbb{E}_{\mathbf{Z}\sim P(\mathbf{Z})}[D(G(\mathbf{Z}))] \tag{2.6}$$

Minimizing the distance between the probability distributions $P(\mathbf{X})$ and $P(\tilde{\mathbf{X}})$ produces data which better represents the original data distribution. However, a limitation of WGAN is the difficulty to enforce the discriminator $D$ to belong to the set of 1-Lipschitz functions (i.e $\mathcal{W}$ in Equation (2.6)). A possible solution to this problem is the inclusion of a gradient penalty term into the objective function of WGAN, resulting in Wasserstein Generative Adversarial Networks with Gradient Penalty (WGAN-GP) [116]. The gradient penalty term enforces the 1-Lipschitz constraint on $D$ by penalizing the model if the value of the gradient norm moves away from 1.

$$\min_G \max_{D\in\mathcal{W}} V(D,G) = \underbrace{\mathbb{E}_{\mathbf{X}\sim P(\mathbf{X})}[D(\mathbf{X})] - \mathbb{E}_{\mathbf{Z}\sim P(\mathbf{Z})}[D(G(\mathbf{Z}))]}_{\text{Critic loss}} + \underbrace{\lambda\mathbb{E}_{\hat{\mathbf{X}}\sim P(\hat{\mathbf{X}})}[(\|\nabla_{\hat{\mathbf{X}}} D(\hat{\mathbf{X}})\|_2 - 1)^2]}_{\text{Gradient penalty}} \tag{2.7}$$

This approach produces high-quality data but has poor computational complexity due

to the addition of extra terms in the loss function.

## 2.2 Causal Discovery Approaches

The idea of employing rule-based or machine learning techniques to identify a graph structure that best represents the dependencies present in observational data has been around since the early 1990s [117]. Over time, the field of causality learning has developed into a well-established scientific domain with five primary approaches to discovering causal relationships from a dataset. These include traditional methods (constraint-based, score-based, and hybrid), continuous optimization, and the more recent efficient structure learning. Elaborations for each of those will be provided in this section.

### 2.2.1 Traditional Methods

Constraint-based methods (CBM) apply various conditional independence tests in order to construct a graph that best represents the independencies within a given joint distribution. The majority of models in this category satisfy the local Markov assumption, resulting in the identification of a set of graphs that describe the same conditional independencies. In other words, the output of these models is a Markov equivalence class (MEC) represented by a completed partially directed acyclic graph (CPDAG), where only the connections between vertices with a unique direction are shown as directed edges. One of the first models capable of producing such a graph is the PC algorithm [28], which is well-known for its three-step approach (skeleton identification, immorality identification and orientation, and edge orientation). Other kernel-based solutions for conditional independence testing include the KCI-test [118] and KLC [119]. When causal sufficiency is not assumed, models such as ION [120], cSAT+ [121], and CCI [122] have demonstrated good performance by applying independence judgments to bidirectional graphs. Other CBM relax the assumptions of faithfulness [123], [124], [125], [126] and acyclicity [127], [128], [129], [130] in order to identify MEC. Lastly, interventional studies have been conducted using constrain-based methods such as IDA [131], backshift [132], CombINE [133], and $\sigma$-CG [134].

The DAG search space in CBM poses a fundamental challenge due to its combinatorial nature. Chickering et al. [22] have demonstrated that discovering DAG through independence tests is NP-hard, rendering this class of algorithms highly inefficient. As a result, a new generation of models, such as FCI [29], RFCI [135], and Parallel-PC [136], has been developed to improve the efficiency of the constraint-based approach. Additionally, the method suffers from unreliability as most conditional independence tests require a substantial amount of data to accurately estimate the independent variables in a given distribution [30].

Score-based methods (SBM) perform similar DAG searches, but they use different techniques. Their objective is to discover a graph that best represents some probability distribution using a scoring function (i.e some metric) [137], [138], [139]. In other words, each graph in the DAG search space is assigned a score, and the one with the best score is considered to accurately describe the underlying causality in a given dataset. Traditional score-based DAG learning focuses on the implementation of rule-based approaches to perform discrete search procedures. These methods aim to provide a discrete optimization solution for the following problem:

$$\max_{\mathbf{G}} \ f_{score}(\mathbf{X}, \mathbf{G}) \text{ s.t. } \mathbf{G} \in \text{discrete DAG}, \tag{2.8}$$

where $f_{score}$ denotes the scoring function, $\mathbf{X}$ the observational data samples and $\mathbf{G}$ the DAG which describes the probability distribution the input data originates from. Notable examples of such functions include Bayesian Information Criterion (BIC) [140], Bayesian Dirichlet equivalence (uniform) (BDe(u)) [25], Bayesian Gaussian equivalent (BGe) [26] and Minimum Description Length (MDL) [27].

An inherent limitation of this method is its inability to effectively handle the super-exponentiality of the DAG search space. To address this, additional assumptions and approximations such as bounded tree-width [141], tree-structure [142], and sampling [143], [144], [145] are often necessary to achieve computational tractability. Another challenge with this approach is the fact that all discrete score-based approaches are non-differentiable, making it impossible to use gradient-based optimization techniques essential for machine learning models. Nevertheless, successful implementa-

tions of the score-based methodology include the original Greedy Equivalence Search (GES) [146] and its improved versions GES-mod [147] and GIES [84]. LiNGAM [31] and its variations [148], [149] recover graphs by assuming a linear non-Gaussian Structural Causal Model. K2 [150] and GCL [151] learn hidden confounders under causal insufficiency. Exceptional results have also been achieved with non-acyclic [152], [153], [154] and interventional [155], [156], [157], [158] solutions. Additionally, the SP [159] model performs causal structure learning by relaxing the faithfulness assumption.

Both CBM and SBM are capable of producing accurate results, but each of them has their own limitations. CBM tend to be unreliable when dealing with a small sample size, while SBM often make additional assumptions and approximations to ensure that the optimization of the score function is computationally feasible. Moreover, both methods are highly inefficient because of the NP-hardness of DAG. As a result, researchers have developed hybrid approaches that combine CBM and SBM to achieve better accuracy and efficiency. Successful models in this domain of causal structure learning include MMHC [34], RELAX [160], ARGES [161], and BiDAG [162]. Unfortunately, due to the discrete nature of the optimization, this approach still has combinatorial computational complexity, making it unsuitable for handling complex datasets containing large volumes of data samples or variables.

### 2.2.2 Continuous Optimization

In 2018, the development of a new model called Non-combinatorial Optimization via Trace Exponantial and Augmented lagRangian for Structure learning (NOTEARS) [12] transformed the discrete nature of SBM into the continuous optimization (CO) of score functions with respect to an explicit acyclicity constraint. The key distinction between this approach and its discrete predecessor is that continuous optimization is a differentiable process, enabling the use of machine learning models for causal structure learning. In a general sense, the CO problem can be defined as follows:

$$\max_{\mathbf{G}} \; f_{score}(\mathbf{X}, \mathbf{G}) \text{ s.t. } h(\mathbf{G}) = 0, \tag{2.9}$$

where $h(\mathbf{G})$ is the acyclicity constraint applied to $\mathbf{G}$. The method proposed by [12] showcased the ability to generate more accurate results in a shorter time frame compared to traditional methods. Nevertheless, a significant drawback of the model is its limited applicability to linear data.

Subsequently, several extensions of NOTEARS have been developed that are nonlinear in nature. Some of these models are based on the Auto-Encoder (AE) architecture [35], [163], [39], [164] or the Generative Adversarial Network (GAN) [63], [165], [37], while others utilize meta-learning [166], [167] and flow-based generative models [168], [169]. Moreover, the ability to discover causality with black-box models has been applied to different types of data, including images [170], [171], tabular data [172], [38], videos [173], and time-series data [174]. However, most continuous optimization algorithms suffer from inefficiencies related to parameter optimization using the augmented Lagrangian. According to [36], the computational complexity of the augmented Lagrangian in a causal structure learning setting is $\mathcal{O}(d^3)$, where $d$ is the number of variables in the data. The poor time-wise performance of these models makes them practically unusable with high-dimensional data. Most of the models mentioned above assume an Additive Noise Model (ANM). However, there are also models that assume a Post-Nonlinear Model (PNL), such as AbPNL [41], Deep PNL [42] CAF-PoNo [175] and MC-PNL [176]. Additionally, there are models that do not assume acyclicity [177], [178]. This work is focused on investigating acyclic causal discovery methods only.

An alternative continuous optimization approach, developed by [44] as a deviation from the NOTEARS framework, demonstrates improved DAG-learning and computational complexity compared to previous approaches. The method is based on learning the correct order of nodes using permutation vectors that form a specific structure called the Permutahedron. Another work in this field is CASPER, developed by [45]. Its authors highlight the flaws in the DAG-independent score functions of previously existing models based on the NOTEARS framework. They argue that not considering the graph structure significantly affects how the DAG search space of an algorithm is defined, resulting in the discovery of substandard DAG. Their model recovers accurate DAG from a dynamic search space through a novel score function. Currently, the latest

work in the field is a novel technique called REX [46] that incorporates Shapley values into the causal discovery process by interpreting feature contributions from machine learning models. REX minimizes the number of features integral to the causal graph by utilizing the connection between Shapley values and causal relationships. This specific technique enables the authors to focus on the most influential variables for subsequent causal analysis, enhancing the effectiveness and precision of identifying causality in complex datasets. This integration not only makes causal models more interpretable but also strengthens the detection of causal relationships by concentrating on features that have a substantial impact.

### 2.2.3 Efficient Structure Learning

In recent times, researchers have developed machine learning models that can discover causality without relying on the augmented Lagrangian method. These methods fall under the category of efficient structure learning (ESL) and can recover connections between variables in a given dataset significantly faster. The algorithms in this class are all based on theoretically developed frameworks for efficiency. One notable approach in this category is DAG-NoCurl [48], which is considered one of the pioneering methods that does not use the augmented Lagrangian. Other examples of such algorithms include VI-DP-DAG [49] which recovers DAG by learning differentiable probability distributions over edges and permutation matrices and ELCS [179] which is an efficient approach utilizing Markov Blankets. Recently, a hybrid model called DAG-NCMLP [50] utilized both the DAG-NoCurl framework and the DAG-Notears-MLP model to enable efficient learning of non-linear, non-parametric causal structures. More specifically, the authors developed a theoretical non-parametric projection formulation for gradient-based adjacency matrices, expanding the projection framework to cover more than just weighted adjacency matrix representations. To instantiate their novel theoretical framework, they build a duo-step algorithm to perform efficient DAG learning relying on non-linear projections. The success of these models has motivated further research in the field of efficient structure learning, making it an active area of study.

## 2.3 Critical Analysis

This section presents a rigorous critical analysis of existing research and state-of-the-art methodologies within the field of causal discovery. It systematically evaluates the strengths, limitations, and underlying assumptions of previous studies to establish a clear understanding of their contributions and shortcomings. Specifically, the author expands on the contents of Table 2.1. Through this investigation, they justify the methodological and theoretical choices underpinning their work, demonstrating how these decisions directly address identified gaps in the literature.

Table 2.1: Limitations of prior works

| **Category** | **Main Issues** |
|---|---|
| **MLE-based Approaches** | Blurry outputs, latent collapse (for latent-based generative models only) |
| **Computational Efficiency** | High computational complexity, poor scalability |
| **Post nonlinear model** | Invertibility, post-nonlinearity, limited research |
| **Single SCM assumption** | Unverifiable structures, in terms of faithfulness to the observed data |
| **Tabular Data Synthesis** | No interpretability, mode collapse, dependence on known or externally learned causal structures |

### 2.3.1 Causal structure learning with MLE-based loss functions

Despite their widespread adoption in models such as NOTEARS [12], DAG-Notears-MLP [38] and GraN-DAG [36], MLE-based loss functions impose several practical and theoretical constraints on causal structure learning with generative modeling algorithms relying on latent variables. By optimizing parameters solely to maximize the likelihood of observed data, these approaches often overlook important aspects of data diversity and structural complexity. As a result, common MLE-based objectives (e.g., Mean Squared Error (MSE) [180], Negative Log-Likelihood (NLL) [181], and Evidence Lower Bound (ELBO) [61]) exhibit several notable limitations, including the following:

1. **Simplicity:** The objective of the loss function is to ensure that the output gen-

erated closely resembles the input data. However, this simplicity can result in blurry results because the model aims to minimize the average error across all data points [56]. In the past, attempts have been made to enhance performance by introducing additional terms as regularizers to the reconstruction loss. However, it should be noted that excessive regularization can also lead to blurry results.

2. **Diversity:** In generative models with latent variables, such as the VAE-GAN and WGAN-GP approaches considered in this thesis, the application of MLE-based reconstruction loss terms tends to reduce the standard deviation (std) of the implicitly learned data distribution toward zero. As a result, sampling relies almost entirely on the mean, producing outputs that represent average data samples. This effect, referred to as latent collapse, severely restricts the diversity of the reconstructed data.

Additionally, MLE-based algorithms are affected by the "curse of dimensionality" [182], which limits their effectiveness with high-dimensional data and complex distributions.

As previously mentioned in Section 1.3, to address these problems people include additional loss terms to regularize the model training process, such as the Wasserstein distance (WD) [57], the Kullback-Leibler divergence (KLD) [58], and the Jensen Shannon divergence (JSD) [59]. From these, WD is currently the least explored in the context of causal structure learning.

The Wasserstein distance can serve as a powerful regularizer for MLE-based causal discovery by shifting the optimization focus from minimizing point-wise discrepancies between observed and reconstructed data toward minimizing the distance between their entire distributions. Unlike MLE, which typically aligns individual data points through likelihood maximization, the Wasserstein-based approach captures the global geometry and structural characteristics of the data distribution. This allows the causal discovery process to incorporate rich distributional features, such as variance, skewness, and multi-modality that MLE tends to overlook. As a result, the learning process evolves from a simple reconstruction task into a generative emulation of the underlying data synthesis mechanisms, enabling the model to more faithfully replicate the causal struc-

ture involved in the generative process responsible for producing the observed data.

Moreover, in practical terms, models that rely on adversarial training are expected to be more effective in identifying causal relationships from observations compared to gradient-based or maximum likelihood estimation (MLE) frameworks. An advantage of this methodology is its ability to scale linearly with increasing data variable size, thus reducing susceptibility to the "curse of dimensionality" [53]. Additionally, generative adversarial models have the ability to model distributions of varying complexity and dimensions [183], [184]. They can also handle noisy or incomplete data and address the issue of latent collapse that is often encountered in MLE-based approaches relying on latent variables during training.

To determine the validity of these claims, researchers integrated adversarial training into the process of learning causal structures, resulting in a novel methodology called Wasserstein Adversarial Causal Discovery (WACD). This approach leverages the Wasserstein distance as a data distribution metric to discover the causal relationships present in a given dataset. Essentially, models within this category aim to minimize the distance between the actual data distribution and the generated data distribution, facilitating the implicit recovery of causal structures. These frameworks involve two key components: a Discriminator $D$ and a Generator $G$. Unlike traditional generators that only focus on generating new samples, $G$ in this context strives to simulate the causal mechanisms necessary to match the underlying causal structure of the original probability distribution. Consequently, this leads to the development of causally aware algorithms capable of generating samples that adhere to causal relationships similar to those observed in the input data.

In a recent survey titled "*D'ya like DAGs? A Survey on Structure Learning and Causal Discovery*" [185], it is suggested that the first model capable of working with tabular data in the field of WACD is Structural Agnostic Modeling (SAM) [63]. On the other hand, models such as Causal Adversarial Network (CAN) [165] and Generative Adversarial Neural Network embedded with causal matrix (CMGAN) [186] have demonstrated the ability to recover causality from images. The increasing number of studies on Wasserstein Adversarial Causal Discovery reflects the popularity of WGAN

in causality learning. However, there is still potential for further application of this approach, particularly in dealing with hidden confounders, mixed-type, time-series, and incomplete data, where limited progress has been made.

Despite the lack of scientific literature volume involving these specific subdomains, this adaptability makes WACD applicable in diverse fields such as healthcare, medicine, and justice, where it can enhance decision-making and foster trust between humans and artificial intelligence. Additionally, ongoing advancements in Wasserstein Generative Adversarial Networks (WGAN) contribute to the continuous refinement of adversarial causal discovery. Consequently, this progress is expected to yield more effective models that surpass the current state-of-the-art in the field. The author contributes to WACD by developing a hybrid model based on adversarial training, MLE-based loss terms and the DAG-GNN architecture, resulting in a significant improvement in causal structure learning accuracy from high-dimensional data - for more details see Chapter 3.

### 2.3.2 Importance of Computational Efficiency

Over the past few decades, various techniques have been developed to recover causal relationships between variables in a dataset, giving rise to numerous causal structure learning algorithms and establishing a novel field of research. As outlined in Section 1.2, models designed to discover causality from data generally fall into one of two categories: 1) rule-based (traditional) approaches; or 2) machine learning methods. Despite their different foundations, both families have demonstrated the ability to recover accurate causal structures from observational data. However, many of these methods suffer from poor computational performance, making them impractical for large-scale applications. The three dominant causes of this inefficiency are the curse of dimensionality [187], the NP-hard nature of learning directed acyclic graphs (DAG) [22] and the formulation of the optimization problem. One notable example of a solution to these challenges is the DAG-NOTEARS [12] framework, which transforms the traditionally discrete and combinatorial process of causal discovery into a continuous optimization problem with an explicit acyclicity constraint. While this represents a significant theoretical advancement, NOTEARS still suffers from substantial computational inefficiencies that

limit its scalability and practical applicability.

One of the main sources of inefficiency in NOTEARS lies in the way it enforces the acyclicity constraint, which ensures the resulting graph has no loops. Instead of relying on simple structural checks, NOTEARS expresses this constraint using complex matrix operations that must be evaluated repeatedly throughout the optimization process. These operations become increasingly expensive as the number of variables grows, leading to long runtimes and high memory usage even on powerful hardware. This makes the method well-suited only for relatively small or medium-sized datasets, while larger systems quickly become computationally infeasible.

Furthermore, the optimization procedure used by NOTEARS is non-convex and requires multiple rounds of iterative updates to converge. Each round involves several inner optimization steps, and convergence can be slow or unstable depending on the initial conditions and tuning parameters. The mathematical precision of the method comes at the cost of computational practicality, often requiring considerable time and manual adjustment to produce reliable results. Despite these limitations, research interest in causal discovery has remained strong, leading to the development of various models aimed at improving computational efficiency, a line of research commonly referred to as efficient structure learning.

Efficient Structure Learning (ESL) is a sub-field of Causal Discovery that focuses on recovering the underlying causal relationships between variables in a dataset in an efficient manner. As the name implies, all of the approaches in this category are computational methods that can handle datasets of various sizes and complexities within a reasonable time frame. In the past, there have been different ways to obtain the causal structure of data, such as score-based and constraint-based methods, hybrid algorithms, and continuous optimization. ESL is considered to be a super-set of all the aforementioned approaches, aiming to improve their computational complexity and optimization techniques, resulting in more efficient causal structure learning. The complexity of the approaches varies, with some methods relying on a set of predefined rules, while others utilizing sophisticated machine learning models trained through parameter optimization. This distinction divides the efforts in efficient structure learning into two

directions. One direction focuses on optimizing the algorithms used to construct the Bayesian Network (underlying structure) of the data [162], [188], [189], by reformulating their individual components or modifying their sequence of steps. The other direction involves developing novel theoretical frameworks [48], [49], [190] for continuous optimization models, which enable faster learning of the connections between variables in a dataset. The frameworks under this category are designed to improve the computational complexity of machine learning models trained using the augmented Lagrangian (cornerstone of the NOTEARS approach). Such models have cubic complexity $\mathcal{O}(d^3)$, due to evaluating a matrix exponential of $\mathbf{A} \in \mathbb{R}^{d \times d}$, where $d$ is the data variable size, involved in the computation of the acyclicity penalty $h(\mathbf{A})$ at each augmented Lagrangian step. Regardless of whether ESL is applied to an existing algorithm or serves as the foundation for a new approach, it always leads to shorter running times.

The importance of efficiency in structure learning cannot be overstated when working with large or complex datasets, as it can have a substantial effect on the computational time required to obtain results. Specifically, most such methodologies have emerged as a response to the computational bottlenecks found in NOTEARS, seeking to preserve theoretical soundness while enhancing its scalability and convergence properties. Various extensions and adaptations, such as sparsity-aware optimization, low-rank approximations, stochastic gradient updates, and distributed or parallelized computation, have been proposed to accelerate the structure learning process. Moreover, several ESL variants relax or approximate the acyclicity constraint introduced in NOTEARS, reducing computational overhead without significantly compromising the accuracy of the learned causal graph in the process. By substantially lowering the computational costs of model training, these developments enhance the scalability of differentiable causal discovery, allowing its deployment in big-data environments and integration in real-world applications.

In essence, the pursuit of efficiency within the NOTEARS framework transcends mere algorithmic refinement, constituting a fundamental prerequisite for the widespread applicability of causal discovery in real-world, data-intensive contexts. By facilitating scalable and computationally tractable inference of causal structures from high-

dimensional observational data, Efficient Structure Learning methodologies effectively broaden the practical and theoretical scope of the NOTEARS-based continuous optimization approach. This, in turn, enables the construction of interpretable and generalizable models capable of informing data-driven decision-making, improving predictive accuracy, and advancing scientific understanding across a diverse range of disciplines. As a result, this specific approach to Causal Structure Learning is an intriguing and important area of research. The author contributes to ESL by conducting an efficiency study in the context of the NOTEARS framework, resulting in a significant decrease in computational complexity and training time - more details are provided in Chapter 4.

### 2.3.3 Causal structure learning under the PNL assumption

Identification and interpretation of the causal dependencies expressed in a dataset are crucial aspects of data analysis, which can lead to significant scientific breakthroughs and an increase in related research. Although both play a role in causal studies, it is important to distinguish between them, as they focus on different areas of causality. More specifically, the technique utilized to discover unique cause-and-effect relationships is called causal structure learning, while causal inference focuses on understanding and explaining the nature of causal relations between variables. These two processes have to be executed sequentially, since to infer causal effects from a dataset one must have knowledge of its interdependencies. The author contributes to causal discovery by investigating the application of the post-nonlinear (PNL) model in learning sparse non-parametric structures from tabular data.

In causal structure learning, randomized control trials are still considered the gold standard for identifying dependencies in data. Such experiments involve manipulation through interventions to reduce confounding factors, facilitating the isolation of specific variable effects on a dataset. Unfortunately, tests of this nature are often impractical or even impossible due to ethical, technical or resource constraints. Addressing this issue has resulted in an increasing demand for uncontrolled causal studies. As a result, it is essential to create frameworks that can extract causal relationships from passive observational data.

Over the past few decades, various methods for observational causal discovery have been developed across numerous scientific fields, including bioinformatics [191], [192], [193], economics [194], biology [195], [196], climate science [197], [198], and social sciences [199]. Many of these studies are based on independence-based algorithms such as PC [28], FCI [29], and RFCI [135] or discrete score-based approaches like GES [146], GES-mod [147], and GIES [84]. In addition, continuous optimization techniques, including NOTEARS [12], DAG-GNN [35], GraN-DAG [36] and DAG-WGAN [75] are also widely used. These methodologies for causal structure learning have undergone rigorous testing, with substantial empirical evidence demonstrating their ability to generate meaningful graphical representations of dependencies within datasets. However, strong performance does not guarantee the structure identifiability (see Definition 1) of causal models. Under such circumstances, multiple directed acyclic graphs can be used to define the same probability distribution, making it impossible to determine its true causal structure.

The inability to correctly identify the ground truth graph of a dataset can have significant consequences. For instance, conducting data analysis with misidentified causal relationships can infer incorrect conclusions about cause and effect. This can lead to various limitations, such as suboptimal decision-making, bias in estimation and inaccurate predictions, just to name a few. To mitigate the impact of these drawbacks, observational studies often assume Structural Causal Models (SCM), parameterized with various equations, to guarantee a unique causal graph can be recovered from a given probability distribution [200]. At present, there are numerous works applying different (mostly) identifiable models to discover causality from observational data. Standout examples include the well-researched linear non-Gaussian acyclic model (LiNGAM) [31], the additive noise model (ANM) [98], which accommodates for limited non-linearity by applying transformations to data variables but assuming the dependencies between them are additive, and the post-nonlinear model (PNL) [32], which is suited for exploring complex non-linear relationships. All of these models have been utilized in both bivariate and multivariate causal structure learning.

Among the previously mentioned SCM, the PNL accounts for both nonlinearities

and distortions when describing how cause(s) influences effect(s) [201]. As a result, it is considered to be a generalization of less complex models, such as LiNGAM and ANM, capable of capturing causal dependencies exhibited in real-world evidence data. Mathematically, the post-nonlinear model can be expressed as follows:

$$\tilde{X} := g_j(f_j(Pa_j) + \mathcal{Z}_j), \forall j, \mathcal{Z}_j \perp\!\!\!\perp f_j(Pa_j), \tag{2.10}$$

where $Pa_j$ denotes the parent(s) of the $j$th data variable and $\mathcal{Z}$ represents a noise vector independent of $Pa_j$. Additionally, the formulation of equation (2.10) indicates that the post-nonlinear model is defined by two functions: 1) an initial function $f_j$ applying nonlinearity to the parent data variables, with subsequent noise being added to all of the transformations; and 2) an invertible (possibly nonlinear) function $g_j$ applying an additional layer of transformations to the result. Despite the PNL model being among the most realistic SCM for representing causal mechanisms in real-world data distributions, it has received less attention than other identifiable models due to difficulties associated with its post-nonlinearity and invertibility constraints.

Several methods have been proposed to explore causal structure learning based on the post-nonlinear assumption. Examples of such models include AbPNL [202], which utilizes an autoencoder architecture to simultaneously learn a function and its inverse by minimizing a combination of independence and reconstruction losses. This is a general approach which applies PNL to causal discovery in both bivariate and multivariate settings. Another similar method, DeepPNL [203] uses multilayer perceptrons to learn both functions associated with the PNL model. Meanwhile, CAF-PoNo [175] investigates the application of normalizing flows to optimize the invertibility constraint of post-nonlinear SCM. Rank-PNL [204] introduces a rank-based approach to estimate the invertible function of the structural causal model. Most recently developed, MC-PNL [176] focuses on achieving efficient structure learning under the PNL assumption by modeling non-linear causal relationships using a novel objective function and block coordinate descent optimization. Despite the latest advances in PNL estimation, learning cause-and-effect relationships with this identifiable causal model remains an ongoing research effort. The author contributes to PNL-based causal discovery by expanding

upon the already exceptional functionality of DAG-Notears-MLP (described in Section 2.4.3) to incorporate structure learning under the PNL causal model assumption - see Chapter 5 for more information.

### 2.3.4 Impact of Structural Model Assumptive Complexity on Causal Discovery

Currently, most state-of-the-art methods for causal discovery rely on the application of a single identifiable causal model to extract dependencies from observational data. However, this approach introduces a significant limitation, as such causal structure learning algorithms cannot verify whether the chosen model accurately represents the true structure of the dataset. Addressing this issue is crucial because misidentifying causal relationships can lead to flawed data analysis, which introduces the problems mentioned earlier in this section.

Assuming multiple structural causal models (SCM) instead of a single one in causal discovery from observational data offers significant advantages in terms of identifiability, robustness, and generality. Under a single-SCM framework, causal discovery is inherently nondeterministic, as a DAG may yield identifiable causal mechanisms, but cannot guarantee the best possible description of the underlying structure of the observational distribution. By contrast, a multi-SCM approach, where distinct causal models represent different semi-parametric assumption sets, introduces distributional variation that can help disentangle genuine causal effects from spurious correlations. This allows researchers to identify invariant causal mechanisms that remain stable even when aspects of the data-generating process change. Furthermore, leveraging multiple SCM mitigates sensitivity to violations of crucial assumptions such as faithfulness, causal sufficiency, or absence of confounding, which may not universally hold in real-world data. As a result, such a framework enhances both the robustness and external validity of inferred causal structures, yielding inferences that are more resilient to model specification and more reflective of the underlying generative mechanisms defining the input observational distribution.

Unlike many approaches that limit the discovery of causality to a single model,

the author can apply their novel methodology (see Chapter 5) to perform structure learning under multiple semi-parametric assumptions. As a result, given any dataset, the author can experimentally identify the most suitable structural causal model for modeling interdependencies in observational data.

### 2.3.5 Application of causal discovery in tabular data synthesis

Tabular data stands out as one of the most widespread mechanisms for representing raw information in an organized manner. Its versatile structure facilitates the representation of features in a variety of formats (continuous, discrete and mixed), making it well-suited for analysis and interpretation. As a result, tabular data plays a pivotal role in extracting insights, essential for informing the decision-making process in fields such as medicine [205], finance [206] and business [207]. However, tabular datasets may sometimes be incomplete, leading to limited availability and poor quality. This weakness raises concerns about the validity of any inferences drawn from such data [208].

Historically, efforts have been made to mitigate the adverse effects of sparse tabular data by synthesizing additional samples modeled using deep neural networks. This approach, known as data generation, employs (deep) generative model optimization [53], [52], [51], aiming to establish an implicit probability distribution that matches the original distribution through end-to-end training. The majority of frameworks for generating tabular data fall under the following two categories [209]: 1) synthesis, which aims to create samples resembling real data (*fidelity*), while ensuring that the distribution of the generated data covers the original distribution as comprehensively as possible (*diversity*); and 2) imputation, which involves generating samples without missing values based on incomplete input data. The author extends the research conducted in tabular data *synthesis* by exploring the concept of causal awareness in Deep Generative Models (DGM).

Currently, a considerable volume of scientific literature discusses the synthesis of tabular data using DGM, categorizing all models utilized in this field into traditional and causal-based approaches. The former relies on statistical patterns and correlations to predict new samples closely resembling the input data. Meanwhile, the latter sim-

ulates the generation process of the original dataset by learning the underlying causal relationships between its variables. Both methodologies have yielded promising results. In the past, works such as MedGAN [210] and CorGAN [211] have demonstrated impressive efficacy in handling Electronic Health Records (EHR) [212] with heterogeneous data types (continuous, discrete and mixed). Furthermore, PATE-GAN [213] focused on addressing privacy concerns related to medical data generation. CTGAN and TVAE proposed by [214] employ a conditional generator to mitigate the limitations of mode collapse and class imbalance. Other models [215], [216] extend the functionality of CTGAN by incorporating a Neural Ordinary Differential Equation (NODE) [217] structure to produce fair synthetic samples at the cost of computational complexity. The outputs of the aforementioned models have undergone rigorous statistical analysis, proving their sufficiency for application in classification and regression problems. However, understanding and interpreting the mechanisms necessary to produce them is a challenging task for people. This lack of explainability presents a significant limitation, raising questions regarding the reliability of the results generated by deep generative models.

Recently, traditional DGM have experienced an improvement in tabular data generation capabilities by leveraging causal inference. Early research into causality [94] suggests at its significance in producing realistic samples by learning the relationships between variables and facilitating the description of their causal dependencies. More specifically, in the context of generative modeling, preserving causation rather than merely modeling correlations provides a principled foundation for generating data that reflects the true underlying mechanisms of its probability distribution, rather than reproducing superficial statistical patterns. Traditional generative models often capture correlational structures without understanding why variables relate, leading to poor generalization under distributional changes. On the other hand, causally-aware generative models explicitly represent the direction and structure of dependencies among variables, allowing for interpretable, modular, and interventionally consistent data generation. This novel causal paradigm enables counterfactual reasoning, supports robust simulation of unseen scenarios, and enhances transferability across domains. There-

fore, preserving causation in generative modeling yields models that are not only more explainable and reliable but also capable of synthesizing data that faithfully replicates the real-world processes from which it arises.

Several causality-based DGM, such as DECAF [218], TabFairGAN [219] and Causal-TGAN [220], have produced tabular datasets by employing this novel methodology. CausalGAN [171] and CausalVAE [170] incorporate causal dependencies into label generation, yielding high-quality images. Alternatively, GCNN [73], DAG-GNN [35], DEAR [164], and DiffAN [221] prioritize causal discovery, producing accurate structures at the cost of data quality and sparsity.

Unfortunately, both the causal structure learning and the tabular data synthesis approaches face challenges in their sample generation techniques. In the case of DAG-GNN, DEAR and DiffAN, incorporating Mean Squared Error (MSE) or its variations (e.g. NLL) produces over-simplified latent representations, resulting in latent collapse during sampling. On the other hand, models such as Causal-TGAN, DECAF, CausalGAN and GCNN assume a known or externally learned causal representation to produce synthetic samples. Working with real-world data makes such sampling procedures unreasonable as they require prior knowledge of the underlying causal structure or the application of independent algorithms to identify the causality within datasets and assess its accuracy before utilizing it for tabular data synthesis.

Recent progress in generative modeling, including Digital Twins and transformer-based multi-attention networks [222], offers novel methodologies to capture complex data relationships. Digital Twin models focus on creating virtual representations of real-world systems, making them particularly useful for generating synthetic data. Similarly, attention-based architectures, such as multi-attention networks, dynamically assess and prioritize dependencies between variables. As generative models become increasingly popular, integrating them with causal structure learning within a unified framework holds great promise for producing more accurate and interpretable data while preserving underlying causal structures [223].

The author resolves the issues with causal discovery and tabular data synthesis by performing the two processes simultaneously using transfer learning to convey informa-

tion between multiple deep neural network instances - further elaboration is provided in Chapter 5.

## 2.4 Relevant Preceding Frameworks

This section provides a brief overview of models closely related to the research described in this thesis. These methods have been compared to other benchmark approaches in the field, such as NOTEARS [12] and GraN-DAG [36], using the Structural Hamming Distance (SHD) metric and have demonstrated capability to produce good results. In particular, the following algorithms are explained: DAG-GNN [35], DAG-NoCurl [48] and DAG-Notears-MLP [38].

### 2.4.1 DAG-GNN

DAG-GNN [35] is a continuous optimization score-based model for causal structure learning that combines a variational autoencoder and graph neural networks. This novel approach extends the capabilities of NOTEARS by handling both linear and non-linear, continuous and discrete data. The model uses an explicit weighted adjacency matrix $\mathbf{A}$ as a learnable parameter and causal structure learning is achieved by minimizing the Evidence Lower BOund (ELBO) [61].

DAG-GNN consists of two models encoder *Enc* and decoder *Dec* each instantiated by shallow neural networks. Both modules can be denoted as

$$\begin{aligned} Enc &\equiv Z = \mathbf{F}_4((I - \mathbf{A}^T)\mathbf{F}_3(\mathbf{X})) \\ Dec &\equiv \tilde{\mathbf{X}} = \mathbf{F}_2((I - \mathbf{A}^T)^{-1}\mathbf{F}_1(Z)), \end{aligned} \tag{2.11}$$

where $\{\mathbf{F}_3, \mathbf{F}_4\}$ and $\{\mathbf{F}_1, \mathbf{F}_2\}$ are the MLPs for the encoder and decoder respectively. In addition, the authors of this model have improved upon the acyclicity constraint of the NOTEARS model making it more efficient, but at the cost of numerical stability. The formulation of their constraint can be expressed as $tr[(I + \alpha\mathbf{A} \circ \mathbf{A})^d] - d = 0$, where $d$ represents the number of nodes in the graph, $\alpha$ is a hyper-parameter, $\circ$ denotes

the Hadamard product of matrices, $\mathbf{A}$ represents the weighted adjacency matrix, $tr$ is the trace exponential and $I$ denotes the identity matrix. Essentially, the acyclicity constraint is a trace exponential that counts the number of cycles detected in the learned graph. Its purpose is to progressively restrict the search space of the graph until the trace yields 0, indicating the absence of cycles and ensuring that the output is a DAG. Moreover, despite its simple architecture, the algorithm is more sophisticated compared to NOTEARS. To ensure proper optimization of the score function and the acyclicity constraint, the authors treat the training of the model as a constrained continuous optimization problem, which can be solved using an augmented Lagrangian approach [47].

### 2.4.2 DAG-NoCurl

DAG-NoCurl [48] is an efficient structure learning framework based on the application of graph Hodge theory [224] and Helmholtz-Hodge Decomposition [225]; [226]; [227] in a causal discovery setting. According to theory, a DAG consists of a harmonic, a divergence-free, and a curl-free component, which represents an acyclic graph. Based on this knowledge, the authors of DAG-NoCurl developed their own theorem, enabling the mapping between weighted adjacency matrices and curl-free components. This means that a recovered graph from this method will have directionality due to its weighted adjacency matrix and acyclicity due to its curl-free component. This leads to the first contribution of DAG-NoCurl, which is an alternative formulation of the DAG search space capable of supporting causal discovery without the use of an augmented Lagrangian. For a more detailed analysis, please refer to [48] however, in short, their Theorem 2.1 proves that both DAG search spaces are equivalent.

The second contribution of this work involves the development of a model that can navigate the equivalent DAG search space and recover graphs from it. To accomplish this, the resulting DAG learning algorithm allows a weighted adjacency matrix $\mathbf{A}$ to be represented as the Hadamard product of a skew-symmetric matrix $W$ and the gradient of a potential function on graph vertices $grad(p)$. By learning this new representation of $\mathbf{A}$, optimization can be performed directly in the DAG space, eliminating the need

for explicit acyclicity constraints and the expensive computation of the augmented Lagrangian. The new model consists of three steps: 1) computing an initial prediction $\mathbf{A}^{pre}$, 2) projecting the initial prediction into the equivalent DAG search space, and 3) obtaining a final DAG $\mathbf{A}^* = W \circ ReLU(grad(p))$.

In order to obtain an initial prediction $\mathbf{A}^{pre}$, the authors solve an unconstrained continuous optimization problem $F(\mathbf{A}, \mathbf{X})$ s.t $\lambda h(\mathbf{A}) = 0$, where, $h(\mathbf{A})$ represents the explicit acyclicity constraint applied to the weighted adjacency matrix $\mathbf{A}$, and $\lambda$ denotes the Lagrangian multiplier, which is set to 10 based on empirical evidence from the hyper-parameter study conducted by the authors. The applied acyclicity constraint is from [35], as it offers faster computation. However, the original constraint proposed by [12] can also be employed. Afterward, the initial prediction is subjected to a thresholding process, with a value of 0.3 being used.

In the second step, the authors project the equivalent representation of $\mathbf{A}^{pre}$ into the new DAG search space. This is achieved by computing the topological ordering $p$ of the initial prediction and using $\mathbf{A}^{pre}$ and $p$ to obtain $W \circ ReLU(grad(p))$. The projection step of DAG-NoCurl allows the direct recovery of $\mathbf{A}^*$ from the DAG search space, without the need for acyclicity constraints. To accomplish this, the authors only optimize $W$ and use a fixed value for $p$ when solving the second unconstrained continuous optimization problem. By keeping $p$ constant, the causal structure remains unchanged, and solving for $W$ refines the strength of the connections within $W \circ ReLU(grad(p))$. This guarantees that the output will be a DAG, but it does not ensure that the distance between the output and the ground truth is minimized, which is a limitation of the approach.

### 2.4.3 DAG-Notears-MLP

DAG-Notears-MLP [38] is another extension of the original NOTEARS model developed by its authors. It is an updated and more generalized version of its predecessor, commonly referred to in the causal structure learning community as NOTEARS+ [185]. The main contributions of this framework lie in its architecture, which includes a new acyclicity constraint and a novel approach to learning weighted adjacency matrices

implicitly. The model is a neural network that consists of an input layer $L_0$ and a sequence of dynamically instantiated locally connected layers $L = \{\alpha(L_1), ..., \alpha(L_d)\}$, where $d$ denotes the number of layers and $\alpha$ is the activation function (e.g. ReLU) applying nonlinearity to each layer. The model is trained using stochastic gradient descent optimization [228], a popular algorithm for learning neural networks. The acyclicity constraint is imposed during training, and the implicit weighted adjacency matrix $W \in \mathbb{R}^{d \times d}$ is obtained from the $L_0$ layer of the Multi-Layer Perceptron (MLP).

Since the model learns the causal graph implicitly, the acyclicity constraints mentioned earlier [12], [35] cannot be applied. To address this issue, the authors of DAG-Notears-MLP propose a new constraint based on partial derivatives [229], which is defined as follows

$$h(W(f)) = 0, [W(j)]_{kj} := ||\partial_k f_j||_2. \tag{2.12}$$

In equation (2.12), $W$ is the weighted adjacency matrix, $\partial_k$ is the partial derivative of $f_j$ with respect to the $k^{th}$ variable and $||.||_2$ is the Ridge Regression norm. Furthermore, the authors investigate the generalization of the model by incorporating non-parametric assumptions. Under such settings, the model assumes the general form of $\mathbb{E}[X_j|X_{Pa_j}] := \mathbb{E}_{\mathcal{Z}}(f_j(X, \mathcal{Z}))$, which encompasses a variety of SCM including additive noise models, index models, generalized linear models and others. Elaboration on how DAG-Notears-MLP performs causal structure learning in each of these cases is provided in their paper [38].

The model has demonstrated an ability to produce good results against other leading models in the field. However, it uses the Mean Squared Error (MSE) loss function as the basis for its parameter optimization process. As a result, DAG-Notears-MLP inherits the limitations of MLE-based approaches in causal discovery, leading to inaccuracies in structure learning with increase in data variable size or introduction of noisy input data.

# Chapter 3

# Adversarial Variational Inference for Causal Discovery

This chapter presents the development of a model that investigates the impact of Wasserstein generative adversarial training on Variational Autoencoder (VAE) architectures within the domain of causal structure learning. Its sections focus on the combination of GAN and VAE for causal structure learning, while also documenting the outcomes, strengths, and limitations of this approach. This description is followed-up by a brief discussion regarding potential enhancements to the base model such as Disentangled Representation Learning (DRL) and Efficient Structure Learning (ESL). The content explored in this chapter has been previously published, and the publications of the author can be found in Section 1.7.

## 3.1 Background Knowledge

As mentioned in Section 2.1.2, the process of discovering causal relationships involves learning the components of a Bayesian Network (BN) [90]. Given a set of observational samples $\mathbf{X} = \{\mathbf{X}_1, ..., \mathbf{X}_n\}$ and latent variables (i.e. hidden) $Z = \{Z_1, ..., Z_{\mathbf{N}}\}$, it is theoretically possible to obtain the correct values for the building blocks of BN by directly applying Bayes's theorem [230] to compute the true posterior distribution $P(Z|\mathbf{X})$:

$$P(Z|\mathbf{X}) = \frac{P(\mathbf{X}|Z)P(Z)}{P(\mathbf{X})} \tag{3.1}$$

Unfortunately, the computation of $P(Z|\mathbf{X})$ using (3.1) is generally intractable, which is why researchers use an approximate solution known as a variational distribution $Q(Z|\mathbf{X}) \approx P(Z|\mathbf{X})$. This approach, called variational inference, provides a practical solution for obtaining a posterior distribution and forms the theoretical foundation for VAE-based models.

Bayesian Networks (BN) can be instantiated through Variational Autoencoders. However, it is important to note that VAE are built on artificial neural networks, which have some differences compared to probabilistic graphical models. The key distinction lies in the assignment of content to the weights. In Bayesian Networks, weights are initialized using probability distributions, while in basic neural networks, each weight is assigned a scalar value. By applying variational inference to a Bayesian network, the focus is on directly modeling probability distributions rather than optimizing individual weight values. This approach leads to the development of various models based on variational Bayes that are capable of learning the structure of BN.

The application of Variational Autoencoders (VAE) in the domain of causal structure learning represents a significant advancement in research. Their importance is second only to the development of the DAG-NOTEARS framework [12], which allows for causal discovery using black-box models. Nevertheless, variational inference was used to facilitate numerous observational studies. In fact, one of the first machine learning models capable of extracting causal relationships from data is known as DAG-GNN [35]. This approach is based on the VAE architecture and has the ability to handle different types of data, such as continuous, vector and discrete (see Section 2.4.1 for more details). Other models that utilize variational Bayes include Graphite [163], which can handle high-dimensional data, Disentangled gEnerative cAusal Representation Learning (DEAR) [164], which works with image data and a known ground truth graph to perform supervised causal structure recovery, Amortized Causal Discovery (ACD) [231], which discovers causality from time-series data, V-CDN [173], which recovers causal structures from video formats, Causal Variational Autoencoder

(CausalVAE) [170], which is a nonlinear extension of NOTEARS capable of working with tabular data, Imputated Causal Learning (ICL) [232], which handles missing data to learn causal relationships, and VI-DP-DAG [49], which is a causal structure learning model that efficiently discovers causal graphs using the VAE architecture. The models mentioned above serve as evidence for the popularity of variational inference in causal structure learning. Despite their numerous contributions, there are still many unexplored potential applications of VAE in causality learning.

Unfortunately, as previously mentioned in Section 2.3.1, traditional MLE-based generative modeling approaches with latent variables (including the VAE framework) possess inherit limitations stemming from their focus on individual data point optimization. In contrast, incorporating the Wasserstein distance as a regularizer refocuses the learning objective from aligning specific data points to minimizing the distance between entire data distributions. This shift enables the capture of the global geometry and richer statistical characteristics of the data, such as variance, skewness, and multimodality, facilitating a more faithful emulation of the underlying causal mechanisms. Building on this foundation enables the development of hybrid models, which leverage adversarial training and a reconstruction process to minimize distributional discrepancies between real and generated data, and recover accurate causal relationships. This advancement marks an exciting and promising research direction in causal structure learning, particularly through the exploration of VAE-GAN architectures for causal discovery.

## 3.2 Causality learning with hybrid generative modeling

The objective of the study is to explore the influence of the Wasserstein distance on variational inference in the context of causal discovery. The research aims to demonstrate the practical significance of this metric by providing empirical support for the hypothesis: " *Will incorporating Wasserstein-1 lead to improved causal recovery through a generative adversarial framework that is trained to synthesize realistic data samples?*". To achieve this goal, the author has developed a novel hybrid generative modeling

framework called DAG-WGAN [75], which is based on the VAE-GAN architecture [62].

The model proposed in this study combines a variational autoencoder and a WGAN-GP architecture. It achieves this by utilizing an encoder-decoder pair for causal discovery and a critic to calculate the Wasserstein distance between the output of the decoder and the input data. To ensure that the recovered causality does not include any cycles, the author incorporates the explicit acyclicity constraint from [35]. The algorithm learns to explicitly model the cause and effect between variables while synthesizing data samples based on recovered causal structures and parameter optimization through end-to-end training.

Extensive testing has been conducted on the model, comparing it to the current state-of-the-art. The experimental results indicate that DAG-WGAN outperforms other models by a significant margin, when dealing with large data variable sizes. In particular, when data attributes have a high cardinality, the causal graphs learned using DAG-WGAN are more accurate than those produced by other models. Additionally, the generated data samples from DAG-WGAN are less noisy and more realistic compared to samples from other data-generating models. The capabilities of the model have been demonstrated on various data types, including linear, non-linear, continuous, and discrete. Furthermore, the method has been tested using data produced from multiple Structural Equation Models (SEM) [233], namely instances of Additive Noise Models (ANM) and Post-Nonlinear Models (PNL). The experimental results suggest that incorporating the Wasserstein distance metric supports causal discovery in the data generative process when working with observational samples produced by applying different SEM assumptions.

Compared to other models in the field, DAG-WGAN has the following advantages:

- **Realistic causal structure learning and data generation** - The model simultaneously performs causal structure learning and data generation to synthesize realistic samples with preserved causality.
- **Multiple data types** - The model is an extension of the original NOTEARS framework capable of working with a variety of data types.

- **Multiple structural equation models** - DAG-WGAN can work with observational data synthesized using instances of additive noise and post-nonlienar models.

### 3.2.1 Model Architecture & Training

This section provides a detailed explanation of the inner workings of DAG-WGAN, focusing on its architecture and training algorithm. The proposed model combines a Variational Autoencoder (VAE) and a Wasserstein Generative Adversarial Network with Gradient Penalty (WGAN-GP). Additionally, the framework incorporates causal discovery by introducing an explicit weighted adjacency matrix $\mathbf{A}$ as a learnable parameter and an acyclicity constraint. For a visual representation of the model architecture, please refer to Figure 3.1. In essence, the model comprises three neural networks that collaborate to recover causal structures and synthesize data samples: 1) an encoder computes the latent representations of the input data; 2) a decoder reconstructs new data samples from the latent representations generated by the encoder; 3) a discriminator ensures that the new data samples are realistic by minimizing the distance between the output of the decoder and the input data.

The decoder plays a crucial role in connecting the other two components of the model. Firstly, it collaborates with the encoder in the variational autoencoder architecture to recover causal structures from observations. Secondly, the decoder also works alongside the discriminator in the WGAN-GP component to generate realistic data samples. This connection between the encoder, decoder, and discriminator is also evident in the training process of DAG-WGAN. The encoder and discriminator are trained using reconstruction and adversarial loss, respectively, while the decoder parameters are optimized using both loss terms. The motivation behind the formulation of this hybrid generative modeling framework is the successful application of VAE-GAN to capture data and feature representations more effectively [62]. DAG-WGAN extends the capabilities of NOTEARS by incorporating multiple data types (e.g., continuous and categorical) and structural equation models (i.e., Additive Noise Models (ANM) and Post-Nonlinear Models (PNL)). For more detailed information, see Section 3.2.3.

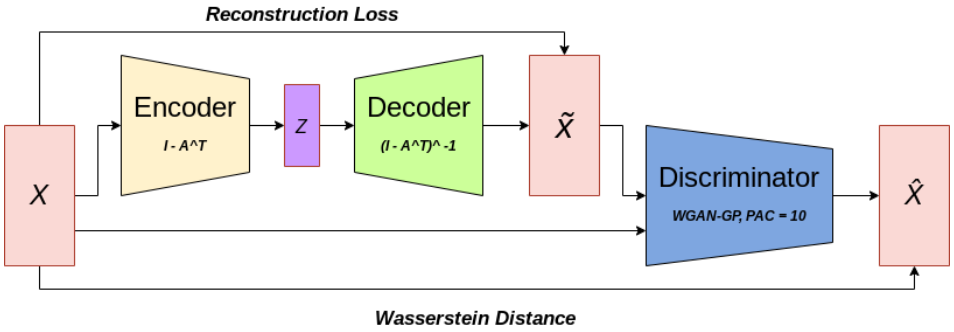



Figure 3.1: DAG-WGAN employs a hybrid architecture composed of two primary components: (1) a Variational AutoEncoder (VAE) and (2) a Wasserstein Generative Adversarial Network with Gradient Penalty (WGAN-GP). The VAE component follows the structure of the DAG-GNN model. Therefore, the key distinction between DAG-WGAN and DAG-GNN is the integration of the additional WGAN-GP architecture, which is implemented through the Discriminator module.

### Variational Autoencoder architecture

Variational autoencoders consist of a pair of interconnected networks, namely an encoder and a decoder. In the default scenario, the encoder $Enc$ takes input $\mathbf{X}$ and generates a latent variable $Z$ by learning a variational posterior $Q_\phi(Z|\mathbf{X})$. On the other hand, the decoder $Dec$ computes a conditional likelihood distribution $P_\theta(\mathbf{X}|Z)$, which is utilized to generate reconstructed samples $\tilde{\mathbf{X}}$. The following mathematical representation captures the aforementioned processes:

$$Enc \equiv \mathbb{E}_{\mathbf{X}\sim P(\mathbf{X})}[Q_\phi(Z|\mathbf{X})] \Rightarrow Z \qquad Dec \equiv \mathbb{E}_{Z\sim Q_\phi(Z|\mathbf{X})}[P_\theta(\mathbf{X}|Z)] \Rightarrow \tilde{\mathbf{X}}, \quad (3.2)$$

where $\phi$ and $\theta$ are the model parameters of $Enc$ and $Dec$ respectively. Moreover, the latent representation $Z$ undergoes regularization to reduce over-fitting, ensuring the latent space contains meaningful information.

DAG-WGAN facilitates the causal structure learning process by assuming structural equations for both $Enc$ and $Dec$ architectures. This allows the encoding of causality in the latent representations that are utilized to reconstruct the data. In order to accomplish the simultaneous recovery of causality and generation of data, the author

modifies (3.2) as follows:

$$
\begin{aligned}
&Enc \equiv Z = \mathbb{E}_{\mathbf{X}\sim P(\mathbf{X})}[Q_\phi(Z|\mathbf{F}_4((I-\mathbf{A}^T)\mathbf{F}_3(\mathbf{X})))] \\
&Dec \equiv \tilde{\mathbf{X}} = \mathbb{E}_{Z\sim Q_\phi(Z|\mathbf{F}_4((I-\mathbf{A}^T)\mathbf{F}_3(\mathbf{X})))}[P_\theta(\mathbf{X}|\mathbf{F}_2((I-\mathbf{A}^T)^{-1}\mathbf{F}_1(Z)))],
\end{aligned} \tag{3.3}
$$

where $(I-\mathbf{A}^T)$ and $(I-\mathbf{A}^T)^{-1}$ are the structural equations for the encoder and the decoder, respectively. $\mathbf{X} \in \mathbb{R}^{n\times d}$ represents observational samples from the distribution $P(\mathbf{X})$, while $Z \in \mathbb{R}^{\mathbf{N}\times d}$ is a latent variable obtained from the distribution $Q_\phi(Z|\mathbf{F}_4((I-\mathbf{A}^T)\mathbf{F}_3(\mathbf{X})))$. The reconstructed data, denoted as $\tilde{\mathbf{X}} \in \mathbb{R}^{n\times d}$, is sampled from the distribution $P_\theta(\mathbf{X}|\mathbf{F}_2((I-\mathbf{A}^T)^{-1}\mathbf{F}_1(Z)))$. The matrix $\mathbf{A} \in \mathbb{R}^{d\times d}$ is an explicitly defined weighted adjacency matrix, with each node corresponding to a variable in $\mathbf{X}$. The functions $\mathbf{F}_1$ to $\mathbf{F}_4$ are parameterized and used to apply (non)linear transformations on $Z$ and $\mathbf{X}$. The architecture is designed in a way that the components in the decoder ($Dec$) can invert the operations performed by the components in the encoder ($Enc$).

**WGAN-GP architecture**

WGAN-GP are a type of generative model that utilize the Wasserstein distance metric. These models consist of two networks, a discriminator $D$ and a generator $G$, which compete against each other to generate realistic data samples. DAG-WGAN deviates from the standard WGAN-GP model by incorporating the decoder from the Variational Autoencoder (VAE) architecture as the generator. Additionally, a critic is employed to calculate the adversarial loss with its gradient penalty. The design of discriminator is based on the PacGAN framework [234] and aims to address the issue of mode collapse. The architecture of $D$ can be described as follows:

$$
\hat{\mathbf{X}} = MLP(\tilde{\mathbf{X}}, \mathbf{X}, leaky-ReLU, Dropout, GP, pac), \tag{3.4}
$$

where $\tilde{\mathbf{X}}$ is the reconstructed data and $\mathbf{X}$ are observational data samples. Leaky-ReLU is the activation function for the model with its negative slope set to 0.01. Dropout [235] is set to 0.5, which accounts for stability and prevents over-fitting. GP is the gradient

penalty term used in the standard WGAN-GP [116] configuration. Pac is a concept related to PacGAN [234] designed to dampen the effect of mode collapse when working with discrete data.

### Training algorithm

The architecture of the variational autoencoder is trained by merging two components, which are the reconstruction and regularization loss, as explained in (2.3). The approximation of the first component is computed using the Gaussian Negative Log-Likelihood (GNLL) [236].

$$\begin{aligned}\mathcal{L}(\mathbf{X}, \tilde{\mathbf{X}}) &= \mathbb{E}_{Z\sim Q_\phi(Z|\mathbf{F}_4((I-\mathbf{A}^T)\mathbf{F}_3(\mathbf{X})))}[\log P_\theta(\mathbf{X}|\mathbf{F}_2((I-\mathbf{A}^T)^{-1}\mathbf{F}_1(Z)))] \\ &\approx -\frac{1}{2}\left[\frac{(\tilde{\mathbf{X}}-\mu(\mathbf{X}))^2}{\sigma(\mathbf{X})^2} + \log\sigma(\mathbf{X})^2\right]\end{aligned} \tag{3.5}$$

The second term, referred to as KL-Divergence, helps prevent overfitting and ensures meaningful information is encoded in the latent space.

$$\begin{aligned}regularizer &= \mathbb{E}_{\mathbf{X}\sim P(\mathbf{X})}[D_{KL}(Q_\phi(Z|\mathbf{F}_4((I-\mathbf{A}^T)\mathbf{F}_3(\mathbf{X})))||P(Z))] \\ &\approx -\frac{1}{2}\left[\log\sigma(Z)^2 - \left(\mu(Z)^2 - \sigma(Z)^2\right) + 1\right]\end{aligned} \tag{3.6}$$

In both (3.5) and (3.6), $\mu$ denotes the mean and $\sigma$ is the standard deviation. Together the two terms form the objective function for training the VAE component of DAG-WGAN:

$$\begin{aligned}\mathbf{R}_{loss}(\mathbf{X}, \tilde{\mathbf{X}}, Z) &= -\mathbb{E}_{Q_\phi(Z|\mathbf{F}_4((I-\mathbf{A}^T)\mathbf{F}_3(\mathbf{X})))}[\log P_\theta(\mathbf{X}|\mathbf{F}_2((I-\mathbf{A}^T)^{-1}\mathbf{F}_1(Z)))] \\ &\quad + \beta\mathbb{E}_{\mathbf{X}\sim P(\mathbf{X})}[D_{KL}(Q_\phi(Z|\mathbf{F}_4((I-\mathbf{A}^T)\mathbf{F}_3(\mathbf{X})))||P(Z))] \\ &\approx -\left(-\frac{1}{2}\left[\frac{(\tilde{\mathbf{X}}-\mu(\mathbf{X}))^2}{\sigma(\mathbf{X})^2} + \log\sigma(\mathbf{X})^2\right]\right) \\ &\quad + \beta\left(-\frac{1}{2}\left[\log\sigma(Z)^2 - \left(\mu(Z)^2 - \sigma(Z)^2\right) + 1\right]\right),\end{aligned} \tag{3.7}$$

where $\beta$ is a hyper-parameter from [237], controlling the influence of the KLD. To account for discrete data the reconstruction loss term in (3.7) is replaced by Cross-

Entropy Loss (CEL) [238]:

$$\begin{aligned}\mathcal{L}(\mathbf{X}, \tilde{\mathbf{X}}) &= \mathbb{E}_{Z\sim Q_\phi(Z|\mathbf{F}_4((I-\mathbf{A}^T)\mathbf{F}_3(\mathbf{X})))}[\log P_\theta(\mathbf{X}|\mathbf{F}_2((I-\mathbf{A}^T)^{-1}\mathbf{F}_1(Z)))] \\ &\approx -\sum_{c=1}^{N}(\mathbf{X}_c \log(\tilde{\mathbf{X}}_c)),\end{aligned} \tag{3.8}$$

where $N$ is the number of categories $c$ present within the data.

Meanwhile, the discriminator $D$ and the generator $Dec$ forming the WGAN-GP architecture are trained via the following adversarial loss term:

$$\begin{aligned}\mathbf{D}_{loss} &= \underbrace{\mathbb{E}_{\tilde{\mathbf{X}}\sim\mathbb{P}_g}[D(\tilde{\mathbf{X}})] - \mathbb{E}_{\mathbf{X}\sim\mathbb{P}_r}[D(\mathbf{X})]}_{\text{Critic loss}} + \underbrace{\lambda\mathbb{E}_{\hat{\mathbf{X}}\sim\mathbb{P}_{\hat{\mathbf{X}}}}[(||\nabla_{\hat{\mathbf{X}}}D(\hat{\mathbf{X}})-1||)^2]}_{\text{Gradient penalty}} \\ \mathbf{G}_{loss} &= \mathbb{E}_{Z\sim Q_\phi(Z|\mathbf{F}_4((I-\mathbf{A}^T)\mathbf{F}_3(\mathbf{X})))}[D(Dec(Z))],\end{aligned} \tag{3.9}$$

where $Z$ is the output of the encoder $Enc$. The hyper-parameter $\lambda$ is responsible for determining the strength of the gradient penalty applied to the Wasserstein distance. In the context of the DAG-WGAN model, the distribution $\mathbb{P}_r$ corresponds to the distribution $P(\mathbf{X})$, while $\mathbb{P}_g$ is equivalent to $P_\theta(\mathbf{X}|\mathbf{F}_2((I-\mathbf{A}^T)^{-1}\mathbf{F}_1(Z)))$. The distribution $\mathbb{P}_{\hat{\mathbf{X}}}$ is obtained by sampling uniformly along a straight line between the real data distribution $\mathbb{P}_r$ and the synthetic data distribution $\mathbb{P}_g$.

Neither minimizing the reconstruction loss nor the adversarial loss guarantees the absence of cycles in the weighted adjacent matrix $\mathbf{A}$. To ensure that $\mathbf{A}$ is acyclic, it is necessary to include an explicit acyclicity constraint in the objective function of the model. This constraint, proposed by the author of [35], is expressed as $h(\mathbf{A}) = tr[(I + \alpha\mathbf{A} \circ \mathbf{A})^d] - d = 0$, where $tr$ represents an exponential trace in the DAG search space, $\alpha$ is a positive hyperparameter, $\circ$ denotes the Hadamard product [239], and $d$ is the number of variables in $\mathbf{A}$. The constraint yields a value that represents the number of cycles found in the recovered graph. Through augmented Lagrangian optimization [47], this constraint can be minimized until the value reaches 0, indicating that the recovered graph is a DAG.

DAG-WGAN is trained through the following loss function.

$$
\begin{aligned}
\mathbf{R}_{loss}(\mathbf{X}, \tilde{\mathbf{X}}, Z) = & \underbrace{-\mathbb{E}_{Z\sim Q_\phi(Z|\mathbf{F}_4((I-\mathbf{A}^T)\mathbf{F}_3(\mathbf{X})))}[\log P_\theta(\mathbf{X}|\mathbf{F}_2((I-\mathbf{A}^T)^{-1}\mathbf{F}_1(Z)))]}_{\text{Reconstruction loss}} \\
& + \underbrace{\beta\mathbb{E}_{\mathbf{X}\sim P(\mathbf{X})}[D_{KL}(Q_\phi(Z|\mathbf{F}_4((I-\mathbf{A}^T)\mathbf{F}_3(\mathbf{X})))||P(Z))]}_{\text{Regularization term}} \\
\mathbf{D}_{loss} = & \underbrace{\mathbb{E}_{\tilde{\mathbf{X}}\sim\mathbb{P}_g}[D(\tilde{\mathbf{X}})] - \mathbb{E}_{\mathbf{X}\sim\mathbb{P}_r}[D(\mathbf{X})]}_{\text{Critic loss}} + \underbrace{\lambda\mathbb{E}_{\hat{\mathbf{X}}\sim\mathbb{P}_{\hat{\mathbf{X}}}}[(||\nabla_{\hat{\mathbf{X}}}D(\hat{\mathbf{X}})-1||)^2]}_{\text{Gradient penalty}} \\
\mathbf{G}_{loss} = & -\underbrace{\mathbb{E}_{Z\sim Q_\phi(Z|\mathbf{F}_4((I-\mathbf{A}^T)\mathbf{F}_3(\mathbf{X})))}[D(Dec(Z))]}_{\text{Generator loss}} \\
& \text{s.t} \quad \underbrace{tr[(I+\alpha\mathbf{A}\circ\mathbf{A})^d]-d=0}_{\text{Acylicity constraint}},
\end{aligned}
\tag{3.10}
$$

where the approximations of the reconstruction and regularization loss term are used - see (3.5) and (3.6).

**Remark.** It is not mandatory to use the acyclicity constraint from [35]. In fact, any function that can be continuously optimized to yield $h(\mathbf{A}) = 0$ can be applied to the loss function of DAG-WGAN.

### 3.2.2 Identifiability analysis

To leverage DAG-WGAN for causal structure learning, it is necessary to determine if the model is capable of recovering unique DAG from data. This property, known as structure identifiability (see Definition 1), is associated with every causal discovery model. Generally, the identifiability of a causal graph is influenced by several factors, including 1) a set of assumptions, 2) the choice of loss functions, 3) the use of SEM in the model architecture, and the generation of input data. However, not all combinations of these factors can result in the discovery of unique causal structures, as certain combinations only allow the identification of a DAG up to its CPDAG superset.

In this section, the author investigates the identifiability of DAG-WGAN by examining its model architecture and objective function. The identifiability of the VAE architecture, where causal structure learning is performed, is discussed first. Prior to this research, there was limited evidence on the exploration of the causal identifiability

of the DAG-GNN architecture. The theoretical analysis of DAG-WGAN concludes by providing mathematical intuition on the identifiability of the hybrid loss function in equation (3.10).

In the meantime, the author also acknowledges the causal sufficiency assumption as one of the most fragile in structure learning from real-world data. This assumption states that all common causes of the observed variables are included in the model, meaning there are no unmeasured confounders. In practice, this is rarely true as many systems involve hidden variables or latent factors that influence multiple observed variables (e.g., socio-economic factors in health studies or environmental variables in economic data). Violations of causal sufficiency can lead to spurious causal relationships and incorrect edge orientations in learned causal graphs, which in turn weakens generalization when applying the model to new settings where these hidden confounders vary.

The faithfulness assumption is another that tends to break down frequently in real-world scenarios. It assumes that all observed independencies arise from the underlying causal structure rather than from specific parameter values or coincidences. In complex systems with feedback loops, nonlinear interactions, or finely tuned parameter values, apparent independencies can emerge that are not structurally meaningful. When faithfulness fails, causal discovery algorithms may miss true edges or incorrectly infer independencies, leading to unreliable causal models that fail to generalize across datasets with slightly different parameterizations.

Ultimately, the fragility of causal sufficiency and faithfulness poses the greatest threat to generalization in real-world data. When these assumptions fail, causal conclusions and predictions derived from one context may not transfer to another, emphasizing the need for careful model validation, sensitivity analyses, and the integration of substantive expertise to ensure more reliable and transferable causal insights.

#### Architecture identifiability

To establish the identifiability of the VAE component in DAG-WGAN, it is necessary to determine the type of SEM employed in the generative model as specified in equation

(3.3). The author leverages the understanding that the VAE learns causal structures by performing (non)linear transformations on a generalized version of linear SEM, as documented in [35], to derive the identifiability of the architecture.

**Lemma 3.2.1.** The Structural Equation Model (SEM) used in the decoder architecture $\tilde{\mathbf{X}} = P_\theta(\mathbf{X}|\mathbf{F}_2((I - \mathbf{A}^T)^{-1}\mathbf{F}_1(Z)))$ belongs to the Additive Noise Model category.

*Proof.* The proof of lemma 3.2.1 is available in Appendix A.1. □

The identifiability of Additive Noise Models has been demonstrated in previous research. Specifically, [240] proves in their Proposition 30 that these models are identifiable if the causal mechanisms $\mathcal{F} = \{f_1, ..., f_d\}$ are three times differentiable, non-constant, and non-linear in all of their arguments. This implies that the decoder has the ability to learn unique DAG and thus is identifiable.

### Loss function identifiability

The analysis carried out in Section 3.6 of [35] indicates that if functions $\mathbf{F}_1$ to $\mathbf{F}_4$ are omitted from the inference and generative model, the architectures $Z = (I - \mathbf{A}^T)\mathbf{X}$ and $\tilde{\mathbf{X}} = (I - \mathbf{A}^T)^{-1}Z$ would achieve perfect data reconstruction. In this case, the accuracy of the output data is solely based on the quality of the latent variable $Z$. Therefore, to achieve lossless reconstruction of $\mathbf{X}$, the objective function (ELBO) of the VAE component of DAG-WGAN is simplified to the least squares loss $\mathbb{E}(Z) = \frac{1}{2}||(I - \mathbf{A}^T)\mathbf{X}||_F^2$, assuming that the standard deviation is not learned and set to a constant value of 1. This function has been demonstrated to produce accurate and unique DAG [12] through end-to-end training, thus establishing ELBO as an identifiable variant of the least square loss.

If VAE alone produce unique causal structures of decent quality, then a sensible question to ask is *What is the contribution of adversarial training to the learning of causal structures?*. To provide an answer, the author develops a mathematical intuition supported by the empirical evidence in Section 3.2.3.

As mentioned earlier, the VAE theoretically achieves perfect reconstruction of the input data by removing the functions $\mathbf{F}_1$ to $\mathbf{F}_4$ from the encoder and decoder architec-

tures. However, in reality, these functions are still present in the architectures, causing the decoder to generate an approximation of the actual data distribution $P(\tilde{\mathbf{X}}) \approx P(\mathbf{X})$. The quality of $P(\tilde{\mathbf{X}})$ depends on how the model parameters are learned. Therefore, the distance between $P(\mathbf{X})$ and $P(\tilde{\mathbf{X}})$ can be further reduced by incorporating additional loss terms. In the case of DAG-WGAN, the added loss term to the ELBO is the Wasserstein distance with a Gradient Penalty.

The Earth Mover distance differs significantly from typical MLE-based loss functions used in causal structure learning. The former aims to minimize the difference between probability distributions, while the latter focuses on maximizing the similarity between individual data points. Under the semi-parametric assumption, it becomes relatively straightforward to discover causality from observational data by applying a Structural Causal Model (SCM) to reconstruct individual data points. However, probability distributions do not provide any information about the relationships between variables in their samples, which makes adversarial causal discovery a challenging task. However, if a causal graph $\mathbf{G_A}$ and a probability distribution $P(.)$ are faithful to each other, they can be considered compatible. In such cases, $\mathbf{G_A}$ represents the causal relationships observed in samples of $P(.)$. Thus, in the case of DAG-WGAN, the distribution of observational data $P(\mathbf{X})$ and the distribution of learned data $P(\tilde{\mathbf{X}})$ can be expressed as follows:

$$
\begin{aligned}
&P_{\mathbf{G_A^0}}(\mathbf{X}) \equiv P(\mathbf{X}) \\
&P_{\mathbf{G_A}}(\tilde{\mathbf{X}}) \equiv \mathbb{E}_{P(\mathbf{X})}[|det(J_{\mathbf{X}\to Z})| Q_\phi(Z|\mathbf{F}_4((I-\mathbf{A}^T)\mathbf{F}_3(\mathbf{X})))],
\end{aligned} \tag{3.11}
$$

where $J$ is the Jacobian matrix [241], $det|J|$ denotes its determinant, $Z \sim Q_\phi(Z|\mathbf{F}_4((I-\mathbf{A}^T)\mathbf{F}_3(\mathbf{X})))$ and $\mathbf{X} \sim P(\mathbf{X})$.

This alternative definition of $P(\mathbf{X})$ and $P(\tilde{\mathbf{X}})$ suggests that minimizing the distance between $P_{\mathbf{G_A^0}}(\mathbf{X})$ and $P_{\mathbf{G_A}}(\tilde{\mathbf{X}})$ will bring the learned causal graph $\mathbf{G_A}$ closer to the ground truth $\mathbf{G_A^0}$, which explains the difference in accuracy between DAG-GNN and DAG-WGAN in the experiments (see Section 3.2.3). Importantly, the rate of improve-

ment varies depending on the data variable size. For datasets with a small number of columns, the reconstruction is already almost perfect, leaving little room for further improvement. Conversely, for high-dimensional data, the reconstruction becomes less accurate, and the contribution from the adversarial loss increases. This is because VAE have inherent difficulty in accurately reconstructing large datasets. The hybrid loss function does not affect the identifiability of $\mathbf{G_A}$ because the Wasserstein Distance with Gradient Penalty is applied to $P_{\mathbf{G_A^0}}(\mathbf{X})$ and $P_{\mathbf{G_A}}(\tilde{\mathbf{X}})$, where $P_{\mathbf{G_A}}(\tilde{\mathbf{X}})$ is the output of the decoder. This essentially means that the distance between the real data and the generated data can be described as $||P_{\mathbf{G_A^0}}(\mathbf{X}) - P_{\mathbf{G_A}}(\mathbf{X}|\mathbf{F}_2((I-\mathbf{A}^T)^{-1}\mathbf{F}_1(Z)))||$, where $\mathbf{G_A}$ is already identified. In other words, the adversarial loss only provides further refinement of $\mathbf{G_A}$, resulting in a closer approximation of $\mathbf{G_A^0}$. It should be noted that $P_{\mathbf{G_A}}(\mathbf{X}|\mathbf{F}_2((I-\mathbf{A}^T)^{-1}\mathbf{F}_1(Z)))$ is still parameterized by $\theta$, but this notation has been omitted for simplicity.

Moreover, as graphs that are faithful to distributions only describe the relationships between variables, they do not contribute to the training process of machine learning models. In the context of DAG-WGAN, this implies that the theoretical results and convergence guarantees of WGAN-GP are applicable.

**Proposition 3.2.2.** Given an (un)known ground truth graph $\mathbf{G_A^0}$ faithful to the observational data distribution $P_{\mathbf{G_A^0}}(\mathbf{X})$, the parameters of the implicitly learned probability distribution $P_{\mathbf{G_A}}(\tilde{\mathbf{X}})$ are refined by the following solution $D : \mathbb{R} \to \mathbb{R}$

$$\underbrace{\mathbb{E}_{\tilde{\mathbf{X}}\sim\mathbb{P}_g}[D(\tilde{\mathbf{X}})] - \mathbb{E}_{\mathbf{X}\sim\mathbb{P}_r}[D(\mathbf{X})]}_{\text{Critic loss}} + \underbrace{\lambda\mathbb{E}_{\hat{\mathbf{X}}\sim\mathbb{P}_{\hat{\mathbf{X}}}}[(||\nabla_{\hat{\mathbf{X}}}D(\hat{\mathbf{X}}) - 1||)^2]}_{\text{Gradient penalty}}$$

$$\underbrace{\mathbb{E}_{Z\sim Q_\phi(Z|\mathbf{F}_4((I-\mathbf{A}^T)\mathbf{F}_3(\mathbf{X})))}[D(Dec(Z))]}_{\text{Generator loss}},$$

where both terms are well-defined, differentiable almost everywhere and converge when $P_{\mathbf{G_A^0}}(\mathbf{X}) = P_{\mathbf{G_A}}(\tilde{\mathbf{X}})$.

*Proof.* The proof of proposition 3.2.2 is available in Appendix A.2. □

### 3.2.3 Experimental results

The performance of DAG-WGAN is evaluated through a series of experiments against some of the best models in the field. The approach is compared directly to DAG-GNN [35] to emphasize the impact of the Wasserstein distance in causal structure learning. Competing against DAG-GNN is justified because both models are based on the same VAE architecture, with the difference lying in the inclusion of adversarial training in DAG-WGAN. Additionally, the author compares their model to DAG-NOTEARS [12] and DAG-NoCurl [48] to provide more comprehensive evidence of the effect of Wasserstein-1 on causal discovery.

Experiments have been conducted using different types of data, such as continuous and categorical. The accuracy of the recovered causality is assessed by calculating the Structural Hamming Distance (SHD) [105] between the ground truth and the output graph. Furthermore, the quality of the generated data is evaluated by comparing the output of DAG-WGAN with data produced by CorGAN [242], and an additional study has been carried out to determine the effect of causal structures on the generation of synthetic data samples.

#### Continuous data

A series of experiments have been conducted to evaluate the performance of DAG-WGAN in the context of continuous data. These experiments utilized synthetic data generated from structural equations belonging to known identifiable causal models. The comparison between the approaches involved in this study, including DAG-GNN [35], DAG-NoCurl [48], DAG-NOTEARS [12], and DAG-WGAN [75], was based on their ability to recover causal structures from samples generated using the same underlying graphs and equations.

The data generation process consists of two main steps. The first step involves determining the ground truth graph, which is done by generating an Erdos-Renyi (ER) [243] DAG with an expected node degree of 3. This graph is represented mathematically by the weighted adjacency matrix $\mathbf{A}$. In the second step, observational samples are synthesized using the ground truth graph and a variety of Structural

Equation Models (SEM). For the linear case, the equation used is $\mathbf{X} = \mathbf{A}^T X + \mathcal{Z}$. For the nonlinear cases, two equations are used: $\mathbf{X} = \mathbf{A}^T cos(X + 1) + \mathcal{Z}$ (non-linear-1) and $\mathbf{X} = 2sin(\mathbf{A}^T(X + 0.5 * 1)) + \mathbf{A}^T(X + 0.5 * 1) + \mathcal{Z}$ (non-linear-2). The two non-linear equations were used to evaluate DAG-WGAN and all other models it was compared against. Furthermore, additional tests have been conducted to assess whether the model architecture can naturally handle the Post Nonlinear Model, which is considered a superset of the Additive Noise Model. The following SEM are used in the experiments: $\mathbf{X} = sinh(\mathbf{A}^T cos(X + 1) + \mathcal{Z})$ (post-nonlinear-1) and $\mathbf{X} = tanh(2sin(\mathbf{A}^T(X + 0.5 * 1)) + \mathbf{A}^T(X + 0.5 * 1) + \mathcal{Z})$ (post-nonlinear-2). The selection of these specific structural equations enables more robust model assessment and a more comprehensive investigation involving DAG-WGAN and DAG-GNN.

The number of samples used in all experiments is 5000 per graph. To assess the scaling capabilities of DAG-WGAN, tests are conducted with varying graph sizes (i.e., 10, 20, 50, and 100). To account for the randomness of the generated samples, each experiment is repeated 5 times per model. For each iteration of a test, the Structural Hamming Distance (SHD) between the learned graph from a model and the ground truth graph is measured. The mean SHD is then calculated for each approach and compared against the average produced from all other methods. Additionally, confidence intervals are used to complement the mean SHD and provide an indication of the consistency of DAG-WGAN. The results of the continuous experiments can be found in Tables 3.1, 3.2, 3.3, 3.4, and 3.5.

Table 3.1: Comparisons of DAG-learning Outcomes with Linear Data Samples

| **Model** | **SHD (5000 linear samples)** | | | |
|---|---|---|---|---|
| | **d = 10** | **d = 20** | **d = 50** | **d = 100** |
| DAG-NOTEARS | 8.4 ± 7.94 | 2.6 ± 1.84 | 25.2 ± 19.82 | 106.56 ± 56.51 |
| DAG-NoCurl | 7.9 ± 7.26 | 2.5 ± 1.93 | 24.6 ± 19.43 | 99.18 ± 55.27 |
| DAG-GNN | 6 ± 7.77 | 3.2 ± 1.6 | 21.4 ± 14.15 | 88.8 ± 47.63 |
| **DAG-WGAN** | **2.2 ± 4.4** | **2 ± 1.1** | **4.8 ± 4.26** | **28.20 ± 12.02** |

Table 3.2: Comparisons of DAG-learning Outcomes with Non-Linear Data Samples 1

| **Model** | **SHD (5000 non-linear-1 samples)** | | | |
|---|---|---|---|---|
| | **d = 10** | **d = 20** | **d = 50** | **d = 100** |
| DAG-NOTEARS | 11.2 ± 4.79 | 19.3 ± 3.14 | 53.7 ± 11.39 | 105.47 ± 13.51 |
| DAG-NoCurl | 10.4 ± 4.42 | 17.4 ± 3.27 | 51.6 ± 11.43 | 105.7 ± 13.65 |
| DAG-GNN | **9.4 ± 0.8** | **15 ± 3.58** | 49.8 ± 7.03 | 104.8 ± 12.84 |
| **DAG-WGAN** | 9.8 ± 2.4 | 16 ± 5.4 | **40.40 ± 10.97** | **80.40 ± 9.09** |

Table 3.3: Comparisons of DAG-learning Outcomes with Non-Linear Data Samples 2

| **Model** | **SHD (5000 non-linear-2 samples)** | | | |
|---|---|---|---|---|
| | **d = 10** | **d = 20** | **d = 50** | **d = 100** |
| DAG-NOTEARS | 9.8 ± 2.61 | 22.9 ± 2.14 | 38.3 ± 13.19 | 125.21 ± 61.19 |
| DAG-NoCurl | 7.4 ± 2.78 | 17.6 ± 2.25 | 33.6 ± 12.53 | 116.8 ± 62.3 |
| DAG-GNN | 2.6 ± 2.06 | 3.80 ± 1.94 | 13.8 ± 6.88 | 112.2 ± 59.05 |
| **DAG-WGAN** | **1 ± 1.1** | **3.4 ± 2.06** | **12.20 ± 7.81** | **20.20 ± 11.67** |

Table 3.4: Comparisons of DAG-learning Outcomes with Post-Non-Linear Data Samples 1

| **Model** | **SHD (5000 post-non-linear-1 samples)** | | | |
|---|---|---|---|---|
| | **d=10** | **d=20** | **d=50** | **d=100** |
| DAG-GNN | 12.7 ± 3.1 | 21.8 ± 5.7 | 65.3 ± 14.4 | 130.2 ± 27.4 |
| **DAG-WGAN** | **10.4 ± 3.2** | **18.2 ± 6** | **51.3 ± 11.8** | **107.8 ± 19.5** |

Table 3.5: Comparisons of DAG-learning Outcomes with Post-Non-Linear Data Samples 2

| **Model** | **SHD (5000 post-non-linear-2 samples)** | | | |
|---|---|---|---|---|
| | **d=10** | **d=20** | **d=50** | **d=100** |
| DAG-GNN | 8.4 ± 5.1 | 14.6 ± 5.2 | 47.8 ± 20.6 | 145.7 ± 77.7 |
| **DAG-WGAN** | **5.6 ± 5.8** | **10.2 ± 6.3** | **35.6 ± 14.4** | **43.3 ± 23.2** |

**Benchmark categorical data**

In order to evaluate the performance of DAG-WGAN on categorical data, the author obtains a set of discrete tabular datasets from the Bayesian Network Repository available at `https://www.bnlearn.com/bnrepository/`. This repository offers datasets of different types, such as Discrete Bayesian Networks, Gaussian Bayesian Networks, and Conditional Linear Gaussian Bayesian Networks, as well as datasets of various sizes,

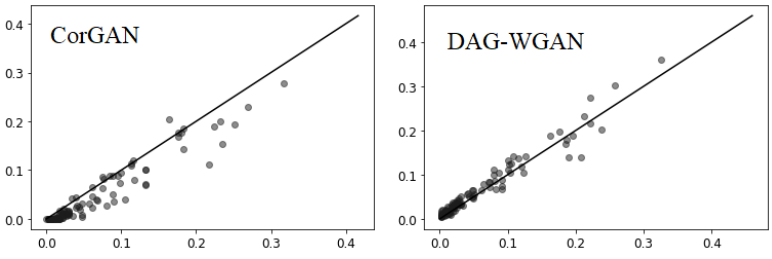
ranging from Small Networks to Massive Networks. To assess the scalability and accuracy of DAG-WGAN when handling categorical data, the author specifically selects the Sachs, Alarm, Child, Hailfinder, and Pathfinder datasets. Since DAG-GNN was the only model capable of working with discrete data at the time, the comparison is made solely between DAG-WGAN and DAG-GNN. The results of the experiment can be found in Table 3.6.

Table 3.6: Comparison of DAG-learning Outcomes with Benchmark Data Samples

| **Dataset** | **Nodes** | **SHD** | |
|---|---|---|---|
| | | **DAG-WGAN** | **DAG-GNN** |
| Sachs | 11 | 17 | 25 |
| Child | 20 | 20 | 30 |
| Alarm | 37 | 36 | 55 |
| Hailfinder | 56 | 73 | 71 |
| Pathfinder | 109 | 196 | 218 |

### Data integrity

Samples generated by DAG-WGAN have been compared with samples from other models to evaluate the data generation performance of each model on a 'dimension-wise probability' basis. This means that the author measures how well each model matches the distribution of observations for each dimension. The tabular dataset used in the experiment is MIMIC-III [244], which has been used in previous studies involving the models DAG-WGAN compares against. The dataset consists of medical measurements and observations, where each row is a patient record containing 1071 entries. The results of the data integrity experiment are shown in Figure 3.2.




(a) CorGAN (b) DAG-WGAN

Figure 3.2: Data integrity experiment outcome

Moreover, DAG-WGAN is compared only with CorGAN [242] as it performs better than other competitors such as medGAN [210] and DBM [245]. A scatter plot is used to represent the outcome of the study, where each point corresponds to one of the 1071 entries. The x and y axes indicate the success rate for real and synthetic data, respectively, while the diagonal line represents the ideal scenario.

The effectiveness of the reconstruction process of DAG-WGAN is also thoroughly examined. Two scenarios are considered to ensure completeness: 1) the output data contains the true causal graph (with SHD equal to 0); 2) the output data represents a causal graph of poor quality (with SHD as far from 0 as possible). The quality of the structure learning has been demonstrated using causal heat maps. The interdependencies between the covariates are studied by analyzing the correlation matrices of both cases. The diversity of the reconstructed data points is also plotted and examined. It should be noted that the input data is generated using the non-linear-2 Structural Equation Model (SEM) and the size of the data variables is set to 10. The author refrains from conducting further experiments since the model architecture and training algorithm remain the same. Therefore, since the only possible change that remains is assuming a different SEM to generate input data, further experiments will yield similar results. The performance of the model is also expected to deteriorate as the data variable size increases, due to the decreased precision in approximating the original data distribution.

In the case where the SHD equals 0, one would expect a perfect graph with no extra, missing or reversed edges to be recovered, exactly as shown in Figure 3.3.

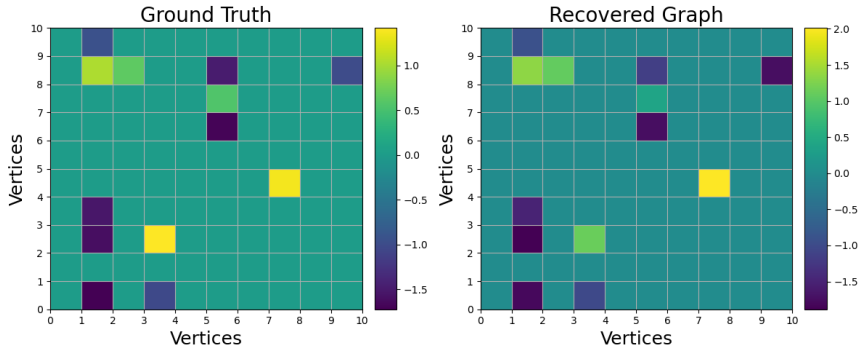



Figure 3.3: Weighted adjacency matrix heat map in the case SHD is 0

A complete recovery of the ground truth graph indicates that the causal connections in the reconstructed data are preserved. If the input data and the generated data share the same causal relationships, then it is reasonable to expect that they will also have similar statistical patterns (i.e., correlations), as illustrated in Figure 3.4.

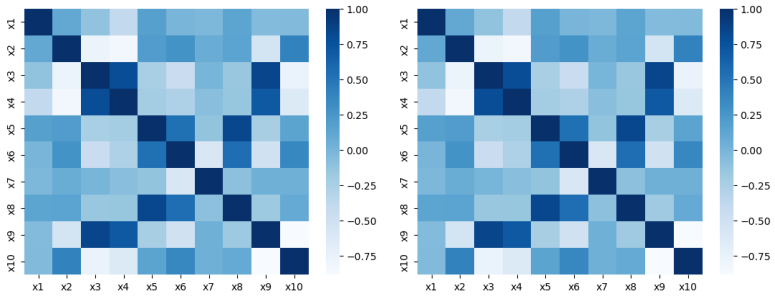



Figure 3.4: Comparison between the correlation matrices across the real (left) and synthetic (right) features, in the case SHD = 0

The investigation reveals that the statistical correlations between the real and fake data are nearly identical across the feature space. This similarity in correlations and preservation of causality leads to a very close approximation of the input data.

The variety of the new data points is low, indicating that the model has an almost perfect reconstruction process with imperceptible deviations from the real data samples - see Figure 3.5.

In the case where the SHD is farthest away from 0, the recovered graph is significantly different from the ground truth - see Figure 3.6.

Nevertheless, despite the inefficacy of causal structure discovery, DAG-WGAN accurately learns the correlations in the input data, while successfully reconstructing data points that closely resemble the original data, as shown in Figures 3.7 and 3.8.

The findings of the study offer valuable information regarding the capacity of DAG-WGAN to carry out both causal structure learning and data generation simultaneously. The most notable aspect is the straightforward design of its VAE component, which allows for precise data reconstruction regardless of the accuracy of the recovered causality. As a result, the ability of the model to learn causal structures relies on reconstructed samples but does not impact the reconstruction process itself.

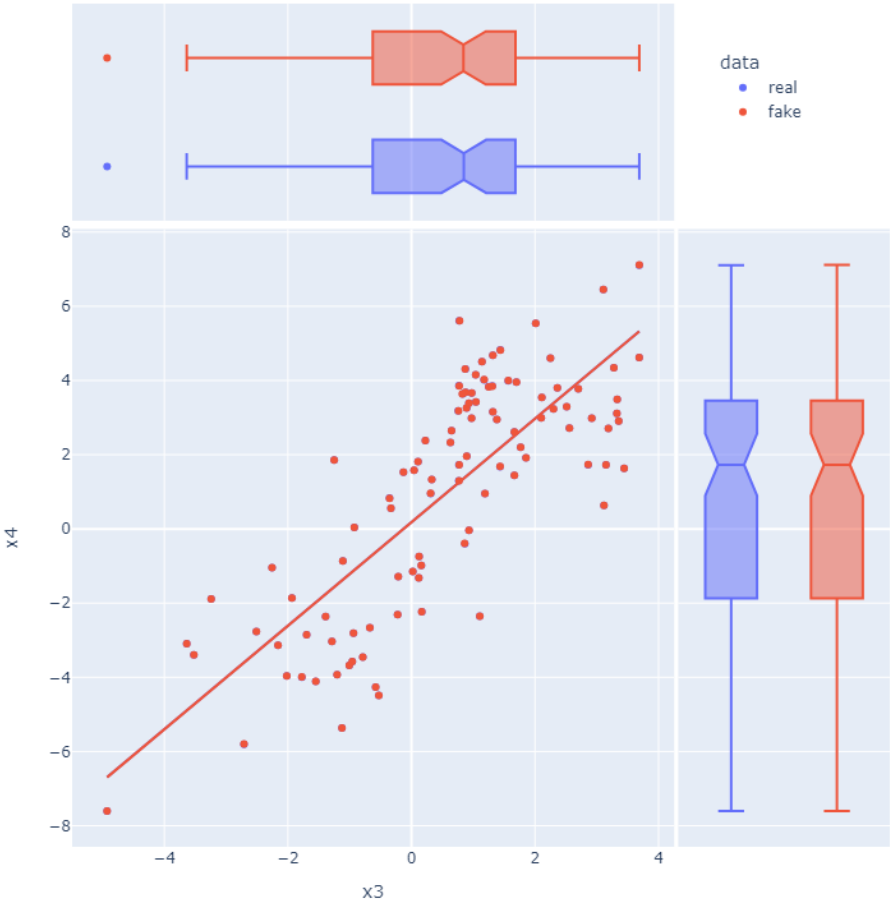



Figure 3.5: Real and synthetic feature distributions (x3,x4), in the case SHD = 0

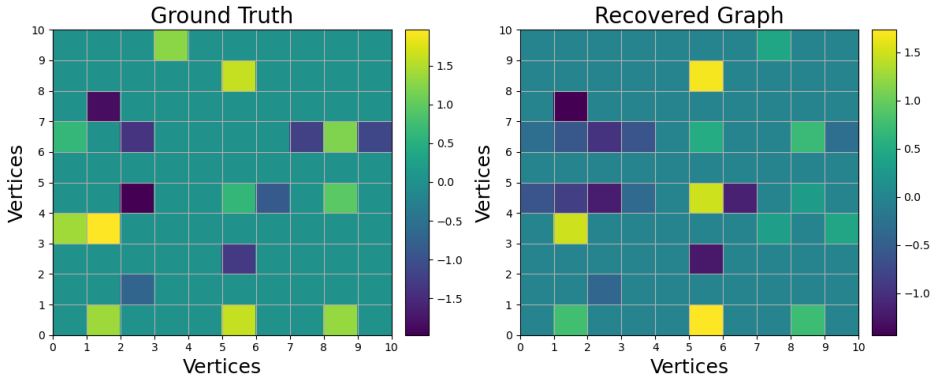



Figure 3.6: Weighted adjacency matrix heat map when SHD is farthest away from 0

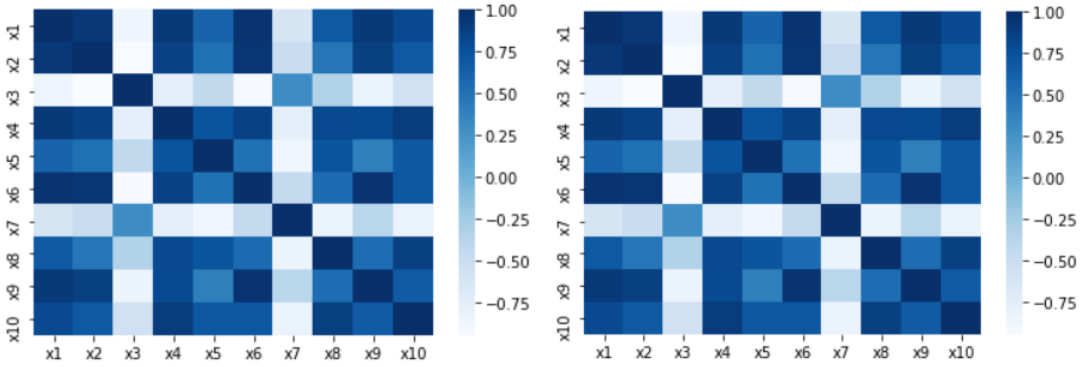



Figure 3.7: Comparison between the correlation matrices across the real (left) and synthetic (right) features when SHD is farthest away from 0

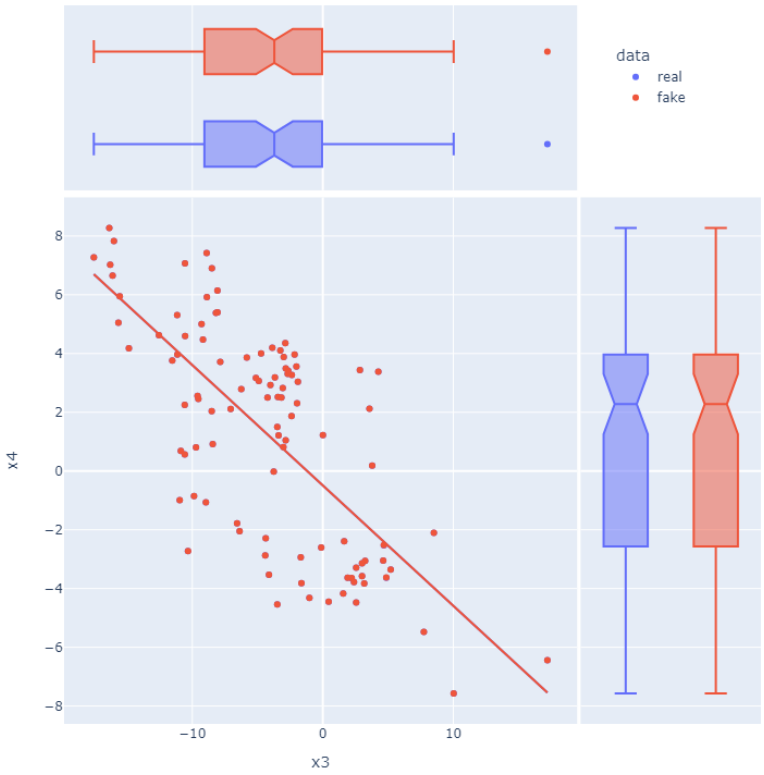



Figure 3.8: Real and synthetic feature distributions (x3,x4) when SHD is farthest away from 0

## 3.3 Discussion

DAG-WGAN has demonstrated competitive performance in all experiments, showcasing its ability to handle various types of data. The continuous results presented in Tables 3.1, 3.2, and 3.3 show that when the Additive Noise Model (ANM) is assumed, the proposed method outperforms DAG-NoCurl and DAG-NOTEARS in all three cases (linear, non-linear-1, and non-linear-2) and across all dimensions. In the non-linear-1 case, DAG-GNN performs better in lower dimensions but is surpassed by DAG-WGAN when the number of variables in the observations is higher. Moreover, the continuous data experiments demonstrate that the method scales better than its counterparts, providing a significant advantage. In the case of the Post-Nonlinear Model (PNL), DAG-WGAN outperforms DAG-GNN in all experiments, although the quality of the results is generally lower than those obtained with ANM. The results from Tables 3.4 and 3.5 reveal two important findings: 1) the architecture of DAG-GNN is not suitable for recovering causal structures when PNL is assumed, and 2) adversarial training is beneficial for discovering causality from data generated using PNL.

The outcomes for categorical data also strongly favor DAG-WGAN. The information presented in Table 3.6 demonstrates that DAG-GNN is less effective than the proposed algorithm in four out of five instances. Specifically, DAG-WGAN yields superior results when applied to the Sachs, Child, Alarm, and Pathfinder datasets, and only slightly worse results than DAG-GNN on Hailfinder. These benchmark experiments offer empirical support for the use of adversarial causal structure learning with discrete data.

Up to this point, the discussion has focused primarily on the ability of DAG-WGAN to learn causal structures. However, the results obtained from the data integrity experiment indicate that the generated data samples produced by the proposed model are also of high quality. Figure 3.2 illustrates the superior quality of the samples generated by DAG-WGAN in terms of their dimensions, compared to those generated by CorGAN, medGAN, and DBM. Furthermore, the completeness study demonstrates that the model is capable of synthesizing high-quality data regardless of the recovered

causal structure, due to the basic architecture of the VAE. This has two important implications: 3) the reconstructed data can be utilized for regression problems and predictions, even if the model has not learned all the connections between variables, and 4) the model has the potential to bridge the gap between Artificial Intelligence and humans - when the recovered graph is relatively accurate (i.e., SHD 0 or close to 0), the model can be trusted as people can comprehend the reasoning behind its predictions.

In general, the experiments on causal structure learning and data integrity showcase the capacity of the model to accurately discover causality from various types of data, all the while upholding a high level of data synthesis. The empirical comparison between DAG-WGAN and DAG-GNN provides evidence that the Wasserstein distance has a beneficial effect on both causal discovery and data generation. Furthermore, the results validate the hypothesis of the author regarding the role of Wasserstein-1 in recovering causality from observational data.

Despite its innovative integration of generative modeling and causal structure learning, and the good performance on the Sachs dataset, the DAG-WGAN framework faces several limitations when applied to real-world data. While the model is designed to infer directed acyclic graphs that capture underlying causal dependencies, its theoretical assumptions often fail to align with the complexities of empirical data. In practice, datasets collected from real-world systems are often noisy, nonlinear, and influenced by latent confounders, whereas DAG-WGAN assumes that all relevant variables are observed and that causal mechanisms can be effectively captured through a generator module. This mismatch can lead to inferred graphs that reflect statistical associations rather than genuine causal relationships, limiting their interpretability and real-world applicability.

When scaled to large real-world datasets, additional challenges emerge. The adversarial training process of DAG-WGAN is computationally intensive, and the acyclicity constraint adds further complexity, limiting scalability to high-dimensional data. This is mainly due to the augmented Lagrangian-based continuous optimization (CO) method [47]. Lachapelle et al. [36] have shown that the computational complexity of the CO approach is $\mathcal{O}(d^3)$, where $d$ represents the number of variables in $\mathbf{X}$. In the

experiments the author conducted, scalability testing has been performed in the range of 10, 20, 50 and 100 data variables per dataset with notable decrease in accuracy and increase in training duration detected as the data variable size increases. This trend is expected to continue indefinitely as the number of variables exceeds beyond the 100 nodes barrier, making the model unusable with big-data. Moreover, training stability remains a major issue, as performance is highly sensitive to hyperparameter choices and optimization dynamics, often resulting in inconsistent outcomes across runs. The purely data-driven approach of the model, without the integration of domain knowledge or structural priors, also restricts its ability to produce interpretable and plausible causal structures at scale.

Last but not least, despite achieving high accuracy in causal structure learning and producing high-quality synthesized data, the performance of DAG-WGAN is limited by its own architectural constraints. One key limitation is that the proposed approach heavily relies on including specific Structural Equation Models (SEM) in both the encoder and decoder modules. This assumption is unreasonable, as it implies that all real-world data must be generated using the same equations as those in the autoencoder architecture, which is highly unlikely. Consequently, DAG-WGAN can only work with data generated using the SEM specified in (3.3). Additionally, the completeness study results indicate that the reconstruction process of DAG-WGAN is overly precise, resulting in a lack of diversity in the generated data samples.

In order to address the limitations of DAG-WGAN, the author will investigate efficient structure learning techniques, remove specific SEM from the architecture, and incorporate Disentangled Representation Learning (DRL) to improve the time complexity, generality of causal structures, and diversity of generated samples. Furthermore, additional experiments will be conducted to determine whether the proposed method can help address the hidden confounder problem [246], [247], [218]. The model will also be extended to handle incomplete and time-series data. Additionally, the use of early stopping techniques will be explored in future iterations of the model to address cases where the optimal DAG is discovered too quickly but the augmented Lagrangian fails to converge. A sensitivity analysis will also be conducted to investigate the performance

of the model when introduced to slight visitations in hyper-parameters.

# Chapter 4

# Efficient Generative Adversarial DAG-Structure Learning

This chapter begins with the author providing the implementation details of the successor to the original DAG-WGAN method. The model architecture remains unchanged, but there are significant improvements in the training algorithm. This follow-up approach called DAG-WGAN+ incorporates efficient frameworks for DAG discovery and disentangled representation learning, resulting in a faster and more accurate method compared to its predecessor. The chapter also explores topics such as data quality, causal identifiability, and computational complexity in the context of DAG-WGAN+. Moreover, the model is capable of handling vector data as well. A series of experiments are conducted to demonstrate the performance of DAG-WGAN+, showing that it can compete with the state-of-the-art in the field (see Section 4.3 for more details). Additionally, an ablation study is conducted to investigate the impact of changes in the training algorithm on the model. The publication resulting from this work is referenced in Section 1.7.

## 4.1 An efficient DAG-WGAN formulation using DAG-NoCurl

Following the significant progress achieved through the use of DAG-NOTEARS [12], the field of causal structure learning has witnessed a surge in research, resulting in the development of several extensions to the framework. Various models, such as DAG-GNN [35], GraN-DAG [36], and DAG-WGAN [75], heavily rely on DAG-NOTEARS and have demonstrated impressive performance. However, these models encounter limitations during training due to the DAG-NOTEARS approach, which affects both their accuracy and the computational time required to obtain results.

Zheng et al. [12] proposed a framework for continuous optimization that incorporates Maximum Likelihood Estimation (MLE) loss terms for model training and augmented Lagrangian to enforce acyclicity. However, as previously mentioned (Section 2.3.1), in architectures such as VAE and WGAN-GP, MLE-based loss terms suffer from a severe lack of diversity in reconstructed data (latent collapse) and function simplicity resulting in a highly accurate reconstruction process leading to synthetic samples overfitting to input data.

The DAG-learning approach of the author called DAG-WGAN+ combines the Evidence Lower BOund (ELBO), Maximum Mean Discrepancy (MMD) and Wasserstein Distance (WD) loss terms under the famous VAE-GAN architecture [62] to learn data probability distributions and recover causal relationships from the training samples. This combination helps to overcome the impact of the MLE limitations inherited by the ELBO loss. The experimental results obtained using DAG-WGAN+ indicate that by jointly optimizing the ELBO and MMD, it is possible to encourage mutual information between observations and latent variables. As a result, the latent space contains meaningful features of the input data, leading to enhanced representation quality, data reconstruction, and causal discovery.

Meanwhile, constraint optimization of DAG learning models using the augmented Lagrangian has been found to be a costly process, making it impractical to apply causal discovery methods to real-world data analysis. To address this issue, researchers have

proposed novel approaches for causality learning that do not involve computationally expensive procedures. One such approach is DAG-NoCurl [48], which implicitly discovers causal relationships in the DAG search space. However, to recover the correct causal structures from observations, this model requires an accurate initial estimate. Failing to meet this requirement may result in inaccurate DAG-learning and impose limitations on the search space, reducing the potential for learning better DAG structures.

DAG-WGAN+ is developed by incorporating a generative adversarial DAG learning approach to an improved version of the DAG-NoCurl efficient structure learning method. Moreover, the model uses Disentangled Representation Learning (DRL) with the help of Maximum Mean Discrepancy (MMD) [67] and allows additional refinement of the initial graph topology to achieve high accuracy and efficiency without any limitations on the DAG search space.

The main objective of the research is to evaluate the performance of DAG-WGAN+ against its predecessor, the original DAG-WGAN model [75], to deduce which of the two methods is superior. Furthermore, the new approach is tested against the current state-of-the-art in a set of experiments, as discussed in Section 4.2. Ultimately, DAG-WGAN+ enables the author to make the following contributions:

- The combination of hybrid generative modeling for causal structure recovery with disentangled representation learning mitigates the limitations of MLE-based loss terms, resulting in higher-quality DAG-discovery.
- Refactoring the original DAG-NoCurl approach enables further refinement of the causal structure obtained from the initial estimation in search of DAG that better fit the input data. Applying this improved version of DAG-NoCurl to generative adversarial DAG-learning results in more efficient and accurate causal discovery.

### 4.1.1 Problem Statement

This model is designed to enhance the efficiency of the constrained continuous optimization process used in structure learning, enabling faster generation of synthetic samples that maintain the causal relationships of the input data through the integration of the

DAG-NoCurl framework. The fundamental concept of this approach can be described as follows: Given a set of $n$ independent and identically distributed (i.i.d.) observations $\mathbf{X}$, DAG-WGAN+ is designed to recover a causal graph $\mathbf{G_A}$, in an efficient manner, by learning the components of an equivalent representation $\mathbf{A}_{init} \in \{\mathcal{G}_{\mathbf{A}_{init}}\} \equiv \mathbb{D}$ of the adjacency matrix $\mathbf{A} \in \mathbb{D}$ that can implicitly produce a new probability distribution $P(\tilde{\mathbf{X}})$ to closely match the original distribution $P(\mathbf{X})$. In this chapter, the notation $P(\tilde{\mathbf{X}}) \equiv P_{\mathbf{G_A}}(\tilde{\mathbf{X}})$ and $P(\mathbf{X}) \equiv P_{\mathbf{G_A^0}}(\mathbf{X})$ are considered equivalent - refer to the segment on the identifiability of the loss function in Section 3.2.2 for further details.

The method is built upon satisfying the set of assumptions outlined in Section 2.1.4. The faithfulness assumption is particularly crucial for the functionality of DAG-WGAN+ [10], as it enables the learning of Directed Acyclic Graphs (DAG) from data distributions. Furthermore, the author employs datasets produced with identifiable structural equation models falling under different Semi-parametric assumption, such as Additive Noise Models (ANM) and Post-Nonlinear Models (PNL).

### 4.1.2 Solution Overview

DAG-WGAN+ tackles the DAG-learning challenge by utilizing a hybrid generative modeling framework that integrates InfoVAE [248] and WGAN-GP [116]. This results in the optimization of model parameters through a combination of reconstruction, regularization loss terms and generative adversarial training. The process of learning causal structures using this method takes place within the auto-encoder architecture, which incorporates an additional learnable parameter $\mathbf{A} \in \mathbb{R}^{d \times d}$. Moreover, the standard reconstruction loss (ELBO) is enhanced by introducing a mutual information term from the training of InfoVAE [249]. The objective function based on the maximum likelihood estimation (MLE) can be expressed as follows:

$$\begin{aligned}\mathbf{R}_{loss}(\mathbf{X}, \tilde{\mathbf{X}}, Z) = &-\mathbb{E}_{Q_\phi(Z|\mathbf{F}_4((I-\mathbf{A}^T)\mathbf{F}_3(\mathbf{X})))}[\log P_\theta(\mathbf{X}|\mathbf{F}_2((I-\mathbf{A}^T)^{-1}\mathbf{F}_1(Z)))] \\ &+ \beta \mathbb{E}_{\mathbf{X}\sim P(\mathbf{X})}[D_{KL}(Q_\phi(Z|\mathbf{F}_4((I-\mathbf{A}^T)\mathbf{F}_3(\mathbf{X})))||P(Z))] \\ &+ \eta \mathbb{E}_{\mathbf{X}\sim P(\mathbf{X})}[\mathbf{I}_{Q_\phi(\mathbf{X},Z)}(\mathbf{X}, Z)],\end{aligned} \tag{4.1}$$

where the latent variable $Z$ is sampled from an implicitly defined variational distribution $Q_\phi(Z|\mathbf{F}_4((I-\mathbf{A}^T)\mathbf{F}_3(\mathbf{X})))$ modeled by the encoder $Enc(\mathbf{X};\mathbf{A};\phi) = Q_\phi(Z|\mathbf{F}_4((I-\mathbf{A}^T)\mathbf{F}_3(\mathbf{X};\phi);\mathbf{A};\phi))$ with parameters $\phi$ and $\mathbf{A}$. The reconstructed data $\tilde{\mathbf{X}}$ is obtained by sampling from $P_\theta(\mathbf{X}|\mathbf{F}_2((I-\mathbf{A}^T)^{-1}\mathbf{F}_1(Z)))$, which represents the probability distribution learned by the decoder $Dec(Z;\theta) = P_\theta(\mathbf{X}|\mathbf{F}_2((I-\mathbf{A}^T)^{-1}\mathbf{F}_1(Z;\theta);\theta))$ with the parameters $\theta$. The multi-layer perceptrons $\{\mathbf{F}_3, \mathbf{F}_4\}$ and $\{\mathbf{F}_1, \mathbf{F}_2\}$ are utilized in $Enc$ and $Dec$ respectively. The term $\mathbf{I}_{Q_\phi(\mathbf{X},Z)}(\mathbf{X}, Z)$ stands for the mutual information loss. Hyperparameters $\beta$ and $\eta$ are employed to adjust the impact of the regularization terms (specifically the KL-Divergence and the mutual information term) on the reconstruction loss. The objective function in (4.1) is also subject to an unconstrained continuous optimization with the acyclicity constraint from [35] $h(\mathbf{A}) = tr[(I+\beta\mathbf{A}\circ\mathbf{A})^d] - d = 0$.

The author incorporates the concept of efficient structure learning by combining generative adversarial DAG-recovery with the DAG-NoCurl framework [48]. Specifically, the model relies on discovering the topology of the variables in $\mathbf{X}$ by computing a potential function $\psi$ and then projecting (non)cyclical structures onto its gradient $\nabla\psi$, thus ensuring that the resulting output is a DAG; as detailed in Section 2.4.2. Within the DAG-WGAN+ framework, both aspects of the DAG structure (the topology and the strength of connections between variables) are jointly optimized, facilitating causal discovery across a broader DAG search space. This approach leads to faster and more precise causal structure learning in comparison to existing methods - for further details, the reader is directed to the experimental findings in Section 4.2.

### 4.1.3 Training algorithm improvements

In this section, the author discusses the changes made to the training algorithm of the initial DAG-WGAN model. Disentangled representation learning details are disclosed, along with the incorporation of an enhanced version of the DAG-NoCurl framework.

#### Disentangled representation learning

The utilization of Disentangled Representation Learning (DRL) for causal discovery stems from a recent investigation [249] carried out using the default VAE configura-

tion. The study findings highlight the limitations of standard Variational Auto-Encoder components. Specifically, the design of the VAE often struggles to capture fine-grained details and variations in the input data, resulting in a noisy and uninterpretable embedding space. As a result, latent variables $Z$ sampled from such a space may fail to capture meaningful representations, leading to inaccuracies in data reconstruction. Furthermore, as noted in Section 2.3.1, the drawbacks of the ELBO loss function can result in erroneous modeling of the approximate posterior $Q(Z|\mathbf{X})$. This issue becomes more apparent when the encoder is exposed to complex or high-dimensional data, which ultimately causes the model to overfit.

DAG-WGAN+ relies on a sophisticated training algorithm that combines the ELBO regularized by DRL with generative adversarial training to extract causality from observations. Specifically, the approach aims to approximate the distribution $P_{\mathbf{G_A}}(\tilde{\mathbf{X}})$ to $P_{\mathbf{G}_\mathbf{A}^0}(\mathbf{X})$ as shown in (3.11). This strategy prevents the model from overfitting by modeling the training distribution with $\tilde{\mathbf{X}}$ instead of $\mathbf{X}$, thus discouraging DAG-WGAN+ from closely matching the input data and focusing on discovering underlying patterns or relationships.

Minimizing (4.1) requires defining the reconstructed data distribution $P_{\mathbf{G_A}}(\tilde{\mathbf{X}})$. However, as indicated in (3.11), computing $P_{\mathbf{G_A}}(\tilde{\mathbf{X}})$ involves the weighted adjacency matrix $\mathbf{A}$, which is one of the parameters in the model. This implies that modeling $P_{\mathbf{G_A}}(\tilde{\mathbf{X}})$ depends on the optimization of $\mathbf{A}$. As a result, the probability distribution $P_{\mathbf{G_A}}(\tilde{\mathbf{X}})$ and the weighted adjacency matrix $\mathbf{A}$ are jointly learned through both reconstruction and adversarial training performed using the following objective function:

$$
\begin{aligned}
\mathbf{R}_{loss}(\mathbf{X}, \tilde{\mathbf{X}}, Z) &= -\mathbb{E}_{Z\sim Q_\phi(Z|\mathbf{F}_4((I-\mathbf{A}^T)\mathbf{F}_3(\mathbf{X})))}[\log P_\theta(\mathbf{X}|\mathbf{F}_2((I-\mathbf{A}^T)^{-1}\mathbf{F}_1(Z)))] \\
&\quad + \beta\mathbb{E}_{\mathbf{X}\sim P(\mathbf{X})}[D_{KL}(Q_\phi(Z|\mathbf{F}_4((I-\mathbf{A}^T)\mathbf{F}_3(\mathbf{X})))||P(Z))] \\
&\quad + \eta\mathbb{E}_{\mathbf{X}\sim P(\mathbf{X})}[\mathbf{I}_{Q_\phi(\mathbf{X},Z)}(\mathbf{X}, Z)] \\
\mathbf{D}_{loss} &= \mathbb{E}_{\tilde{\mathbf{X}}\sim\mathbb{P}_g}[D(\tilde{\mathbf{X}})] - \mathbb{E}_{\mathbf{X}\sim\mathbb{P}_r}[D(\mathbf{X})] + \lambda\mathbb{E}_{\hat{\mathbf{X}}\sim\mathbb{P}_{\hat{\mathbf{X}}}}[(||\nabla_{\hat{\mathbf{X}}}D(\hat{\mathbf{X}}) - 1||)^2] \\
\mathbf{G}_{loss} &= -\mathbb{E}_{Z\sim Q_\phi(Z|\mathbf{F}_4((I-\mathbf{A}^T)\mathbf{F}_3(\mathbf{X})))}[D(Dec(Z))] \\
&\quad \text{s.t} \quad tr[(I+\alpha\mathbf{A}\circ\mathbf{A})^d] - d = 0,
\end{aligned}
\tag{4.2}
$$

where $\mathbf{R}_{loss}$, $\mathbf{D}_{loss}$ and $\mathbf{G}_{loss}$ are the reconstruction, discriminator and generator losses, respectively. $\mathbb{P}_r$ and $\mathbb{P}_g$ represent $P(\mathbf{X})$ and $P(\tilde{\mathbf{X}})$ and thus are equivalent to $P_{\mathbf{G}_\mathbf{A}^0}(\mathbf{X})$ and $P_{\mathbf{G}_\mathbf{A}}(\tilde{\mathbf{X}})$ as well. Importantly, the only difference in the objective function between DAG-WGAN+ (4.2) and DAG-WGAN (3.10) is the inclusion of the mutual information term. The expanded form of (4.2) reveals the final loss function used to train DAG-WGAN+:

$$
\begin{aligned}
\mathbf{A}^*, \theta^*, \phi^*, \omega^* = \operatorname{argmin}_{\mathbf{A},\theta,\phi} \max_\omega \mathcal{L}_{DAG-WGAN+}(\mathbf{A}, \theta, \phi, \omega) \\
\mathcal{L}_{DAG-WGAN+} = -\mathbb{E}_{Z\sim Q_\phi(Z|\mathbf{F}_4((I-\mathbf{A}^T)\mathbf{F}_3(\mathbf{X})))}[\log P(\mathbf{X}|Z;\theta)] \\
+(1-\beta)\mathbb{E}_{\mathbf{X}\sim P(\mathbf{X})}[D_{KL}(Q(Z|\mathbf{X};\mathbf{A},\phi)||P(Z))] \\
+(\gamma+\beta-1)D_{KL}(Q(Z;\mathbf{A},\phi)||P(Z)) \\
+\mathbb{E}_{\mathbf{X}\sim P(\mathbf{X}),Z\sim Q_\phi(Z|\mathbf{F}_4((I-\mathbf{A}^T)\mathbf{F}_3(\mathbf{X})))}[D(\mathbf{X};\omega) - D(Dec(Z;\theta);\omega)] \\
+\lambda\mathbb{E}_{\hat{\mathbf{X}}\sim P(\hat{\mathbf{X}})}\left[(||\nabla_{\hat{\mathbf{X}}} D(\hat{\mathbf{X}})||_2 - 1)^2\right] \\
+\mathcal{K}(tr[(I+\alpha\mathbf{A}\circ\mathbf{A})^d] - d) \\
\text{such that} \\
\begin{cases}
\mathbb{E}_{Z\sim Q_\phi(Z|\mathbf{F}_4((I-\mathbf{A}^T)\mathbf{F}_3(\mathbf{X})))}[\log P(\mathbf{X}|Z;\theta)] \approx -\frac{1}{2}\left[\frac{(\tilde{\mathbf{X}}-\mu(\mathbf{X}))^2}{\sigma(\mathbf{X})^2} + \log\sigma(\mathbf{X})^2\right] \text{ if } \mathbf{X} \\
\text{is continuous} \\
\mathbb{E}_{Z\sim Q_\phi(Z|\mathbf{F}_4((I-\mathbf{A}^T)\mathbf{F}_3(\mathbf{X})))}[\log P(\mathbf{X}|Z;\theta)] \approx -\sum_{c=1}^N(\mathbf{X}_c\log(\tilde{\mathbf{X}}_c)) \text{ if } \mathbf{X} \text{ is discrete}
\end{cases} \\
\mathbb{E}_{\mathbf{X}\sim P(\mathbf{X})}[D_{KL}(Q(Z|\mathbf{X};\mathbf{A},\phi)||P(Z))] \approx -\frac{1}{2}\left[\log\sigma(Z)^2 - \left(\mu(Z)^2 - \sigma(Z)^2\right) + 1\right],
\end{aligned}
\tag{4.3}
$$

where the following set of parameters $\{\mathbf{A}, \phi\}$, $\{\theta\}$ and $\{\omega\}$ are used to optimize the encoder $Enc(\mathbf{X}, \mathbf{A}, \phi)$, the decoder $Dec(Z, \theta)$ and the discriminator $D(\mathbf{X}, \tilde{\mathbf{X}}, \omega)$. The $1^{st}$ term represents the reconstruction loss. The $4^{th}$ and $5^{th}$ terms are responsible for computing the Wasserstein-1 metric and its gradient penalty, whereas the $2^{nd}$ and $3^{rd}$ terms introduce regularization to the loss function. In both of the latter terms, $P(Z)$ denotes a Gaussian prior. The distance between $P(Z)$ and both $Q(Z|\mathbf{X};\mathbf{A},\phi)$ and $Q(Z)$ is computed and minimized using KL-Divergence (KLD) and Maximum Mean Discrepancy (MMD) [250]; [112]. The final term [35] is utilized to ensure the acyclicity

of the recovered graph. Furthermore, by incorporating $D(\mathbf{X}, \tilde{\mathbf{X}}, \omega)$ and employing the min-max optimization described in (4.3), the refinement of the reconstruction loss is facilitated through adversarial training. Consequently, the $4^{th}$ and $5^{th}$ terms are involved in modeling the parameters of the discriminator, while the encoder and decoder are trained using the $1^{st}$, $2^{nd}$, $3^{rd}$ and $6^{th}$ terms.

**Proposition 4.1.1.** Given some input $\mathbf{X}$ and latent variables $Z$, for any fixed value of the mutual information term $\mathbf{I}_{Q_\phi(\mathbf{X},Z)}(\mathbf{X}, Z)$, $\mathcal{L}_{DAG-WGAN+}$ reaches global optimum when the decoder distribution $P_\theta(\mathbf{X}|\mathbf{F}_2((I - \mathbf{A}^T)^{-1}\mathbf{F}_1(Z)))$ matches the observational data distribution $P(\mathbf{X})$.

*Proof.* The proof of proposition 4.1.1 is available in Appendix A.3. □

Causal structure identifiability is another crucial aspect of DAG-WGAN+ - see Definition 1. To guarantee that the DAG recovered are identifiable, the author relies on the following assumptions: 1) employing an identifiable structural equation model to generate the observational data samples and 2) applying an Additive Noise Model (ANM) [240] as the structural causal model in *Dec* under the semi-parametric assumption.

**Proposition 4.1.2.** Given a generated data distribution $P_{\mathbf{G_A}}(\tilde{\mathbf{X}})$, defined using a causal graph $\mathbf{G_A}$ belonging to the set of identifiable causal graphs $S_{\mathbf{G_A}}$, and the true underlying causal structure of the input data denoted as $\mathbf{G_A^0}$. Assuming that $\mathbf{G_A^0}$ is also a member of $S_{\mathbf{G_A}}$, then a learned causal graph $\mathbf{G_A}$ contains the same structure as $\mathbf{G_A^0}$ i.f.f. $P_{\mathbf{G_A}}(\tilde{\mathbf{X}})$ matches the original data distribution $P_{\mathbf{G_A^0}}(\mathbf{X})$.

*Proof.* The proof of proposition 4.1.2 is available in Appendix A.4. □

As stated in the beginning of this chapter, there is no architectural distinction between DAG-WGAN and its enhanced iteration. Therefore, the structure of DAG-WGAN+ has already been proven to be identifiable; for further information, refer to Section 3.2.2. Modifications are solely present in the loss function of the VAE module, where an extra mutual information component, represented by MMD, is incorporated

into the ELBO. Given that ELBO is identifiable, the focus of the identifiability assessment of DAG-WGAN+ is the MMD loss term.

The role of MMD in DAG-WGAN+ is consistent with Disentangled Representation Learning in other VAE variants, influencing the training procedure by maximizing the mutual information between observations $\mathbf{X}$ and their latent representation $Z$. This leads to the ability to discover latent data features that may be concealed by the complexity of the input data, facilitating the identification of hidden confounders within $\mathbf{X}$. In the DAG-WGAN+ framework, causal insufficiency is not assumed; rather, MMD enhances the quality of $Z$, subsequently improving the performance of the decoder, thus explaining the superior outcomes of DAG-WGAN+ over its predecessor. As the mutual information term is applied solely to the encoder output, MMD does not directly impact the causal graph faithful to the probability distribution of the reconstructed data. As a result, causal structure learning with DAG-WGAN+ yields identifiable outcomes due to the VAE architecture and other loss terms employed during training.

It is important to note that the identifiability of causal structures is also dependent on the assumptions made during their learning process. Among the assumptions used through this work (see Section 2.1.4), faithfulness plays a crucial role, as it ensures that the observed statistical independencies accurately describe the underlying causal relationships in data. When this assumption is violated, the mapping between dependencies and causal structure breaks down, directly undermining causal identifiability. In such cases, different causal graphs can generate the same observed independencies, making it impossible to uniquely recover the true causal structure from observational datasets alone. This violation can affect DAG-WGAN+, which relies on faithfulness to discover accurate causal graphs: genuine causal relationships may be masked, spurious dependencies may appear, and the learned structure may no longer represent the true causal mechanisms. Under such circumstances, even if the model fits the observed data distribution well, its causal interpretations and counterfactual reasoning become unreliable. To mitigate these issues, robustness measures such as incorporating sparsity or regularization constraints, applying stability-based or ensemble methods, introducing domain knowledge through soft constraints, and leveraging interventional data or causal

invariance principles can help recover more stable and interpretable causal structures despite potential violations of faithfulness.

**Efficient causal structure learning with DAG-NoCurl**

Currently, a significant proportion of machine learning models used to recover causality from data are trained through constrained continuous optimization methods employing the Augmented Lagrangian technique [47]. Although this approach has been shown to produce accurate results for causal structure learning, its performance comes at the expense of computational efficiency and substantial time consumption [36]. A recent advancement that addresses the constraints of Augmented Lagrangian optimization is the DAG-NoCurl framework [48], which directly models causal structures within the DAG search space. The framework is heavily dependent on an equivalent weighted adjacency matrix representation $\mathbf{A} = W \circ ReLU(\nabla\psi)$ - as outlined in Theorem 2.1 of [48], where $W \in \mathbb{R}^{d\times d}$ represents an upper triangular matrix, $\circ$ denotes the Hadamard product [239], ReLU [251] signifies the rectified linear unit activation function, $\psi$ is a potential function, and $\nabla$ symbolizes its gradient. Efficiency is achieved by computing an initial estimate $\mathbf{A}_{pre}$ to derive the potential function $\psi$, which subsequently maps $\mathbf{A}_{pre}$ to a causal graph $\mathbf{G}_{\mathbf{A}_{\mathbf{init}}} \in \{\mathcal{G}_{\mathbf{A}_{init}}\} \equiv \mathbb{D}$. Following this step, further enhancement of $W$ is carried out with a fixed value for $\psi$. However, a drawback of this model is that the optimization of $W$ solely impacts the edge weights and not the topology of the graph represented by $\mathbf{A}_{init}$, leading to a restricted search space for DAG. Consequently, while the DAG-NoCurl approach guarantees acyclicity, it does not ensure the accuracy of the output graph.

The proposed model integrates the DAG-NoCurl framework with a generative adversarial DAG-Structure learning approach. Initially, the estimation $\mathbf{A}_{pre}$ is obtained through an unconstrained augmented Lagrangian optimization of the loss function in (4.3) while adhering to the acyclicity constraint $h(\mathbf{A}) = tr[(I + \alpha\mathbf{A} \circ \mathbf{A})^m] - m = 0$, with a fixed Lagrange multiplier value of 10. Subsequently, $\mathbf{A}_{pre}$ is projected into the equivalent graph space $\{\mathcal{G}_{\mathbf{A}_{init}}\} \equiv \mathbb{D}$ by determining a value for the potential function $\psi$ to solve $\mathbf{A}_{init} = W \circ ReLU(\nabla\psi)$. In the following refinement phase, the author opts

not to constrain the value of $\psi$, but rather improves the projected causal graph $\mathbf{G}_{\mathbf{A_{init}}}$ by solving (4.3) under the acyclicity constraint $h(\mathbf{A}) = 0$. This refinement step eliminates the constraints imposed on $\mathbf{A}_{init}$ by the fixed value of $\psi$, enabling the recovery of DAG structures from a wider graph search space. Notably, the Lagrange multiplier is not updated during the aforementioned steps, which significantly reduces computational complexity. Additionally, the original DAG-NoCurl algorithm yields $\tilde{W}$ after the refinement phase. As $\tilde{W}$ is not a DAG, an additional step $\tilde{\mathbf{A}} = \tilde{W} \circ ReLU(\nabla\psi)$ is included to derive the final DAG. In contrast, DAG-WGAN+ directly computes a weighted adjacency matrix $\mathbf{A}$ by optimizing both $W$ and $\psi$, eliminating the need for post-processing steps to obtain the final DAG structure. The detailed sequence of steps is provided in Algorithm 1.

**Remark.** The same additional post-processing computation step from the DAG-NoCurl framework can be included in the model, however, doing so significantly reduces accuracy. More information on this matter is found in the Ablation Study - 4.2.6.

**Algorithm 1** Efficient adversarial structure learning with DAG-NoCurl

Step 1: Compute an initial prediction $\mathbf{A}_{pre}$ by optimizing (4.3) with a fixed Lagrangian multiplier for the acyclic constraint.

Step 2: Use $\mathbf{A}_{pre}$ to compute the potential function $\psi = -L^{+}\nabla^{T}\big(0.5 * (C(\mathbf{A}_{pre}) - C(\mathbf{A}_{pre})^{T})\big)$, where $L^{+}$ is the Moore-Penrose pseudo-inverse of the graph Laplacian matrix $L$, $\nabla^{T}$ is the transpose of the gradient matrix $\nabla$, $C(\mathbf{A}_{pre})$ is the connection matrix of $\mathbf{A}_{pre}$ – see [48] for more details.

Step 3: Compute the $W$ matrix by converting each non-zero entry (indexed by in row $i$ and column $j$, $\psi(j) > \psi(i)$) of $\mathbf{A}_{pre}$ to the entry of $W$ by scaling it with a factor $\mathbf{A}_{pre}(i,j)/(\psi(j) - \psi(i))$ - see Equation (10) in the DAG-NoCurl [48] paper.

Step 4: Compute the initial weighted adjacency matrix $\mathbf{A}_{init} = W \circ ReLU(\nabla\psi)$

Step 5: Update $\mathbf{A}$ by optimising (4.3) with the initialisation of $\mathbf{A} = \mathbf{A}_{init}$

Furthermore, the author employs thresholding at the end of Steps 1 and 5 to reduce

the number of false discoveries. Consistent with the approach introduced by [12], a threshold value of 0.3 is chosen. Although different threshold values could be utilized, practical experiments conducted by the author indicated that the value aligned with [12] and several other models such as [36], [38], [75] yields the most consistent results.

### 4.1.4 Computational Complexity

The development of complex functions or algorithms is often accompanied by discussions of their resource utilization. In fact, a common question regarding any software is "*How long does it take to execute this program?*". However, providing a precise answer to this question is challenging due to variables like hardware quality, concurrent program count, programming language, etc. Instead, in the area of computer science, a simpler question is posed: "*How does the execution time of a piece of code change as the input size grows?*". This question is valuable as the time needed to run an algorithm or function varies based on input size. Computer scientists refer to this concept as computational complexity [252], [253], which delves into the resources needed to run computer programs, with a primary focus on computational time (time complexity) and memory requirements (space complexity).

In this section, a computational analysis is performed to investigate how variations in input dimensionality and the incorporation of the DAG-NoCurl framework impact the runtime of DAG-WGAN+. The focus of the investigation is on DAG-NoCurl, outlined in Algorithm 1, which comprises a total of five steps. The initial and final steps involve optimization, while the intermediary steps are crucial for the functionality of the theoretical framework. To assess the time complexity of DAG-WGAN+, the author calculates the total resources needed to execute each step and adds them together. The space complexity is equivalent to that of NOTEARS and its related extensions (such as DAG-GNN), which is $\mathcal{O}(d)$.

Optimization in Steps 1 and 5 uses stochastic gradient descent (SGD) [228]. The computational complexity of SGD is computed by taking into account the number of iterations $k$, the data batch size $n$, and the number of variables in the input data $d$. In a typical scenario, the time complexity of SGD is $\mathcal{O}(knd)$. However, in the case of

DAG-WGAN+ the hyperparameters $k$ and $n$ remain constant throughout the process, thus the complexity of Steps 1 and 5 simplifies to $\mathcal{O}(d)$.

Furthermore, Steps 2, 3, and 4 entail individual computations that are performed based on specific equations and modifications of matrix values, as illustrated in Algorithm 1. Each of these steps comprises a series of instructions executed once per run. Step 2 calculates $\psi$ by applying the formula $-L^{+}\nabla^{T}\big(0.5*(C(\mathbf{A}_{pre})-C(\mathbf{A}_{pre})^{T})\big)$, where all components are fixed except for the dimensions of $\mathbf{A}_{pre} \in \mathbb{R}^{d\times d}$. As this step involves matrix subtraction, its time complexity is $\mathcal{O}(d^2)$. Step 3 involves a function that encompasses a sequence of matrix operations involving subtraction and scaling. Similarly to the previous step, the runtime of Step 3 is directly proportional to the size of $\mathbf{A}_{pre} \in \mathbb{R}^{d\times d}$, resulting in a computational complexity of $\mathcal{O}(d^2)$. Step 4 calculates $\mathbf{A}_{init}$ based on $W \circ ReLU(\nabla\psi)$, where all parameters are fixed except for $W \in \mathbb{R}^{d\times d}$. The symbol $\circ$ denotes the Hadamard product of matrices, which is a variant of matrix multiplication. As a result, the computational complexity of Step 4 is also $\mathcal{O}(d^2)$. The time complexity of the thresholding procedures at the conclusion of Steps 1 and 5 also increases quadratically as their respective inputs grow in size. Given that Steps 1 and 5 exhibit linear complexity while Steps 2, 3, and 4 demonstrate quadratic complexity, the time complexity of DAG-WGAN+ is $\mathcal{O}(2d + 3d^2)$, which simplifies to $\mathcal{O}(d + d^2)$. Since computational complexity is estimated based on the term that is most rapidly growing, the time complexity of DAG-WGAN+ is $\mathcal{O}(d^2)$, which is lower than that of the original DAG-WGAN model, $\mathcal{O}(d^3)$, therefore proving that the replacement of the augmented Lagrangian with DAG-NoCurl in the training algorithm of DAG-WGAN+ leads to more efficient causal structure learning.

## 4.2 Experiments

In this section, the author presents a set of experiments that demonstrate how their method outperforms current best practices. Additionally, comparison is made between DAG-WGAN+ and its predecessor, as they share the same structure. This allows for an examination of how the MMD-based mutual information term from [67] and the efficient structure learning with DAG-NoCurl by [48] may enhance the process of

generative adversarial DAG learning.

Three sets of experiments have been conducted utilizing continuous, discrete, and vector data formats on both synthetic and benchmark datasets. The accuracy of the recovered DAG structures is evaluated through the Structural Hamming Distance (SHD) [105]. Furthermore, the computational time of each model is documented and the data reconstruction performance of DAG-WGAN+ is evaluated. The results of all these experiments are discussed in Section 4.3.

### 4.2.1 Continuous experiments

In order to assess the accuracy of a causal graph $\mathbf{G_A}$ identified from observational data $\mathbf{X}$, it is essential to compare it to the true underlying graph $\mathbf{G_A^0}$ of $\mathbf{X}$. To this end, to ensure fair comparisons, experiments involve the use of synthetic datasets (with their respective ground truth graphs) that are generated from the same structural equations as those utilized in the experiments of DAG-GNN [35], DAG-NoCurl [48], DAG-WGAN [75] and GraN-DAG [36].

The process of generating synthetic data follows the methodology outlined in the original DAG-WGAN approach. Initially, an Erdos-Renyi (ER) graph [243] with an expected node degree of 3 is generated. Subsequently, a series of Structural Equation Models (SEM) are employed to generate both linear and non-linear data observations. Specifically, the equations utilized include: linear SEM ($\mathbf{X} = \mathbf{A}^T X + \mathcal{Z}$), non-linear-1 SEM ($\mathbf{X} = \mathbf{A}^T cos(X+1)+\mathcal{Z}$), non-linear-2 SEM ($\mathbf{X} = 2sin(\mathbf{A}^T(X+0.5*1))+\mathbf{A}^T(X+0.5*1)+\mathcal{Z}$), post-non-linear-1 SEM ($\mathbf{X} = sinh(\mathbf{A}^T cos(X+1)+\mathcal{Z})$), and post-non-linear-2 SEM ($\mathbf{X} = tanh(2sin(\mathbf{A}^T(X+0.5*1))+\mathbf{A}^T(X+0.5*1)+\mathcal{Z})$). Further details regarding the experimental setup for continuous data, such as graph dimensions, sample sizes, and repetitions per model, are provided in Section 3.2.3 under the Continuous Data segment. The outcomes of the investigation are presented in Tables 4.1, 4.2, 4.3, 4.4, and 4.5.

Table 4.1: Efficient Generative Adversarial DAG Learning from Linear Scalar Data Samples

| Model | SHD (5000 linear samples) | | | | |
|---|---|---|---|---|---|
| | **d = 10** | **d = 20** | **d = 30** | **d = 50** | **d = 100** |
| DAG-NOTEARS | 8.6 ± 7.74 | 3.8 ± 8.2 | 8.4 ± 9.1 | 26.6 ± 28.3 | 102.6 ± 52.4 |
| DAG-NoCurl | 7.4 ± 7.46 | 3.2 ± 7.6 | 7 ± 8.2 | 24.2 ± 25.4 | 97.1 ± 48.6 |
| DAG-NOTEARS-MLP | 2.4 ± 4.2 | 2.8 ± 5.1 | 5.6 ± 5.8 | 17.4 ± 15.3 | 84.4 ± 27.2 |
| DAG-GNN | 4 ± 6.47 | 2.6 ± 8.8 | 4.4 ± 9.4 | 21 ± 21.6 | 77.8 ± 33.4 |
| GAE | 3.5 ± 4.4 | 2.5 ± 5.7 | 4 ± 5.1 | 18.3 ± 11.2 | 64.9 ± 19.8 |
| GraN-DAG | 1 ± 5.1 | 1.5 ± 6.9 | 3.8 ± 7.35 | 16.8 ± 13.5 | 53.7 ± 21.7 |
| VI-DP-DAG | **0.6 ± 4.3** | **1.4 ± 4.7** | 3.5 ± 4.2 | 13.4 ± 10.8 | 42.8 ± 18.3 |
| DAG-WGAN | 3 ± 3.9 | 2.4 ± 2.1 | 3.2 ± 2.5 | 9.6 ± 8.26 | 24 ± 16.4 |
| **DAG-WGAN+** | 1.3 ± 3.1 | 1.8 ± 1.5 | **2.8 ± 1.8** | **7.2 ± 7.6** | **16.8 ± 11.2** |

Table 4.2: Efficient Generative Adversarial DAG Learning from Non-Linear-1 Scalar Data Samples

| Model | SHD (5000 non-linear-1 samples) | | | | |
|---|---|---|---|---|---|
| | **d = 10** | **d = 20** | **d = 30** | **d = 50** | **d = 100** |
| DAG-NOTEARS | 11.4 ± 4.6 | 19.8 ± 7.2 | 41 ± 9.6 | 53.4 ± 26.3 | 107.8 ± 43.5 |
| DAG-NoCurl | 10.8 ± 4.4 | 17.3 ± 6.8 | 27 ± 8.3 | 51.6 ± 21 | 105.6 ± 40.8 |
| DAG-NOTEARS-MLP | 8.4 ± 3.3 | 15.6 ± 5.2 | 25.2 ± 6.4 | 42.7 ± 16.9 | 91.3 ± 27.3 |
| DAG-GNN | 8.8 ± 4 | 12.4 ± 6.1 | 27.6 ± 7.7 | 44.3 ± 19.7 | 84 ± 33.8 |
| GAE | 8 ± 3.2 | 11.7 ± 4.2 | 25.8 ± 4.9 | 40.6 ± 13.4 | 81.5 ± 20.2 |
| GraN-DAG | 4.6 ± 3.8 | 6.2 ± 4.7 | 23 ± 5.8 | 38.5 ± 15.3 | 77.9 ± 22.6 |
| VI-DP-DAG | **3.2 ± 2.9** | **4.8 ± 4** | 21.7 ± 4.5 | 34.3 ± 12.6 | 70.6 ± 18.9 |
| DAG-WGAN | 7.4 ± 2.4 | 10.6 ± 3.6 | 20.4 ± 4.3 | 31.4 ± 11.2 | 65.4 ± 17.8 |
| **DAG-WGAN+** | 5.6 ± 1.9 | 7.7 ± 2.3 | **16.6 ± 3.1** | **22.2 ± 8.4** | **46.8 ± 13.1** |

Table 4.3: Efficient Generative Adversarial DAG Learning from Non-Linear-2 Scalar Data Samples

| Model | SHD (5000 non-linear-2 samples) | | | | |
|---|---|---|---|---|---|
| | **d = 10** | **d = 20** | **d = 30** | **d = 50** | **d = 100** |
| DAG-NOTEARS | 9.1 ± 4.4 | 23.8 ± 7.6 | 36.6 ± 10 | 41.8 ± 25.8 | 121.8 ± 44.5 |
| DAG-NoCurl | 8.4 ± 4.2 | 19.4 ± 7.2 | 28 ± 8.6 | 37.5 ± 20.5 | 113.2 ± 41.8 |
| DAG-NOTEARS-MLP | 5.2 ± 3.1 | 12.3 ± 5.6 | 23.4 ± 6.7 | 22.6 ± 16.4 | 104.2 ± 28.3 |
| DAG-GNN | 3.2 ± 3.8 | 5.4 ± 6.5 | 14.3 ± 8.1 | 16.2 ± 19.2 | 90.8 ± 34.8 |
| GAE | 2.9 ± 2.5 | 4.6 ± 4.1 | 13.2 ± 5.2 | 15.3 ± 12.4 | 76.1 ± 21.9 |
| GraN-DAG | 1.8 ± 3.6 | 3.2 ± 5.1 | 12.4 ± 6.2 | 14.7 ± 14.8 | 55.6 ± 23.6 |
| VI-DP-DAG | **1 ± 2.4** | **2.6 ± 3.8** | 11.5 ± 4.9 | 12.9 ± 11.1 | 37.2 ± 20.7 |
| DAG-WGAN | 2.6 ± 2.2 | 3.6 ± 3.3 | 10.4 ± 4.6 | 12 ± 10.7 | 22.6 ± 19.8 |
| **DAG-WGAN+** | 2.2 ± 1.7 | 3.4 ± 2.7 | **6.4 ± 3.4** | **11.2 ± 7.9** | **19.3 ± 12.2** |

Table 4.4: Efficient Generative Adversarial DAG Learning from Post-Non-Linear-1 Scalar Data Samples

| **Model** | **SHD (5000 post-non-linear-1 samples)** | | | | |
|---|---|---|---|---|---|
| | **d=10** | **d=20** | **d=30** | **d=50** | **d=100** |
| DAG-GNN | 11.2 ± 7.5 | 18.6 ± 8 | 36.7 ± 11.4 | 60.1 ± 28.8 | 114.3 ± 48.2 |
| GAE | 10.3 ± 5.6 | 16.6 ± 6.2 | 33.4 ± 9.8 | 53.2 ± 22.7 | 97.8 ± 35.2 |
| DAG-WGAN | 8.7 ± 3.3 | 13.4 ± 4.5 | 26.5 ± 7.3 | 41.3 ± 16.2 | 85.6 ± 27.8 |
| **DAG-WGAN+** | **6.8 ± 2.2** | **10.7 ± 3.4** | **21.7 ± 6.1** | **30.6 ± 12.5** | **63.4 ± 19.7** |

Table 4.5: Efficient Generative Adversarial DAG Learning from Post-Non-Linear-2 Scalar Data Samples

| **Model** | **SHD (5000 post-non-linear-2 samples)** | | | | |
|---|---|---|---|---|---|
| | **d=10** | **d=20** | **d=30** | **d=50** | **d=100** |
| DAG-GNN | 9.3 ± 7.8 | 14.2 ± 10.7 | 25.7 ± 13.3 | 34.8 ± 28.5 | 125.4 ± 46.4 |
| GAE | 8.1 ± 5.5 | 12.8 ± 8.6 | 22.3 ± 10.4 | 30 ± 23.7 | 103.9 ± 36.1 |
| DAG-WGAN | 6. ± 4.7 | 10.6 ± 5.3 | 16.8 ± 8.2 | 24.1 ± 17.8 | 45.2 ± 32.5 |
| **DAG-WGAN+** | **4 ± 3.4** | **7.9 ± 4.8** | **12.7 ± 6.6** | **20.4 ± 12.1** | **38.6 ± 26.7** |

### 4.2.2 Vector experiments

Vector experiments are also carried out using synthetic continuous data, where the sizes of the graph and the quantity of samples remain the same. The data generation process is the same as with the one described in Section 4.2.1. Leveraging the architecture of the DAG-GNN [35] framework, the model can naturally handle vector data by expanding the column dimension to more than 1 for each variable in the observations. In this research, the column dimension is specified as 5, enabling direct comparison with DAG-GNN [35], GAE [39], and DAG-WGAN [75], which were the only three models recognized for managing vector data at the time of the experiment. The outcomes of the study are presented in Tables 4.6, 4.7, 4.8, 4.9, and 4.10.

Table 4.6: Efficient Generative Adversarial DAG Learning from Linear Vector Data Samples

| **Model** | **SHD (5000 linear samples)** | | | | |
|---|---|---|---|---|---|
| | **d = 10** | **d = 20** | **d = 30** | **d = 50** | **d = 100** |
| DAG-GNN | 3.6 ± 2.4 | 10.2 ± 8.8 | 16.4 ± 15.3 | 32.2 ± 24.7 | 65.8 ± 44.1 |
| GAE | 3.5 ± 2.3 | 9.6 ± 8.1 | 14.2 ± 13.4 | 28.3 ± 22.5 | 61.2 ± 40.6 |
| DAG-WGAN | 3.3 ± 2.1 | 9.2 ± 7.4 | 12.8 ± 11.2 | 24.7 ± 21.1 | 59.8 ± 38.3 |
| **DAG-WGAN+** | **3 ± 1.8** | **8.6 ± 6** | **10.5 ± 9.4** | **21.8 ± 15.3** | **51.3 ± 30.2** |

Table 4.7: Efficient Generative Adversarial DAG Learning from Non-Linear-1 Vector Data Samples

| Model | SHD (5000 non-linear-1 samples) | | | | |
|---|---|---|---|---|---|
| | **d = 10** | **d = 20** | **d = 30** | **d = 50** | **d = 100** |
| DAG-GNN | 5.8 ± 4.6 | 11.2 ± 9.8 | 23.6 ± 18.4 | 42.8 ± 32.9 | 95.2 ± 56.3 |
| GAE | 4.2 ± 3.8 | 10.4 ± 7.3 | 22.5 ± 15 | 41.3 ± 29.8 | 90 ± 49.5 |
| DAG-WGAN | 3.8 ± 2.2 | 8.4 ± 6 | 19.2 ± 12.7 | 40.2 ± 27.4 | 86.4 ± 43.2 |
| **DAG-WGAN+** | **3.2 ± 1.7** | **7.6 ± 5.2** | **15.4 ± 8.9** | **35.7 ± 18.7** | **77.6 ± 31.8** |

Table 4.8: Efficient Generative Adversarial DAG Learning from Non-Linear-2 Vector Data Samples

| Model | SHD (5000 non-linear-2 samples) | | | | |
|---|---|---|---|---|---|
| | **d = 10** | **d = 20** | **d = 30** | **d = 50** | **d = 100** |
| DAG-GNN | 4 ± 2.8 | 7.2 ± 4.8 | 15.6 ± 11.9 | 39 ± 30.4 | 92.4 ± 51.7 |
| GAE | 3.6 ± 2.4 | 6.8 ± 4.2 | 14.8 ± 10.2 | 36.5 ± 28.3 | 88.3 ± 46.9 |
| DAG-WGAN | 3.2 ± 2 | 6.4 ± 3.6 | 13.2 ± 8.5 | 33.3 ± 26.2 | 85.8 ± 42.4 |
| **DAG-WGAN+** | **2.8 ± 1.6** | **5.1 ± 2.7** | **11.7 ± 6.3** | **28.4 ± 16.7** | **74.3 ± 29.6** |

Table 4.9: Efficient Generative Adversarial DAG Learning from Post-Non-Linear Vector Data Samples 1

| Model | SHD (5000 post-non-linear-1 samples) | | | | |
|---|---|---|---|---|---|
| | **d=10** | **d=20** | **d=30** | **d=50** | **d=100** |
| DAG-GNN | 9.2 ± 6.9 | 16.8 ± 11.2 | 33.7 ± 21.5 | 66.3 ± 43.7 | 125.2 ± 71.1 |
| GAE | 7.5 ± 5 | 13.4 ± 8.3 | 28.6 ± 18.4 | 58.4 ± 39.8 | 111.7 ± 58.2 |
| DAG-WGAN | 5.9 ± 3.1 | 10.2 ± 6.6 | 23.3 ± 16.7 | 50.7 ± 31.3 | 98.1 ± 49.5 |
| **DAG-WGAN+** | **4.4 ± 2.5** | **9 ± 5.4** | **19.5 ± 10.6** | **41.7 ± 24.2** | **87.3 ± 39.7** |

Table 4.10: Efficient Generative Adversarial DAG Learning from Post-Non-Linear Vector Data Samples 2

| Model | SHD (5000 post-non-linear-2 samples) | | | | |
|---|---|---|---|---|---|
| | **d=10** | **d=20** | **d=30** | **d=50** | **d=100** |
| DAG-GNN | 8.9 ± 4.1 | 15.6 ± 8.2 | 27.4 ± 19.2 | 63.8 ± 41.6 | 118.7 ± 66.7 |
| GAE | 6.7 ± 3.8 | 12.7 ± 7.6 | 24.3 ± 16.4 | 51.2 ± 38.5 | 106.4 ± 57.2 |
| DAG-WGAN | 5.4 ± 3.1 | 9.8 ± 5.9 | 19.7 ± 12.5 | 43.1 ± 32.8 | 98.3 ± 48.6 |
| **DAG-WGAN+** | **3.7 ± 2.4** | **7.3 ± 4.1** | **15.4 ± 9.8** | **34.5 ± 22.6** | **81.6 ± 37.5** |

### 4.2.3 Benchmark data experiments

The benchmark data experiments involve the use of datasets such as Child, Alarm, Hailfinder, and Pathfinder, along with their corresponding ground truths sourced from the Bayesian Network Repository `https://www.bnlearn.com/bnrepository`. These

datasets are acquired specifically for scalability assessment and to guarantee a fair comparison with the state-of-the-art. Furthermore, this experimental configuration allows for direct assessment of the influence of MMD by comparing this approach with DAG-WGAN [75]. The results are detailed in Table 4.11.

Table 4.11: Efficient Generative Adversarial DAG Learning with Benchmark Data Samples

| **Dataset** | **Nodes** | **SHD** | | |
|---|---|---|---|---|
| | | **DAG-WGAN** | **DAG-GNN** | **DAG-WGAN+** |
| Child | 20 | 20 | 30 | 19 |
| Alarm | 37 | 36 | 55 | 35 |
| Hailfinder | 56 | 73 | 71 | 66 |
| Pathfinder | 109 | 196 | 218 | 194 |

### 4.2.4 Real data experiments

In order to establish the practical applicability of their algorithm in a real-world scenario, the author showcases the effectiveness of DAG-WGAN+ using a dataset related to genetic protein and phospholipids [21]. This dataset called Sachs, obtained from `https://www.bnlearn.com/bnrepository/`, comprises 11 variables and approximately 7450 samples. The results of the conducted experiments can be found in Table 4.12.

Table 4.12: Real Data Experiments conducted on the Sachs Dataset

| **Model** | **Sachs Datatset** |
|---|---|
| | **SHD / Time Estimation** |
| DAG-WGAN | 17 (00:15:33) |
| DAG-GNN | 25 (00:13:57) |
| GAE | 20 (00:09:18) |
| GraN-DAG | 17 (00:12:28) |
| VI-DP-DAG | 16 (00:04:35) |
| **DAG-WGAN+** | **15 (00:03:09)** |

### 4.2.5 Time-wise performance

The author recorded the time needed to achieve the accuracy of the discovered graphs reported in the preceding sections. To achieve this, the author used the 'time' library in Python. Specifically, the code for DAG-WGAN+ was encapsulated between two lines: 1) 't = time.time()', which records the current time in seconds and is used as a starting timestamp; and 2) ' print("Programm finished in: "+ str(time.strftime("%H:%M:%S", time.gmtime(time.time() - t))))', which calculates how much time has passed since the first timestamp and converts the elapsed time into an easily readable string-based time structure in hours, minutes, and seconds. There is no notable difference in the time required to learn causal structures in datasets with lower dimensions. However, a substantial discrepancy is evident in higher dimensions, where DAG-WGAN+ achieves comparable results in significantly less time - less than thirty minutes compared to one or a few hours for all the other methods. It is important to note that these results are not definitive but only indicative of good performance as they are produced using the following hardware: 13th Gen Intel(R) Core(TM) i7-13700H (2.40 GHz), 32.0 GB RAM, NVIDIA GeForce RTX 4060 GPU with 8 GB VRAM. As a result, by relying on supercomputers or higher-quality hardware one can reduce these times even further. The outcomes are provided in Tables 4.13 - 4.22.

Table 4.13: Time Duration Comparison with Linear Vector Data Samples

| **Model** | **Time Duration (5000 linear samples)** | | | | |
|---|---|---|---|---|---|
| | **d = 10** | **d = 20** | **d = 30** | **d = 50** | **d = 100** |
| DAG-GNN | 00:25:06 | 00:39:40 | 01:00:27 | 01:29:15 | 02:19:29 |
| GAE | 00:18:39 | 00:35:21 | 00:49:37 | 01:01:35 | 01:49:38 |
| DAG-WGAN | 00:27:46 | 00:44:56 | 01:21:27 | 01:53:46 | 03:11:39 |
| **DAG-WGAN+** | **00:16:10** | **00:16:21** | **00:16:29** | **00:17:18** | **00:22:07** |

Table 4.14: Time Duration Comparison with Non-Linear-1 Vector Data Samples

| **Model** | **Time Duration (5000 non-linear-1 samples)** | | | | |
|---|---|---|---|---|---|
| | **d = 10** | **d = 20** | **d = 30** | **d = 50** | **d = 100** |
| DAG-GNN | 00:28:39 | 00:38:50 | 00:57:19 | 01:20:32 | 02:23:22 |
| GAE | 00:19:24 | 00:33:56 | 00:51:13 | 01:09:47 | 01:52:31 |
| DAG-WGAN | 00:32:29 | 00:45:01 | 01:25:38 | 01:58:33 | 03:05:23 |
| **DAG-WGAN+** | **00:16:29** | **00:16:48** | **00:16:53** | **00:17:32** | **00:21:41** |

Table 4.15: Time Duration Comparison with Non-Linear-2 Vector Data Samples

| **Model** | **Time Duration (5000 non-linear-2 samples)** | | | | |
|---|---|---|---|---|---|
| | **d = 10** | **d = 20** | **d = 30** | **d = 50** | **d = 100** |
| DAG-GNN | 00:24:28 | 00:35:00 | 00:55:24 | 01:24:59 | 02:25:14 |
| GAE | 00:17:26 | 00:30:55 | 00:47:13 | 01:11:21 | 01:55:24 |
| DAG-WGAN | 00:31:14 | 00:43:29 | 01:22:41 | 01:49:07 | 02:56:19 |
| **DAG-WGAN+** | **00:15:55** | **00:16:08** | **00:16:16** | **00:17:20** | **00:22:53** |

Table 4.16: Time Duration Comparison with Post-Non-Linear-1 Vector Data Samples

| **Model** | **Time Duration (5000 non-linear-1 samples)** | | | | |
|---|---|---|---|---|---|
| | **d = 10** | **d = 20** | **d = 30** | **d = 50** | **d = 100** |
| DAG-GNN | 00:28:59 | 00:39:18 | 00:58:55 | 01:21:54 | 02:26:17 |
| GAE | 00:19:45 | 00:34:39 | 00:52:49 | 01:11:09 | 01:54:26 |
| DAG-WGAN | 00:32:59 | 00:45:44 | 01:27:14 | 01:59:55 | 03:07:19 |
| **DAG-WGAN+** | **00:16:49** | **00:17:31** | **00:18:29** | **00:18:54** | **00:23:36** |

Table 4.17: Time Duration Comparison with Post-Non-Linear-2 Vector Data Samples

| **Model** | **Time Duration (5000 non-linear-2 samples)** | | | | |
|---|---|---|---|---|---|
| | **d = 10** | **d = 20** | **d = 30** | **d = 50** | **d = 100** |
| DAG-GNN | 00:24:50 | 00:35:43 | 00:57:03 | 01:26:21 | 02:27:09 |
| GAE | 00:17:48 | 00:31:38 | 00:48:49 | 01:12:43 | 01:57:19 |
| DAG-WGAN | 00:31:36 | 00:44:12 | 01:24:17 | 01:50:29 | 02:58:14 |
| **DAG-WGAN+** | **00:16:17** | **00:16:51** | **00:17:53** | **00:18:42** | **00:24:48** |

Table 4.18: Time Duration Comparison with Linear Scalar Data Samples

| **Model** | **Time Duration (5000 linear samples)** | | | | |
|---|---|---|---|---|---|
| | **d = 10** | **d = 20** | **d = 30** | **d = 50** | **d = 100** |
| DAG-NOTEARS-MLP | 00:16:02 | 00:35:21 | 00:49:48 | 05:38:57 | 10:25:19 |
| DAG-GNN | 00:23:20 | 00:32:15 | 01:01:15 | 01:16:27 | 03:03:06 |
| GAE | 00:18:36 | 00:23:45 | 00:37:41 | 00:59:11 | 02:31:29 |
| GraN-DAG | 00:25:12 | 00:37:41 | 01:39:38 | 02:11:29 | 04:09:56 |
| DAG-WGAN | 01:45:42 | 01:45:34 | 02:25:11 | 03:24:36 | 5:06:34 |
| VI-DP-DAG | 00:17:22 | 00:20:51 | 00:23:17 | 00:27:33 | 00:31:16 |
| **DAG-WGAN+** | **00:14:32** | **00:15:35** | **00:16:58** | **00:17:30** | **00:19:25** |

Table 4.19: Time Duration Comparison with Non-Linear-1 Scalar Data Samples

| **Model** | **Time Duration (5000 non-linear-1 samples)** | | | | |
|---|---|---|---|---|---|
| | **d = 10** | **d = 20** | **d = 30** | **d = 50** | **d = 100** |
| DAG-NOTEARS-MLP | 00:17:56 | 00:36:21 | 00:44:31 | 05:53:36 | 10:48:54 |
| DAG-GNN | 00:22:50 | 00:35:54 | 00:51:12 | 01:18:40 | 02:56:54 |
| GAE | 00:18:53 | 00:32:17 | 00:42:34 | 01:06:22 | 02:24:39 |
| GraN-DAG | 00:24:18 | 00:39:23 | 01:44:15 | 02:39:41 | 04:16:26 |
| DAG-WGAN | 01:54:12 | 02:19:23 | 02:32:43 | 03:51:10 | 5:22:34 |
| VI-DP-DAG | 00:16:53 | 00:21:48 | 00:24:47 | 00:28:11 | 00:32:15 |
| **DAG-WGAN+** | **00:15:29** | **00:16:05** | **00:17:33** | **00:18:08** | **00:19:17** |

Table 4.20: Time Duration Comparison with Non-Linear-2 Scalar Data Samples

| Model | Time Duration (5000 non-linear-2 samples) | | | | |
|---|---|---|---|---|---|
| | **d = 10** | **d = 20** | **d = 30** | **d = 50** | **d = 100** |
| DAG-NOTEARS-MLP | 00:34:08 | 01:57:28 | 04:03:41 | 07:54:48 | 09:54:10 |
| DAG-GNN | 00:24:16 | 00:29:44 | 00:49:28 | 01:21:32 | 02:25:27 |
| GAE | 00:17:44 | 00:25:46 | 00:40:57 | 01:03:42 | 01:57:39 |
| GraN-DAG | 00:29:12 | 00:42:37 | 01:20:30 | 01:44:35 | 02:41:13 |
| DAG-WGAN | 01:37:00 | 02:25:12 | 02:44:34 | 03:50:04 | 04:18:29 |
| VI-DP-DAG | 00:15:33 | 00:20:14 | 00:25:03 | 00:27:43 | 00:31:22 |
| **DAG-WGAN+** | **00:14:26** | **00:15:55** | **00:17:16** | **00:19:04** | **00:20:04** |

Table 4.21: Time Duration Comparison with Post-Non-Linear-1 Scalar Data Samples

| Model | Time Duration (5000 non-linear-1 samples) | | | | |
|---|---|---|---|---|---|
| | **d = 10** | **d = 20** | **d = 30** | **d = 50** | **d = 100** |
| DAG-GNN | 00:23:52 | 00:35:57 | 00:54:10 | 01:18:40 | 02:58:49 |
| GAE | 00:19:15 | 00:33:00 | 00:44:10 | 01:07:44 | 02:26:34 |
| DAG-WGAN | 01:55:34 | 02:20:06 | 02:34:19 | 03:52:32 | 5:23:29 |
| **DAG-WGAN+** | **00:15:51** | **00:16:48** | **00:19:09** | **00:19:30** | **00:21:12** |

Table 4.22: Time Duration Comparison with Post-Non-Linear-2 Scalar Data Samples

| Model | Time Duration (5000 non-linear-2 samples) | | | | |
|---|---|---|---|---|---|
| | **d = 10** | **d = 20** | **d = 30** | **d = 50** | **d = 100** |
| DAG-GNN | 00:24:16 | 00:30:27 | 00:51:04 | 01:23:16 | 02:27:22 |
| GAE | 00:18:06 | 00:26:29 | 00:42:33 | 01:05:04 | 01:59:34 |
| DAG-WGAN | 01:37:22 | 02:25:55 | 02:46:10 | 03:51:26 | 04:20:24 |
| **DAG-WGAN+** | **00:14:48** | **00:16:28** | **00:18:52** | **00:20:26** | **00:21:59** |

It is crucial to note that the outcomes mentioned do not demonstrate the quadratic complexity of the model. This is due to the fact that Steps 2, 3, and 4 of Algorithm 1 are performed only once per execution, with the main focus of this method being on optimization (specifically Steps 1 and 5). As a result, DAG-WGAN+ exhibits a behavior more consistent with a linear growth in the duration of time.

### 4.2.6 Ablation study

The author has conducted an additional ablation study to determine the impact of various aspects of the model on the results of causality learning. These experiments encompass: 1) Comparing model training with the generative adversarial loss (Wasserstein distance) against training solely with the reconstruction loss (referred to as **w/o GAN**) to assess the role of generative adversarial training; 2) Contrasting model train-

ing with and without the encoder in the model architecture (referred to as **w/o AE**) to evaluate the contribution of the encoder; 3) Introducing an extra step to achieve a final approximate solution (referred to as **6 steps** - for further details, see Efficient causal structure learning with DAG-NoCurl in Section 4.1.3); 4) Analyzing model training with and without considering the mutual information between data and latent variables to understand the impact of the MMD loss (referred to as **w/o MMD**); and 5) Implementing model training as described in Section 4.1.3 (referred to as **default case**). The study was carried out using the Sachs dataset [21].

Table 4.23: Ablation Studies conducted on our model with Sachs Dataset

| **Model** | **Sachs Datatset** |
|---|---|
| | **SHD / Time Estimation** |
| w/o GAN | 25 (00:03:00) |
| w/o AE | 22 (00:02:51) |
| 6 steps | 19 (00:03:11) |
| w/o MMD | 16 (00:03:05) |
| default case | **15 (00:03:09)** |

### 4.2.7 Data quality

The data reconstruction capabilities of DAG-WGAN+ have also been investigated. To achieve this, the author replicated the 'dimension-wise' and completeness experiments as detailed in the Data Quality segment of Section 3.2.3 using the same datasets to ensure a fair comparison between DAG-WGAN+ and its predecessor. This experimental setup also enables the assessment of the impact of MMD on the data reconstruction process. The results in terms of recovered causal graphs, correlation matrices, feature importance, and data integrity remain consistent and are not provided in this section; readers are directed to the Data integrity analysis of DAG-WGAN for more details. Regarding data diversity, a slight advantage is observed in favor of DAG-WGAN+, as illustrated in Figure 4.1.

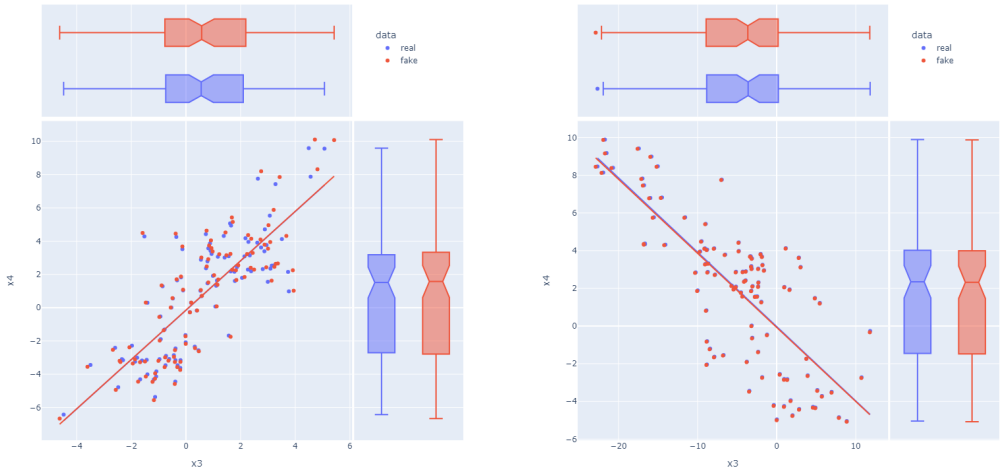



Figure 4.1: Real and synthetic feature distributions (x3,x4), in the case SHD = 0 (left) and when SHD is farthest away from 0 (right)

## 4.3 Result Analysis

The outcomes presented in Section 4.2 illustrate the capability of DAG-WGAN+ to compete against and surpass the current leading methods in causal structure learning. Specifically, Tables 4.1 - 4.3 demonstrate the superiority of the model over all other approaches in every scenario (linear, non-linear-1, and non-linear-2) when working with high-dimensional continuous data under the assumption of the Additive Noise Model. It is only surpassed by GraN-DAG [36] and VI-DP-DAG [49] in the case of low-dimensional data. Moreover, as indicated by the findings in Tables 4.4 and 4.5, DAG-WGAN+ excels over its competitors in all instances (post-non-linear-1 and post-non-linear-2) under the assumption of the Post-Nonlinear Model.

The outcomes of the vector experiments illustrated in Tables 4.6 - 4.10 demonstrate that DAG-WGAN+ outperforms other methods when utilized with vector data, irrespective of the assumed structural equation model. The approach of the author consistently generates the most accurate DAG compared to all other approaches examined in the experiment, in various dimensions and cases. Similar to the continuous data study, the difference in accuracy between DAG-WGAN+ and its rivals becomes

more apparent as the size of the data variables increases.

The proposed method also surpasses all other models evaluated in the benchmark and real-world experiments. The findings presented in Table 4.11 indicate that, when applied to the Child, Alarm, Hailfinder, and Pathfinder datasets, DAG-WGAN+ successfully reconstructed the most accurate causal structures. Similarly, the data in Table 4.12 demonstrate that DAG-WGAN+ exhibited superior performance in terms of both accuracy and computational efficiency on the Sachs dataset [21]. Moreover, the timings documented in Tables 4.13 - 4.22 reveal a consistent pattern where the model identifies accurate graphs significantly faster than any other cutting-edge method examined in the research, often by orders of magnitude.

The reconstructed data exhibits outstanding quality as well. In particular, the 'dimension-wise', feature importance and correlation experiments produce consistent results with those in the Data Integrity segment of Section 3.2.3. The key distinction is the variety in the reconstructed samples, a result of utilizing DRL in the training process. As illustrated in Figure 4.1, there is a minor increase in the standard deviation of the distribution of reconstructed data, leading to a slightly less precise but still satisfactory and more diverse reconstruction outcome.

Last but not least, the results of the ablation study reveal the optimal configuration of loss terms that form the objective function of the approach. The data in Table 4.23 demonstrate that DAG-WGAN+ achieves the highest precision in recovering DAG when trained using a sophisticated loss function that encompasses reconstruction, adversarial, and MMD components. In essence, the efficient generative adversarial approach for learning DAG structures yields superior outcomes when compared to the original DAG-WGAN model [75] across all experiments. Furthermore, it surpasses its rivals in most scenarios and demonstrates the ability to produce results much quicker than the current state-of-the-art methods, as it does not rely on the augmented Lagrangian continuous optimization technique during the training process.

Although DAG-WGAN+ demonstrates potential for combining efficient causal structure learning with disentangled representations and adversarial training, several limitations arise when the model is applied to real-world data (e.g., Sachs dataset). Specif-

ically, a major drawback is the lack of guaranteed semantic alignment. In practice, unsupervised disentanglement often produces latent factors that are statistically independent but do not correspond to meaningful or interpretable real-world variables. This reduces the practical utility of the representations for domains that require clear, actionable insights. Another primary issue is the difficulty associated with evaluating disentanglement in real-world contexts. Unlike synthetic benchmarks, where ground-truth factors are known, real-world data rarely provide clear references for evaluating how well the latent factors correspond to true causal variables.

On the efficient structure learning side, DAG-NoCurl relies on a linear projection of the initial adjacency estimate to enforce acyclicity and refine the graph. While this approach reduces computational complexity, it struggles to capture the nonlinear dependencies, noise, and latent confounding commonly present in real-world data. As a result, the inferred causal graph may include spurious edges or omit true causal relationships, limiting the reliability and interpretability of the learned structure. Together, these challenges highlight that, while DAG-WGAN+ provides a powerful framework for integrating fast causal discovery, disentangled representation learning and adversarial training, its practical applicability remains constrained in complex, noisy, and partially observed real-world settings.

Furthermore, based on the experiments conducted, DAG-WGAN+ effectively tackles some of the issues of its predecessor concerning efficient usability, data quality, and management of diverse data formats (e.g., vector data). However, several drawbacks of the original DAG-WGAN model persist and are inherent in this updated approach as well. The key challenges that still need to be resolved are: 1) **simplicity of architecture** - This leads to discrepancies between data quality and causal discovery, allowing the model to produce high-quality data while struggling to recover accurate causal relationships; 2) **causal generality** - The specific SEM utilized in both the inference and generative models of the Variational Autoencoder (VAE) component restrict causality learning to the semi-parametric assumptions of Additive Noise Models (ANM), limiting its applicability to real-world scenarios; and 3) **mixed data types** - Although the model can currently handle discrete and continuous data separately, it

lacks the capability to manage both types simultaneously in a single dataset, making it unsuitable for complex tabular datasets.

Future research will focus on overcoming the constraints outlined earlier. In particular, a promising direction to improve the disentangled representation component is the introduction of mechanisms that promote semantic alignment between latent factors and meaningful real-world variables, through weakly supervised learning or domain-informed regularization. To that end, the author will employ more reliable evaluation metrics that do not rely on known ground-truth factors, such as the Modularity and Explicitness scores, DCI (Disentanglement-Completeness-Informativeness) framework, Separated Attribute Predictability (SAP) score, Mutual Information Gap (MIG), or Interventional Robustness Score (IRS), to assess disentanglement performance in practical scenarios.

Moreover, on the structure learning side, the author intends on extending the current DAG-NoCurl framework beyond the linear projection of the initial adjacency estimate, which will enable the model to better capture nonlinear dependencies and reduce sensitivity to noise. The author also plans to expand the model to accommodate time-series, mixed and incomplete data. The significance of DAG-WGAN+ in capturing valuable latent features is crucial for understanding causal relationships in scenarios with common causes among variables. Therefore, upcoming studies will explore the capability of the model to tackle the issue of hidden confounders. Subsequent versions of DAG-WGAN+ will have no specific SEM in their architecture. Additionally, they are going to be enhanced with further experiments like sensitivity analysis and investigations into hyper-parameters to identify an optimal configuration that enhances the performance of the approach.

# Chapter 5

# Nonparametric structure learning with nonlinear causal models

This chapter begins with the author revealing the theory and implementation details behind a novel Directed Acyclic Generative Adversarial Framework (DAGAF), designed for joint poly-assumptive causal structure learning and generation of tabular datasets. Specifically, the algorithm explores the application of the PNL model and its subsets, which include LiNGAM and ANM. The recovered causality is utilized in tabular data synthesis to investigate whether the following hypothesis holds: *Is it possible to simultaneously learn an accurate approximation of the original causal mechanisms in a probability distribution and apply them to define a synthetic distribution that produces realistic data samples?* Crucially, the author disentangles causality learning and tabular data generation, eliminating the issues with parallel causal discovery and sample production in a single model instance - for more information see Section 5.1. In addition, a comprehensive theoretical analysis has been conducted to investigate the contribution of the loss terms involved in the training process of their framework and how its identifiability is influenced by non-i.i.d., discrete or incomplete data. DAGAF has been extensively evaluated against leading models in causal structure learning, with empirical evidence indicating its effectiveness in identifying accurate causal approximations from observational data under multiple structural causal model assumptions. Furthermore, an in-depth analysis of the generated data reveals that DAGAF is capable of producing

samples of remarkably high quality. The findings from the research conducted based on this work have been published and can be found in Section 1.7.

## 5.1 DAGAF: A Directed Acyclic Generative Adversarial Framework for joint Structure Learning and Tabular Data Synthesis

The framework of the author is designed to produce synthetic samples by learning the underlying generative process of input data. To accomplish this, DAGAF models a directed acyclic graph (DAG) $\mathbf{G_A}$ that captures the causal relationships within a dataset $\chi$, facilitating the synthesis of realistic samples with minimal loss of fidelity and diversity. Furthermore, the model not only yields testable results on synthetic data, but also demonstrates performance on real-world datasets, as outlined in Section 5.2.3. The objective of the approach is formalized as follows.

**Goal**: Given $n$ i.i.d. observations $\mathbf{X} \sim P(\mathbf{X}) \in \chi$, the framework models $\mathbf{G_A} \approx \mathbf{G_A^0} \in \mathbb{D}$ to learn the set of structural equations $\mathcal{F} = \{f_1, ...f_d\}$, such that $\tilde{X}_j := f_j(Pa_j, \mathcal{Z}_j)$ results in $\tilde{\mathbf{X}} \sim P_{\mathbf{G_A}}(\tilde{\mathbf{X}}) \in \tilde{\chi}$ matching the input.

Initially, the author attempted to achieve this goal by performing simultaneous causal structure learning and adversarial data generation all within a single model instance. This approach proved challenging, as it required the application of loss terms (namely Mean Squared Error (MSE) and Wasserstein Distance (WD)), which are practically incompatible within the context of causality-based adversarial data generation. On the one hand, MSE is essential for causal discovery, but applying the reconstruction loss directly to adversarial training could limit the range of noise needed to generate fake samples, causing significant latent collapse. Conversely, relying solely on the adversarial loss to create fake data can hinder causality modeling, resulting in noisy structures. To overcome these limitations, the author employs a framework based on a divide-and-conquer approach, involving transfer learning to distribute responsibility across

multiple model instances established over a sequence of steps. In summary, an SCM models causal mechanism approximations to describe the structure of the observational data distribution. This causality is then transferred into a DGM, which produces tabular data by emulating the generative process of the observational samples. Figure 5.1 offers a visual representation of the framework pipeline utilized by the model of the author.

Section 5.1.1 elaborates on Step 1, focusing on the recovery of causal structures from $\mathbf{X}$. Moreover, because the framework identifies causal structures by modeling the underlying data generative process of $\mathbf{X}$, it is inherently suitable for sample synthesis. However, this involves an additional training phase (Step 2) requiring the development of a separate Deep Generative Model (DGM) consisting of a discriminator and a generator, as detailed in Section 5.1.3. For a comprehensive overview of the training methodology, refer to Algorithm 2. The architecture and training approach of DAGAF are thoroughly outlined in Section 5.1.4.

**Algorithm 2** DAGAF training algorithm

**Require:** Sample $n$ observational data points $\{\mathbf{X}_1, \ldots, \mathbf{X}_n\}$ from the training data and $d$ noise vectors $\{\mathbf{Z}_1, \ldots, \mathbf{Z}_d\}$ from normal or uniform distributions. Generate $n$ synthetic data samples $\{\tilde{\mathbf{X}}_1, \ldots, \tilde{\mathbf{X}}_n\}$, with data attributes $\tilde{X} := f(X) + \mathcal{Z}$, $\tilde{X}_j := f_j(Pa_j) + \mathcal{Z}_j$ or $\tilde{X}_j := g_j(f_j(Pa_j) + \mathcal{Z}_j)$ depending on whether LiNGAM, ANM or PNL is assumed.

**Ensure:** The acyclicity constraint value $h(\mathbf{A}^{L_0}(f))$ is higher than its tolerance of error $h\_tol$ set to 1e-8. Each step during training has its own instance of DAG-Notears-MLP. Causal information is transferred from the SCM into the DGM architecture.

**Step 1**: Poly-assumptive causal structure learning
LiNGAM, ANM $\rightarrow$ learn $f$ by minimizing a combination of loss terms including adversarial loss (5.1), Mean Squared Error (5.2), Kullback-Lieber divergence (5.3), Maximum Mean Discrepancy (5.4) and the acyclicity constraint from [38].
PNL $\rightarrow$ learn both $f$ and $g^{-1}$ by solving (5.8)
This step recovers a graph representation $\mathbf{G_A}$ of the causal mechanisms in $\mathbf{X}$

**Step 2**: Generative process simulation under multiple structural causal model assumptions
LiNGAM, ANM $\rightarrow$ learn $f$ by computing (5.1)
PNL $\rightarrow$ learn $f$ and $g$ by finding the optimal value for (5.1)
This step models a generative process involving $\mathbf{G_A}$ through adversarial training, producing new data samples.

It is important to note that this framework does not assume any specific model for each step. In fact, any combination of models is possible as long as the following requirements are met:

1. An SCM is employed to learn a graph representation (i.e. an adjacency matrix)

of causal structures from observational data with acyclicity enforced explicitly or implicitly.

2. Causal knowledge is transferable in a meaningful representation from the first to the second step.

3. A DGM performs tabular data synthesis using the discovered causal mechanisms from the first step to generate new samples.

In the rest of this section, the author provides the implementation details of DAGAF and discusses how it integrates within Algorithm 2.

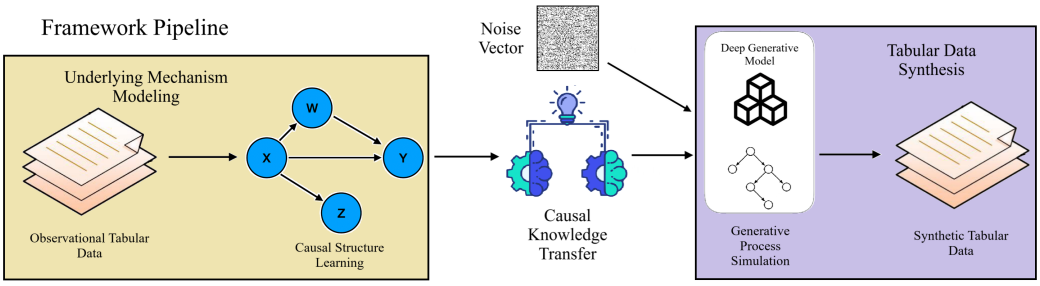



Figure 5.1: Pipeline of the framework for joint causal discovery and tabular data synthesis. Initially, the modeling of the underlying mechanisms describing the observational distribution is performed through a process known as causal structure learning, resulting in an implicit graphical representation (weighted adjacency matrix) consisting of model parameters. Afterwards, tabular data synthesis is achieved by simulating the generative process of the input data by modeling each causal mechanism using parent variables defined in the weighted adjacency matrix from the previous step. Weight (parameter) transfer between model instances facilitates the communication of causal knowledge between the two stages, making the framework heavily reliant on the 'transfer learning' methodology.

### 5.1.1 Modelling causal structure approximations

The DAGAF framework is designed to approximate the underlying causal mechanisms $\{f_j(Pa_j, \mathcal{Z}_j)\}$ that generate the observed data $\mathbf{X}$. According to the (semi) parametric assumptions detailed in Section 2.1.4, each node $X_j \in \mathbf{G_A}$ is defined as a function $f_j : \mathbb{R}^d \to \mathbb{R}$. In this context, the general nonparametric form $\mathbb{E}[X_j|X_{Pa_j}] := \mathbb{E}_{\mathcal{Z}}(f_j(X, \mathcal{Z}))$ simplifies to one of the following models: (i) **Linear non-Gaussian Acyclic Models**

**(LiNGAM)**: $\tilde{X} := f(X) + \mathcal{Z}$, where $f(X)$ is a linear function of $X$ and $\mathcal{Z}$ represents a non-Gaussian noise term that is independent of $X$; (ii) **Additive Noise Models (ANM)**: $\tilde{X}_j := f_j(Pa_j) + \mathcal{Z}_j$, where $f_j$ is a nonlinear function of the parent variables $Pa_j$, and $\mathcal{Z}_j$ is Gaussian and independent of $f_j(Pa_j)$; (iii) **Post-Nonlinear Models (PNL)**: $\tilde{X}_j := g_j(f_j(Pa_j) + \mathcal{Z}_j)$, where $g_j$ is a nonlinear function and $\mathcal{Z}_j$ is Gaussian and independent of $f_j(Pa_j)$.

In the initial phase of DAGAF training, the aim is to learn Directed Acyclic Graphs (DAG) by computing an optimal solution to a sophisticated objective function that blends together various loss terms relevant to causal structure learning. The basic framework encompases the LiNGAM and ANM structural causal models, leveraging adversarial training and a reconstruction loss supplemented by regularization terms to facilitate the synthesis of $\tilde{\mathbf{X}}$ from $\mathbf{X}$. A key advantage of this framework is its adaptability, enabling the basic approach to be extended to support causal discovery under the PNL assumption without major difficulties. The enhanced form broadens the functionality of DAGAF to include PNL by introducing an additional reconstruction loss to model the parameters of the non-linear function $g_j$.

### Adversarial loss with gradient penalty

DAGAF simulates $\mathbf{X}$ by learning how to generate $\tilde{\mathbf{X}}$ through approximations of the causal mechanisms $\{f_j(Pa_j, \mathcal{Z}_j)\} \in P(\mathbf{X})$. Instead of directly modeling $\tilde{\mathbf{X}}$, the emphasis is placed on recovering the set of causal mechanisms $\mathcal{F} = \{f_1, \ldots, f_d\}$, where each $f_j$ is expressed as $f_j(Pa_j; W_j^1, \ldots, W_j^L) + \mathcal{Z}_j$ (see Section 5.1.4 for details). This process involves identifying the immediate parents of each variable, which are encoded within the causal structure of $\mathbf{X}$.

To achieve this, the framework of the author minimizes the Wasserstein distance $\mathbb{W}_p(P(\mathbf{X}), P_{\mathbf{G_A}}(\tilde{\mathbf{X}}))$ by applying adversarial training, which implicitly refines the causal structure $\mathbf{G_A}$ and facilitates the discovery of the underlying causal mechanisms. In DAGAF, adversarial training is formulated as a min-max optimization, where an SCM-based generator $\mathcal{M}$ learns to generate synthetic data to minimize the discrepancy measured by a discriminator $D(.)$, while $D(.)$ is trained to maximize $\mathbb{W}p(P(\mathbf{X}), P_{\mathbf{G_A}}(\tilde{\mathbf{X}}))$.

As a result, DAGAF identifies causal relationships from observational data by learning both the reconstruction process and the distributional asymmetries of $P(\mathbf{X})$. The Wasserstein distance with gradient penalty loss term is defined as follows:

$$\begin{aligned}\mathcal{L}_{\text{adv}}(\mathbf{X}, \tilde{\mathbf{X}}) &= \sup_{\|\phi\|_L \leq 1} \mathbb{E}_{\mathbf{X} \sim P(\mathbf{X})}[\phi(\mathbf{X})] - \mathbb{E}_{\tilde{\mathbf{X}} \sim P_{\mathbf{G_A}}(\tilde{\mathbf{X}})}[\phi(\tilde{\mathbf{X}})] \\ &= \mathbb{E}_{\mathbf{X} \sim P(\mathbf{X})}[D(\mathbf{X})] - \mathbb{E}_{\tilde{\mathbf{X}} \sim P_{\mathbf{G_A}}(\tilde{\mathbf{X}})}[D(\tilde{\mathbf{X}})] \\ &+ \mathbb{E}_{\hat{\mathbf{X}} \sim P(\hat{\mathbf{X}})}[(||\nabla_{\hat{\mathbf{X}}} D(\hat{\mathbf{X}})||_2 - 1)^2],\end{aligned} \tag{5.1}$$

where $\phi(\mathbf{X})$ is a 1-Lipschitz function used to approximate the Wasserstein distance $\mathbb{W}p(P(\mathbf{X}), P_{\mathbf{G_A}}(\tilde{\mathbf{X}}))$. The function $D(\mathbf{X})$ is trained adversarially to learn $\phi(\mathbf{X})$ and distinguish between real $\mathbf{X} \sim P(\mathbf{X})$ and generated samples $\tilde{\mathbf{X}} \sim P_{\mathbf{G_A}}(\tilde{\mathbf{X}})$.

Hence, computing the optimal solution for the loss term (5.1) across all samples from the input and synthetic distributions results in overlap between $P(\mathbf{X})$ and $P_{\mathbf{G_A}}(\tilde{\mathbf{X}})$. This ensures that the synthetic data $\tilde{\mathbf{X}}$ becomes indistinguishable from the original data $\mathbf{X}$ , effectively approximating its generative process - provided the causal structure in $\mathbf{G_A}$ is correctly identified.

**Proposition 5.1.1.** Let the ground-truth graph $\mathbf{G_A^0}$ be the only structure that can generate $P(\mathbf{X})$, then, under the assumption of causal identifiability, applying adversarial training ensures the following: 1) the implicitly generated distribution $P_{\mathbf{G_A}}(\tilde{\mathbf{X}})$ matches $P(\mathbf{X})$ and 2) the causal graph $\mathbf{G_A}$ used to define $P_{\mathbf{G_A}}(\tilde{\mathbf{X}})$ is identical to $\mathbf{G_A^0}$.

$$\mathbb{W}_p(P(\mathbf{X}), P_{\mathbf{G_A}}(\tilde{\mathbf{X}})) = 0 \implies P_{\mathbf{G_A}}(\tilde{\mathbf{X}}) = P(\mathbf{X}) \implies \mathbf{G_A} = \mathbf{G_A^0}.$$

*Proof.* The proof of Proposition 5.1.1 is available in Appendix A.5. □

**Reconstruction with Mean Squared Error**

To improve causal structure learning, the author incorporates a reconstruction loss to the training algorithm of DAGAF. The choice of this particular loss term is predicated upon the need for a suitable metric to assess the distance between $P(\mathbf{X})$ and $P_{\mathbf{G_A}}(\tilde{\mathbf{X}})$ in the context of causal discovery. Previous research have investigated various metrics for measuring distances between data distributions, including Wasserstein-1 [115] and

maximum mean discrepancy (MMD) [67], among others. However, to prevent overparameterization which can skew the causality learning results further away from $\mathbf{G_A^0}$, the author only approximates the means of both distributions, disregarding their unit variance. In this context, the mean squared error (MSE) is considered an appropriate reconstruction loss term.

$$\mathcal{L}_{\text{MSE}}(\mathbf{X}, \tilde{\mathbf{X}}) = \mathbb{E}_{\mathbf{X},\tilde{\mathbf{X}}}(||\mathbf{X} - \tilde{\mathbf{X}}||_2) = \frac{1}{n}\sum_{i=1}^{n}\sum_{j=1}^{d} ||X_{ij} - \{f_j(Pa_j; W_j^1, ..., W_j^L) + \mathcal{Z}_j\}_i||_2 \tag{5.2}$$

By optimizing the parameters of DAGAF using (5.2), the residual distance between individual samples $||\mathbf{X} - \tilde{\mathbf{X}}||$ is minimized, leading the framework to generate $\tilde{\mathbf{X}} \sim P_{\mathbf{G_A}}(\tilde{\mathbf{X}})$ by implicitly identifying the causal relationships of $\mathbf{X}$ encoded in $\mathbf{G_A}$. This reconstruction process effectively leads to a more accurate representation of the causal mechanisms underlying $\mathbf{X}$.

**Proposition 5.1.2.** Incorporating a reconstruction loss term into adversarial training ensures that the distance between individual data points from both synthetic $P_{\mathbf{G_A}}(\tilde{\mathbf{X}})$ and observational $P(\mathbf{X})$ data distributions is minimized. This reduction in noise prevents significant gradient fluctuations, resulting in more stable adversarial convergence.

$$\min_{\mathbf{G_A} \in \mathbb{D}} \mathcal{L}_{\text{MSE}}(\mathbf{X}, \tilde{\mathbf{X}}) = 0 \Rightarrow \forall i, \tilde{\mathbf{X}}_i = \mathbf{X}_i$$

*Proof.* The proof of Proposition 5.1.2 is available in Appendix A.6. □

The experiments of the author highlight the significance of the MSE loss in DAG learning. This observation is consistent with the majority of existing studies in the field, which predominantly employ MSE as their loss function of choice.

### Kullback–Leibler Divergence

Utilizing MSE as a reconstruction loss can result in overfitting to $\mathbf{X}$ and lead to inaccuracies in identifying the causal mechanisms within the generative process of $\tilde{\mathbf{X}}$. To mitigate this issue, the author incorporates Kullback–Leibler divergence (KLD) [58]

as a regularization term. Commonly applied in Variational Autoencoders (VAE), the KLD is a standard component of the Evidence Lower Bound (ELBO) loss function for latent variable regularization. It is defined as $D_{KL}\left(\mathcal{N}(\mu, \sigma^2) \| \mathcal{N}(0,1)\right) = \frac{1}{2}\sum_{i=1}^{n}\left(\sigma_i^2 + \mu_i^2 - \log(\sigma_i^2) - 1\right)$, where $\mu$ and $\sigma$ denote the mean and standard deviation of $\tilde{\mathbf{X}}$. During DAGAF training, this term is used to regularize $\tilde{\mathbf{X}}$. Furthermore, since the model is designed to model only the mean of $P_{\mathbf{G_A}}(\tilde{\mathbf{X}})$ and sets its variance to 1, the regularization function simplifies to:

$$\mathcal{L}_{\text{KLD}}(\mathbf{X}, \tilde{\mathbf{X}}) = D_{KL}(P(\mathbf{X}) || P_{\mathbf{G_A}}(\tilde{\mathbf{X}})) = \frac{1}{2}\sum_{i=1}^{n}(\mu_i^2). \tag{5.3}$$

The author incorporates the Kullback–Leibler divergence (KLD) as a regularization term for $\tilde{\mathbf{X}}$, the model-generated data, to emulate an additive noise scenario where noise is introduced to each data point. This application of KLD encourages the model to generate $\tilde{\mathbf{X}}$ that closely resembles the true data distribution while accounting for the variability introduced by noise. This approach prevents overfitting by ensuring that the generated data captures the natural variations of the real data, resulting in more robust and realistic samples. Since the model is designed to learn causal mechanisms, this regularization technique also helps prevent it from inferring incorrect causal structures, such as mistakenly identifying child nodes as parent nodes.

**Proposition 5.1.3.** The $\mathcal{L}_{\text{KLD}}(\mathbf{X}, \tilde{\mathbf{X}})$ regularization imposes a statistical prior on $P_{\mathbf{G_A}}(\tilde{\mathbf{X}})$, ensuring that the learned distribution remains close to a predefined Gaussian. Moreover, it enhances optimization stability, particularly under additive Gaussian noise, by preventing $P_{\mathbf{G_A}}(\tilde{\mathbf{X}})$ from deviating excessively from a normal distribution, mitigating erratic behavior. By complementing adversarial and MSE losses, it ensures both the alignment and smoothness of $P_{\mathbf{G_A}}(\tilde{\mathbf{X}})$.

*Proof.* The proof of Proposition 5.1.3 is available in Appendix A.7. □

Note that this does not apply to the LiNGAM causal model because, under that specific assumption, the noise term $\mathcal{Z}$ is non-Gaussian.

### Maximum Mean Discrepancy

The reconstruction loss and its regularization term focus exclusively on capturing the mean of $P(\mathbf{X})$, neglecting its variance. This oversight makes the reconstruction process in DAGAF particularly sensitive to rare events or outliers in $P(\mathbf{X})$. To resolve this issue, the author further minimizes the residual discrepancy between the input distribution $\mathbf{X} \sim P(\mathbf{X})$ and the generated data distribution $\tilde{\mathbf{X}} \sim P_{\mathbf{G_A}}(\tilde{\mathbf{X}})$ by incorporating the Maximum Mean Discrepancy (MMD) metric [67]. The kernel trick [254] is employed to efficiently compute the solution for this approach.

$$\begin{aligned}\mathcal{L}_{\text{MMD}}(\mathbf{X}, \tilde{\mathbf{X}}) &= ||\mathbb{E}_{\mathbf{X}\sim P(\mathbf{X})}[k(\mathbf{X})] - \mathbb{E}_{\tilde{\mathbf{X}}\sim P_{\mathbf{G_A}}(\tilde{\mathbf{X}})}[k(\tilde{\mathbf{X}})]||^2_{\mathcal{H}} \\ &= \frac{1}{n}\sum_{i\neq j}^{n} k(\mathbf{X}_i, \mathbf{X}_j) - \frac{2}{n}\sum_{i\neq j}^{n} k(\mathbf{X}_i, \tilde{\mathbf{X}}_j) + \frac{1}{n}\sum_{i\neq j}^{n} k(\tilde{\mathbf{X}}_i, \tilde{\mathbf{X}}_j),\end{aligned} \tag{5.4}$$

where $\mathcal{H}$ denotes the reproducing kernel Hilbert space (RKHS) and $k \in \mathcal{H}$ is a kernel function.

The MMD maximizes mutual information between $P(\mathbf{X})$ and $P_{\mathbf{G_A}}(\tilde{\mathbf{X}})$, ensuring the two distributions match in both their means and overall shapes. By aligning their shapes, the MMD term also helps to reduce discrepancies in their variances. Therefore, applying (5.4) indirectly models the standard deviation of $P_{\mathbf{G_A}}(\tilde{\mathbf{X}})$, addressing latent collapse in $\tilde{\mathbf{X}}$ and discovering the causal mechanisms that generate its outliers.

**Proposition 5.1.4.** Minimizing the Maximum Mean Discrepancy (MMD) loss term $\mathcal{L}_{\text{MMD}}(\mathbf{X}, \tilde{\mathbf{X}})$ encourages the alignment of higher-order moments between the input distribution $P(\mathbf{X})$ and the synthetic distribution $P_{\mathbf{G_A}}(\tilde{\mathbf{X}})$, which supports the adversarial loss in achieving overall distributional alignment.

*Proof.* The proof of Proposition 5.1.4 is available in Appendix A.8. □

The ablation study conducted in support of DAGAF confirms that incorporating the MMD term, as introduced in DAG-GAN [37], contributes to causal discovery.

**Model training under the Post-Nonlinear SCM assumption**

Up to this point, the author has explored the loss terms computed for the LiNGAM and ANM scenarios, where the model output $\tilde{\mathbf{X}}$ is synthesized using causal mechanism approximations $\tilde{X} := f(X) + \mathcal{Z}$ or $\tilde{X}_j = f_j(Pa_j) + \mathcal{Z}_j$. These generated samples are modeled to resemble the training data $\mathbf{X}$ by minimizing $||P(\mathbf{X}) - P_{\mathbf{G_A}}(\tilde{\mathbf{X}})||$. A significant strength of DAGAF lies in its adaptability, enabling it to be extended for handling Post-Nonlinear Models (PNL).

Post-Nonlinear Models (PNL) play a vital role in causal discovery by providing a more realistic framework for capturing non-linear causal relationships in observational data. Additionally, PNL is regarded as a broader, more general framework that includes other identifiable models, such as ANM [240] and LiNGAM [31], as special cases.

$$X_j := g_j(f_j(Pa_j) + \mathcal{Z}_j), \forall j, \mathcal{Z}_j \perp\!\!\!\perp f_j(Pa_j) \tag{5.5}$$

Without loss of generality, the author rearranges (5.5) into

$$\mathcal{Z}_j = g_j^{-1}(X_j) - f_j(Pa_j), \tag{5.6}$$

where $g^{-1}$ is the inverse of $g$. Under this setting (from the rearranged equation), the problem has been broken into two parts, which are to learn $f(\cdot)$ and $g^{-1}(\cdot)$ respectively.

The process of learning $f(\cdot)$ remains the same as in the ANM and LiNGAM cases, as outlined in the previous sections on loss terms. However, learning $g^{-1}(\cdot)$ represents a unique step specific to the PNL case. In practice, these functions $g^{-1}(\cdot)$ and $f(\cdot)$ are implemented using two separate neural networks, where $f(\cdot)$ follows the same approach as before, and $g^{-1}(\cdot)$ is modeled as the inverse of a general MLP. Moreover, the training procedure incorporates an additional Mean Squared Error (MSE) term, which the author defines as follows:

$$\mathcal{L}_{\text{PNL}}(\hat{\mathbf{X}}, \tilde{\mathbf{X}}) = MSE(\hat{\mathbf{X}}, \tilde{\mathbf{X}}) = \frac{1}{n}\sum_{i=1}^{n}\sum_{j=1}^{d} ||g_j^{-1}(X_j)_i - f_j(Pa_j)_i||_2, \tag{5.7}$$

where $\hat{\mathbf{X}}$ is the output of $g^{-1}$.

It is important to highlight that the loss terms from the previous sections, where $f(.)$ serves as the final output of the model, can also be applied in the PNL case due to the use of skip connections, similar to those in ResNet. Although $f(\cdot)$ is not the final output in the PNL setting, DAGAF can bypass the final function $g(\cdot)$, effectively allowing the same loss terms to be used as in the ANM and LiNGAM cases. For further details on skip connections, refer to [255].

#### Causal structure acyclicity

Finding optimal values for the reconstruction and adversarial loss terms does not ensure that $\mathbf{G_A}$ will be acyclic. Additionally, explicit acyclicity constraints, such as those used in [12] and [48], fail due to the implicit nature of the contents in $\mathbf{G_A}$. This means that to prevent cycles in the learned causal structures, the author applies the implicit acyclicity constraint from [38], defined as $h(\mathbf{A}^{L_0}(f)) = 0$, where $\mathbf{A}^{L_0} \in \mathbb{R}^{d \times d}$ represents the weighted adjacency matrix implicitly defined by the model weights. Further details are available in Section 2.4.3.

### 5.1.2 Causal identifiability

Discovering $\mathbf{G_A}$ from $\mathbf{X}$ does not necessarily guarantee that its content accurately represents the causal mechanisms underlying the observational data. Theory suggests that a qualitative approximation of $\mathcal{F} = \{f_1, \ldots, f_d\}$ depends on whether it is determined to be a unique set of structural equations capable of producing samples that closely resemble $\mathbf{X}$ [256]. Considering this, the author assumes identifiable causal models (refer to Definition 1) and shows that the generative process of $\mathbf{X}$ can be replicated through end-to-end optimization.

More specifically, the author demonstrates that, when identifiable models are applied, the global minimum of the distance $||P(\mathbf{X}) - P_{\mathbf{G_A}}(\tilde{\mathbf{X}})||$ can only be achieved if the true causal structure is correctly identified, leading to $P(\mathbf{X}) = P_{\mathbf{G_A}}(\tilde{\mathbf{X}})$. For further details, see Section A.4.

As previously mentioned, DAGAF applies several types of models, including: Linear non-Gaussian Acyclic Models (LiNGAM), Additive Noise Models (ANM), and Post-

Nonlinear Models (PNL). It has been demonstrated that each of these models is causally identifiable under certain assumptions.

- **LiNGAM**: The causal identifiability of Linear non-Gaussian Acyclic Models (LiNGAM) is assured when the noise terms are assumed to be non-Gaussian. In particular, if the noise variables are non-Gaussian and independent of $X$, it has been proven that the underlying causal structure can be uniquely identified [31].
- **ANM**: Additive Noise Models (ANM) assume that the noise term $\mathcal{Z}_j$ is independent of the parent variables $Pa_j$. This assumption of independence allows for the identification of the causal direction between variables. Furthermore, the function $f_j(\cdot)$ must be non-linear and three times differentiable to guarantee that applying this model leads to a unique identification of the causal direction between variables [98].
- **PNL**: Post-Nonlinear Models (PNL) build upon the ANM framework by adding an additional non-linear transformation, $g_j(\cdot)$, after the function $f_j(\cdot)$. The key assumptions for identifiability in PNL include the independence of the noise terms and the non-linear, invertible nature of the function $g_j(\cdot)$. With these conditions in place, the causal structure can be identified, even when complex non-linear interactions are present [32].

Furthermore, the theoretical analysis supporting the DAGAF framework demonstrates that, under the assumptions of LiNGAM, ANM, and PNL, the learnable DAG model $\mathbf{G_A}$ is identifiable.

**Proposition 5.1.5.** Assuming the Additive Noise Model (ANM), Linear non-Gaussian Acyclic Model (LiNGAM), or Post-Nonlinear Model (PNL), there is a unique DAG $\mathbf{G_A^0}$ that defines the observed joint distribution $P(\mathbf{X})$.

*Proof.* The proof of Proposition 5.1.5 is available in Appendix A.9. □

**Corollary 5.1.5.1.** According to Proposition 4.1.2 and Lemmas A.9.1, A.9.2 that constitute the proof of Proposition 5.1.5, the uniqueness property of $\mathbf{G_A}$ enables the author to reconstruct the generative process of $\mathbf{X}$.

Corollary 5.1.5.1 indicates that, given the causal model assumptions applied in DAGAF, the author can generate synthetic data samples that maintain the original causal structures, which is only achievable if $\mathbf{G_A} = \mathbf{G_A^0}$. Therefore, this means that the implicitly generated distribution $P_{\mathbf{G_A}}(\tilde{\mathbf{X}})$ matches the observed distribution $P(\mathbf{X})$. Thus, it has been established that a single unique DAG can accurately represent the probability distribution of both the input and the synthetic data.

It should be noted that the analysis assumes the data is continuous and follows an independently and identically distributed (i.i.d.) pattern. The author acknowledges this as a limitation since such conditions are rarely encountered in real-world datasets. Therefore, the author examines the performance of the loss terms used to train DAGAF under more challenging scenarios, including cases with non-i.i.d. data, missing values, and discrete variables.

**Impact of Non-i.i.d. Conditions on Causal Identifiability**

Consider a scenario with real-world data, where the samples $\{\mathbf{X}_1, ..., \mathbf{X}_n\}$ are not independent (i.e., there exist correlations between $\mathbf{X}_i$ and $\mathbf{X}_j$) and each sample belongs to a different heterogeneous distribution $P_i(\mathbf{X})$. In such a context, the empirical distribution $P'(\mathbf{X})$ does not accurately represent the true distribution $P(\mathbf{X})$, which in turn affects the optimization process.

More specifically, the author assumes that both the true distribution and the implicitly generated distribution can be expressed as $P'(\mathbf{X}) = P(\mathbf{X}) + \delta(\mathbf{X})$ and $P'_{\mathbf{G_A}}(\tilde{\mathbf{X}}) = P_{\mathbf{G_A}}(\tilde{\mathbf{X}}) + \delta(\tilde{\mathbf{X}})$, where the terms $\delta(\mathbf{X})$ and $\delta(\tilde{\mathbf{X}})$ reflect deviations from the i.i.d. assumptions. Under these conditions, the author investigates whether the identifiability associated with the loss terms applied in the objective function of DAGAF holds or breaks down.

**1) Adversarial Loss and Identifiability:** When the data are not i.i.d., the adversarial loss becomes:

$$\mathcal{L}'_{\text{adv}}(\mathbf{X}, \tilde{\mathbf{X}}) = D(P'(\mathbf{X}) || P_{\mathbf{G_A}}(\tilde{\mathbf{X}})).$$

For $\mathbf{G_A}$ to remain identifiable, $\delta(\mathbf{X})$ must not interfere with the optimization of $\mathcal{L}_{\text{adv}}(\mathbf{X}, \tilde{\mathbf{X}}, \mathbf{G_A})$. However, if $\delta(\mathbf{X})$ introduces spurious dependencies between variables (e.g., time-series correlations), then $\mathbf{G_A}$ may include additional edges. Moreover, if $\delta(\mathbf{X})$ skews the marginal distributions $P'(\mathbf{X}_i)$, the inferred functional relationships $\tilde{X}_j = f_j(Pa_j) + \mathcal{Z}_j$ or $\tilde{X}_j = g_j(f_j(Pa_j) + \mathcal{Z}_j)$ may no longer match the true ones. Therefore, in the non-i.i.d. data case, the learned graph $\mathbf{G_A}$ is minimizing $D(P'(\mathbf{X})||P_{\mathbf{G_A}}(\tilde{\mathbf{X}}))$, which may differ from the true graph $\mathbf{G_A^0}$, due to the bias $\delta(\mathbf{X})$.

In the case described above, the bias term $\delta(\mathbf{X})$ impacts the gradient of this loss, which is defined as follows:

$$\nabla_\phi \mathcal{L}'_{\text{adv}}(\mathbf{X}, \tilde{\mathbf{X}}) = \nabla_\phi D(P(\mathbf{X})||P_{\mathbf{G_A}}(\tilde{\mathbf{X}})) + \nabla_\phi D(\delta(\mathbf{X})||P_{\mathbf{G_A}}(\tilde{\mathbf{X}})).$$

The additional term, $\nabla_\phi D(\delta(\mathbf{X})||P_{\mathbf{G_A}}(\tilde{\mathbf{X}}))$, may destabilize optimization by introducing unintended gradient components due to data dependencies or heterogeneity, as well as by amplifying the sensitivity to noise.

Therefore, the violation of i.i.d. assumptions introduces a bias $\delta(\mathbf{X})$ in the empirical distribution $P'(\mathbf{X})$, which impacts the identifiability of $\mathbf{G_A^0}$ through the adversarial loss. This can lead to spurious dependencies, overfitting $\mathbf{G_A}$ to noise or correlations, leading to averaging out domain-specific causal structures, which can reduce the uniqueness of $\mathbf{G_A}$.

**2) Mean Squared Error Loss and Identifiability:** Under non-i.i.d. conditions, the mean squared error loss is modified as follows:

$$\mathcal{L}'_{\text{MSE}}(\mathbf{X}, \tilde{\mathbf{X}}) = \mathcal{L}_{\text{MSE}}(\mathbf{X}, \tilde{\mathbf{X}}) + \delta(\mathbf{X}).$$

If $\delta(\mathbf{X})$ induces correlations between samples $\mathbf{X}_i$ and $\mathbf{X}_j$, this disrupts the assumption that the noise terms $\mathcal{Z}_j$ are independent. Therefore, the altered MSE loss term $\mathcal{L}'_{\text{MSE}}(\mathbf{X}, \tilde{\mathbf{X}})$ might erroneously capture false patterns across samples, leading the output of $f_j(Pa_j)$ to fail in representing the true functional relationship.

Additionally, heterogeneous distributions $P_i(\mathbf{X})$ imply that $X_j$ and $Pa_j$ may follow varying conditional relationships. This causes $\mathcal{L}'_{\text{MSE}}(\mathbf{X}, \tilde{\mathbf{X}}, \mathbf{G_A})$ to average over

different functional relationships $f_j(.)$, losing the specificity of $f_j(.)$. This can have ramifications in the form of the learned graph $\mathbf{G_A}$ failing to reflect the true causal structure $\mathbf{G_A^0}$, as the functional forms are no longer consistent across samples.

Furthermore, the above statement is also supported by investigating the gradient of $\mathcal{L}'_{\text{MSE}}(\mathbf{X}, \tilde{\mathbf{X}})$ with respect to $\theta$ is:

$$\nabla_\theta \mathcal{L}'_{\text{MSE}}(\mathbf{X}, \tilde{\mathbf{X}}) = \nabla_\theta \mathcal{L}_{\text{MSE}}(\mathbf{X}, \tilde{\mathbf{X}}) + \nabla_\theta \delta(\mathbf{X}).$$

The additional term $\nabla_\theta \delta(\mathbf{X})$ destabilizes optimization by introducing misleading gradients caused by sample dependencies and noise arising from heterogeneity. As a result, the optimization process becomes more sensitive to initialization and hyperparameter choices, ultimately reducing the reliability of convergence.

**3) Kullback-Leibler Divergence Loss and Identifiability:** Under non-i.i.d. conditions, the author defines the empirical KLD subject to the application of a first-order Taylor expansion $P(\mathbf{X}_i)$ using the following expression:

$$\mathcal{L}'_{\text{KLD}}(\mathbf{X}, \tilde{\mathbf{X}}) = \frac{1}{n} \sum_{i=1}^{n} \log \frac{P_{\mathbf{G_A}}(\tilde{\mathbf{X}}_i)}{P'(\mathbf{X}_i)} \implies$$
$$\mathcal{L}'_{\text{KLD}}(\mathbf{X}, \tilde{\mathbf{X}}) \approx \mathcal{L}_{\text{KLD}}(\mathbf{X}, \tilde{\mathbf{X}}) - \frac{1}{n} \sum_{i=1}^{n} \frac{\delta(\mathbf{X}_i)}{P(\mathbf{X}_i)}$$

The ratio $\frac{\delta(\mathbf{X}_i)}{P(\mathbf{X}_i)}$ introduces bias, especially when $\delta(\mathbf{X}_i)$ varies considerably among samples. This bias distorts the optimization of $P_{\mathbf{G_A}}(\tilde{\mathbf{X}})$, potentially resulting in an approximate distribution that diverges from $P(\mathbf{X})$.

Further evidence in support of the above is reflected in the gradient for the KLD loss term, defined as follows:

$$\nabla_\theta \mathcal{L}'_{\text{KLD}}(\mathbf{X}, \tilde{\mathbf{X}}) \approx \nabla_\theta \mathcal{L}_{\text{KLD}}(\mathbf{X}, \tilde{\mathbf{X}}) - \int \nabla_\theta P_{\mathbf{G_A}}(\tilde{\mathbf{X}}) \frac{\delta(\mathbf{X})}{P(\mathbf{X})} d\mathbf{X}.$$

The additional term $\int \nabla_\theta P_{\mathbf{G_A}}(\tilde{\mathbf{X}}) \frac{\delta(\mathbf{X})}{P(\mathbf{X})} d\mathbf{X}$ introduces noise into the gradients, decreasing optimization stability. This can create misleading directions in the parameter space,

making it more difficult to converge to the true distribution $P(\mathbf{X})$.

More specifically, dependence among samples causes correlated gradients, leading to oscillations or poor convergence during training. Heterogeneity in distributions results in gradients that do not align with the true target distribution, further destabilizing the learning process. Therefore, the KLD term is minimized when $P_{\mathbf{G_A}}(\tilde{\mathbf{X}}) = P(\mathbf{X})$ under i.i.d. assumptions. Non-i.i.d. effects, however, can lead to multiple minima or local optima, reducing the identifiability of $P(\mathbf{X})$.

**4) Maximum Mean Discrepancy Loss and Identifiability:** Due to perturbations introduced by the non-i.i.d. term $\Delta$, the author defines the empirical MMD term as:

$$\begin{aligned}\mathcal{L}'_{\text{MMD}}(\mathbf{X}, \tilde{\mathbf{X}}) \approx \mathcal{L}_{\text{MMD}}(\mathbf{X}, \tilde{\mathbf{X}}) + \Delta \quad \text{s.t} \\ \Delta = \frac{1}{n}\sum_{i\neq j}^{n} \Delta_{P(\mathbf{X})}(\mathbf{X}_i, \mathbf{X}_j) \\ - \frac{2}{n}\sum_{i\neq j}^{n} \Delta_{P(\mathbf{X}), P_{\mathbf{G_A}}(\tilde{\mathbf{X}})}(\mathbf{X}_i, \tilde{\mathbf{X}}_j) \\ + \frac{1}{n}\sum_{i\neq j}^{n} \Delta_{P_{\mathbf{G_A}}(\tilde{\mathbf{X}})}(\tilde{\mathbf{X}}_i, \tilde{\mathbf{X}}_j),\end{aligned}$$

where expanding each kernel function $k(.)$ associated with the loss term yields a set of equations.

$$\begin{aligned}k(\mathbf{X}_i, \mathbf{X}_j) &= k(P(\mathbf{X}_i), P(\mathbf{X}_j)) + \Delta_{P(\mathbf{X})}(\mathbf{X}_i, \mathbf{X}_j), \\ k(\mathbf{X}_i, \tilde{\mathbf{X}}_j) &= k(P(\mathbf{X}_i), P_{\mathbf{G_A}}(\tilde{\mathbf{X}}_j)) + \Delta_{P(\mathbf{X}), P_{\mathbf{G_A}}(\tilde{\mathbf{X}})}(\mathbf{X}_i, \tilde{\mathbf{X}}_j), \\ k(\tilde{\mathbf{X}}_i, \tilde{\mathbf{X}}_j) &= k(P_{\mathbf{G_A}}(\tilde{\mathbf{X}}_i), P_{\mathbf{G_A}}(\tilde{\mathbf{X}}_j) + \Delta_{P_{\mathbf{G_A}}(\tilde{\mathbf{X}})}(\tilde{\mathbf{X}}_i, \tilde{\mathbf{X}}_j),\end{aligned}$$

The terms $\Delta_{P(\mathbf{X})}(\mathbf{X}_i, \mathbf{X}_j)$, $\Delta_{P(\mathbf{X}), P_{\mathbf{G_A}}(\tilde{\mathbf{X}})}(\mathbf{X}_i, \tilde{\mathbf{X}}_j)$ and $\Delta_{P_{\mathbf{G_A}}(\tilde{\mathbf{X}})}(\tilde{\mathbf{X}}_i, \tilde{\mathbf{X}}_j)$ represent perturbations due to non-i.i.d. effects. Due to the inclusion of the $\Delta$ term, the empirical MMD estimate becomes biased, and as a result, it might not converge to the true population MMD even when the sample size $n$ goes to infinity.

This is also theoretically implied in the gradient of $\mathcal{L}'_{\text{MMD}}(\mathbf{X}, \tilde{\mathbf{X}})$ with respect to model parameters $\theta$, which is defined as follows:

$$\nabla_\theta \mathcal{L}'_{\text{MMD}}(\mathbf{X}, \tilde{\mathbf{X}}) = 2\Big(\mathbb{E}_{\mathbf{X},\mathbf{X}'\sim P'(\mathbf{X})}[\nabla_\theta k(\mathbf{X}, \mathbf{X}')] - \mathbb{E}_{\mathbf{X}\sim P'(\mathbf{X}),\tilde{\mathbf{X}}\sim P'_{\mathbf{G_A}}(\tilde{\mathbf{X}})}[\nabla_\theta k(\mathbf{X}, \tilde{\mathbf{X}})]\Big).$$

The extra perturbations $\Delta_{P(\mathbf{X})}$, $\Delta_{P(\mathbf{X}),P_{\mathbf{G_A}}(\tilde{\mathbf{X}})}$, and $\Delta_{P_{\mathbf{G_A}}(\tilde{\mathbf{X}})}$ add noise to the gradient computations, which may destabilize the optimization process and hinder convergence.

Therefore, under i.i.d. assumptions, minimizing MMD ensures $P(\mathbf{X}) = P_{\mathbf{G_A}}(\tilde{\mathbf{X}})$. However, under non-i.i.d. conditions, the perturbed kernel computations may lead to local optima where $P'(\mathbf{X}) \neq P'_{\mathbf{G_A}}(\tilde{\mathbf{X}})$.

### DAG identifiability in Discrete Variables

Even though models such as Additive Noise Models (ANM), LiNGAM, and Post-Nonlinear Models (PNL) are identifiable in continuous settings, their DAG are not necessarily unique in discrete settings due to *symmetry* and *observational equivalence* between different causal structures in the discrete domain.

In the discrete framework, different DAG can lead to identical joint distributions, making it challenging to uniquely pinpoint the true DAG $\mathbf{G_A^0}$. For example, consider two structurally distinct DAG, $\mathbf{G_1}^{\mathbf{0}}_{\mathbf{A_1}}$ and $\mathbf{G_2}^{\mathbf{0}}_{\mathbf{A_2}}$, which nevertheless yield the *same* joint distribution. In such *discrete settings*, the symmetry inherent in causal relationships means that actions such as reversing the direction of edges or reparameterizing certain dependencies do not alter the resulting joint distribution. More formally, this can be expressed as:

$$\begin{aligned} P(X_i \mid \text{Pa}(X_i)) = P(X_j \mid \text{Pa}(X_j)) \quad &\text{for some } (X_i, X_j) \\ \text{s.t} \qquad X_j \in \text{Pa}(X_i) \text{ or } &X_i \in \text{Pa}(X_j). \end{aligned}$$

If the functional forms $f_j$ and $f_k$ are linear or have similar forms (e.g., $f_j = W_j X_k + B_j$), the reparameterization of the weights (e.g., $W_j$) or the reversal of causal edges (e.g., from $X_j \to X_k$ to $X_k \to X_j$) may result in the same *conditional distributions* $P(X_j|X_k)$. Therefore, for DAG $\mathbf{G_1}^{\mathbf{0}}_{\mathbf{A_1}}$ and $\mathbf{G_2}^{\mathbf{0}}_{\mathbf{A_2}}$, the following holds:

$$P(X_1|X_2) = P(X_2|X_1) \quad \text{for certain values of} \quad X_1, X_2$$

This symmetry ensures that the conditional distributions in both DAG are identical. Consequently, in the discrete setting, the *identifiability of the DAG* is compromised because the conditional distributions remain *equivalent*, despite differences in the underlying structural graph.

### Impact of Missing Data

Real-world datasets often contain significant missing data, which can affect the identifiability (uniqueness) of the causal structure under the Post-Nonlinear (PNL) model or other causal discovery frameworks. In the remainder of this section, the author examines the effects of missing data on DAG identifiability.

Missing data in a real-world dataset may be caused by different mechanisms, including: **1) Missing Completely at Random:** If the probability of missingness is unrelated to any variable in the dataset, it simply reduces the sample size. Identifiability may still hold with sufficient remaining data. However, smaller sample sizes weaken statistical patterns. **2) Missing at Random:** If the probability of missingness depends only on observed variables, biases may be introduced into conditional independence tests and noise independence checks. DAG discovery remains theoretically identifiable if robust imputation is used. Practical performance may still suffer due to bias. **3) Missing Not at Random:** This the most problematic type of missingness. It depends on unobserved or missing variables and, therefore, the dataset is no longer representative of the true causal structure. Identifiability often fails because dependencies in the observed data may not reflect the true DAG. Additionally, hidden biases introduced by missingness can also create spurious relationships.

The uniqueness of the true DAG $\mathbf{G_A^0}$ critically depends on accurately testing conditional independence (for instance, verifying that $\mathcal{Z}_j \perp\!\!\!\perp Pa_j$ in the PNL model), missing data undermines the statistical strength of these tests. Losing significant portions of data can lead to conditional independence tests that are unreliable or incorrect. Additionally, imputation methods or biased sampling might introduce false dependencies or independencies. Since models like ANM, LiNGAM, and PNL assume that the noise term $\mathcal{Z}_j$ is independent of the set of parent variables ($\mathcal{Z}_j \perp\!\!\!\perp Pa_j$), missing data can

obscure or distort the observed relationships, making it challenging to distinguish noise from the contributions of the model.

Furthermore, it is assumed that the functional forms $f_j$ (nonlinear for ANM and linear for LiNGAM) and $g_j$ (nonlinear for PNL) are either known or can be learned. However, the incomplete nature of real-world data often breaks this assumption. Specifically, missing data can bias the noise estimates $\mathcal{Z}_j$, disrupting the independence of residuals. In the case of LiNGAM, this makes testing for non-Gaussian noise even more challenging.

The ability to identify the correct model depends on accurately estimating the marginal distributions. When data is incomplete, especially, if parent variables or structural nodes are missing more frequently, these estimates can be significantly distorted.

### 5.1.3 Simulating data generative processes

In the second stage of Algorithm 2, the focus shifts to generating realistic tabular data samples using the causal graph $\mathbf{G_A}$ obtained from Step 1. This data generation process relies on a separate instance of the SCM $\mathcal{M}$ used during the causal discovery phase, referred to here as the generator $G$. Causal knowledge is transferred between SCM instances by loading $W^{L_0}$ from $\mathcal{M}$ to $L_0 \in G$. To facilitate tabular data synthesis, the architecture of the generator is augmented with an additional noise vector $\mathbf{Z} = \{\mathbf{Z}_1, \ldots, \mathbf{Z}_d\}$, sampled from $\mathcal{N}(0, 1)$.

In this step, the models are trained adversarially to ensure that $P_{\mathbf{G_A}}(\tilde{\mathbf{X}})$ closely matches $P(\mathbf{X})$. Specifically, the generator network $G$ produces synthetic samples while competing with a discriminator $D : \mathbb{R}^d \to \mathbb{R}$, which aims to distinguish between synthetic and real observational samples. The training process leverages the Wasserstein-1 distance with a gradient penalty, enabling the Deep Generative Model (DGM) to produce realistic samples that are indistinguishable from $\mathbf{X}$. The loss function used is the same as Equation (5.1).

More precisely, the transferred weights $W^{L_0}$ form $\mathbf{A} \in L_0 \in G$, which is then subsequently thresholded to form a binary mask $M$ that specifies parent–child relationships.

This mask is then used to define a set of dynamically instantiated locally connected layers, where each node-specific layer $f_j(.)$ receives only its parent variables $Pa_j$ as inputs to define $\tilde{X}_j = f_j(Pa_j, \mathcal{Z}_j), Pa_j = \{i : M_{ij} = 1\}$. This guarantees the global causal structure inferred from $\mathbf{A}$ is transferred into the set of locally connected layers, while making sure that each layer models only causally relevant dependencies

Furthermore, each connected layer $\alpha(L_j) \in \{\alpha(L_1), \ldots, \alpha(L_d)\}$ is treated as an individual generator $G_j(\mathbf{Z}_j) \in \{G_1(\mathbf{Z}_1), \ldots, G_d(\mathbf{Z}_d)\}$. This allows each causal mechanism $f_j \in \{f_1, \ldots, f_d\}$ to be modeled such that $\tilde{X}_j$ is generated in one of three forms: $\tilde{X} := G(X) + \mathcal{Z}$, $\tilde{X}_j := G_j(Pa_j) + \mathbf{Z}_j$, or $\tilde{X}_j := g_j(G_j(Pa_j) + \mathbf{Z}_j)$, depending on the assumed model - LiNGAM, ANM, or PNL, respectively. In this way, a synthetic tabular dataset $\tilde{\mathbf{X}} \in \tilde{\chi} \subseteq \mathbb{R}^{n \times d} = \mathcal{F}(\mathbf{Z}) = f_j(Pa_j, \mathbf{Z}_j)$ is generated.

During training, only the parameters $W = \{W^1, \ldots, W^L\}$ of the locally connected hidden layers are updated. The weights of $L_0$ are not modified to preserve the structural equations $\mathcal{F}$ used to produce $\tilde{\mathbf{X}}$.

The experiments described in Section 5.2.4 indicate that the DGM employed by the author can produce high-quality data under both the ANM and PNL structural assumptions.

### 5.1.4 Model architecture and training specifications

Figure 5.2 illustrates the overall architecture of the DAGAF framework. In Figure 5.2a, the ANM and LiNGAM settings are depicted, where the input data $\mathbf{X}$ is transformed by function $f$ to generate $\tilde{\mathbf{X}}$. The optimization process is governed by multiple loss terms: $\mathcal{L}_{\text{adv}}(\mathbf{X}, \tilde{\mathbf{X}})$, $\mathcal{L}_{\text{MSE}}(\mathbf{X}, \tilde{\mathbf{X}})$, $\mathcal{L}_{\text{KLD}}(\mathbf{X}, \tilde{\mathbf{X}})$, and $\mathcal{L}_{\text{MMD}}(\mathbf{X}, \tilde{\mathbf{X}})$, with $\mathcal{L}_{\text{KLD}}(\mathbf{X}, \tilde{\mathbf{X}})$ being omitted in the LiNGAM case. Figure 5.2b expands upon Figure 5.2a by integrating the PNL model. The right-hand pathway remains consistent with Figure 5.2a, while an additional left-hand pathway applies $g^{-1}$ to invert $\mathbf{X}$. This inversion is involved in computing $\mathcal{L}_{\text{PNL}}(\hat{\mathbf{X}}, \tilde{\mathbf{X}})$, which is then combined with the loss terms from the right-hand pathway, forming a unified optimization framework. Figure 5.2c illustrates the data generation process used to create synthetic data, showcasing how the framework enables structured data synthesis.

The author adopts the Multi-Layer Perceptron (MLP) from [38] as the Structural Causal Model (SCM) $\mathcal{M}$ to represent $f$ during the causal structure learning step. This MLP consists of two main components: (i) an initial linear layer, $L_0$, which implicitly defines the causal graph $\mathbf{G_A}$ and enables the modeling of causal structures, and (ii) a set of locally connected hidden layers, $L = \{\alpha(L_1), ..., \alpha(L_d)\}$, where $\alpha$ applies a nonlinear transformation to each layer. These hidden layers are designed to learn an accurate approximation of the causal mechanisms $\mathcal{F} = \{f_1, ..., f_d\}$ within $\mathbf{G_A}$.

In contrast, $g$ is a general-purpose MLP consisting of five linear layers arranged as [$d$ - $10d$ - $10d$ - $10d$ - $d$] (one input layer, three hidden layers, and one output layer), with nonlinearity applied via the ReLU activation function (used specifically in the PNL case). Figure 5.2 provides an overview of this architecture.

More specifically, each feature in $\mathbf{X}$ is modeled by a neural network with $L$ hidden layers, represented as $f_j(Pa_j, \mathcal{Z}_j; W_j^1, ..., W_j^L)$ for $j \in [1, d]$, where $W_j^l$ represents the parameters of the $l^{th}$ layer. Let $W_j^{(0)} \in \mathbb{R}^{h \times d}$ denote the weight matrix in $L_0$ connecting to the local neural network modeling $X_j$, where $h$ is the latent size and $d$ is the number of input variables. For any pair of variables $X_j$ and $X_k$, the Ridge regression norm of the weights connecting $X_k$ to all latent units in the network for $X_j$ is calcualted as: $\mathbf{A}_{jk} = \left\| W_{j,k,:}^{(1)} \right\|_2 = \sqrt{\sum_{m=1}^{h} \left( W_{j,k,m}^{(1)} \right)^2}$, where $W_{j,k,m}^{(1)}$ represents the weight connecting the $k$-th input variable $X_k$ to the $m$-th latent unit in the first layer of the network for $X_j$.

During training, a learning rate of $3 \times 10^{-3}$ is used, along with a batch size of 1000. Ridge regression regularization is incorporated in both steps by setting the weight decay for both discriminators to $1 \times 10^{-6}$. The models in DAGAF are optimized iteratively, with their parameters updated using gradient descent.

The adversarial loss is applied to the reconstructed distribution $P_{\mathbf{G_A}}(\tilde{\mathbf{X}})$, meaning that no noise vector is used during training in the causal structure learning step. Once the parameters in $\mathbf{A}^{L_0}$ are updated, $\mathbf{A}^{L_0}$ is converted into $\mathbf{G_A}$ through a post-processing step: $\mathbf{G_A} = \sqrt{\mathbf{A}^{L_0}(f)}$, where $w^2 jk \in \mathbf{A}^{L_0}(f)$, followed by thresholding at a value of 0.3, as recommended by prior works such as DAG-GNN [35], GAE [39], and others. These final steps are essential to recover the weights $w_{jk} \in \mathbf{G_A}$ from $\mathbf{A}^{L_0}(f)$ and to

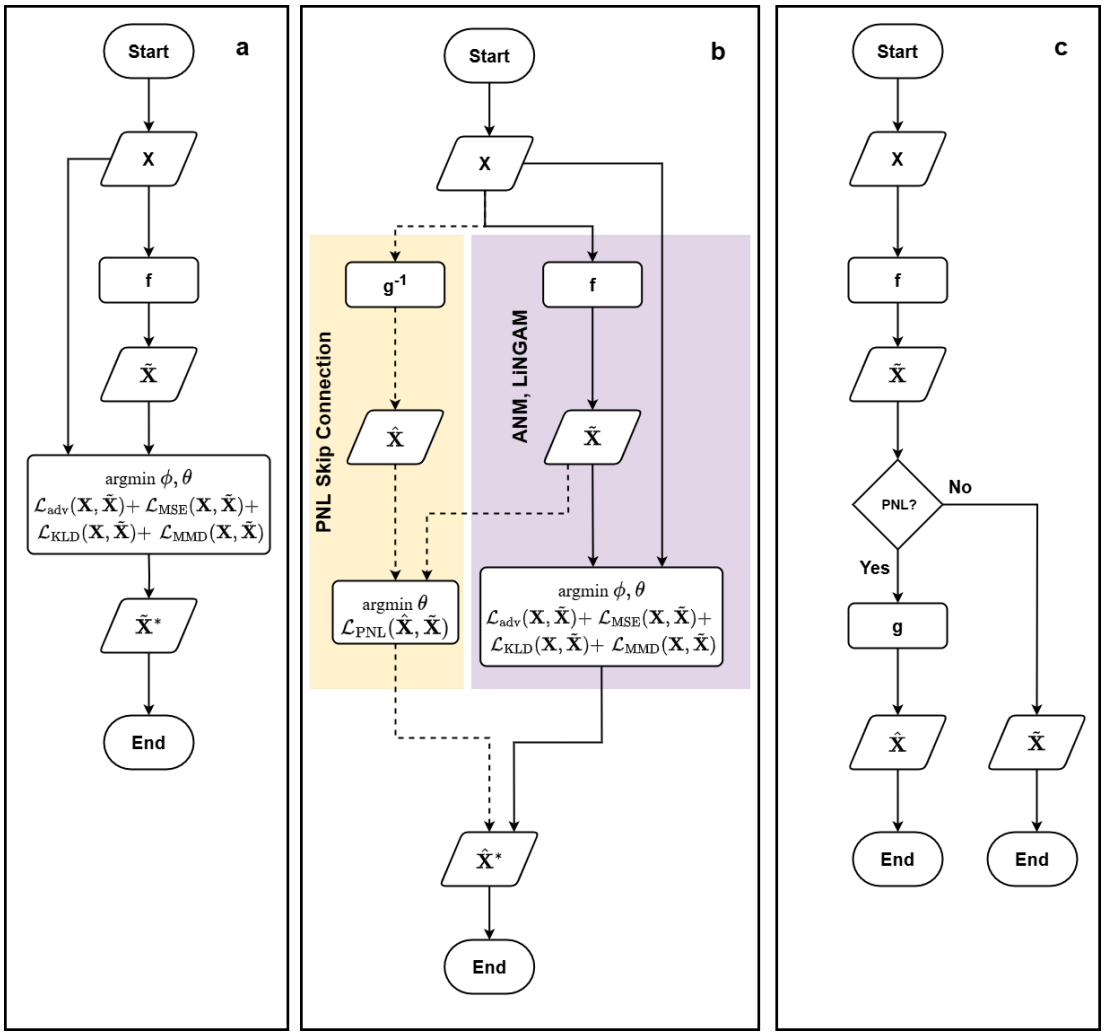



Figure 5.2: A Visual Representation of DAGAF. (a) The optimization structure under ANM and LiNGAM, where input data is processed to reconstruct $\tilde{\mathbf{X}}$ using multiple loss terms, excluding $\mathcal{L}_{\text{KLD}}$ in the LiNGAM case. (b) The extended framework integrating ANM, LiNGAM, and PNL, where an additional inversion function $g^{-1}$ is introduced to compute $\mathcal{L}_{\text{PNL}}$, unifying the optimization process. The dashed line signifies the skip connection. When PNL is not assumed the advanced form of the framework reverts back to its basic form capable of handling only ANM and LiNGAM by solely learning $f$. (c) The synthetic data generation process, illustrating how the framework enables structured data synthesis while preserving underlying causal relationships.

minimize false discoveries in $\mathbf{G_A}$.

To learn $g^{-1}$ for the PNL case, the architecture and training procedure of $g$ are reversed so that $\tilde{\mathbf{X}}$ serves as the input to reconstruct the original $\mathbf{X}$. However, since $g$ is a general model, inverting its architecture does not require any changes to its

configuration. Therefore, the focus is placed solely on the training algorithm.

**Remark.** The output data $\tilde{\mathbf{X}}$ from Step 1 is used exclusively for calculating the loss terms during training and is then discarded. This is because the reconstruction loss employed to learn the causal structure of $\mathbf{X}$ greatly restricts the range of the generated samples, producing $\tilde{\mathbf{X}}$ with high fidelity but limited diversity.

The training process is formulated as a constrained continuous optimization problem due to the need to simultaneously update the model weights and the parameters associated with the acyclicity constraint. To address this, the author adopts a modified version of the augmented Lagrangian method [47], as utilized in DAG-Notears-MLP. The complete training objective for Step 1 is defined as follows:

$$
\begin{aligned}
\mathcal{L}_{REC}(\mathbf{X}, \tilde{\mathbf{X}}) &= \underbrace{\frac{1}{n}\sum_{i=1}^{n} ||\mathbf{X}_i - \tilde{\mathbf{X}}_i||_2}_{\text{Mean Squared Error}} + \underbrace{\frac{1}{2}\sum_{i=1}^{n}(\mu_i^2)}_{\text{KL Divergence}} \\
&\quad + \underbrace{\frac{1}{n}\sum_{i\neq j}^{n} k(\mathbf{X}_i, \mathbf{X}_j) - \frac{2}{n}\sum_{i\neq j}^{n} k(\mathbf{X}_i, \tilde{\mathbf{X}}_j) + \frac{1}{n}\sum_{i\neq j}^{n} k(\tilde{\mathbf{X}}_i, \tilde{\mathbf{X}}_j)}_{\text{Maximum Mean Discrepancy}} \\
\mathcal{L}_{adv}(\mathbf{X}, \tilde{\mathbf{X}}) &= \underbrace{\mathbb{E}_{\mathbf{X}\sim P(\mathbf{X})}[D(\mathbf{X})] - \mathbb{E}_{\tilde{\mathbf{X}}\sim P_{\mathbf{G_A}}(\tilde{\mathbf{X}})}[D(\tilde{\mathbf{X}})]}_{\text{Discriminator loss}} + \underbrace{\mathbb{E}_{\hat{\mathbf{X}}\sim P(\hat{\mathbf{X}})}[(||\nabla_{\hat{\mathbf{X}}} D(\hat{\mathbf{X}})||_2 - 1)^2]}_{\text{Gradient Penalty}} \\
\mathcal{L}_G(\mathbf{X}) &= -\underbrace{\mathbb{E}_{\mathbf{X}\sim P(\mathbf{X})}[D(G(\mathbf{X}))]}_{\text{Generator loss}} \\
\mathcal{L}_{PNL}(\hat{\mathbf{X}}, \tilde{\mathbf{X}}) &= \underbrace{\frac{1}{n}\sum_{i=1}^{n} ||\hat{\mathbf{X}}_i - \tilde{\mathbf{X}}_i||_2}_{\text{PNL loss term}} \qquad \text{i.f.f the assumed SCM is PNL} \\
&\text{s.t} \quad \underbrace{h(\mathbf{A}^{L_0}(f)) = 0, [\mathbf{A}^{L_0}(j)]_{ij} := ||\partial_i f_j||_2}_{\text{Acyclicity constraint}}
\end{aligned}
\tag{5.8}
$$

### 5.1.5 Computational Complexity Analysis

The DAGAF framework consists of three individual models (FCM/Generator $\mathcal{M}/G$, Discriminator $D$ (ANM, LiNGAM setting) and an additional MLP (PNL) case) trained with an algorithm involving three interconnected components (Causal Structure Learning, Tabular Data Synthesis and Augmented Lagrangian-based Continuous Optimiza-

tion). This intricate architecture and training process make DAGAF considerably more complicated compared to other state-of-the-art methods, such as DAG-GNN [35], GraN-DAG [36], DECAF [218], and Causal-TGAN [220], which only focus on causal discovery or tabular data synthesis and involve fewer models. This complexity motivated the author to evaluate the efficiency and practicality of their approach.

They investigate how much resources DAGAF requires to perform causal structure learning and tabular data synthesis simultaneously. To achieve this, the author provides pseudo-code for Algorithm 2 and conducts a time complexity analysis on it. The alternative representation of the training process for their framework is available below.

$\lambda \leftarrow 0$, $c \leftarrow 1$
$current\_h(\mathbf{A}^{L_0}(f)) \leftarrow \infty$, $h\_tol \leftarrow 1e-8$
$k\_max\_iter \leftarrow 100$, $epochs \leftarrow 300$
**for** k $<$ $k\_max\_iter$ **do**
  **while** $c < 1e+20$ **do**
    **for** epoch $<$ $epochs$ **do**

      **if** $pnl == True$ **then** ▷ The beginning of the Causal Discovery (CD) Step
        $\tilde{X} := \{g_1(f_1(Pa_1; W_1^1, ..., W_1^L) + \mathcal{Z}_1), ..., g_d(f_d(Pa_d; W_d^1, ..., W_d^L) + \mathcal{Z}_d)\}$
      **else**
        $\tilde{X} := \{f_1(Pa_1; W_1^1, ..., W_1^L) + \mathcal{Z}_1, ..., f_d(Pa_d; W_d^1, ..., W_d^L) + \mathcal{Z}_d\}$
      **end if**

      DiscLoss = $\mathcal{L}_{adv}(\mathbf{X}, \tilde{\mathbf{X}})$
      GenLoss = $\mathcal{L}_G(\mathbf{X})$
      RecLoss = $\mathcal{L}_{REC}(\mathbf{X}, \tilde{\mathbf{X}}) + \frac{c}{2}|h(\mathbf{A}^{L_0})|^2 + \lambda h(\mathbf{A}^{L_0})$
      PnlLoss = $\mathcal{L}_{PNL}(\hat{\mathbf{X}}, \tilde{\mathbf{X}})$ ▷ if PNL is assumed

      DiscGradients = DiscLoss.backward()
      GenGradients = GenLoss.backward()
      RecGradients = RecLoss.backward()
      PnlGradients = PnlLoss.backward() ▷ if PNL is assumed

      DiscParameters = DiscParameters - $1e-3$ * DiscGradients
      GenParameters = GenParameters - $1e-3$ * GenGradients
      RecParameters = RecParameters - $1e-3$ * RecGradients
      PnlParameters = PnlParameters - $1e-3$ * PnlGradients ▷ if PNL is assumed

      $DS\{W^{L_0}\} \leftarrow CD\{W^{L_0}\}$ ▷ Parameter transfer between steps

**if** $pnl == True$ **then** ▷ The beginning of the Data Synthesis (DS) Step
$\tilde{X} := \{g_1(G_1(Pa_1; W_1^1, ..., W_1^L) + Z_1), ..., g_d(G_d(Pa_d; W_d^1, ..., W_d^L) + Z_d)\}$
**else**
$\tilde{X} := \{G_1(Pa_1; W_1^1, ..., W_1^L) + Z_1, ..., G_d(Pa_d; W_d^1, ..., W_d^L) + Z_d\}$
**end if**

DiscLoss = $\mathcal{L}_{adv}(\mathbf{X}, \tilde{\mathbf{X}})$
GenLoss = $\mathcal{L}_G(Z)$

DiscGradients = DiscLoss.backward()
GenGradients = GenLoss.backward()

DiscParameters = DiscParameters - $1e-3$ * DiscGradients
GenParameters = GenParameters - $1e-3$ * GenGradients

**end for**
**if** $h(\mathbf{A}^{L_0}(f)) > 0.25$ **then**
$c \leftarrow c * 10$
**else**
*break*
**end if**
**end while**
$current\_h(\mathbf{A}^{L_0}(f)) \leftarrow h(\mathbf{A}^{L_0}(f))$
$\lambda \leftarrow c * current\_h(\mathbf{A}^{L_0}(f))$
**if** $current\_h(\mathbf{A}^{L_0}(f)) \leq h\_tol$ **then**
*break*
**end if**
**end for**

The space complexity of DAGAF is $\mathcal{O}(d)$, where $d$ is the number of variables in $\mathbf{X}$, which is consistent with that of Notears and its extensions. For more theoretical details, the reader is referred to [12].

To conduct a comprehensive time complexity analysis on their framework, the author investigates the efficiency of each stage in Algorithm 2 individually. Additionally, they include the augmented Lagrangian and the causal knowledge transfer in their study. The total computational complexity is calculated by adding the individual complexities of each component of Algorithm 2 and deducing which is the most resource-demanding. The training procedure of DAGAF begins with an initial stage, involving declarations of variables, hyperparameters and model instances, all of which

are considered atomic operations taking constant time $\mathcal{O}(1)$.

Afterwards, the training procedure is applied by entering the augmented Lagrangian, which consists of three nested loops (1: controlled by $k\_max\_iter$, 2: constrained by the range of values for $c$ and 3: managed by the number of *epochs* in the training process). Since, in the worst case, all of them will run until their respective limiting values are reached, individually each of the loops has linear complexity. If the range for each loop is considered constant, then optimizing the augmented Lagrangian parameters relies solely on the number of data variables in the input dataset, yielding a time complexity of $\mathcal{O}(d)$, where $d$ is the number of variables in the observational data. Since there are three nested loops and parameter optimization (taking constant time $\mathcal{O}(1)$) involved in the augmented Lagrangian, its computational complexity is cubic $\mathcal{O}(d)^3$.

Within the augmented Lagrangian, the training algorithm divides into two parts: a causal structure learning stage and a tabular data synthesis stage with an additional operation to transfer the causal knowledge between steps taking constant time $\mathcal{O}(1)$. Both sections of the training procedure employ stochastic gradient decent (SGD) to perform model parameter optimization. Typically, the computational complexity of SGD is $\mathcal{O}(knd)$, where $k$ is the number of epochs, $n$ is the number of samples and $d$ is the data variable size of $\mathbf{X}$. In the case of DAGAF, both $k$ and $n$ are constant hyperparameters, which means that the complexity of the optimization technique depends only on the number of data attributes present in the input. Hence, the total computational complexity of both parts is linear $\mathcal{O}(d)$.

The time complexity of Algorithm 2 can be expressed as $\mathcal{O}(d)^3 + 2\mathcal{O}(d)$, which reduces to $\mathcal{O}(d)^3$ since researchers are only interested in the fastest growing term. The results of the analysis indicate that DAGAF exhibits a cubic computation complexity, which is an outcome consistent with findings reported in other research studies [12], [36].

## 5.2 Experimental Results

The author performs a series of experiments on their general framework for causality-based tabular data synthesis. These tests utilize various datasets comprising continuous

and discrete data types to evaluate the following factors:

- Structure learning accuracy, which evaluates how well the model captures and represents the relationships between features in observational data.
- Synthetic data quality, which analyzes the standard of the samples produced using the learned generative process.
- An ablation study and sensitivity analysis are conducted to evaluate the impact of the loss term configuration and the hyper-parameter settings on the training process - for more information, the reader is referred to Sections 5.2.6 and 5.2.7.

To assess structure learning, DAGAF is compared against several state-of-the-art Directed Acyclic Graph (DAG) learning methods, including DAG-WGAN [75], DAG-WGAN+ [77], DAG-Notears-MLP [38], Dag-Notears [12], DAG-GNN [35], GraN-DAG [36], GAE [39], CAREFL [168], DAG-NF [257], DCRL [258] and VI-DP-DAG [49]. The quality of the discovered causality is assessed using the Structural Hamming Distance (SHD) [105] as the primary metric across all experiments. However, it is important to note that SHD is not the only metric for evaluating the accuracy of learned structures. Alternative measures, such as Area Under Curve (AUC) and Area Over Curve (AOC), can also be applied.

The author further assesses the quality of the synthetic data produced by DAGAF by conducting several tests to analyze the statistical properties of $\tilde{\mathbf{X}}$. To compare $P(\mathbf{X})$ with $P_{\mathbf{G_A}}(\tilde{\mathbf{X}})$, they utilize boxplot analysis and examine marginal distributions. Furthermore, correlation matrices are calculated for both $\chi$ and $\tilde{\chi}$ to evaluate the relationships among their covariates.

### 5.2.1 Continuous data

The author conducts experiments on continuous data types using simulated datasets derived from predefined structural equations and Directed Acyclic Graph (DAG) structures. To achieve this, the author generates an Erdos-Renyi [243] causal graph with an expected node degree of 3, which is used as the ground-truth DAG $\mathbf{G_A^0}$ and can be

described by a weighted adjacency matrix $\mathbf{A}$. Each experiment involves 5,000 observational data samples generated using various equations applied to the causal mechanisms in $\mathbf{G_A^0}$, including linear ($\tilde{\mathbf{X}} := \mathbf{A}^T X + \mathcal{Z}$), non-linear-1 ($\tilde{\mathbf{X}} := \mathbf{A}\cos(X+1) + \mathcal{Z}$), non-linear-2 ($\tilde{\mathbf{X}} := 2\sin(\mathbf{A}(X+0.5)) + \mathbf{A}(X+0.5) + \mathcal{Z}$), post-non-linear-1 ($\tilde{\mathbf{X}} := \sinh(\mathbf{A}\cos(X+1) + \mathcal{Z})$), and post-non-linear-2 ($\tilde{\mathbf{X}} := \tanh(2\sin(\mathbf{A}(X+0.5)) + \mathbf{A}(X+0.5) + \mathcal{Z})$). These structural equations have been extensively utilized in various studies on DAG learning, including models such as, DAG-GNN [35], Gran-DAG [36], GAE [39], DAG-WGAN [75], DAG-WGAN+ [77] and Notears-MLP [38], among others. Their widespread use enables a thorough and reliable comparison with other state-of-the-art models in the field. The last two equations are modified versions of the second and third equations, specifically designed to serve as appropriate test cases for experiments related to the PNL assumption. It is also important to note that the list of equations used in the experiments is by no means exhaustive, other equations can also be applied.

The approach of the author aligns with the methodology used in most state-of-the-art DAG learning models, including DAG-GNN, GraN-DAG, DAG-Notears and GAE. To assess the scalability of DAGAF, tests are performed on datasets with 10, 20, 50, and 100 columns. Each experiment is repeated five times to account for sample randomness and the average Structural Hamming Distance (SHD) is recorded. The findings are summarized in Tables 5.1, 5.2, 5.3, 5.4, and 5.5.

Table 5.1: Non-parametric DAG structures recovered from linear data samples

| Model | SHD (5000 linear samples) | | | |
|---|---|---|---|---|
| | d=10 | d=20 | d=50 | d=100 |
| DAG-Notears | 8.6 ± 7.2 | 13.8 ± 9.6 | 41.8 ± 29.4 | 102.8 ± 53.2 |
| DAG-Notears-MLP | 4.6 ± 4.3 | 7.6 ± 6.3 | 29.6 ± 18.5 | 74 ± 30.6 |
| DAG-GNN | 6 ± 6.9 | 11.4 ± 8.2 | 33.6 ± 21.2 | 85.4 ± 46.4 |
| GAE | 5.5 ± 4.9 | 10.3 ± 7.2 | 31.3 ± 13.8 | 80.2 ± 24.6 |
| GraN-DAG | 3.4 ± 5.2 | 6.4 ± 7.5 | 25.2 ± 14.6 | 68.4 ± 25.8 |
| CAREFL | 2.7 ± 4.8 | 5.9 ± 7.1 | 24.9 ± 14.1 | 66.9 ± 24.7 |
| DAG-NF | 2.4 ± 4.6 | 5.2 ± 6.9 | 23.1 ± 13.4 | 64.2 ± 24.3 |
| VI-DP-DAG | 2.1 ± 4.5 | 4.5 ± 6.7 | 22.4 ± 12.7 | 63.7 ± 23.5 |
| DCRL | 1.8 ± 2.7 | 3.1 ± 4.8 | 18.7 ± 11.9 | 53.3 ± 21.9 |
| DAG-WGAN | 5.2 ± 3.8 | 9.2 ± 5.7 | 19.6 ± 12.3 | 58.6 ± 22.7 |
| DAG-WGAN+ | 3.7 ± 3.1 | 5.6 ± 4.9 | 17.2 ± 10.5 | 49.1 ± 20.1 |
| DAGAF | **1.4 ± 2.3** | **2 ± 4.4** | **16.4 ± 9.8** | **38.8 ± 18.3** |

Table 5.2: Non-parametric DAG structures recovered from non-linear-1 data samples

| Model | SHD (5000 non-linear-1 samples) | | | |
|---|---|---|---|---|
| | d=10 | d=20 | d=50 | d=100 |
| DAG-Notears | 11.4 ± 4.5 | 28.2 ± 10.2 | 55 ± 23.1 | 105.6 ± 48.3 |
| DAG-Notears-MLP | 5.2 ± 1.8 | 15.4 ± 4.6 | 43.8 ± 15.4 | 86.2 ± 29.8 |
| DAG-GNN | 9.2 ± 3.3 | 23.4 ± 8.4 | 50.2 ± 19.5 | 98.6 ± 37.6 |
| GAE | 8.6 ± 2.2 | 20 ± 5.7 | 47.5 ± 10.2 | 92.3 ± 18.9 |
| GraN-DAG | 4 ± 2.4 | 11.2 ± 6.5 | 36.4 ± 11.9 | 72.8 ± 21.7 |
| CAREFL | 3.8 ± 2.2 | 10.9 ± 6.2 | 34.1 ± 11.2 | 71.7 ± 19.1 |
| DAG-NF | 3.4 ± 2.1 | 10.4 ± 5.6 | 31.6 ± 10.7 | 69.5 ± 17.3 |
| VI-DP-DAG | 3.1 ± 2 | 9.8 ± 5.1 | 28.7 ± 9.3 | 68.1 ± 16.5 |
| DCRL | 2.9 ± 1.7 | 7.5 ± 4 | 24.3 ± 7.8 | 61.4 ± 14.9 |
| DAG-WGAN | 6.4 ± 1.4 | 18.6 ± 3.7 | 22 ± 8.6 | 64.6 ± 15.2 |
| DAG-WGAN+ | 4.9 ± 1.2 | 14.2 ± 3.3 | 20.5 ± 6.9 | 57.1 ± 14.5 |
| DAGAF | **2.6 ± 1** | **5.2 ± 2.8** | **18.8 ± 6.2** | **50.2 ± 13.4** |

Table 5.3: Non-parametric DAG structures recovered from non-linear-2 data samples

| Model | SHD (5000 non-linear-2 samples) | | | |
|---|---|---|---|---|
| | d=10 | d=20 | d=50 | d=100 |
| DAG-Notears | 10.4 ± 3.9 | 22.4 ± 8.1 | 47.6 ± 21.2 | 112.8 ± 57.8 |
| DAG-Notears-MLP | 5.4 ± 1.5 | 13.8 ± 4.3 | 30.4 ± 15.7 | 85.6 ± 35.6 |
| DAG-GNN | 8.4 ± 3.2 | 19.2 ± 7.7 | 36.2 ± 18.6 | 91.8 ± 49.3 |
| GAE | 7.3 ± 1.8 | 17.4 ± 5.1 | 33.7 ± 13.7 | 88.4 ± 26.6 |
| GraN-DAG | 4.2 ± 2.1 | 11.6 ± 5.6 | 25.2 ± 14.5 | 71.6 ± 29.7 |
| CAREFL | 3.8 ± 1.8 | 10.5 ± 5.3 | 24.8 ± 13.8 | 69.9 ± 26.1 |
| DAG-NF | 3.3 ± 1.7 | 9.7 ± 4.9 | 24.3 ± 13.1 | 68.1 ± 24.3 |
| VI-DP-DAG | 2.8 ± 1.6 | 9.3 ± 4.7 | 23.8 ± 13.3 | 67.3 ± 23.8 |
| DCRL | 2.2 ± 1.3 | 7.1 ± 2.9 | 15.1 ± 9.4 | 59.5 ± 17.2 |
| DAG-WGAN | 6.6 ± 1.2 | 15.2 ± 3.4 | 22.6 ± 12.9 | 64.2 ± 21.5 |
| DAG-WGAN+ | 5.1 ± 1.1 | 12.3 ± 2.5 | 17.5 ± 10.2 | 56.7 ± 18.4 |
| DAGAF | **1.4 ± 0.9** | **5.8 ± 2.2** | **14.2 ± 8.3** | **51.8 ± 16.2** |

Table 5.4: Non-parametric DAG structures recovered from post-non-linear-1 data samples

| Model | SHD (5000 post-non-linear-1 samples) | | | |
|---|---|---|---|---|
| | d=10 | d=20 | d=50 | d=100 |
| DAG-GNN | 13.7 ± 9.2 | 21.7 ± 10.4 | 63.7 ± 31.2 | 118.6 ± 50.1 |
| GAE | 12.3 ± 8.1 | 19.1 ± 8.8 | 56.2 ± 24.6 | 101.3 ± 37.4 |
| CAREFL | 11.8 ± 6.4 | 18.5 ± 7.9 | 52.1 ± 22.8 | 97.2 ± 34.9 |
| DAG-NF | 11.2 ± 5.3 | 16.2 ± 6.1 | 47.3 ± 19.5 | 92.5 ± 31.3 |
| DAG-WGAN | 10.5 ± 4.7 | 15.6 ± 5.8 | 44.5 ± 17.7 | 88.7 ± 29.6 |
| DAG-WGAN+ | 8.4 ± 3.3 | 12.8 ± 4.3 | 32.8 ± 13.6 | 66.1 ± 21.2 |
| DAGAF | **5.6 ± 2.5** | **7.3 ± 3.2** | **25.4 ± 11.3** | **52.4 ± 15.7** |

Table 5.5: Non-parametric DAG structures recovered from post-non-linear-2 data samples

| Model | SHD (5000 post-non-linear-2 samples) | | | |
|---|---|---|---|---|
| | d=10 | d=20 | d=50 | d=100 |
| DAG-GNN | 10.8 ± 8.7 | 16.1 ± 11.9 | 37.1 ± 30.3 | 128.3 ± 48.2 |
| GAE | 9.1 ± 6.3 | 14.3 ± 9.5 | 31.5 ± 24.8 | 105.7 ± 34.4 |
| CAREFL | 8.3 ± 5.8 | 13.5 ± 8.3 | 29.8 ± 22.4 | 92.1 ± 32.3 |
| DAG-NF | 7.7 ± 5.5 | 12.8 ± 7.4 | 28.4 ± 21.7 | 84.8 ± 28.5 |
| DAG-WGAN | 7.2 ± 5.2 | 11.4 ± 6.2 | 25.2 ± 18.6 | 76.5 ± 27.6 |
| DAG-WGAN+ | 4.5 ± 3.6 | 8.6 ± 5.1 | 21.7 ± 12.3 | 69.4 ± 19.1 |
| DAGAF | **2.9 ± 2.4** | **5.7 ± 3.6** | **18.6 ± 10.5** | **47.2 ± 14.7** |

### 5.2.2 Benchmark experiments

In their experiments, the author incorporated discrete datasets as part of an empirical study to evaluate how the DAGAF framework performs on such data. However, as discussed in the theoretical analysis conducted in Section 5.1.2, they acknowledge that applying this method to discrete datasets introduces identifiability challenges.

In conducting experiments with discrete data, the author utilized benchmark datasets including Child, Alarm, Hailfinder, and Pathfinder, available with their ground truths from the Bayesian Network Repository at `https://www.bnlearn.com/bnrepository`. These datasets are meticulously prepared to facilitate scalability testing and allow for a fair comparison with leading-edge techniques. The author compared their model against DAG-GNN and both versions of DAG-WGAN, with the experimental results shown in Table 5.6.

Table 5.6: Non-parametric DAG structures recovered from benchmark data samples

| Datasets | Nodes | SHD | | | |
|---|---|---|---|---|---|
| | | DAG-WGAN | DAG-WGAN+ | DAG-GNN | DAGAF |
| Child | 20 | 20 | 19 | 30 | **17** |
| Alarm | 37 | 36 | **35** | 55 | 43 |
| Hailfinder | 56 | 73 | 66 | 71 | **63** |
| Pathfinder | 109 | 196 | 194 | 218 | **181** |

### 5.2.3 Real data experiments

Up to this point, simulations based on artificial data suggest that the model can yield satisfactory outcomes. However, such findings are not entirely conclusive because the simulations do not perfectly reflect real-world scenarios. To mitigate this issue, the author conducted experiments on the acclaimed Sachs dataset [21], which is respected within the research community. This dataset comprises 7466 samples across 11 variables, with its ground-truth underlying structure presumed to contain roughly 20 connections. Additionally, DAGAF was employed with both Additive Noise Model (ANM) and Post Non-linear (PNL) assumptions to compare the Structural Hamming Distance (SHD) produced by these Structural Causal Models (SCM), deducing whether the post-nonlinear model performs better with real-world data. The findings are provided in Table 5.7.

Table 5.7: Non-parametric DAG structures from real data samples

| Model | Sachs Dataset |
|---|---|
| | SHD |
| DAG-WGAN | 17 |
| DAG-WGAN+ | 15 |
| DAG-NF | 15 |
| DAG-GNN | 25 |
| GAE | 20 |
| GraN-DAG | 17 |
| VI-DP-DAG | 16 |
| DAGAF | ANM **9** / PNL **8** |

### 5.2.4 Synthetic data quality

This study argues that the method of the author outperforms the current best models in the field of causal discovery by integrating DAG learning with synthetic data production. To support this assertion, they analyze features (d=10) drawn from two sets of simulated data based on the ANM and PNL assumptions, then compare these against features generated by their technique. The author considers the special scenario in which their model attains a SHD of 0 on the simulation data, resulting in the highest quality samples because of the comprehensive understanding of causal mechanisms

within the generative process.

The author conducts multiple analyses to evaluate the similarity between the original and synthetic data. These experiments involve computing the correlation matrices, visualizing joint and marginal distributions, performing Principal Component Analysis (PCA) [259] to study distributional consistency and performing machine learning regression to compare the feature importance in both datasets. The findings demonstrate that the synthetic data generated by the proposed framework possesses adequate predictive information for regression applications (Figure, 5.3). Additionally, the joint and marginal distributions of the features (Figure 5.4) present in the input data are also captured by the generated data. Moreover, the produced samples preserve the fundamental patterns and structure of the original dataset (Figure 5.5) and accurately reflect the correlations present within (Figures 5.7 and 5.8).

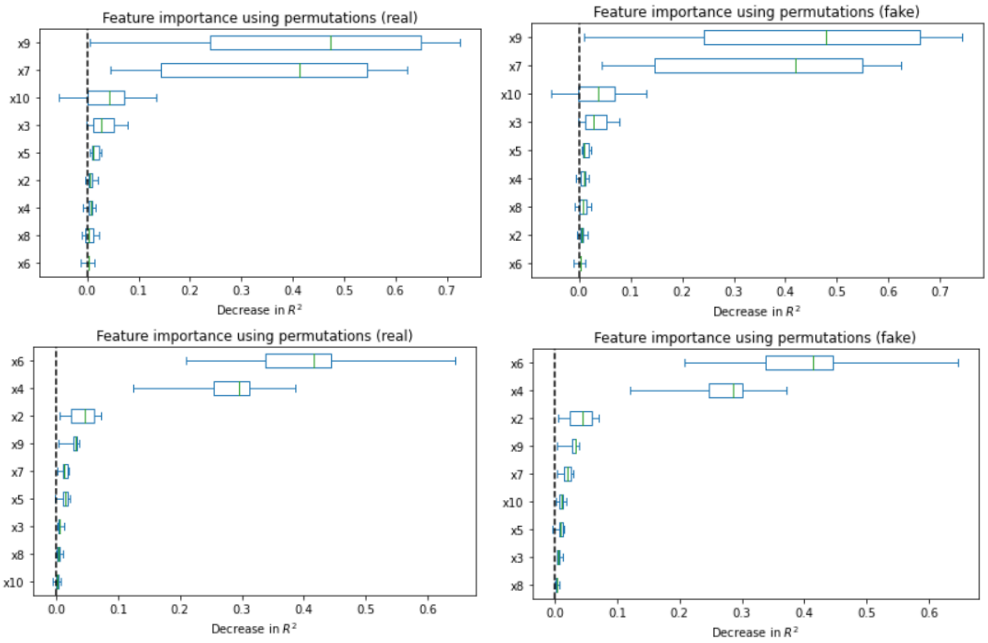



Figure 5.3: Feature importance comparison between real (left) and synthetic (right) data, in both the ANM (first row) and the PNL (second row) case. The synthetic features with their relevance are indistinguishable from the original ones, allowing for their application in regression tasks.

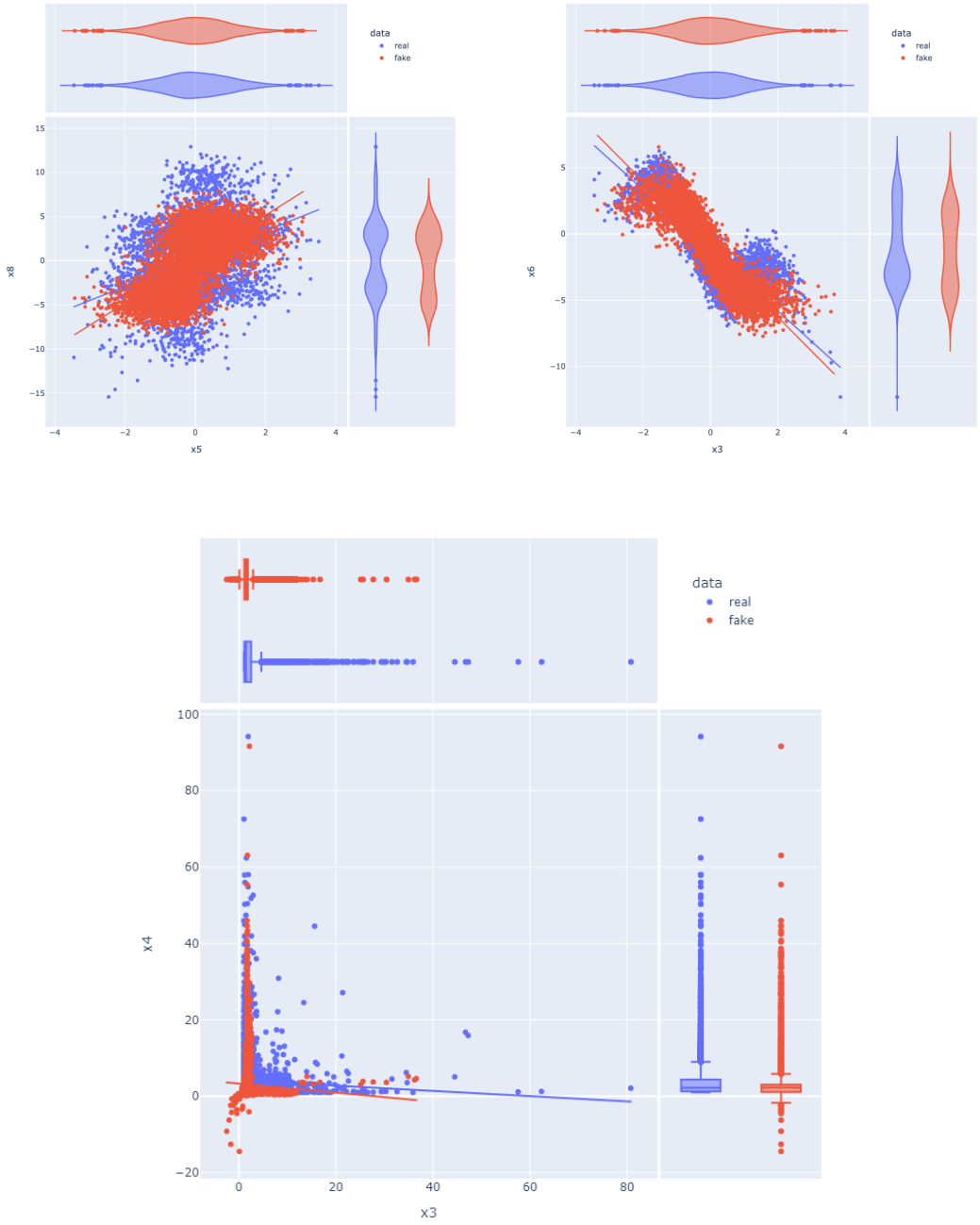



Figure 5.4: Visualizing the distributions of the real and synthetic features, the author plotted x5 against x8 (left), x3 against x6 (right), in the case of ANM, and x3 against x4 for the PNL case. The joint and marginal distributions are accurately modeled with no significant differences between the real and synthetic features.

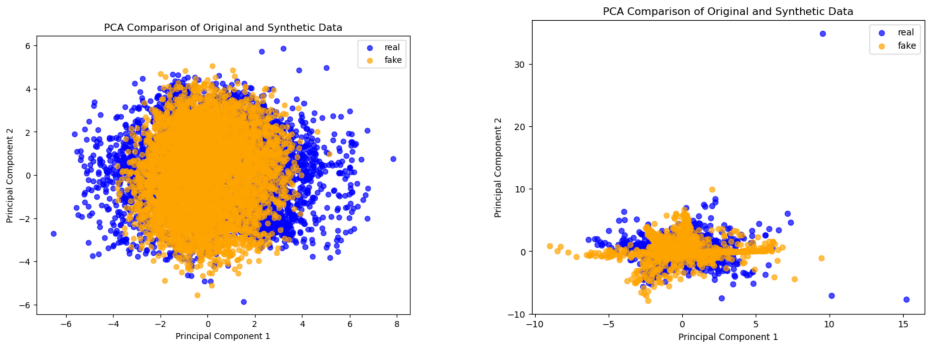



Figure 5.5: Principal Component Analysis (PCA) between the original and synthetic samples for both the ANM (left) and the PNL (right) case. The author observes both the input and the synthetic samples have similar clusters and outliers. The results indicate that the implicitly generated distribution resembles the original distribution in both mean and standard deviation, making them indistinguishable from each other.

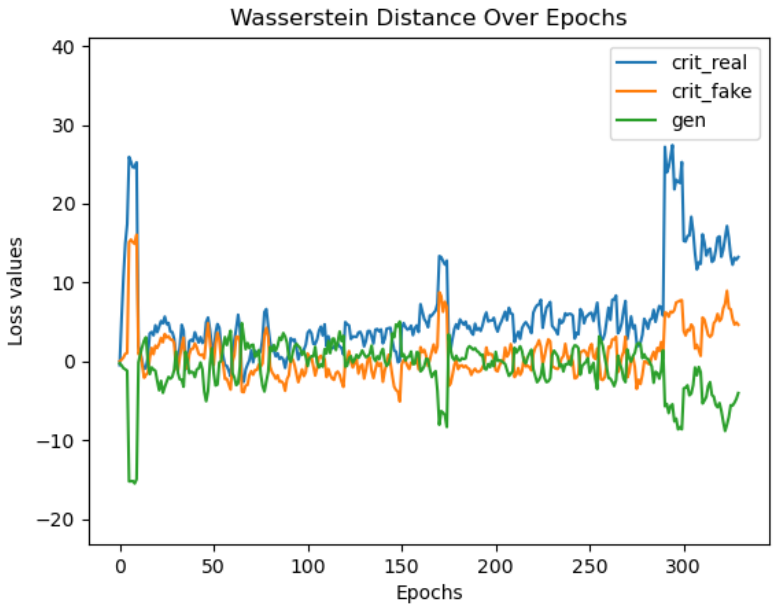



Figure 5.6: Visualizing the Wasserstein distance between the original and synthetic data over the course of the augmented Lagrangian algorithm. The significant discrepancy between the real and the generated samples (165-170 and from 300 epochs onward) occurs because of fluctuations in the SHD, courtesy of the parameter-tuning for the continuous optimization approach. Conversely, the lowest SHD is detected when the Wasserstein Distance is at its lower conversions (50-150 and 175 - 275 epochs).

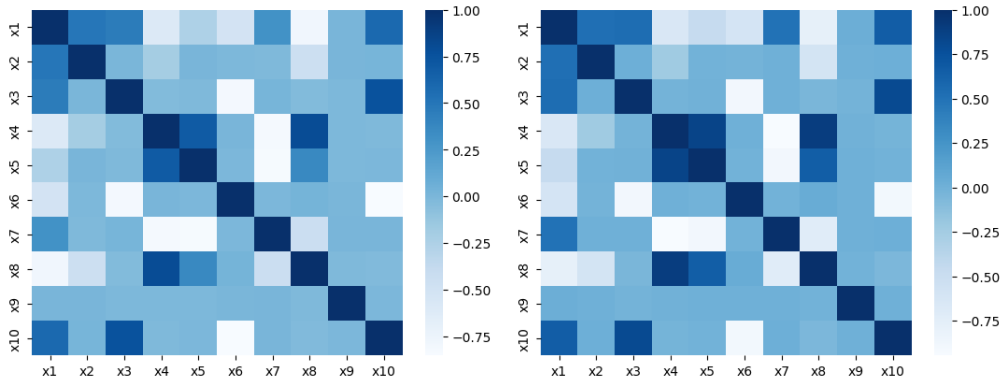



Figure 5.7: Comparison of the correlation matrices for real (left) and synthetic (right) features reveals that the statistical correlations across the feature space for both real and synthetic data are nearly identical, in the ANM case.

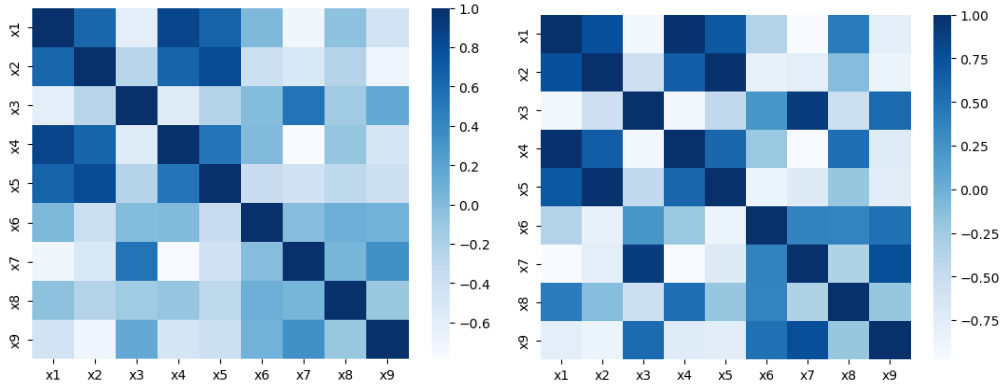



Figure 5.8: Comparison of the correlation matrices for real (left) and synthetic (right) features reveals that the statistical correlations across the feature space for both real and synthetic data are nearly identical, in the PNL case.

### 5.2.5 Additional results

The author enhances the analysis from earlier experiments by incorporating more examples. These include real-vs-synthetic statistical comparisons for each feature (Table 5.8), additional visual representations of synthetic feature distributions (Figure 5.9), and the remaining outcomes from machine learning regression models (Figures 5.10 and 5.11). Additionally, they offer examples of suboptimal results to demonstrate the repercussions when causal structure learning or tabular data synthesis do not yield adequate results.

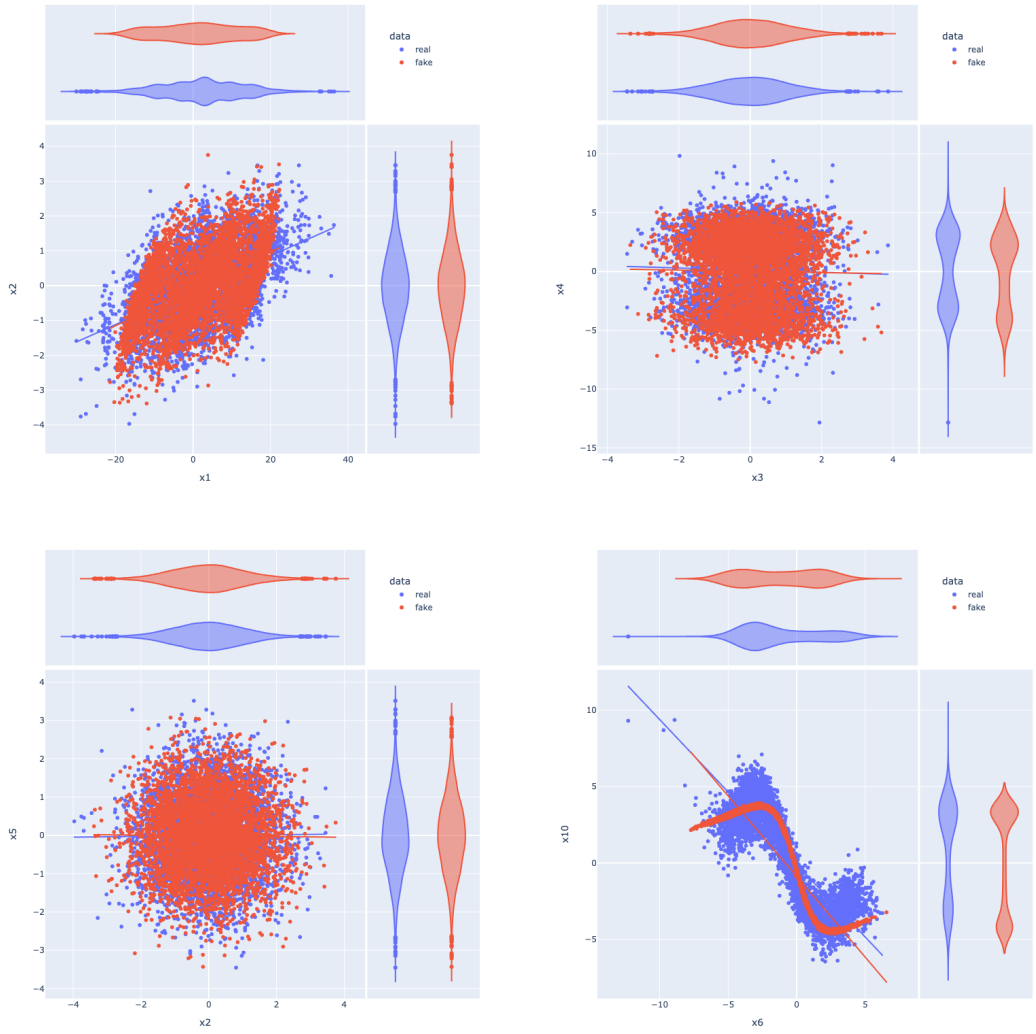



Figure 5.9: Further examples of the synthetic joint and marginal distributions for the method of the author on the dataset presented in Section 5.2.4. The author observes multiple cases with different distribution shapes. Additionally, they depict one case of severe latent collapse (bottom-right corner) in the produced data from DAGAF.

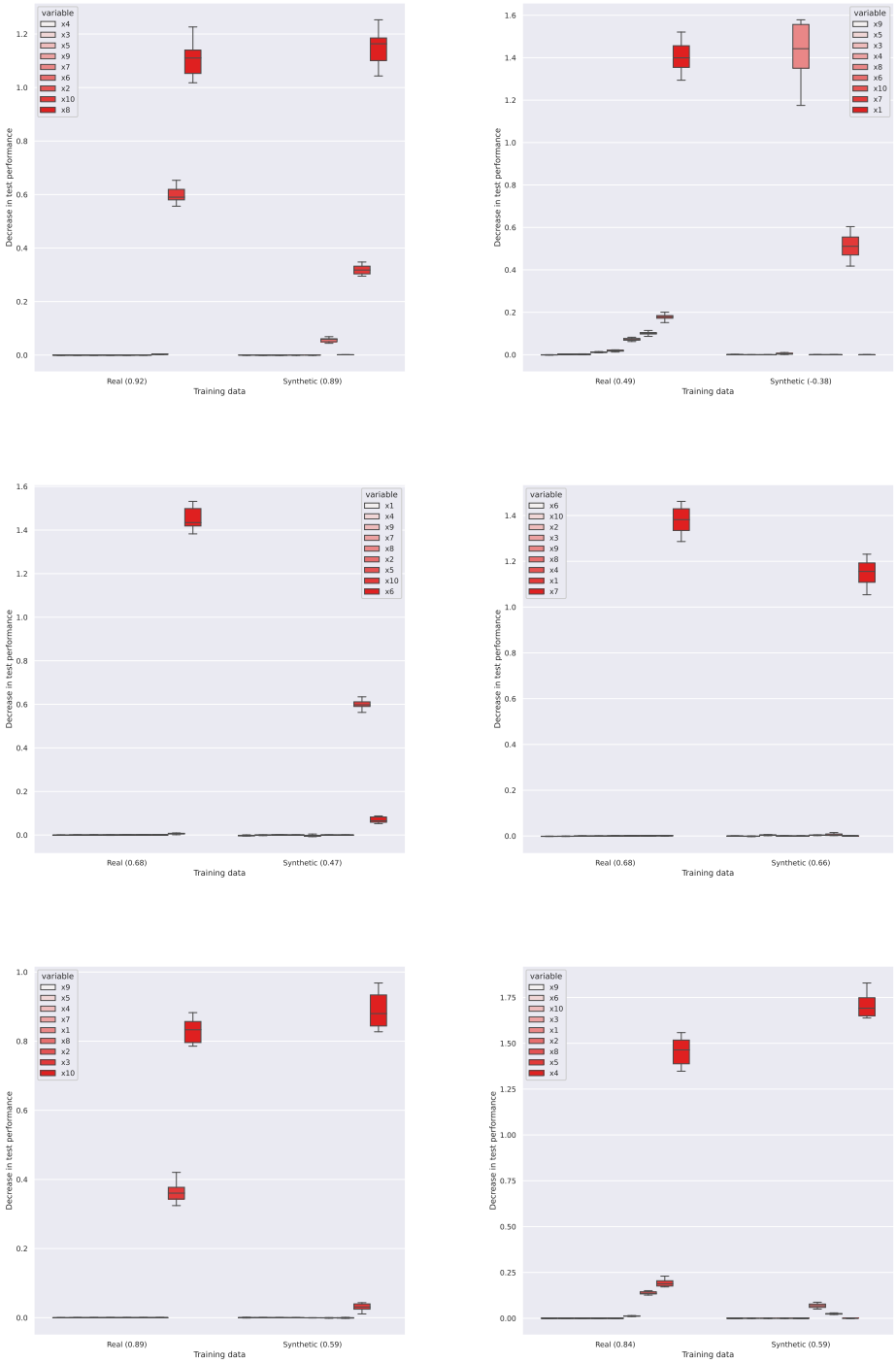



Figure 5.10: Remaining examples of feature importances (x1-x6) to supplement the results in Section 5.2.4. The author observes some failure cases, where the synthetic features differ significantly from their real counterparts.

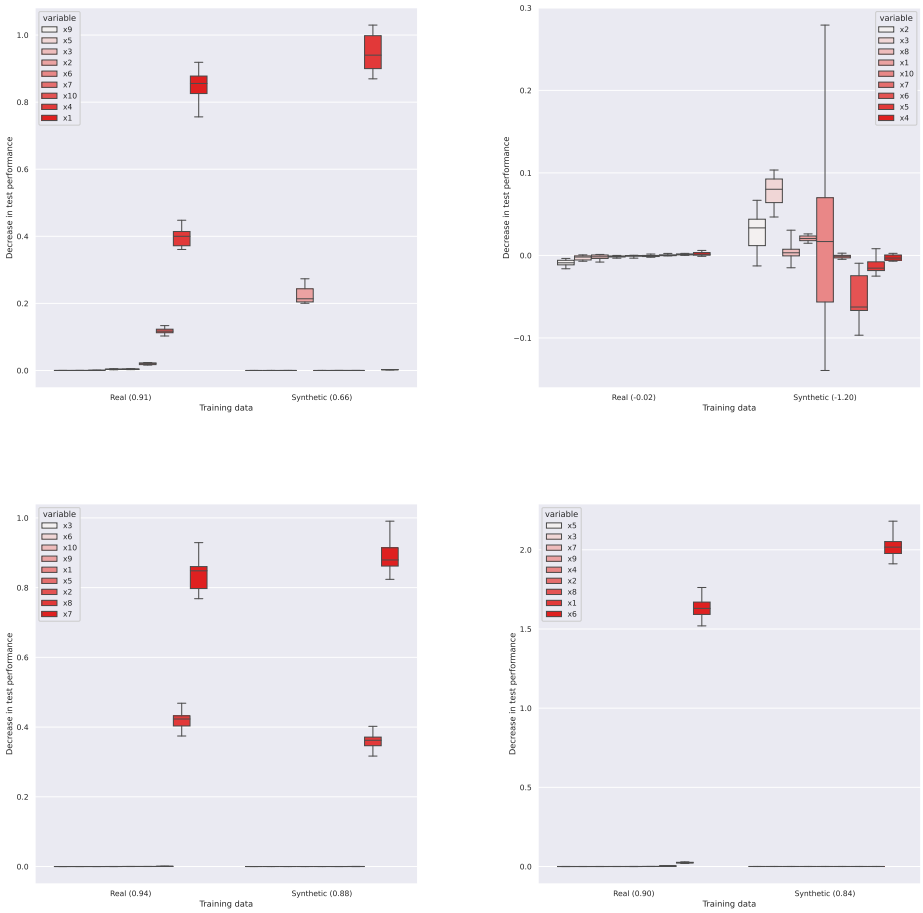



Figure 5.11: Remaining examples of feature importances (x7-x10) to supplement the results in Section 5.2.4. The author observes some failure cases, where the synthetic features differ significantly from their real counterparts.

Table 5.8: Mann-Whitney t-test results for all real and synthetic features to supplement Figure 5.4. The author observes some failure cases, where the real and synthetic features differ significantly ($p < 0.05$).

| Feature | $p$-value |
|---|---|
| x1 | 7.7952e-07 |
| x2 | 0.5004 |
| x3 | 0.1683 |
| x4 | 0.0020 |
| x5 | 0.8563 |
| x6 | 0.9127 |
| x7 | 0.0364 |
| x8 | 0.1747 |
| x9 | 0.2089 |
| x10 | 6.4502e-26 |

### 5.2.6 Ablation study

In an ablation study, the author aimed to find the optimal combination of terms within the loss function for Step 1. Using the Sachs, ECOLI70, MAGIC-IRRI and ARTH150 datasets available in `https://www.bnlearn.com/bnrepository`, nine distinct experiments were conducted, exploring different mixtures of loss terms. Each setup included the Wasserstein-1 distance. The first configuration, labeled "**w/o recon loss**", excludes the reconstruction loss and its regularization from the training process. The remaining configurations were identified by the specific terms involved in the reconstruction loss for $W$, such as MSE [180] and NLL [260]. Furthermore, the author examined combinations with additional terms like MMD [67] and KLD [58]. The outcomes of these trials are detailed in Table 5.9.

Table 5.9: DAGAF ablation study

| Loss function | SHD | | | |
|---|---|---|---|---|
| | Sachs | ECOLI70 | MAGIC-IRRI | ARTH150 |
| w/o recon loss | 21 | 109 | 194 | 352 |
| recon loss (MSE) | 14 | 85 | 148 | 263 |
| recon loss (NLL) | 16 | 100 | 163 | 295 |
| MSE + MMD | 10 | 51 | 111 | 164 |
| NLL + MMD | 14 | 85 | 148 | 263 |
| MSE + KLD | 12 | 63 | 130 | 196 |
| NLL + KLD | 12 | 63 | 130 | 196 |
| MSE + KLD + MMD | **9** | **46** | **102** | **150** |
| NLL + KLD + MMD | 11 | 54 | 117 | 172 |

### 5.2.7 Sensitivity analysis

In order to evaluate the robustness of the model, the author conducts a sensitivity analysis to examine how changes in hyper-parameter configurations influence model training. This study measures the accuracy of DAG modeling (denoted as SHD) under varying hyper-parameters, including the learning rate and dropout rate (**lr, dropout**), the size of the noise vector, and the batch size (**z-size, batch-size**). The analysis begins with a baseline set at **lr = 0.001, dropout = 0.5, z-size = 1, batch-size =**

**100**, modifying each parameter one at a time to determine their impact on the SHD. The experiments utilized the Sachs dataset, with the outcomes compiled in Table 5.10.

Table 5.10: DAGAF sensitivity analysis

| Hyper-parameters | Sachs Dataset |
|---|---|
| | SHD |
| lr = 3e-3, dropout = 0.5, z-size = 1, batch-size = 100 | 9 |
| lr = 3e-3, dropout = 0.0, z-size = 1, batch-size = 100 | 10 |
| lr = 3e-3, dropout = 0.5, z-size = 2, batch-size = 100 | 10 |
| lr = 3e-3, dropout = 0.5, z-size = 5, batch-size = 100 | 11 |
| lr = 3e-3, dropout = 0.5, z-size = 1, batch-size = 500 | 9 |
| lr = 3e-3, dropout = 0.5, z-size = 1, batch-size = 1000 | 10 |
| lr = 2e-4, dropout = 0.5, z-size = 1, batch-size = 100 | 11 |
| lr = 1e-3, dropout = 0.5, z-size = 1, batch-size = 100 | 12 |

## 5.3 Discussion & Future Work

Tables 5.1 through 5.5 indicate that the proposed general framework for causality-driven tabular data synthesis consistently surpasses current state-of-the-art methods in DAG learning across all test scenarios (linear, nonlinear-1, nonlinear-2, post-nonlinear-1, and post-nonlinear-2) and data dimensionalities, regardless of whether ANM or PNL assumptions are applied. Notably, the SHD difference between DAGAF and other models becomes more pronounced as data dimensionality increases, underscoring the enhanced performance of the approach for DAG learning in datasets with numerous variables, courtesy of adversarial training.

Table 5.6 showcases the benchmark experiment results, highlighting the exceptional performance of DAGAF. Notably, it consistently surpasses DAG-GNN across all four datasets: Child, Alarm, Hilfinder, and Pathfinder. Moreover, both DAG-WGAN and its improved version, DAG-WGAN+, deliver poorer outcomes than the approach of the author in three out of the four datasets. Similar patterns emerge in experiments on continuous datasets, where the SHD gap between the method of the author and others widens with more data variables.

Up to this point, the author has focused solely on the performance of their model

using continuous and benchmark datasets. While these outcomes show strong performance, assessing the approach with real-world datasets is essential for a comprehensive evaluation. The experiment using the Sachs dataset illustrates that DAGAF is proficient at accurately determining DAG structures from real data. As indicated in Table 5.7, the method of the author significantly surpasses all other prominent models employed in the study. Furthermore, empirical data suggest that assuming PNL enables the framework to derive a more precise approximation of the causal structure compared to other identifiable causal models.

In DAGAF, the process of learning DAG structures from observational data is conducted alongside the generation of high-quality synthetic datasets. This is evidenced by the results in Figures 5.7, 5.8, 5.3, 5.6, 5.5 and 5.4, applicable to both ANM and PNL scenarios. When the model accurately identifies the true structure in a dataset (indicated by SHD = 0), the gap between the distributions of the real and synthetic data is minimized. Additionally, the model accurately mirrors the statistical dependencies from the real dataset in the synthetic data it generates. These findings demonstrate the ability of DAGAF to produce varied data samples while preserving the integrity of DAG structures.

The ablation study together with the sensitivity analysis identifies the best combination of loss functions for the framework and illustrates the effect of hyperparameters on model training. As shown in Table 5.9, the optimal loss terms for Step 1 include MSE, KLD, MMD, and adversarial training. Additionally, the data in Table 5.10 reveal that lowering the learning and dropout rates substantially enhances model performance. Conversely, expanding the dimensions of the noise vector and input batch size leads to only minor variations in the accuracy of the algorithm.

The results of the experiments demonstrate that the proposed approach adeptly handles various types of data (numerical and categorical) to consistently reconstruct DAG structures given the ANM and PNL assumptions while generating realistic data samples. Notably, DAGAF significantly surpasses the performance of the latest DAG-learning methods. The research highlights that the incorporation of the Wasserstein distance substantially enhances the process of DAG-learning.

Although DAG-WGAN+ demonstrates strong potential in causal structure learning, several limitations become evident when applying the model to benchmarks or data produced by real-world systems (i.e., Child, Alarm, Sachs, Hailfinder, Pathfinder, etc.). One of the primary challenges lies in the nature of real-world data, which often contain noise, potential measurement errors and mixed data types. DAG-WGAN+, as a generative machine learning-based approach, can be sensitive to these issues, risking overfitting or misrepresenting causal relationships when data quality is suboptimal or the dataset itself is inherently challenging. Furthermore, the model assumes causal sufficiency, which means that all relevant variables are observed and measured. In practical settings, this assumption rarely holds, and the presence of latent confounders or unobserved factors can lead to biased or incomplete causal structures. In their future work, the author intends to address these challenges by integrating robust regularization techniques, domain adaptation methods, or hybrid approaches that explicitly model latent variables. Additionally, pre-processing strategies to better handle mixed data types and denoise observations will be utilized to improve the stability and reliability of learned causal structures in real-world datasets.

Model-related limitations also constrain DAG-WGAN+ in real-world application. The framework enforces acyclicity through continuous relaxations, which restricts its ability to represent systems with feedback loops or cyclic dependencies. Additionally, the performance of DAG-WGAN+ depends heavily on hyperparameter tuning, network architecture, and adversarial training stability. Small changes in these configurations can result in significantly different inferred graphs, undermining reproducibility and interpretability. Consequently, while the adversarial component allows the model to capture complex nonlinear dependencies, it can also introduce training instability, making convergence difficult and outcomes inconsistent across runs. This occurs primarily due to the delicate balance required between the generator and discriminator during training, where small imbalances can lead to higher oscillating losses or mode collapse. Future research will explore additional stabilization techniques for adversarial training, such as adaptive learning rates or curriculum learning approaches. Alternative formulations for enforcing acyclicity, as well as architectures capable of approximating

cyclic dependencies, will also be explored with the aim of enhancing the robustness and expressiveness of DAG-WGAN+ while improving reproducibility across runs.

Moreover, the outcomes discussed have been derived utilizing LiNGAM, ANM, or PNL causal models, which are recognized as identifiable SCM [31], [98], [261], [32]. Nevertheless, the scope of current experiments is limited to these models, presenting a challenge. Future research endeavors will investigate a broader array of identifiable structures, including generalized linear models, polynomial regression, and index models. Additionally, experiments involving the synthesis of tabular data have also been somewhat limited, focusing largely on the basic attributes of datasets. Ongoing work will aim to broaden these investigations by evaluating DAGAF against other causality-based tabular data generation methods [218], [219], [220]. This evaluation will utilize more appropriate metrics like the Cross-Validation Score (CVS) [262], Kolmogorov-Smirnov (KS) test [263] or Chi-Square test [264] to enable a more comprehensive assessment of the data generation proficiency of DAGAF.

The proposed approach identifies DAG structures by integrating MLE with adversarial loss components while applying an acyclicity constraint through an augmented Lagrangian. Consequently, DAGAF is characterized by substantial computational demands and a complex loss function. The author plans to explore more efficient methods for structure learning and adversarial loss training to create a faster model that mainly uses the Wasserstein distance. Additionally, the PNL model instance is limited since the neural network designed to learn $g^{-1}$ features a simple architecture, and there is inadequate regularization on the loss term that governs the parameter learning of the invertible function. Future research will involve experiments to ascertain whether a more elaborate architecture and loss function can be utilized in training $g^{-1}$ to discover more accurate causal structures.

The proposed causal learning-based framework for synthetic data generation is closely linked to recent advancements in generative modeling, including Digital Twins and transformer-based architectures. DAG learning inherently captures the core idea of attention mechanisms by identifying the direct causal parents of each variable, much like how transformers dynamically assign importance to relevant dependencies. Further-

more, the approach aligns with the principles of Digital Twins, which aim to replicate real-world systems and generate data that accurately represent their causal structures. This study introduces a unified framework for causal discovery and generative modeling, incorporating adversarial learning, MSE, MMD, and KLD regularization to ensure robust structure learning and high-fidelity synthetic data generation.

In their future work, the author will implement various strategies to address missing data. This includes data imputation techniques such as mean/mode imputation, multiple imputation, and more advanced methods such as matrix completion and variational autoencoders (VAE), while recognizing that imputation inherently introduces assumptions about missingness that could bias results. Additionally, they will incorporate structural information by utilizing partial knowledge of the directed acyclic graph (DAG), informed by domain expertise, to mitigate the impact of missing data. Another approach involves explicitly modeling missingness by introducing a missingness variable within the DAG to indicate whether a particular variable is absent. Furthermore, the author will apply causal inference methods, including latent variable models and specialized techniques tailored for incomplete data, to enhance the robustness and accuracy of their analyses.

Finally, as part of their future work, the author will explore the flexibility of their framework by experimenting with various combinations of SCM and DGM to determine the optimal configuration for improving output quality and extending its applicability to time-series data. To that end, emerging concepts, such as the digital twin layer utilizing multi-attention networks [222], [223], present promising directions for further investigation. Their capacity to handle mixed-variable datasets, align higher-order statistical distributions, and dynamically capture multimodal dependencies can enhance the causal discovery framework proposed in this study. Future research could focus on integrating these mechanisms to strengthen the robustness and scalability of causal discovery and synthetic data generation for complex real-world datasets. Such integration would bridge theoretical foundations with practical applications, addressing challenges like non-i.i.d. data and variable heterogeneity while enabling the creation of high-fidelity synthetic datasets for downstream tasks.

A thorough investigation of hyper-parameters will underpin the novel setup to identify their optimal values, leading to more realistic data samples produced by a more precisely simulated generative process.

# Chapter 6

# Conclusion

In this chapter, the author concludes their thesis by reflecting on the impact of their research on causal structure learning. They also share their perspective on recent advancements and the current state of the field, highlighting how their findings can shape future research directions. Finally, the author provides closing remarks on the contributions and significance of their work within the context of the thesis.

## 6.1 Advancements in Causal Structure Learning through the Wasserstein Distance

The research described in Chapter 3 provides compelling evidence that incorporating the Wasserstein distance metric can substantially improve causal structure learning from tabular data. By addressing the limitations of traditional MLE-based causal discovery methods, the study demonstrates how adversarial training, guided by the Wasserstein metric, can enhance both the accuracy of causal inference and the quality of data generation. To achieve this, the author introduces DAG-WGAN, a novel framework that seamlessly integrates a Wasserstein-based adversarial loss with an autoencoder architecture and an acyclicity constraint. This combination enables the model to simultaneously learn causal relationships and produce realistic synthetic data that better represent the underlying data distribution. Comprehensive experimental results indicate that DAG-WGAN consistently surpasses existing methods that exclude

the Wasserstein distance, particularly when applied to high-cardinality datasets. The model demonstrates remarkable scalability, achieving improved performance across scenarios involving 50 to 100 nodes, along with enhanced training stability that produces a 99.9% improvement across all experiments conducted and a Structural Hamming Distance (SHD) of 17 compared to 25 achieved by state-of-the-art models on real-world data. Moreover, DAG-WGAN exhibits a notable advantage in data generation quality, as illustrated in Figures 3.2 – 3.8. The results collectively suggest that the improved fidelity of the synthesized data not only enhances the interpretability and robustness of causal discovery but also contributes to a more accurate and dependable data generation process overall.

## 6.2 Optimizing Causal Structure Learning with Generative Adversarial Networks and DAG-NoCurl

The research described in Chapter 4 introduces a generative adversarial DAG learning framework that advances causal structure discovery by integrating adversarial training, disentangled representations and efficient structure learning techniques. Building on recent methods that reformulate causality learning as an optimization problem with a continuous acyclicity constraint, the proposed approach called DAG-WGAN+ leverages generative adversarial networks to overcome the limitations of maximum likelihood estimation and improve both accuracy and efficiency. Assuming the identifiability of the true causal model, the framework learns a causal structure capable of generating data distributions consistent with the observed data, further enhanced through integration with InfoVAE to encourage mutual information between latent and observed variables. Theoretical analysis demonstrates that, for a fixed level of mutual information, the model achieves global optimality when it accurately recovers the data distribution. Additionally, by adapting a modified version of the DAG-NoCurl framework, the proposed method achieves substantial speed improvements while avoiding restrictions tied to initial estimations, allowing continued refinement of the recovered DAG. Extensive experiments on benchmark datasets confirm that the model outperforms most state-

of-the-art approaches in both learning quality (99.9% improvement across all cases, SHD 15 vs. >= 16 for the state-of-the-art on real-world data) and computational performance (reduced computational complexity from cubic to quadratic).

## 6.3 DAGAF insights towards integrating Causal Discovery and Data Synthesis

The research described in Chapter 5 introduces DAGAF, a comprehensive and robust dual-step framework for multivariate causal structure learning and high-fidelity tabular data synthesis. Unlike conventional approaches that rely on a single identifiable causal model, DAGAF unifies multiple structural causal models (Additive Noise Model (ANM), Linear non-Gaussian Acyclic Model (LiNGAM), and Post-Nonlinear Model (PNL)) within a single architecture capable of learning complex causal dependencies. By leveraging Directed Acyclic Graphs (DAG) to represent inter-variable relationships, the framework models the underlying generative mechanisms of data, enabling it to produce realistic samples that closely match true data distributions. A rigorous theoretical analysis demonstrates how the Wasserstein-1 distance metric serves as an effective measure for guiding structure learning, while the integration of regularization and reconstruction loss terms strengthens the ability of the framework to recover meaningful causal relationships from observational data. Extensive experimental evaluations on both real-world and benchmark datasets reveal that DAGAF consistently outperforms state-of-the-art DAG-learning methods, achieving significantly lower Structural Hamming Distance (SHD) scores (Sachs: 47%, Child: 11%, Hailfinder: 5%, and Pathfinder: 7% improvements), while simultaneously generating diverse, high-quality synthetic samples. These findings highlight a profound connection between the accurate recovery of DAG structures and the generation of realistic, representative data, underscoring that the synthesis of authentic tabular datasets is inherently linked to the discovery of meaningful causal mechanisms within the data.

## 6.4 Future directions

The field of causal structure learning has made significant advancements in recent years, with research primarily focusing on enhancing continuous optimization-based methods and establishing robust theoretical frameworks for efficient causal discovery. These developments have enabled researchers to accurately identify causal relationships between data variables within a reasonable amount of time. The author envisions the next phase of progress in this field to be its practical application in industry. To that end, they propose directing future research efforts to solving the fundamental challenges associated with the integration of causal machine learning methods in industry described below. Each research problem is presented with a case study showcasing potential solutions and real-world applications, facilitating the translation of causal inference into diverse industrial domains.

- **Scalability to High-Dimensional and Big Data** – A key challenge is *scalability*, as current methods frequently face difficulties with computational efficiency and accuracy when applied to high-dimensional datasets, such as those in genomics or social networks. Future research will aim to address this by developing scalable algorithms that utilize sparsity, distributed computing, and approximation techniques such as adversarial training to efficiently manage large-scale systems.

  **Decoding Cancer Causality at Genomic Scale:** In genomics, large-scale projects such as The Cancer Genome Atlas (TCGA) have leveraged sparse Bayesian networks and adversarial training to discover causal relationships between gene mutations and tumor progression, enabling personalized cancer treatment strategies [265]. Scalable causal inference methods have also been applied in social network analysis to identify influential nodes that drive information diffusion and polarization across massive user networks.

- **Causal Structure Learning in Dynamic and Temporal Systems** – Increasing emphasis is being placed on causal structure learning from *dynamic* and *temporal* systems, as many real-world phenomena, such as climate patterns and

neural activity, involve complex temporal processes that are challenging to model causally. Progress in this area requires extending causal discovery techniques to time-series data, managing feedback loops, and addressing non-stationary behavior. Approaches applying dynamic Bayesian networks are anticipated to play a significant role in these advancements.

**Mapping Climate Feedback Loops Through Time** In climate science, researchers have employed Dynamic Bayesian Networks (DBN) to model causal interactions between $CO_2$, temperature, and ocean currents, improving predictions of climate feedback mechanisms [266]. Similarly, in neuroscience, the Human Connectome Project has used DBN and Granger causality to identify dynamic causal pathways between brain regions, deepening understanding of disorders such as epilepsy and schizophrenia.

- **Multi-Modal and Heterogeneous Data Integration:** – Integrating *multi-modal* and *heterogeneous* data presents a major challenge. Real-world datasets often encompass diverse formats, including text, images, and tabular data. Future approaches are expected to concentrate on identifying causal relationships across these varied modalities, potentially leveraging embeddings and feature representations to establish a cohesive causal framework.

  **Unifying Brain Imaging and Clinical Data in Alzheimer's Research:** The Alzheimer's Disease Neuroimaging Initiative (ADNI) combines MRI, PET, genetic, and clinical data to discover cross-modal causal relationships related to cognitive decline. Through multi-view causal representation learning, researchers identified biomarkers predictive of Alzheimer's progression, improving early detection and interpretability [267]. Similar techniques are also used in autonomous systems and disaster response to fuse sensory, textual, and environmental data for causal event modeling.

- **Causal Discovery in Noisy, Biased or Missing Data** – Managing *noisy, biased, or missing* data continues to be a significant challenge, as real-world datasets are frequently incomplete or contain errors, making causal inference

more complex. Future research will strive to develop robust algorithms capable of addressing noise, hidden confounders, and selection bias, while accurately imputing missing data without compromising the underlying causal relationships.

**Recovering Medical Causality from Imperfect Health Records:** In healthcare, causal discovery from Electronic Health Records (EHR) often involves incomplete or biased data. Robust causal models using graph-based imputation and Bayesian inference have been employed to reveal medication–outcome relationships in chronic disease management [268]. In economics, causal techniques with bias correction are used by institutions such as the IMF and World Bank to infer policy impacts from incomplete and noisy global indicators.

- **Applications in Real-World Domains** – The application of causal structure learning in *real-world* domains presents immense potential. Areas such as healthcare, economics, and environmental science stand to gain from actionable insights into causal relationships. Future work will focus on partnering with domain experts to develop customized causal discovery tools and showcase their effectiveness in addressing complex societal challenges, such as crafting public health strategies or informing policy decisions.

  **Causal Insights from COVID-19 Policy Interventions:** During the COVID-19 pandemic, causal Bayesian networks were applied to the Oxford COVID-19 Government Response Tracker to evaluate the effectiveness of interventions across more than 180 countries. The study identified which measures, such as lockdowns and mask mandates, had the strongest causal effect on transmission reduction [269]. Beyond public health, similar causal modeling frameworks are now used in energy systems for predictive maintenance and in economic policy design for assessing taxation and welfare impacts.

- **Ethics, Bias and Policy Implications** – The field must address the *ethical* and societal challenges associated with causal reasoning. Applying causal models in sensitive areas, such as hiring practices or criminal justice, raises important concerns regarding *bias* and fairness. It will be essential to develop methods that

promote ethical applications and produce unbiased results, especially as causal models are increasingly employed in decision-making and policy development.

**Reassessing Fairness in Algorithmic Justice:** The ProPublica COMPAS study exposed racial bias in recidivism prediction systems. Subsequent research on counterfactual fairness [270] applied causal reasoning to separate legitimate from spurious causal pathways, leading to fairer decision frameworks. Similar causal debiasing approaches have been adopted by organizations such as LinkedIn and IBM Research to ensure equitable outcomes in hiring and recommendation algorithms.

Moreover, the author provides additional case studies and real-world applications for addressing specific scientific problems within industry domains they are interested in, such as Physics, Astronomy, Large Language Models (LLM), and Biomedical Sciences.

- **Causal Discovery in Complex Physical Systems** – Understanding causality in physical and astrophysical data remains difficult due to non-linearity, noise, and temporal dependencies inherent to large observational datasets. Extending causal discovery to handle dynamic, multivariate signals is key to improving physical interpretability.

  **Tracing Cosmic Evolution through Causal Graphs:** Researchers applied causal structure learning to cosmological simulations (e.g., CAMELS) to discover how dark matter distribution causally influences galactic formation and star evolution [271], [272]. Graph neural networks and dynamic Bayesian models were used to infer causal dependencies across temporal snapshots, enhancing the interpretability of simulation-based inference in astrophysics.

- **Causal Reasoning and Fairness in LLM** – As large language models are increasingly used in decision-support systems, ensuring causal consistency and fairness is critical. Most LLM excel at correlational pattern recognition but struggle with true causal inference or counterfactual reasoning.

  **Probing Causal Understanding in Large Language Models:** Recent evaluations show that GPT-4 and similar LLM exhibit systematic biases in causal

judgment tasks, such as direction-of-causation or counterfactual inference [273]. Hybrid models integrating structural causal models with LLM aim to improve causal reasoning and reduce bias, advancing interpretability and ethical deployment.

- **Causal Discovery in Real-World Clinical Data** – Handling bias, noise, and missingness in health records remains a major challenge in clinical causal inference. Integrating causal discovery with generative patient modeling can emulate and validate clinical trials.

  **Emulating Real-World GLP-1 Efficacy in Type 2 Diabetes through Causal Learning and Virtual Patients:** A virtual trial framework combined causal structure learning with generative modeling to emulate RCT of GLP-1 receptor agonists [274]. Using 5,476 patient records, virtual patients were generated via a causal-WGAN to reproduce treatment effect rankings, demonstrating scalable and generalizable estimation of real-world treatment efficacy.

Ultimately, the connection between causality and adversarial training underscores a fundamental shift in modern machine learning toward models that prioritize robustness, interpretability, and causal validity over superficial correlation fitting. While causality seeks to discover the true generative mechanisms that govern observed samples, adversarial training reinforces this objective by exposing models to carefully constructed perturbations that emulate counterfactual or interventional scenarios. This process forces models to distinguish between features that are causally relevant and those that are merely coincidental or distribution-specific. In essence, adversarial perturbations function as empirical probes, similar to causal interventions, that reveal the stability and invariance of learned representations under various manipulations. By aligning the empirical rigor of adversarial robustness with the conceptual foundations of causal inference, researchers can accelerate the development of learning systems that not only withstand adversarial or out-of-distribution challenges but also capture the mechanism-driven regularities underlying real-world data. Therefore, the integration of causal reasoning and adversarial training represents a promising pathway toward achieving more

reliable, interpretable, and firmly grounded in underlying scientific principles artificial intelligence.

## 6.5 Closing thoughts

This thesis delves into various theoretical and practical aspects of causal discovery, focusing on continuous optimization-based models, efficient structure learning algorithms, and frameworks designed to capture multivariate causality under multiple causal model assumptions. It introduces three approaches (namely DAG-WGAN, DAG-WGAN+ and DAGAF), which have led to four successful publications (two in conferences and two in journals), with a fifth paper currently in progress. Through these contributions, the author has made a substantial impact on key areas of causal structure learning, including adversarial-based causal discovery, resource-efficient structure learning, and the simultaneous approximation of causal mechanisms and tabular data synthesis.

The thesis provides a comprehensive account of the implementation details of these models, supported by theoretical analyses that include mathematical proofs and intuitive explanations. Additionally, the author presents extensive empirical evidence from various experiments to validate their theoretical claims and test their underlying hypotheses. Finally, the results highlight that the proposed models significantly outperform state-of-the-art approaches, showcasing their effectiveness and superiority.

# Appendix A

# Theoretical Proofs

This appendix serves as a designated space for presenting proofs of various statements made throughout this work.

## A.1 Proof of lemma 3.2.1

**Lemma 3.2.1.** The Structural Equation Model (SEM) used in the decoder architecture $\tilde{\mathbf{X}} = P_\theta(\mathbf{X}|\mathbf{F}_2((I - \mathbf{A}^T)^{-1}\mathbf{F}_1(Z)))$ belongs to the Additive Noise Model category.

*Proof.* For simplicity, the derivation of this proof requires only the architecture of the generative model, denoted as $\tilde{\mathbf{X}} = \mathbf{F}_2((I - \mathbf{A}^T)^{-1}\mathbf{F}_1(Z))$. According to [35], under the assumption that $\mathbf{F}_2$ is invertible, $\tilde{\mathbf{X}} = \mathbf{F}_2((I-\mathbf{A}^T)^{-1}\mathbf{F}_1(Z)) \equiv \mathbf{F}_2^{-1}(\tilde{\mathbf{X}}) = \mathbf{A}^T\mathbf{F}_2^{-1}(\mathbf{X}) + \mathbf{F}_1(Z)$. Furthermore, if the functions $\mathbf{F}_2$ and $\mathbf{F}_1$ are omitted, the architecture simplifies to $\tilde{\mathbf{X}} = (I - \mathbf{A}^T)^{-1}Z \equiv \mathbf{A}^T\mathbf{X} + Z$, where $\mathbf{A}^T\mathbf{X} + Z$ is the linear SEM.

The linear SEM can be represented as a Generalised Linear Model (GLM) $X_j = g_j(f_j(X))$, where $X_j = g_j(f_j(X)) \equiv X_j = \mathbf{A}_j^T X + \mathcal{Z}_j$ under the assumption that the function $g_j$ just adds noise to its input and $f_j = \mathbf{A}_j^T X$ (i.e linear). However, the parameterized functions $\mathbf{F}_2$ and $\mathbf{F}_1$ apply nonlinearity to the linear structural equation model. Therefore, the SEM applied in the VAE component of DAG-WGAN is a special case of GLM, where $X_j = g_j(f_j(X))$ assumes the general form of $X_j = f_j(X) + \mathcal{Z}_j$, which falls under the Additive Noise Model (ANM) category [98] due to $f$ being nonlinear, thus concluding the proof. □

## A.2 Proof of proposition 3.2.2

**Proposition 3.2.2.** Given an (un)known ground truth graph $\mathbf{G_A^0}$ faithful to the observational data distribution $P_{\mathbf{G_A^0}}(\mathbf{X})$, the parameters of the implicitly learned probability distribution $P_{\mathbf{G_A}}(\tilde{\mathbf{X}})$ are refined by the following solution $D : \mathbb{R} \to \mathbb{R}$

$$\underbrace{\mathbb{E}_{\tilde{\mathbf{X}} \sim \mathbb{P}_g}[D(\tilde{\mathbf{X}})] - \mathbb{E}_{\mathbf{X} \sim \mathbb{P}_r}[D(\mathbf{X})]}_{\text{Critic loss}} + \underbrace{\lambda \mathbb{E}_{\hat{\mathbf{X}} \sim \mathbb{P}_{\hat{\mathbf{X}}}}[(||\nabla_{\hat{\mathbf{X}}} D(\hat{\mathbf{X}}) - 1||)^2]}_{\text{Gradient penalty}}$$

$$\underbrace{\mathbb{E}_{Z \sim Q_\phi(Z|\mathbf{F}_4((I-\mathbf{A}^T)\mathbf{F}_3(\mathbf{X})))}[D(Dec(Z))]}_{\text{Generator loss}},$$

where both terms are well-defined, differentiable almost everywhere and converge when $P_{\mathbf{G_A^0}}(\mathbf{X}) = P_{\mathbf{G_A}}(\tilde{\mathbf{X}})$.

*Proof.* DAG-WGAN is a VAE-GAN approach, where the adversarial architecture is WGAN-GP. Notice also that the discriminator $D$ used in DAG-WGAN is very similar to the one used in the standard WGAN-GP (see the WGAN-GP architecture subsection in Section 3.2.1), and both are trained using the same adversarial loss. This allows the author to derive the proof of their proposition from other existing ones.

The convergence of the terms in the proposition relies on the fact that all variations of WGAN converge when the critic cannot distinguish real from fake data samples, at which point the Wasserstein distance is 0. Theoretically speaking, to achieve an Earth-Mover distance of 0 means that the generator must synthesize new data samples which are indistinguishable from the input. To this end, the following must be correct: **Given a fixed optimal 1-Lipschitz continuous discriminator $D^*$, the generator $G$ converges if and only if $P_{\mathbf{G_A^0}}(\mathbf{X}) = P_{\mathbf{G_A}}(\tilde{\mathbf{X}})$.** The above statement is intuitively true because $D^*$ always produces a Wasserstein distance of 0, which is only possible if the samples produced by $G$ belong to a probability distribution $P_{\mathbf{G_A}}(\tilde{\mathbf{X}})$ identical to the observational distribution $P_{\mathbf{G_A^0}}(\mathbf{X})$. Therefore, the generator will converge if and only if its fake samples do not violate the converging condition of the critic, which occurs only when $P_{\mathbf{G_A^0}}(\mathbf{X}) = P_{\mathbf{G_A}}(\tilde{\mathbf{X}})$, thus completing the proof.

□

## A.3 Proof of proposition 4.1.1

**Proposition 4.1.1.** Given some input $\mathbf{X}$ and latent variables $Z$, for any fixed value of the mutual information term $\mathbf{I}_{Q_\phi(\mathbf{X},Z)}(\mathbf{X}, Z)$, $\mathcal{L}_{DAG-WGAN+}$ reaches global optimum when the decoder distribution $P_\theta(\mathbf{X}|\mathbf{F}_2((I - \mathbf{A}^T)^{-1}\mathbf{F}_1(Z)))$ matches the observational data distribution $P(\mathbf{X})$.

*Proof.* The author considers the InfoVAE objective used in the context of the DAG-WGAN+ decoder model, defined as:

$$\begin{aligned}\mathcal{L}_{\text{VAE}} = &-\mathbb{E}_{Q_\phi(Z|\mathbf{F}_4((I-\mathbf{A}^T)\mathbf{F}_3(\mathbf{X})))}[\log P_\theta(\mathbf{X}|\mathbf{F}_2((I - \mathbf{A}^T)^{-1}\mathbf{F}_1(Z)))] \\ &+ (1-\beta)\mathbb{E}_{\mathbf{X}\sim P(\mathbf{X})}[D_{KL}(Q_\phi(Z|\mathbf{F}_4((I - \mathbf{A}^T)\mathbf{F}_3(\mathbf{X})))||P(Z))] \\ &+ (\gamma+\beta-1)D_{\text{KL}}(Q_\phi(Z)\|P(Z)).\end{aligned}$$

More specifically, they aim to show that this objective achieves a global minimum when the joint distribution $Q_\phi(\mathbf{X}, Z)$ of the encoder matches the model joint distribution $P_\theta(\mathbf{X}, Z)$ induced by the decoder:

$$\begin{aligned}Q_\phi(\mathbf{X}, Z) \equiv P(\mathbf{X})Q_\phi(Z|\mathbf{F}_4((I - \mathbf{A}^T)\mathbf{F}_3(\mathbf{X}))) = \\ P_\theta(\mathbf{X}, Z) \equiv P(Z)P_\theta(\mathbf{X}|\mathbf{F}_2((I - \mathbf{A}^T)^{-1}\mathbf{F}_1(Z))).\end{aligned} \tag{A.1}$$

Under this condition, several consequences follow (by properties of joint distributions):

- The marginals match: $Q_\phi(\mathbf{X}) = P_\theta(\mathbf{X})$.
- The conditionals match: $Q_\phi(Z|\mathbf{F}_4((I-\mathbf{A}^T)\mathbf{F}_3(\mathbf{X}))) = P_\theta(\mathbf{F}_2((I-\mathbf{A}^T)^{-1}\mathbf{F}_1(Z))|\mathbf{X})$ and $Q_\phi(\mathbf{F}_4((I - \mathbf{A}^T)\mathbf{F}_3(\mathbf{X}))|Z) = P_\theta(\mathbf{X}|\mathbf{F}_2((I - \mathbf{A}^T)^{-1}\mathbf{F}_1(Z)))$.
- The latent marginal also matches: $Q_\phi(Z) = P_\theta(Z)$.

Substituting into the loss:

1. The reconstruction term becomes:

$$-\mathbb{E}_{Q_\phi(Z|\mathbf{F}_4((I-\mathbf{A}^T)\mathbf{F}_3(\mathbf{X})))}[\log P_\theta(\mathbf{X}|\mathbf{F}_2((I - \mathbf{A}^T)^{-1}\mathbf{F}_1(Z)))],$$

which achieves its minimum when $P_\theta(\mathbf{X}|\mathbf{F}_2((I-\mathbf{A}^T)^{-1}\mathbf{F}_1(Z))) = Q_\phi(\mathbf{F}_4((I-\mathbf{A}^T)\mathbf{F}_3(\mathbf{X}))|Z)$ - that is, the decoder correctly models the conditional distribution of the data given the latent variables.

2. The KL divergence terms vanish:

$$D_{\mathrm{KL}}(Q_\phi(Z|\mathbf{F}_4((I-\mathbf{A}^T)\mathbf{F}_3(\mathbf{X})))\|P(Z)) = 0 \quad \text{and} \quad D_{\mathrm{KL}}(Q_\phi(Z)\|P(Z)) = 0,$$

because the respective distributions match under the joint equality. In other words, the first term yields 0 because $Q_\phi(Z|\mathbf{F}_4((I-\mathbf{A}^T)\mathbf{F}_3(\mathbf{X}))) = P_\theta(\mathbf{F}_2((I-\mathbf{A}^T)^{-1}\mathbf{F}_1(Z))|\mathbf{X}) = P(Z)$, while the second equation computes a value of 0 due to $Q_\phi(Z) = P_\theta(Z) = P(Z)$.

To justify why this implies recovery of the true data distribution: note that the encoder is trained on samples from the true data distribution $P(\mathbf{X})$, so the joint distribution $Q_\phi(\mathbf{X}, Z) = P(\mathbf{X})Q_\phi(Z|\mathbf{F}_4((I-\mathbf{A}^T)\mathbf{F}_3(\mathbf{X})))$ is grounded in the true data. If the decoder achieves $P_\theta(\mathbf{X}, Z) = Q_\phi(\mathbf{X}, Z)$, then its marginal over $\mathbf{X}$ is also:

$$\begin{aligned} P_\theta(\mathbf{X}) &= \int P_\theta(\mathbf{X}|\mathbf{F}_2((I-\mathbf{A}^T)^{-1}\mathbf{F}_1(Z)))P(Z)\,dZ \\ &= \int Q_\phi(\mathbf{F}_4((I-\mathbf{A}^T)\mathbf{F}_3(\mathbf{X}))|Z)P(Z)\,dZ. \end{aligned}$$

But since $Q_\phi(\mathbf{F}_4((I-\mathbf{A}^T)\mathbf{F}_3(\mathbf{X}))|Z)$ was derived from $P(\mathbf{X})$ via the encoder, it follows that:

$$P_\theta(\mathbf{X}) = \int Q_\phi(\mathbf{F}_4((I-\mathbf{A}^T)\mathbf{F}_3(\mathbf{X}))|Z)P(Z)\,dZ = P(\mathbf{X}).$$

Thus, the decoder distribution $P_\theta(\mathbf{X})$ matches the true observational distribution. This means that the decoder

$$P_\theta(\mathbf{X}|\mathbf{F}_2((I-\mathbf{A}^T)^{-1}\mathbf{F}_1(Z)))$$

has successfully learned to generate samples indistinguishable from the true data distribution $P(\mathbf{X})$.

Therefore, the InfoVAE objective is globally minimized when $Q_\phi(\mathbf{X}, Z) = P_\theta(\mathbf{X}, Z)$,

and under this condition, the decoder $P_\theta(\mathbf{X}|\mathbf{F}_2((I-\mathbf{A}^T)^{-1}\mathbf{F}_1(Z)))$ recovers the true observational distribution $P(\mathbf{X})$, thus concluding the proof. □

## A.4 Proof of proposition 4.1.2

**Proposition 4.1.2.** Given a generated data distribution $P_{\mathbf{G_A}}(\tilde{\mathbf{X}})$, defined using a causal graph $\mathbf{G_A}$ belonging to the set of identifiable causal graphs $S_{\mathbf{G_A}}$, and the true underlying causal structure of the input data denoted as $\mathbf{G_A^0}$. Assuming that $\mathbf{G_A^0}$ is also a member of $S_{\mathbf{G_A}}$, then a learned causal graph $\mathbf{G_A}$ contains the same structure as $\mathbf{G_A^0}$ i.f.f. $P_{\mathbf{G_A}}(\tilde{\mathbf{X}})$ matches the original data distribution $P_{\mathbf{G_A^0}}(\mathbf{X})$.

*Proof.* If $\mathbf{G_A^0}$ is contained in $S_{\mathbf{G_A}}$, then according to Definition 1 the following statement must also be true: Under the same set of assumptions $\mathcal{A}$, there exists only one causal graph $\mathbf{G_A}$ capable of defining $P_{\mathbf{G_A}}(\tilde{\mathbf{X}})$. Hence, if a causal structure learning model $\mathcal{M}$ recovers a causal graph $\mathbf{G_A}$ that matches $\mathbf{G_A^0}$, then $P_{\mathbf{G_A}}(\tilde{\mathbf{X}}) = P_{\mathbf{G_A^0}}(\mathbf{X})$, thus concluding the proof. □

## A.5 Proof of proposition 5.1.1

**Proposition 5.1.1.** Let the ground-truth graph $\mathbf{G_A^0}$ be the only structure that can generate $P(\mathbf{X})$, then, under the assumption of causal identifiability, applying adversarial training ensures the following: 1) the implicitly generated distribution $P_{\mathbf{G_A}}(\tilde{\mathbf{X}})$ matches $P(\mathbf{X})$ and 2) the causal graph $\mathbf{G_A}$ used to define $P_{\mathbf{G_A}}(\tilde{\mathbf{X}})$ is identical to $\mathbf{G_A^0}$.

$$\mathbb{W}_p(P(\mathbf{X}), P_{\mathbf{G_A}}(\tilde{\mathbf{X}})) = 0 \implies P_{\mathbf{G_A}}(\tilde{\mathbf{X}}) = P(\mathbf{X}) \implies \mathbf{G_A} = \mathbf{G_A^0}.$$

*Proof.* Consider $\tilde{\mathbf{X}} \sim P_{\mathbf{G_A}}(\tilde{\mathbf{X}})$ as the distribution induced by the DAG $\mathbf{G_A}$. Assume that the observational data distribution $\mathbf{X} \sim P(\mathbf{X})$ is defined using the ground-truth graph $\mathbf{G_A^0}$. Furthermore, let the adversarial loss term $\mathcal{L}_{\text{adv}}(\mathbf{X}, \tilde{\mathbf{X}})$, describing the Wasserstein distance $\mathbb{W}_p(P(\mathbf{X}), P_{\mathbf{G_A}}(\tilde{\mathbf{X}}))$, correspond to the formulation given in Equation (5.1). Then, achieving the global optimum for $\mathcal{L}_{\text{adv}}(\mathbf{X}, \tilde{\mathbf{X}})$ guarantees distributional overlap between $P_{\mathbf{G_A}}(\tilde{\mathbf{X}})$ and $P(\mathbf{X})$.

$$\mathbb{W}_p(P(\mathbf{X}), P_{\mathbf{G_A}}(\tilde{\mathbf{X}})) = 0 \longrightarrow P_{\mathbf{G_A}}(\tilde{\mathbf{X}}) = P(\mathbf{X})$$

Additionally, by considering the reverse perspective one can observe how distributional alignment implies that the optimal solution for Equation (5.1) is discovered.

$$P_{\mathbf{G_A}}(\tilde{\mathbf{X}}) = P(\mathbf{X}) \implies \mathbb{W}_p(P(\mathbf{X}), P_{\mathbf{G_A}}(\tilde{\mathbf{X}})) = 0$$

When $\mathbf{G_A} \neq \mathbf{G_A^0}$, the synthetic and observational distributions cannot be matched, which means that $P_{\mathbf{G_A}}(\tilde{\mathbf{X}}) \neq P(\mathbf{X})$, because $\mathbf{G_A}$ is incorrect. As a result, there are fundamental structural discrepancies between $P_{\mathbf{G_A}}(\tilde{\mathbf{X}})$ and $P(\tilde{\mathbf{X}})$, resulting from the application of incorrect causal mechanisms in the generation of the synthetic distribution. These differences are reflected in their samples, leading to an increase in Earth Mover's distance:

$$\mathbb{W}_p(P(\mathbf{X}), P_{\mathbf{G_A}}(\tilde{\mathbf{X}})) > 0.$$

Therefore, minimizing $\mathcal{L}_{\text{adv}}(\mathbf{X}, \tilde{\mathbf{X}})$ ensures that $P_{\mathbf{G_A}}(\tilde{\mathbf{X}})$ aligns with $P(\mathbf{X})$, and the identifiability assumption guarantees that this alignment occurs exclusively when $\mathbf{G_A} = \mathbf{G_A^0}$, thus concluding the proof. □

## A.6 Proof of proposition 5.1.2

**Proposition 5.1.2.** Incorporating a reconstruction loss term into adversarial training ensures that the distance between individual data points from both synthetic $P_{\mathbf{G_A}}(\tilde{\mathbf{X}})$ and observational $P(\mathbf{X})$ data distributions is minimized. This reduction in noise prevents significant gradient fluctuations, resulting in more stable adversarial convergence.

$$\min_{\mathbf{G_A} \in \mathbb{D}} \mathcal{L}_{\text{MSE}}(\mathbf{X}, \tilde{\mathbf{X}}) = 0 \Rightarrow \forall i, \tilde{\mathbf{X}}_i = \mathbf{X}_i$$

*Proof.* Based on the mathematical formulation presented in Equation (5.2), the optimal solution for the loss term $\mathcal{L}_{\text{MSE}}(\mathbf{X}, \tilde{\mathbf{X}})$ can only be obtained when the squared difference between each corresponding pair of data points $\mathbf{X}_i \sim P(\mathbf{X})$ and $\tilde{\mathbf{X}}_i \sim P_{\mathbf{G_A}}(\tilde{\mathbf{X}})$, where

$i \in \{1, ..., n\}$, equals zero. This implies that there is no difference between the predicted data $\tilde{\mathbf{X}}$ and the original data $\mathbf{X}$, resulting in a perfect match between the means of their respective distributions $P_{\mathbf{G_A}}(\tilde{\mathbf{X}}) = P(\mathbf{X})$ w.r.t. $\mu$.

The gradient of $\mathcal{L}_{\text{MSE}}(\mathbf{X}, \tilde{\mathbf{X}})$ with respect to the SCM parameters $\theta$, which model $\mathbf{G_A}$ and subsequently $P_{\mathbf{G_A}}(\tilde{\mathbf{X}})$, can be expressed as the following:

$$\nabla_\theta \mathcal{L}_{\text{MSE}}(\mathbf{X}, \tilde{\mathbf{X}}) = \frac{1}{n} \sum_{i=1}^{n} 2 \cdot ||\mathbf{X}_i - \tilde{\mathbf{X}}_i|| \cdot \nabla_\theta \tilde{\mathbf{X}}_i.$$

As the predicted data points $\tilde{\mathbf{X}}_i$ begin to approximate the observations $\mathbf{X}_i$ more accurately, the residual distance $\|\mathbf{X}_i - \tilde{\mathbf{X}}_i\|$ reduces further. As a result, the loss term $\mathcal{L}_{\text{MSE}}(\mathbf{X}, \tilde{\mathbf{X}})$ is forced to approach its infimum, leaving little room for parameter optimization with each subsequent iteration of model training.

$$\|\mathbf{X}_i - \tilde{\mathbf{X}}_i\| \to 0 \quad \Longrightarrow \quad \mathcal{L}_{\text{MSE}}(\mathbf{X}, \tilde{\mathbf{X}}) \to 0 \quad \Longrightarrow \quad \nabla_\theta \mathcal{L}_{\text{MSE}}(\mathbf{X}, \tilde{\mathbf{X}}) \to 0.$$

This behavior occurs because the residual distance $\|\mathbf{X}_i - \tilde{\mathbf{X}}_i\|$ directly influences the gradient magnitude. As $\tilde{\mathbf{X}}_i$ approaches $\mathbf{X}_i$ the gradient diminishes, leading to smaller updates during optimization. Therefore, the $\mathcal{L}_{\text{MSE}}(\mathbf{X}, \tilde{\mathbf{X}})$ loss ensures stable optimization through smooth gradients. Its steady convergence as $\tilde{\mathbf{X}}_i \to \mathbf{X}_i$ prevents oscillatory behavior, thus concluding the proof. □

## A.7 Proof of proposition 5.1.3

**Proposition 5.1.3.** The $\mathcal{L}_{\text{KLD}}(\mathbf{X}, \tilde{\mathbf{X}})$ regularization imposes a statistical prior on $P_{\mathbf{G_A}}(\tilde{\mathbf{X}})$, ensuring that the learned distribution remains close to a predefined Gaussian. Moreover, it enhances optimization stability, particularly under additive Gaussian noise, by preventing $P_{\mathbf{G_A}}(\tilde{\mathbf{X}})$ from deviating excessively from a normal distribution, mitigating erratic behavior. By complementing adversarial and MSE losses, it ensures both the alignment and smoothness of $P_{\mathbf{G_A}}(\tilde{\mathbf{X}})$.

*Proof.* The application of the $\mathcal{L}_{\text{KLD}}(\mathbf{X}, \tilde{\mathbf{X}})$ loss term enforces the Gaussianity of the

probability distribution $P(\mathcal{Z}_j|Pa_j)$ from which the residual noise $\mathcal{Z}_j$ is sampled. The noise terms belonging to this conditional distribution can be defined as follows: $\mathcal{Z}_j = X_j - f_j(Pa_j)$. Therefore, converging on the global optimum of $\mathcal{L}_{\text{KLD}}(\mathbf{X}, \tilde{\mathbf{X}})$ guarantees that DAGAF models the causal mechanism $f_j$ in a way that ensures $\mathcal{Z}_j \sim P(\mathcal{Z}_j|Pa_j) \approx \mathcal{N}(0, \sigma_j^2)$. Enforcing the Gaussianity of $\mathcal{Z}_j$ guarantees that the observed deviations from the functional relationship $X_j = f_j(Pa_j)$ follow the Gaussian noise assumption, which is crucial for causal discovery under both the ANM and the PNL assumptions.

Consider $\mathcal{L}_{\text{KLD}}(\mathbf{X}, \tilde{\mathbf{X}})$ as a regularizer, penalizing the model when the conditional residual noise distribution $P(\mathcal{Z}_j|Pa_j)$ deviates from the standard normal distribution $\mathcal{N}(0, \sigma_j^2)$. Additionally, express the gradient for $\mathcal{L}_{\text{KLD}}(\mathbf{X}, \tilde{\mathbf{X}})$ w.r.t. $\mathbf{G_A}$ as the following:

$$\nabla_{\mathbf{G_A}} \mathcal{L}_{\text{KLD}}(\mathbf{X}, \tilde{\mathbf{X}}) = \sum_{j=1}^{d} \mathbb{E}_{Pa_j} \left[ \nabla_{\mathbf{G_A}} \log \frac{P(\mathcal{Z}_j \mid Pa_j)}{\mathcal{N}(\mathcal{Z}_j; 0, \sigma_j^2)} \right].$$

From the above equation, it is evident that the term $\log \mathcal{N}(\mathcal{Z}_j; 0, \sigma_j^2)$ is quadratic in $\mathcal{Z}_j$, yielding a smooth gradient $\nabla_{\mathbf{G_A}} \mathcal{L}_{\text{KLD}}(\mathbf{X}, \tilde{\mathbf{X}})$ that is robust against minor perturbations in $\mathbf{G_A}$. This limits overfitting to the noise present in $X_j$ and stabilizes the modeling of $f_j$. As a result, the $\mathcal{L}_{\text{KLD}}(\mathbf{X}, \tilde{\mathbf{X}})$ term enhances the overall stability of the model by aligning the implicitly generated distribution $P_{\mathbf{G_A}}(\tilde{\mathbf{X}})$ with a normal (Gaussian) distribution.

The $\mathcal{L}_{\text{KLD}}(\mathbf{X}, \tilde{\mathbf{X}})$ term can also be applied to other components of the objective function used in the training process of DAGAF. For example, the adversarial loss $\mathcal{L}_{\text{adv}}(\mathbf{X}, \tilde{\mathbf{X}})$ encourages overlap between $P(\mathbf{X})$ and $P_{\mathbf{G_A}}(\tilde{\mathbf{X}})$, however, it does not explicitly impose the additive Gaussian assumption. On the other hand, the reconstruction loss $\mathcal{L}_{\text{MSE}}(\mathbf{X}, \tilde{\mathbf{X}})$ minimizes the distance between the original $\mathbf{X}_i$ and the synthetic $\tilde{\mathbf{X}}_i$ data points individually, but fails to take into consideration the statistical properties of $\mathcal{Z}_j$. The KLD regularization term $\mathcal{L}_{\text{KLD}}(\mathbf{X}, \tilde{\mathbf{X}})$ directly imposes a Gaussian structure on $\mathcal{Z}_j$, ensuring that it adheres to the additive Gaussian assumption. This constraint discourages $f_j$ from overfitting to non-Gaussian noise, thus completing the proof. □

## A.8 Proof of proposition 5.1.4

**Proposition 5.1.4.** Minimizing the Maximum Mean Discrepancy (MMD) loss term $\mathcal{L}_{\text{MMD}}(\mathbf{X}, \tilde{\mathbf{X}})$ encourages the alignment of higher-order moments between the input distribution $P(\mathbf{X})$ and the synthetic distribution $P_{\mathbf{G_A}}(\tilde{\mathbf{X}})$, which supports the adversarial loss in achieving overall distributional alignment.

*Proof.* Based on the MMD definition provided in Equation (5.4), the author gives the gradient of the term $\mathcal{L}_{\text{MMD}}(\mathbf{X}, \tilde{\mathbf{X}})$ w.r.t. $\theta$, which is used to define $\mathbf{G_A}$, as follows:

$$\begin{aligned}\nabla_\theta \mathcal{L}_{\text{MMD}}(\mathbf{X}, \tilde{\mathbf{X}}) = 2(&\mathbb{E}_{\tilde{\mathbf{X}} \sim P_{\mathbf{G_A}}(\tilde{\mathbf{X}})}[\nabla_\theta k(\tilde{\mathbf{X}}_i, \tilde{\mathbf{X}}_j)] \\ &- \mathbb{E}_{\mathbf{X} \sim P(\mathbf{X}), \tilde{\mathbf{X}} \sim P_{\mathbf{G_A}}(\tilde{\mathbf{X}})}[\nabla_\theta k(\mathbf{X}_i, \tilde{\mathbf{X}}_j)]),\end{aligned}$$

where $\tilde{\mathbf{X}} \sim P_{\mathbf{G_A}}(\tilde{\mathbf{X}})$ represents samples drawn from the model-generated distribution, while $\mathbf{X} \sim P(\mathbf{X})$ denotes samples from the true distribution. The function $k(\mathbf{X}, \tilde{\mathbf{X}})$ serves as a positive-definite kernel, commonly selected as a Gaussian kernel or another characteristic kernel (e.g., RBF or Polynomial).

The kernel $k(\mathbf{X}, \tilde{\mathbf{X}})$ inherently encodes the higher-order statistics of both the true distribution $P(\mathbf{X})$ and the synthetic data distribution $P_{\mathbf{G_A}}(\tilde{\mathbf{X}})$. The function achieves this by encouraging internal consistency within the implicitly generated model distribution, as evidenced by the third term in $\mathcal{L}_{\text{MMD}}(\mathbf{X}, \tilde{\mathbf{X}})$, $\mathbb{E}_{\tilde{\mathbf{X}} \sim P_{\mathbf{G_A}}(\tilde{\mathbf{X}})}[k(\tilde{\mathbf{X}}_i, \tilde{\mathbf{X}}_j)]$, which aligns the fake data points $\tilde{\mathbf{X}}_i$ and $\tilde{\mathbf{X}}_j$ to ensure that their higher-order moments are consistent. Additionally, the kernel function facilitates alignment with the true distribution through the second term, $\mathbb{E}_{\mathbf{X} \sim P(\mathbf{X}), \tilde{\mathbf{X}} \sim P_{\mathbf{G_A}}(\tilde{\mathbf{X}})}[k(\mathbf{X}_i, \tilde{\mathbf{X}}_j)]$.

The loss term $\mathcal{L}_{\text{MMD}}(\mathbf{X}, \tilde{\mathbf{X}})$ directly targets higher-order differences using kernel-induced feature mappings $k(.)$. This mechanism complements the adversarial loss by ensuring that both general and detailed aspects of $P(\mathbf{X})$ and $P_{\mathbf{G_A}}(\tilde{\mathbf{X}})$ are matched. Therefore, the combination of $\mathcal{L}_{\text{MMD}}(\mathbf{X}, \tilde{\mathbf{X}})$ and $\mathcal{L}_{\text{adv}}(\mathbf{X}, \tilde{\mathbf{X}})$ provides a robust framework for distributional alignment, addressing both large-scale discrepancies and higher-order mismatches, thus completing the proof.

□

## A.9 Proof of proposition 5.1.5

**Proposition 5.1.5.** Assuming the Additive Noise Model (ANM), Linear non-Gaussian Acyclic Model (LiNGAM), or Post-Nonlinear Model (PNL), there is a unique DAG $\mathbf{G_A^0}$ that defines the observed joint distribution $P(\mathbf{X})$.

*Proof.* The proof for this proposition constitutes two different derivations due to fundamental differences between the assumptions involved in the definition of the causal models used to perform causal discovery under the DAGAF framework. As a result, the author proceeds to first investigate the LiNGAM and ANM cases, and afterwords addresses the PNL case.

**Lemma A.9.1.** Assuming either the additive noise model (ANM) or the linear non-Gaussian acyclic model (LiNGAM) condition holds, the ground-truth directed acyclic graph (DAG) $\mathbf{G_A^0}$ can be uniquely determined from the distribution $P(\mathbf{X})$.

$$P(\mathbf{X}) \neq P'(\mathbf{X}) \implies \mathbf{G_A^0} \neq \mathbf{G'^0_{A'}}.$$

*Proof.* Consider a dataset $\chi$ comprising data attributes $X = \{X_1, ..., X_d\}$, where each attribute $X_j$ is produced under either the ANM or LiNGAM assumption, as represented by the following equation:

$$X_j := f_j(Pa_j) + \mathcal{Z}_j.$$

Each function $f_j : \mathbb{R}^d \to \mathbb{R}$ is deterministic (e.g., nonlinear in the ANM case and linear for the LiNGAM scenario). Additionally, the noise terms $\mathcal{Z}_j \sim P(\mathcal{Z})$ are independent - non-Gaussian in LiNGAM and Gaussian in ANM, while $Pa_j$ denotes the set of immediate parents of $X_j$ in the DAG.

In both ANM and LiNGAM, the independence of the noise term $\mathcal{Z}_j$ from the parent set $Pa_j$ is fundamental, expressed as $\mathcal{Z}_j \perp\!\!\!\perp Pa_j$. This independence in the true DAG $\mathbf{G_A^0}$ places significant restrictions on the functional relationships between the causal mechanisms within $\mathbf{G_A^0}$:

$$P(\mathcal{Z}_j) = P_{\mathcal{Z}_j}(X_j - f_j(Pa_j)).$$

Assuming $\mathbf{G'^0_{A'}}$ is different from $\mathbf{G^0_A}$, then the functions $f'_j$ used to define $\mathbf{G'^0_{A'}}$ must satisfy the following conditions:

$$P(\mathcal{Z}'_j) = P_{\mathcal{Z}'_j}(X_j - f'_j(Pa'_j)),$$

However, if $\mathbf{G'^0_{A'}}$ differs from $\mathbf{G^0_A}$, then the corresponding causal mechanism functions $f'_j$ will not match their true counterparts $f_j$ present in the ground-truth DAG. Moreover, the new noise terms $\mathcal{Z}'_j$ will lose their independence from their parent sets $Pa'_j$, since that independence is unique to the actual causal structure in $\mathbf{G^0_A}$. Therefore, $\mathbf{G'^0_{A'}}$ cannot simultaneously satisfy the same independence conditions as $\mathbf{G^0_A}$, leading to a contradiction.

Hence, given the ANM with nonlinear functions and independent noise or the LiNGAM model with linear functions and non-Gaussian noise, no alternative DAG $\mathbf{G'^0_{A'}}$ distinct from $\mathbf{G^0_A}$ can generate the same observational data distribution $P(\mathbf{X})$. Therefore, this confirms that the true DAG $\mathbf{G^0_A}$ is uniquely identifiable from $P(\mathbf{X})$, thus completing the proof. □

Next, the author provides theoretical analysis regarding the identifiability of the Post-Nonlinear (PNL) assumption.

**Lemma A.9.2.** Assuming the Post-Nonlinear (PNL) causal model assumption holds, there exists a uniquely identifiable DAG $\mathbf{G^0_A}$ that produces the joint distribution observed for the data variables $\{X_1, ..., X_d\}$.

*Proof.* Let $\chi$ be a data set consisting of data attributes $\{X_1, ..., X_d\}$, where each $X_j$ is associated with a set of parent nodes $Pa_j$ and an independent Gaussian noise term $\mathcal{Z}_j$, satisfying $\mathcal{Z}_j \perp\!\!\!\perp Pa_j$. Moreover, causal mechanisms represented by nonlinear functions $f_j$ are applied to model parent contributions, while $g_j$ denotes a nonlinear function applied post-summation:

$$X_j := g_j(f_j(Pa_j) + \mathcal{Z}_j), \forall j, \mathcal{Z}_j \perp\!\!\!\perp f_j(Pa_j), \mathcal{Z}_j \sim \mathcal{N}(\mu, \sigma_j^2).$$

Additionally, assume that the input $\rho_j$ to the post-nonlinear function $g_j$ is expressed as follows:

$$\rho_j = f_j(Pa_j) + \mathcal{Z}_j.$$

Assuming that $Pa_j$ is the correct set of parent nodes and the function $g_j$ does not affect the independence structure. Then, the residual noise $\mathcal{Z}_j$ remains independent of the parent variables, which is formally expressed as:

$$\mathcal{Z}_j \perp\!\!\!\perp Pa_j.$$

Within the Post-Nonlinear causal model, the statistical connection between $X_j$, its parent nodes $Pa_j$, and the residual noise $\mathcal{Z}_j$ exhibits specific invariances. In particular, the conditional probability $P(X_j|Pa_j)$ (which is determined by the PNL structure) and the marginal probability $P(Pa_j)$ together define the joint distribution as follows:

$$P(X_j, Pa_j) = P(X_j|Pa_j)P(Pa_j).$$

Now, consider an alternative parent set $Pa'_j$ that does not match the true set $Pa_j$. For this incorrect set, the residual noise $\mathcal{Z}_j$ is computed by subtracting the function $f_j(Pa'_j)$ from $\rho_j$, expressed as:

$$\mathcal{Z}_j = \rho_j - f_j(Pa'_j).$$

In this scenario, the fundamental independence condition $\mathcal{Z}_j \perp\!\!\!\perp Pa'_j$ does not hold. As a result, when the parent set is defined incorrectly, the residual noise $\mathcal{Z}_j$ becomes statistically dependent on the variables in $Pa'_j$. This dependency means that the conditional distribution $P(X_j|Pa'_j)$ cannot preserve the same invariance because of introduced dependencies, thus completing the proof.

□

The proofs above demonstrate that under each of the three causal model assumptions (LiNGAM, ANM, and PNL) a given probability distribution can be represented by only one unique DAG, thus concluding the proof.

□

# Appendix B

# Data quality evaluation notebook

In this section, we analyse the quality of the synthetic data generated by the model. We conduct the following tests:

1. **Statistical properties**: We compare the closeness-of-fit between the real and synthetic data distributions using boxplot analysis, marginal distributions and principal component analysis. We additionally compute the correlation matrices across both sets of data to study the interdependencies between the covariates.

2. **Machine learning regression**: We train separate Random forest regressors on the real and synthetic datasets and compare their corresponding regression performances. We additionally plot the feature importances using permutations.

In [1]:
```
import pickle
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sb
import plotly.express as px
from scipy import stats
from sklearn.ensemble import RandomForestRegressor
from sklearn.inspection import permutation_importance
from sklearn.model_selection import train_test_split
from sklearn.metrics import mean_squared_error,
    r2_score
```

```
from sklearn.decomposition import PCA
from sklearn.preprocessing import StandardScaler
```

In [2]:
```
#load in the real dataset
var_names = ['x1', 'x2','x3', 'x4', 'x5', 'x6', 'x7', 'x8', 'x9', 'x10']
''' if data stored as a .csv file
'''
fname_real = './real_data.csv'
real_df = pd.read_csv(fname_real, names=var_names)
real_df.drop(index=0, inplace=True)
real_df['data'] = 'real'

''' if data stored as DataLoader
'''
# with open(r"./train_loader.pkl", "rb") as input_file:
#     train_loader = pickle.load(input_file)

# real_tensor_data = train_loader.dataset.tensors[0].squeeze()
# real_df = pd.DataFrame(real_tensor_data.numpy(), columns=var_names)
# real_df['data'] = 'real'
display(real_df)
```

In [3]:
```
#load in the fake dataset (of equal size to the real one)
fname_fake = './generated_data.csv'
fake_df = pd.read_csv(fname_fake, names=var_names)
fake_df.drop(index=0, inplace=True)
fake_df['data'] = 'fake'
display(fake_df)
```

In [4]:
```
#combine both the real and fake data into a single dataset to allow comparisons
df_all = pd.concat([real_df, fake_df], ignore_index=True)
```

```
display(df_all)
df_all.to_csv('./real_&_fake_dataframe.csv')
```

In [5]:

```
#given the column names for the covariates we want to visualise, plot the real vs fake and run a t-test on them.
def show_data_properties(df, x_var, y_var, plot_type, key='data'):
    '''
    description
    @author: calmac
    @date: 16.05.23
    '''
    # get features of interest
    xreal, yreal = df[x_var].loc[df[key]=='real'], df[y_var].loc[df[key]=='real']
    xfake, yfake = df[x_var].loc[df[key]=='fake'], df[y_var].loc[df[key]=='fake']

    # run a t-test between real/fake features
    ttest_x = stats.mannwhitneyu(xreal, xfake)
    ttest_y = stats.mannwhitneyu(yreal, yfake)
    print('Mann-Whitney U-test (real vs fake):')
    print('\t{}: p-value={}'.format(x_var, ttest_x[1]))
    print('\t{}: p-value={}'.format(y_var, ttest_y[1]))

    # visualise results
    fig = px.scatter(df, x=x_var, y=y_var,
                     marginal_x=plot_type, marginal_y=plot_type,
                     color=key, width=800, height=800,
                     trendline='ols'
                     )
    fig.update_layout(legend=dict(
        yanchor='top', y=0.95,
        xanchor='right', x=0.9
    ))
```

```
29
30      fig.show()
```

In [6]:
```
x_var = 'x3'
y_var = 'x4'
plot_type = 'box'

show_data_properties(df_all, x_var, y_var, plot_type)
```

In [7]:
```
#we alternatively plot correlation matrices to
    visualise the dependencies between all covariates.
def plot_correlation(data, names, cmap='Blues',
    annotations=False):
    '''
    Description

    @author: calmac
    @date: 16.05.23
    '''
    data = pd.DataFrame(data, columns=names)
    corr = data.corr()
    sb.heatmap(corr, cmap=cmap, annot=annotations,
    xticklabels=names, yticklabels=names)
    plt.show()
```

In [8]:
```
#real data:correlation matrix
plot_correlation(real_df.iloc[:,:-1], real_df.columns
    [:-1])
```

In [9]:
```
#fake data:correlation matrix
plot_correlation(fake_df.iloc[:,:-1], fake_df.columns
    [:-1])
```

In [10]:
```
#machine learning regression
def runRandomForestRegression(real_data, fake_data,
    outcome, seed=42, feature_importance=True,
    bootstrap=False):
```

```
    '''
    Description

    @author: calmac
    @date: 16.05.23
    '''

    # find all features except the outcome: these
    become the covariates
    feature_names = real_data.columns[~real_data.
    columns.isin([outcome])]
    print('Outcome of interest: {}'.format(outcome))
    print('Covariates: {}'.format(list(feature_names)))

    ''' Random forest regressor fit to the REAL data
    '''
    # real data
    Xr, yr = real_data[feature_names], real_data[
    outcome]
    Xr_train, Xr_test, yr_train, yr_test =
    train_test_split(Xr, yr, test_size=0.1,
    random_state=seed)

    # real model
    model_real = RandomForestRegressor(n_estimators
    =1000, max_depth=5, random_state=seed)
    model_real.fit(Xr_train, yr_train)

    ''' Random forest regressor fit to the FAKE data
    '''
    # fake data
    Xf, yf = fake_data[feature_names], fake_data[
    outcome]
    Xf_train, Xf_test, yf_train, yf_test =
    train_test_split(Xf, yf, test_size=0.1,
    random_state=seed)

```

```
    # fake model
    model_fake = RandomForestRegressor(n_estimators
    =1000, max_depth=5, random_state=seed)
    model_fake.fit(Xf_train, yf_train)

    '''
    Given two trained RF models (one trained on real,
    the other on fake data)
    run both models on the same test set of real data
    and compare their performances.
    '''
    yr_pred = model_real.predict(Xr_test)
    print('Results (real):')
    print('\tR2 score: {}'.format(r2_score(yr_test,
    yr_pred)))
    print('\tMSE: {}'.format(mean_squared_error(yr_test
    , yr_pred)))

    yf_pred = model_fake.predict(Xr_test)
    print('Results (fake):')
    print('\tR2 score: {}'.format(r2_score(yr_test,
    yf_pred)))
    print('\tMSE: {}'.format(mean_squared_error(yr_test
    , yf_pred)))


    if feature_importance:
        '''
        Now examine whether the real and fake RFs use
    similar features to
        predict the outcome of interest.
        '''
        # Real
        result_real = permutation_importance(model_real
    , Xr_test, yr_test, n_repeats=10, random_state=seed
    )
        sorted_imp_idx_real = result_real.
```

```
    importances_mean.argsort()
58      imp_real = pd.DataFrame(
59          data = result_real.importances[
    sorted_imp_idx_real].T,
60          columns = feature_names[sorted_imp_idx_real
    ]
61      )
62      ax = imp_real.plot.box(vert=False, whis=10)
63      ax.set_title('Feature importance using
    permutations (real)')
64      ax.axvline(x=0, color='k', linestyle='--')
65      ax.set_xlabel(r'Decrease in $R^2$')
66      ax.figure.tight_layout()
67      plt.show()
68
69      # Fake
70      result_fake = permutation_importance(model_fake
    , Xr_test, yr_test, n_repeats=10, random_state=seed
    )
71      sorted_imp_idx_fake = result_fake.
    importances_mean.argsort()
72      imp_fake = pd.DataFrame(
73          data = result_fake.importances[
    sorted_imp_idx_fake].T,
74          columns = feature_names[sorted_imp_idx_fake
    ]
75      )
76      ax = imp_fake.plot.box(vert=False, whis=10)
77      ax.set_title('Feature importance using
    permutations (fake)')
78      ax.axvline(x=0, color='k', linestyle='--')
79      ax.set_xlabel(r'Decrease in $R^2$')
80      ax.figure.tight_layout()
81      plt.show()
```

In [11]:

```
1 outcome = 'x1'
2 runRandomForestRegression(real_data=real_df.iloc[:,
```

```
    :-1],
fake_data=fake_df.iloc[:, :-1], outcome=outcome)
```

In [12]:

```
features_std = StandardScaler().fit_transform(df_all.
    iloc[:, :-2])

pca = PCA(n_components=2)
principalComponents = pca.fit_transform(features_std)

pca_df = pd.DataFrame(data=principalComponents, columns
    =['PC1', 'PC2'])
pca_df['data'] = df_all['data']
pca_df

plt.figure(figsize=(8, 6))
for label, color in zip(['real', 'fake'], ['blue', '
    orange']):
    subset = pca_df[pca_df['data'] == label]
    plt.scatter(subset['PC1'], subset['PC2'], label=
    label, alpha=0.7, color=color)

plt.title('PCA Comparison of Original and Synthetic
    Data')
plt.xlabel('Principal Component 1')
plt.ylabel('Principal Component 2')
plt.legend()
plt.show()
```

# Bibliography


[1] C. E. Rasmussen and C. K. I. Williams, *Gaussian processes for machine learning.* MIT Press, 2005.

[2] Aristotle, *Metaphysics*, W. D. Ross, Ed. The Internet Classics Archive, 350BCE.

[3] Aristotle, E. S. Forster, and H. Tredennick, *Posterior Analytics/Topica.* Harvard University Press, Cambridge, Mass., 1960.

[4] A. B. S. Hill, "The environment and disease: Association or causation?" *Journal of the Royal Society of Medicine*, vol. 58, pp. 295 – 300, 1965.

[5] T. C. Chalmers, H. Smith, B. Blackburn, B. Silverman, B. Schroeder, D. Reitman, and A. Ambroz, "A method for assessing the quality of a randomized control trial." *Controlled clinical trials*, vol. 21, pp. 31–49, 1981.

[6] W. G. Jennings, *Experimental Criminology.* New York, NY: Springer, 2014.

[7] J. Loong, "Applying the bradford hill criteria to economics and policy," https://joshualoong.com/2020/07/21/applying-the-bradford-hill-criteria-to-economics-and-policy/, 2020.

[8] J. Howick, P. Kelly, and M. P. Kelly, "Establishing a causal link between social relationships and health using the bradford hill guidelines," *SSM - Population Health*, vol. 8, 2019.

[9] J. D. Sargent, S. Cukier, and T. F. Babor, "Alcohol marketing and youth drinking: Is there a causal relationship, and why does it matter?" *Journal of Studies on Alcohol and Drugs. Supplement*, pp. 5 – 12, 2020.

[10] J. Pearl, *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Los Angeles, California: Morgan Kaufmann Publishers, 1988.

[11] P. Spirtes, C. Glymour, and R. Scheines, *Computation, Causation, and Discovery*. Cambridge, Massachusetts: AAAI Press, 1985.

[12] X. Zheng, B. Aragam, P. Ravikumar, and E. P. Xing, "Dags with no tears: Continuous optimization for structure learning." *Conference on Neural Information Processing Systems*, March 2018.

[13] J. Aldrich, "Correlations genuine and spurious in pearson and yule," *Statistical Science*, vol. 10, pp. 364–376, 1995.

[14] E. R. Tufte, "The cognitive style of powerpoint: Pitching out corrupts within," in *Graphics Pr*, 2003.

[15] C. Xu, S. M. Brown, and C. E. Grant, "Detecting simpson's paradox," in *FLAIRS*, 2018.

[16] G. Shmueli and I. Yahav, "Tackling simpson's paradox in big data using classification & regression trees," in *ECIS*, 2014.

[17] G. Imbens and D. B. Rubin, *Causal Inference for Statistics, Social, and Biomedical Sciences: An Introduction*. Cambridge University Press, 2015.

[18] D. Lopez-Paz, K. Muandet, B. Scholkopf, and I. O. Tolstikhin, "Towards a learning theory of cause-effect inference," in *International Conference on Machine Learning*, 2015.

[19] J. M. Mooij, J. Peters, D. Janzing, J. Zscheischler, and B. Scholkopf, "Distinguishing cause from effect using observational data: Methods and benchmarks," *ArXiv*, vol. abs/1412.3773, 2014.

[20] S. Ott, S. Imoto, and S. Miyano, "Finding optimal models for small gene networks," *In Pacific symposium on biocomputing*, February 2004.

[21] K. Sachs, O. D. Perez, D. Pe'er, D. A. Lauffenburger, and G. P. Nolan, "Causal protein-signaling networks derived from multiparameter single-cell data," *Science*, vol. 308, pp. 523 – 529, 2005.

[22] D. M. Chickering, D. Heckerman, and C. Meek, "Large sample learning of bayesian networks is np-hard." *Journal of Machine Learning Research*, vol. 5, p. 1287–1330, 2004.

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

[24] J. Zhang, "On the completeness of orientation rules for causal discovery in the presence of latent confounders and selection bias," *Artificial Intelligence*, vol. 172 (16-17), pp. 1873–1896, 2008.

[25] D. Heckerman, D. Geiger, and D. M. Chickering, "Learning bayesian networks: The combination of knowledge and statistical data." *Machine Learning*, vol. 20, p. 197–243, 1995.

[26] J. Kuipers, G. Moffa, and D. Heckerman, "Addendum on the scoring of gaussian directed acyclic graphical models." *The Annals of Statistics*, vol. 42 (4), p. 1689–1691, 2014.

[27] R. Bouckaert, "Probabilistic network construction using the minimum description length principle." *European conference on symbolic and quantitative approaches to reasoning and uncertainty*, p. 41–48, 1993.

[28] P. L. Spirtes, C. Glymour, and R. Scheines, *Causation, Prediction, and Search, 2nd Edition.* Cambridge, Massachusetts: MIT press, 2001.

[29] P. Spirtes, C. Glymour, R. Scheines, S. A. Kauffman, V. Aimale, and F. C. Wimberly, "Constructing bayesian network models of gene expression networks from microarray data," in *Atlantic Symposium on Computational Biology*, 2000.

[30] R. D. Shah and J. Peters, "The hardness of conditional independence testing and the generalised covariance measure," *The Annals of Statistics*, 2018.

[31] S. Shimizu, P. O. Hoyer, A. Hyvärinen, and A. J. Kerminen, "A linear non-gaussian acyclic model for causal discovery," *J. Mach. Learn. Res.*, vol. 7, pp. 2003–2030, 2006.

[32] K. Zhang and A. Hyvärinen, "On the identifiability of the post-nonlinear causal model," in *Conference on Uncertainty in Artificial Intelligence*, 2009.

[33] H. Ma, K. Aihara, and L. Chen, "Detecting causality from nonlinear dynamics with short-term time series," *Scientific Reports*, vol. 4, 2014.

[34] I. Tsamardinos, L. E. Brown, and C. F. Aliferis, "The max-min hill-climbing bayesian network structure learning algorithm," *Machine Learning*, vol. 65 (1), p. 31–78, 2006.

[35] Y. Yu, J. Chen, T. Gao, and M. Yu, "Dag-gnn: Dag structure learning with graph neural networks," *International Conference on Machine Learning*, April 2019.

[36] S. Lachapelle, P. Brouillard, T. Deleu, and S. Lacoste-Julien, "Gradient-based neural dag learning," *ArXiv*, vol. abs/1906.02226, 2020.

[37] Y. Gao, L. Shen, and S.-T. Xia, "Dag-gan: Causal structure learning with generative adversarial nets," *ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 3320–3324, 2021.

[38] X. Zheng, C. Dan, B. Aragam, P. Ravikumar, and E. P. Xing, "Learning sparse nonparametric dags," *ArXiv*, vol. abs/1909.13189, 2020.

[39] I. Ng, S. Zhu, Z. Chen, and Z. Fang, "A graph autoencoder approach to causal structure learning," *ArXiv*, vol. abs/1911.07420, 2019.

[40] I. Ng, A. Ghassami, and K. Zhang, "On the role of sparsity and dag constraints for learning linear dags," *ArXiv*, vol. abs/2006.10201, 2020.

[41] K. Uemura and S. Shimizu, "Estimation of post-nonlinear causal models using autoencoding structure," *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 3312–3316, 2020.

[42] Y. Chung, "Post-nonlinear causal model with deep neural networks," in *Carnegie Mellon University*, 2019.

[43] S. Zhu, I. Ng, and Z. Chen, "Causal discovery with reinforcement learning," *ArXiv*, vol. abs/1906.04477, 2019.

[44] V. Zantedeschi, L. Franceschi, J. Kaddour, M. J. Kusner, and V. Niculae, "Dag learning on the permutahedron," *ArXiv*, vol. abs/2301.11898, 2023.

[45] F. Liu, W. Ma, A. Zhang, X. Wang, Y. Duan, and T.-S. Chua, "Discovering dynamic causal space for dag structure learning," *ArXiv*, vol. abs/2306.02822, 2023.

[46] J. Renero, I. Ochoa, and R. Maestre, "Rex: Causal discovery based on machine learning and explainability techniques," *ArXiv*, vol. abs/2501.12706, 2025.

[47] D. Bertsekas, *Nonlinear Programming*. 2nd edition: Athena Scientific, 1999.

[48] N. Y. Yue Yu, Tian Gao and Q. Ji, "Dags with no curl: An efficient dag structure learning approach," *Proceedings of the 38th International Conference on Machine Learning*, 2021.

[49] B. Charpentier, S. Kibler, and S. Günnemann, "Differentiable dag sampling," *ArXiv*, vol. abs/2203.08509, 2022.

[50] N. Yin, Y. Yu, T. Gao, and Q. Ji, "Efficient nonlinear dag learning under projection framework," in *International Conference on Pattern Recognition*, 2024.

[51] G. Papamakarios, E. T. Nalisnick, D. J. Rezende, S. Mohamed, and B. Lakshminarayanan, "Normalizing flows for probabilistic modeling and inference," *J. Mach. Learn. Res.*, vol. 22, pp. 57:1–57:64, 2019.

[52] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," *CoRR*, vol. abs/1312.6114, 2013.

[53] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. C. Courville, and Y. Bengio, "Generative adversarial nets," *NIPS*, 2014.

[54] Z. Chang, G. A. Koulieris, and H. P. H. Shum, "On the design fundamentals of diffusion models: A survey," *ArXiv*, vol. abs/2306.04542, 2023.

[55] R. J. Rossi, *Mathematical Statistics*. New York: New York: John Wiley & Sons., 2018.

[56] M. Zhao, Y. Cong, S. Dai, and L. Carin, "Bridging maximum likelihood and adversarial learning via $\alpha$-divergence," *arXiv: Learning*, 2020.

[57] L. V. Kantorovich, "Mathematical methods of organizing and planning production," *Management Science*, vol. 6, pp. 366–422, 1960.

[58] S. Kullback and R. A. Leibler, "On information and sufficiency," *Annals of Mathematical Statistics*, vol. 22, pp. 79–86, 1951.

[59] D. M. Endres and J. E. Schindelin, "A new metric for probability distributions," *IEEE Transactions on Information Theory*, vol. 49, pp. 1858–1860, 2003.

[60] F. Huszár, "How (not) to train your generative model: Scheduled sampling, likelihood, adversary?" *ArXiv*, vol. abs/1511.05101, 2015.

[61] D. P. Kingma and M. Welling, "An introduction to variational autoencoders," *ArXiv*, vol. abs/1906.02691, 2019.

[62] A. B. L. Larsen, S. K. Sønderby, H. Larochelle, and O. Winther, "Autoencoding beyond pixels using a learned similarity metric," *ArXiv*, vol. abs/1512.09300, 2015.

[63] D. Kalainathan, O. Goudet, I. R. Subramanian, D. Lopez-Paz, and M. Sebag, "Structural agnostic modeling: Adversarial learning of causal graphs," *arXiv: Machine Learning*, 2018.

[64] Y. Gao and Q. Cai, "A wgan-based missing data causal discovery method," *2023 4th International Conference on Big Data, Artificial Intelligence and Internet of Things Engineering (ICBAIE)*, pp. 136–139, 2023.

[65] K. Uemura, T. Takagi, K. Takayuki, H. Yoshida, and S. Shimizu, "A multivariate causal discovery based on post-nonlinear model," in *CLEaR*, 2022.

[66] G. Keropyan, D. Strieder, and M. Drton, "Rank-based causal discovery for post-nonlinear models," *ArXiv*, vol. abs/2302.12341, 2023.

[67] I. O. Tolstikhin, B. K. Sriperumbudur, and B. Schölkopf, "Minimax estimation of maximum mean discrepancy with radial kernels," *NIPS*, 2016.

[68] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Köpf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala, "Pytorch: An imperative style, high-performance deep learning library," 2019, paper presented at 33rd Conference on Neural Information Processing Systems, Vancouver, Canada.

[69] K. Zhang, Z. Wang, J. Zhang, and B. Schölkopf, "On estimation of functional causal models: general results and application to the post-nonlinear causal model," *ACM Transactions on Intelligent Systems and Technology (TIST)*, vol. 7, no. 2, pp. 1–22, 2015.

[70] K. Zhang, Z. Wang, and B. Schölkopf, "On estimation of functional causal models: post-nonlinear causal model as an example," in *2013 IEEE 13th International Conference on Data Mining Workshops*, 2013, pp. 139–146.

[71] K. Zhang and A. Hyvärinen, "Nonlinear functional causal models for distinguishing cause from effect," *Statistics and causality: Methods for applied empirical research*, pp. 185–201, 2016.

[72] K. Zhang, J. Zhang, B. Huang, B. Schölkopf, and C. Glymour, "On the identifiability and estimation of functional causal models in the presence of outcome-dependent selection." in *UAI*, 2016.

[73] O. Goudet, D. Kalainathan, P. Caillou, I. M. Guyon, D. Lopez-Paz, and M. Sebag, "Causal generative neural networks," *arXiv: Machine Learning*, 2017.

[74] Y. Wang, F. Cao, K. Yu, and J. Liang, "Local causal discovery in multiple manipulated datasets," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 34, pp. 7235–7247, 2022.

[75] H. Petkov, C. Hanley, and F. Dong, "Dag-wgan: Causal structure learning with wasserstein generative adversarial networks," *ArXiv*, vol. abs/2204.00387, 2022.

[76] H. Petkov, F. Dong, and C. Hanley, "Causality learning with wasserstein generative adversarial networks," *ArXiv*, vol. abs/2206.01496, 2022.

[77] H. H. Petkov and F. Dong, "Efficient generative adversarial dag learning with no-curl," *2023 International Conference Automatics and Informatics (ICAI)*, pp. 164–169, 2023.

[78] C. R. MacLellan, C. McKeag, H. Petkov, F. Dong, D. J. Lowe, R. Maguire, S. Moschoyiannis, J. Armes, S. S. Skene, and C. Sainsbury, "Ai-powered clinical trials: Emulating real-world glp-1 efficacy with synthetic patient populations using causal effect learning," 2023, research poster presented at Diabetes Technology Society in Virtual Diabetes Technology Meeting, Tuesday, November 7th.

[79] H. Petkov, C. MacLellan, and F. Dong, "Dagaf: a directed acyclic generative adversarial framework for joint structure learning and tabular data synthesis," *Applied Intelligence*, 2025.

[80] C. R. MacLellan, H. Petkov, C. McKeag, F. Dong, D. J. Lowe, R. Maguire, S. Moschoyiannis, J. Armes, S. Skene, A. Finlinson *et al.*, "Emulating real-world glp-1 efficacy in type 2 diabetes through causal learning and virtual patients," *PLOS digital health*, 2025.

[81] K. Thulasiraman and M. N. S. Swamy, *Graphs - theory and algorithms.* John Wiley and Sons, 1992.

[82] J. Bang-Jensen and G. Gutin, *Digraphs - theory, algorithms and applications.* Springer Monographs in Mathematics (2nd ed.): Springer-Verlag, 2002.

[83] J. J. Seidel, "Strongly regular graphs with (-1, 1, 0) adjacency matrix having eigenvalue 3," *Linear Algebra and its Applications*, vol. 1, pp. 281–298, 1968.

[84] A. Hauser and P. Bühlmann, "Characterization and greedy learning of interventional markov equivalence classes of directed acyclic graphs," *Journal of machine learning research*, vol. 13 (1), p. 2409–2464, 2011.

[85] B. Gao and Y. Cui, "Learning directed acyclic graphical structures with genetical genomics data," *Bioinformatics*, vol. 31 (24), pp. 3953–60, 2015.

[86] S. Greenland, J. Pearl, and J. M. Robins, "Causal diagrams for epidemiologic research." *Epidemiology*, vol. 10 (1), pp. 37–48, 1999.

[87] M. Joffe, M. Gambhir, M. Chadeau-Hyam, and P. Vineis, "Causal diagrams in systems epidemiology," *Emerging Themes in Epidemiology*, vol. 9, pp. 1 – 1, 2012.

[88] M. Velikova, J. T. van Scheltinga, P. J. Lucas, and M. E. Spaanderman, "Exploiting causal functional relationships in bayesian network modelling for personalised healthcare," *Int. J. Approx. Reason.*, vol. 55, pp. 59–73, 2014.

[89] M. D. Garvey, S. Carnovale, and S. Yeniyurt, "An analytical framework for supply network risk propagation: A bayesian network approach," *Eur. J. Oper. Res.*, vol. 243, pp. 618–627, 2015.

[90] J. Pearl, "Bayesian networks," in *Encyclopedia of Social Network Analysis and Mining. 2nd Ed.*, 1998.

[91] D. Koller and N. Friedman, *Probabilistic Graphical Models.* MIT Press, 2009.

[92] G. Rebane and J. Pearl, "The recovery of causal poly-trees from statistical data," *Int. J. Approx. Reason.*, 1987.

[93] S. Wright, "Correlation and causation," *Journal of Agricultural Research.*, vol. 20, p. 557–585, 1921.

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

[95] M. H. Maathuis, M. Kalisch, and P. Bühlmann, "Estimating high-dimensional intervention effects from observational data," *The Annals of Statistics*, vol. 37, no. 6A, pp. 3133–3164, 2009.

[96] G. Moffa, G. Catone, J. Kuipers, E. Kuipers, D. Freeman, S. Marwaha, B. R. Lennox, M. R. Broome, and P. Bebbington, "Using directed acyclic graphs in epidemiological research in psychosis: an analysis of the role of bullying in psychosis," *Schizophrenia bulletin*, vol. 43, no. 6, pp. 1273–1279, 2017.

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

[98] P. O. Hoyer, D. Janzing, J. M. Mooij, J. Peters, and B. Schölkopf, "Nonlinear causal discovery with additive noise models," in *NIPS*, 2008.

[99] C. Meek, "Strong completeness and faithfulness in bayesian networks," *ArXiv*, vol. abs/1302.4973, 1995.

[100] D. Geiger and J. Pearl, "On the logic of causal models," *ArXiv*, vol. abs/1304.2355, 2013.

[101] T. Verma and J. Pearl, "Equivalence and synthesis of causal models," *Probabilistic and Causal Inference*, 1990.

[102] S. A. Andersson, D. Madigan, and M. D. Perlman, "A characterization of markov equivalence classes for acyclic digraphs," *Annals of Statistics*, vol. 25, pp. 505–541, 1997.

[103] D. M. Chickering, "Learning equivalence classes of bayesian-network structures," *J. Mach. Learn. Res.*, vol. 2, pp. 445–498, 1996.

[104] C. Heinze-Deml, M. H. Maathuis, and N. Meinshausen, "Causal structure learning," in *Annual Review of Statistics and Its Application 5*, 2017.

[105] M. de Jongh and M. J. Druzdzel, "A comparison of structural distance measures for causal bayesian network models," in *Recent Advances in Intelligent Information Systems*, 2009, p. 443–456.

[106] J. Peters and P. Buhlmann, "Structural intervention distance (sid) for evaluating causal graphs," *arXiv: Machine Learning*, 2013.

[107] G. Yu, G. Sapiro, and S. Mallat, "Solving inverse problems with piecewise linear estimators: From gaussian mixture models to structured sparsity," *IEEE Transactions on Image Processing*, vol. 21, no. 5, pp. 2481–2499, 2012.

[108] T. Starner and A. Pentland, "Real-time american sign language recognition from video using hidden markov models," in *Proceedings of International Symposium on Computer Vision - ISCV*, 1995, pp. 265–270.

[109] J. Xie, S.-C. Zhu, and Y. N. Wu, "Learning energy-based spatial-temporal generative convnets for dynamic patterns," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 43, no. 2, pp. 516–531, 2021.

[110] J. J. Hopfield, "Neural networks and physical systems with emergent collective computational abilities." *Proceedings of the National Academy of Sciences of the United States of America*, vol. 79, pp. 2554–2558, 1982.

[111] J. J. Zhao, M. Mathieu, and Y. LeCun, "Energy-based generative adversarial network," *ArXiv*, vol. abs/1609.03126, 2016.

[112] Y. Li, K. Swersky, and R. S. Zemel, "Generative moment matching networks," in *International Conference on Machine Learning*, 2015.

[113] G. K. Dziugaite, D. M. Roy, and Z. Ghahramani, "Training generative neural networks via maximum mean discrepancy optimization," in *Conference on Uncertainty in Artificial Intelligence*, 2015.

[114] E. Levina and P. J. Bickel, "The earth mover's distance is the mallows distance: some insights from statistics," *Proceedings Eighth IEEE International Conference on Computer Vision. ICCV 2001*, vol. 2, pp. 251–256, 2001.

[115] M. Arjovsky, S. Chintala, and L. Bottou, "Wasserstein gan," *ArXiv*, vol. abs/1701.07875, 2017.

[116] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, and A. C. Courville, "Improved training of wasserstein gans," in *Neural Information Processing Systems*, 2017.

[117] M. Drton and M. H. Maathuis, "Structure learning in graphical modeling," *arXiv: Methodology*, 2016.

[118] K. Zhang, J. Peters, D. Janzing, and B. Scholkopf, "Kernel-based conditional independence test and application in causal discovery," *ArXiv*, vol. abs/1202.3775, 2011.

[119] X. Sun, D. Janzing, B. Scholkopf, and K. Fukumizu, "A kernel-based causal learning algorithm," in *International Conference on Machine Learning*, 2007.

[120] R. E. Tillman, D. Danks, and C. Glymour, "Integrating locally learned causal structures with overlapping variables," in *Neural Information Processing Systems*, 2008.

[121] S. Triantafillou, I. Tsamardinos, and I. G. Tollis, "Learning causal structure from overlapping variable sets," in *International Conference on Artificial Intelligence and Statistics*, 2010.

[122] E. V. Strobl, "A constraint-based algorithm for causal discovery with cycles, latent variables and selection bias," *International Journal of Data Science and Analytics*, vol. 8, pp. 33 – 56, 2018.

[123] J. Ramsey, J. Zhang, and P. Spirtes, "Adjacency-faithfulness and conservative causal inference," *ArXiv*, vol. abs/1206.6843, 2006.

[124] K. D. Yang, A. Katoff, and C. Uhler, "Characterizing and learning equivalence classes of causal dags under interventions," in *International Conference on Machine Learning*, 2018.

[125] M. Kocaoglu, A. Jaber, K. Shanmugam, and E. Bareinboim, "Characterization and learning of causal graphs with latent variables from soft interventions," in *Neural Information Processing Systems*, 2019.

[126] A. Jaber and M. Kocaoglu, "Causal discovery from soft interventions with unknown targets: Characterization and learning," in *Neural Information Processing Systems*, 2020.

[127] T. S. Richardson, "A discovery algorithm for directed cyclic graphs," in *Conference on Uncertainty in Artificial Intelligence*, 1996.

[128] A. Hyttinen, P. O. Hoyer, F. Eberhardt, and M. Järvisalo, "Discovering cyclic causal models with latent variables: A general sat-based procedure," *ArXiv*, vol. abs/1309.6836, 2013.

[129] D. Colombo and M. H. Maathuis, "Order-independent constraint-based causal structure learning," *J. Mach. Learn. Res.*, vol. 15, pp. 3741–3782, 2012.

[130] B. Huang, K. Zhang, J. Zhang, J. Ramsey, R. Sanchez-Romero, C. Glymour, and B. Scholkopf, "Causal discovery from heterogeneous/nonstationary data," *J. Mach. Learn. Res.*, vol. 21, pp. 89:1–89:53, 2019.

[131] M. H. Maathuis, M. Kalisch, and P. Buhlmann, "Estimating high-dimensional intervention effects from observational data," *Annals of Statistics*, vol. 37, pp. 3133–3164, 2008.

[132] D. Rothenhäusler, C. Heinze, J. Peters, and N. Meinshausen, "Backshift: Learning causal cyclic graphs from unknown shift interventions," in *Neural Information Processing Systems*, 2015.

[133] S. Triantafillou and I. Tsamardinos, "Constraint-based causal discovery from multiple interventions over overlapping variable sets," *ArXiv*, vol. abs/1403.2150, 2014.

[134] P. Forré and J. M. Mooij, "Constraint-based causal discovery for non-linear structural causal models with cycles and latent confounders," *ArXiv*, vol. abs/1807.03024, 2018.

[135] D. Colombo, M. H. Maathuis, M. Kalisch, and T. S. Richardson, "Learning high-dimensional directed acyclic graphs with latent and selection variables," *Annals of Statistics*, vol. 40, pp. 294–321, 2011.

[136] T. D. Le, T. Hoang, J. Li, L. Liu, H. Liu, and S. Hu, "A fast pc algorithm for high dimensional causal discovery with multi-core pcs," *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 16, pp. 1483–1495, 2015.

[137] B. Huang, K. Zhang, Y. Lin, B. Schölkopf, and C. Glymour, "Generalized score functions for causal discovery," *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2018.

[138] A. Hyvärinen and S. M. Smith, "Pairwise likelihood ratios for estimation of non-gaussian structural equation models," *Journal of machine learning research : JMLR*, pp. 111–152, 2013.

[139] S. Imoto, T. Goto, and S. Miyano, "Estimation of genetic networks and functional structures between genes by using bayesian networks and nonparametric regression," *Pacific Symposium on Biocomputing. Pacific Symposium on Biocomputing*, pp. 175–86, 2001.

[140] D. M. Chickering and D. Heckerman, "Efficient approximations for the marginal likelihood of bayesian networks with hidden variables." *Machine Learning*, vol. 29 (2-3), p. 181–212, 1997.

[141] S. Nie, D. Maua, C. de Campos, and Q. Ji, "Advances in learning bayesian networks of bounded treewidth," *Advances in Neural Information Processing Systems 27 (NIPS 2014)*, June 2014.

[142] C. Chow and C. Liu, "Approximating discrete probability distributions with dependence trees." *IEEE transactions on Information Theory*, vol. 14 (3), p. 462–467, 1968.

[143] R. He, J. Tian, and H. Wu, "Structure learning in bayesian networks of a moderate size by efficient sampling," *Journal of Machine Learning Research*, vol. 17 (1), p. 3483–3536, 2016.

[144] D. Madigan, J. York, and D. Allard, "Bayesian graphical models for discrete data," *International Statistical Review / Revue Internationale de Statistique*, vol. 63 (2), pp. 215–232, 1995.

[145] N. Friedman and D. Koller, "Being bayesian about network structure. a bayesian approach to structure discovery in bayesian networks," *Machine Learning*, vol. 50 (1-2), p. 95–125, 2003.

[146] D. M. Chickering, "Optimal structure identification with greedy search," *J. Mach. Learn. Res.*, vol. 3, pp. 507–554, 2003.

[147] J. I. Alonso-Barba, L. de la Ossa, J. A. Gamez, and J. M. Puerta, "Scaling up the greedy equivalence search algorithm by constraining the search space of equivalence classes," in *International Journal of Approximate Reasoning*, 2011.

[148] P. O. Hoyer, S. Shimizu, A. J. Kerminen, and M. Palviainen, "Estimation of causal effects using linear non-gaussian causal models with hidden variables," *Int. J. Approx. Reason.*, vol. 49, pp. 362–378, 2008.

[149] A. Shahbazinia, S. Salehkaleybar, and M. Hashemi, "Paralingam: Parallel causal structure learning for linear non-gaussian acyclic models," *J. Parallel Distributed Comput.*, vol. 176, pp. 114–127, 2021.

[150] G. F. Cooper and E. H. Herskovits, "A bayesian method for the induction of probabilistic networks from data," *Machine Learning*, vol. 9, pp. 309–347, 1992.

[151] G. Varando, "Learning dags without imposing acyclicity," *ArXiv*, vol. abs/2006.03005, 2020.

[152] N. Meinshausen and P. Buhlmann, "High-dimensional graphs and variable selection with the lasso," *Annals of Statistics*, vol. 34, pp. 1436–1462, 2006.

[153] J. Friedman, T. J. Hastie, and R. Tibshirani, "Sparse inverse covariance estimation with the graphical lasso." *Biostatistics*, vol. 9 3, pp. 432–41, 2008.

[154] O. Banerjee, L. E. Ghaoui, E. Banerjee, Ghaoui, and Aspremont, "Model selection through sparse max likelihood estimation model selection through sparse maximum likelihood estimation for multivariate gaussian or binary data," in *Journal of Machine Learning Research 9*, 2008.

[155] A. Shojaie and G. Michailidis, "Penalized likelihood methods for estimation of sparse high-dimensional directed acyclic graphs." *Biometrika*, vol. 97 3, pp. 519–538, 2009.

[156] F. Fu and Q. Zhou, "Learning sparse causal gaussian networks with experimental intervention: Regularization and coordinate descent," *Journal of the American Statistical Association*, vol. 108, pp. 288 – 300, 2013.

[157] S. W. Han, G. Chen, M.-S. Cheon, and H. Zhong, "Estimation of directed acyclic graphs through two-stage adaptive lasso for gene network inference," *Journal of the American Statistical Association*, vol. 111, pp. 1004 – 1019, 2016.

[158] J. Gu, F. Fu, and Q. Zhou, "Penalized estimation of directed acyclic graphs from discrete data," *Statistics and Computing*, vol. 29, pp. 161 – 176, 2014.

[159] G. Raskutti and C. Uhler, "Learning directed acyclic graph models based on sparsest permutations," *Stat*, vol. 7, 2018.

[160] A. Fast and D. Jensen, "Constraint relaxation for learning the structure of bayesian networks," *University of Massachusetts Amherst*, April 2009.

[161] P. Nandy, A. Hauser, and M. H. Maathuis, "High-dimensional consistency in score-based and hybrid structure learning," in *Annals of Statistics*, 2017.

[162] J. Kuipers, P. Suter, and G. Moffa, "Efficient sampling and structure learning of bayesian networks," *Journal of Computational and Graphical Statistics*, vol. 31, pp. 639 – 650, 2018.

[163] A. Grover, A. Zweig, and S. Ermon, "Graphite: Iterative generative modeling of graphs," *ArXiv*, vol. abs/1803.10459, 2018.

[164] X. Shen, F. Liu, H. Dong, Q. Lian, Z. Chen, and T. Zhang, "Disentangled generative causal representation learning," *ArXiv*, vol. abs/2010.02637, 2020.

[165] R. Moraffah, B. Moraffah, M. Karami, A. J. Raglin, and H. Liu, "Can: A causal adversarial network for learning observational and interventional distributions," *ArXiv*, vol. abs/2008.11376, 2020.

[166] Y. Bengio, T. Deleu, N. Rahaman, N. R. Ke, S. Lachapelle, O. Bilaniuk, A. Goyal, and C. J. Pal, "A meta-transfer objective for learning to disentangle causal mechanisms," *ArXiv*, vol. abs/1901.10912, 2019.

[167] N. R. Ke, J. X. Wang, J. Mitrovic, M. Szummer, and D. J. Rezende, "Amortized learning of neural causal representations," *ArXiv*, vol. abs/2008.09301, 2020.

[168] I. Khemakhem, R. P. Monti, R. Leech, and A. Hyvärinen, "Causal autoregressive flows," *ArXiv*, vol. abs/2011.02268, 2020.

[169] P. Brouillard, S. Lachapelle, A. Lacoste, S. Lacoste-Julien, and A. Drouin, "Differentiable causal discovery from interventional data," *ArXiv*, vol. abs/2007.01754, 2020.

[170] M. Yang, F. Liu, Z. Chen, X. Shen, J. Hao, and J. Wang, "Causalvae: Disentangled representation learning via neural structural causal models," *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9588–9597, 2020.

[171] M. Kocaoglu, C. Snyder, A. G. Dimakis, and S. Vishwanath, "Causalgan: Learning causal implicit generative models with adversarial training," *ArXiv*, vol. abs/1709.02023, 2017.

[172] I. Ng, Z. Fang, S. Zhu, and Z. Chen, "Masked gradient-based causal structure learning," in *SDM*, 2019.

[173] Y. Li, A. Torralba, A. Anandkumar, D. Fox, and A. Garg, "Causal discovery in physical systems from videos," *ArXiv*, vol. abs/2007.00631, 2020.

[174] R. Pamfil, N. Sriwattanaworachai, S. Desai, P. Pilgerstorfer, P. Beaumont, K. Georgatzis, and B. Aragam, "Dynotears: Structure learning from time-series data," *ArXiv*, vol. abs/2002.00498, 2020.

[175] N. Hoang, B. Duong, and T. Nguyen, "Enabling causal discovery in post-nonlinear models with normalizing flows," *arXiv preprint arXiv:2407.04980*, 2024.

[176] T. Zhang, F. Yin, and Z.-Q. Luo, "Post-nonlinear causal relationship with finite samples: A maximal correlation perspective," *ILCR*, 2024.

[177] M. G. Sethuraman, R. Lopez, R. V. Mohan, F. Fekri, T. Biancalani, and J.-C. Hutter, "Nodags-flow: Nonlinear cyclic causal structure learning," in *International Conference on Artificial Intelligence and Statistics*, 2023.

[178] J.-C. Hütter and P. Rigollet, "Estimation rates for sparse linear cyclic causal models," in *Conference on Uncertainty in Artificial Intelligence*, 2019.

[179] S. Yang, H. Wang, K. Yu, F. Cao, and X. Wu, "Towards efficient local causal structure learning," *IEEE Transactions on Big Data*, vol. 8, no. 6, pp. 1592–1609, 2022.

[180] P. J. Bickel and K. A. Doksum, "Mathematical statistics: Basic ideas and selected topics, volume i, second edition," in *If we use quadratic loss, our risk function is called the mean squared error (MSE) ...*, 2015.

[181] E. R. Ziegel, E. L. Lehmann, and G. Casella, *Theory of point estimation*. Springer, 1950.

[182] R. Bellman, "Dynamic programming," *Science*, vol. 153, pp. 34 – 37, 1957.

[183] D. Lopez-Paz and M. Oquab, "Revisiting classifier two-sample tests," *arXiv: Machine Learning*, 2016.

[184] T. Karras, T. Aila, S. Laine, and J. Lehtinen, "Progressive growing of gans for improved quality, stability, and variation," *ArXiv*, vol. abs/1710.10196, 2017.

[185] M. J. Vowels, N. C. Camgoz, and R. Bowden, "D'ya like dags? a survey on structure learning and causal discovery," *ACM Computing Surveys*, vol. 55, pp. 1 – 36, 2021.

[186] W. Zhang, J. Liao, Y. C. Zhang, and L. Liu, "Cmgan: A generative adversarial network embedded with causal matrix," *Applied Intelligence*, vol. 52, pp. 16 233 – 16 245, 2022.

[187] R. Bellman, "Adaptive control processes - a guided tour (reprint from 1961)," in *Princeton Legacy Library*, 2015.

[188] T. Deleu, A. G'ois, C. C. Emezue, M. Rankawat, S. Lacoste-Julien, S. Bauer, and Y. Bengio, "Bayesian structure learning with generative flow networks," *ArXiv*, vol. abs/2202.13903, 2022.

[189] M. Leonelli and G. Varando, "Highly efficient structural learning of sparse staged trees," in *European Workshop on Probabilistic Graphical Models*, 2022.

[190] J. Berrevoets, N. Seedat, F. Imrie, and M. van der Schaar, "Differentiable and transportable structure learning," *ArXiv*, vol. abs/2206.06354, 2022.

[191] J. Choi, R. S. Chapkin, and Y. Ni, "Supplementary material of "bayesian causal structural learning with zero-inflated poisson bayesian networks"," in *NIPS*, 2020.

[192] R. Foraita, J. Friemel, K. Günther, T. Behrens, J. Bullerdiek, R. Nimzyk, W. Ahrens, and V. Didelez, "Causal discovery of gene regulation with incomplete data," *Journal of the Royal Statistical Society: Series A (Statistics in Society)*, vol. 183, 2020.

[193] X. Shen, S. Ma, P. Vemuri, and G. J. Simon, "Challenges and opportunities with causal discovery algorithms: Application to alzheimer's pathophysiology," *Scientific Reports*, vol. 10, 2020.

[194] A. Moneta, D. Entner, P. O. Hoyer, and A. Coad, "Causal inference by independent component analysis: Theory and applications," *Econometrics: Econometric & Statistical Methods - Special Topics eJournal*, 2013.

[195] R. Opgen-Rhein and K. Strimmer, "From correlation to causation networks: a simple approximate learning algorithm and its application to high-dimensional plant gene expression data," *BMC Systems Biology*, vol. 1, pp. 37 – 37, 2007.

[196] A. Londei, A. D'Ausilio, D. Basso, and M. O. Belardinelli, "A new method for detecting causality in fmri data of cognitive processing," *Cognitive Processing*, vol. 7, pp. 42–52, 2006.

[197] I. Ebert-Uphoff and Y. Deng, "Causal discovery for climate research using graphical models," *Journal of Climate*, vol. 25, pp. 5648–5665, 2012.

[198] J. Runge, S. Bathiany, E. M. Bollt, G. Camps-Valls, D. Coumou, E. R. Deyle, C. Glymour, M. Kretschmer, M. D. Mahecha, J. Muñoz-Marí, E. H. van Nes, J. Peters, R. Quax, M. Reichstein, M. Scheffer, B. Scholkopf, P. Spirtes, G. Sugihara, J. Sun, K. Zhang, and J. Zscheischler, "Inferring causation from time series in earth system sciences," *Nature Communications*, vol. 10, 2019.

[199] S. L. Morgan and C. Winship, "Counterfactuals and causal inference: Methods and principles for social research," in *Cambridge University Press*, 2007.

[200] J. Peters, J. Mooij, D. Janzing, and B. Schölkopf, "Identifiability of causal graphs using functional models," *arXiv preprint arXiv:1202.3757*, 2012.

[201] K. Zhang and A. Hyvärinen, "Distinguishing causes from effects using nonlinear acyclic causal models," in *Causality: Objectives and Assessment*. PMLR, 2010, pp. 157–164.

[202] K. Uemura, T. Takagi, K. Takayuki, H. Yoshida, and S. Shimizu, "A multivariate causal discovery based on post-nonlinear model," in *Conference on Causal Learning and Reasoning*. PMLR, 2022, pp. 826–839.

[203] Y. Chung, J. Kim, T. Yan, and H. Zhou, "Post-nonlinear causal model with deep neural networks," 2019.

[204] G. Keropyan, D. Strieder, and M. Drton, "Rank-based causal discovery for post-nonlinear models," in *International Conference on Artificial Intelligence and Statistics*. PMLR, 2023, pp. 7849–7870.

[205] D. Ulmer, L. Meijerink, and G. Ciná, "Trust issues: Uncertainty estimation does not enable reliable ood detection on medical tabular data," *ArXiv*, vol. abs/2011.03274, 2020.

[206] F. Tan, X. Hou, J. Zhang, Z. Wei, and Z. Yan, "A deep learning approach to competing risks representation in peer-to-peer lending," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 30, pp. 1565–1574, 2019.

[207] N. Damij, "Business process modelling using diagrammatic and tabular techniques," *Business process management journal*, vol. 13, no. 1, pp. 70–90, 2007.

[208] T. J. Plewes and R. Tourangeau, "Nonresponse in social science surveys: A research agenda," 2013.

[209] A. M. Alaa, B. van Breugel, E. S. Saveliev, and M. van der Schaar, "How faithful is your synthetic data? sample-level metrics for evaluating and auditing generative models," in *International Conference on Machine Learning*, 2021.

[210] E. Choi, S. Biswal, B. A. Malin, J. D. Duke, W. F. Stewart, and J. Sun, "Generating multi-label discrete patient records using generative adversarial networks," in *MLHC*, 2017.

[211] A. Torfi and E. A. Fox, "Corgan: Correlation-capturing convolutional generative adversarial networks for generating synthetic healthcare records," in *The Florida AI Research Society*, 2020.

[212] C. Kimble, "Electronic health records: Cure-all or chronic condition?" *Health Economics eJournal*, 2014.

[213] J. Jordon, J. Yoon, and M. van der Schaar, "Pate-gan: Generating synthetic data with differential privacy guarantees," in *International Conference on Learning Representations*, 2018.

[214] L. Xu, M. Skoularidou, A. Cuesta-Infante, and K. Veeramachaneni, "Modeling tabular data using conditional gan," in *Advances in Neural Information Processing Systems*, 2019.

[215] J. Lee, "Invertible tabular gans: Killing two birds with onestone for tabular data synthesis," in *Neural Information Processing Systems*, 2022.

[216] J. Kim, J. Jeon, J. Lee, J. Hyeong, and N. Park, "Oct-gan: Neural ode-based conditional tabular gans," *Proceedings of the Web Conference 2021*, 2021.

[217] T. Q. Chen, Y. Rubanova, J. Bettencourt, and D. K. Duvenaud, "Neural ordinary differential equations," in *Neural Information Processing Systems*, 2018.

[218] B. van Breugel, T. Kyono, J. Berrevoets, and M. van der Schaar, "Decaf: Generating fair synthetic data using causally-aware generative networks," in *Neural Information Processing Systems*, 2021.

[219] A. Rajabi and O. O. Garibay, "Tabfairgan: Fair tabular data generation with generative adversarial networks," *Mach. Learn. Knowl. Extr.*, vol. 4, pp. 488–501, 2021.

[220] B. Wen, L. O. Colon, K. Subbalakshmi, and R. Chandramouli, "Causal-tgan: Generating tabular data using causal generative adversarial networks," *ArXiv*, vol. abs/2104.10680, 2021.

[221] P. Sanchez, X. Liu, A. Q. O'Neil, and S. A. Tsaftaris, "Diffusion models for causal discovery via topological ordering," *ArXiv*, vol. abs/2210.06201, 2022.

[222] D. Połap and A. Jaszcz, "Sonar digital twin layer via multiattention networks with feature transfer," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 62, pp. 1–10, 2024.

[223] C. K. Thomas, W. Saad, and Y. Xiao, "Causal semantic communication for digital twins: A generalizable imitation learning approach," *IEEE Journal on Selected Areas in Information Theory*, vol. 4, pp. 698–717, 2023.

[224] W. V. D. Hodge, "The theory and applications of harmonic integrals," in *CUP Archive*, 1941.

[225] X. Jiang, L.-H. Lim, Y. Yao, and Y. Ye, "Statistical ranking and combinatorial hodge theory," *Mathematical Programming*, vol. 127, pp. 203–244, 2008.

[226] H. Bhatia, G. Norgard, V. Pascucci, and P.-T. Bremer, "The helmholtz-hodge decomposition—a survey," *IEEE Transactions on Visualization and Computer Graphics*, vol. 19, pp. 1386–1404, 2013.

[227] L.-H. Lim, "Hodge laplacians on graphs," *ArXiv*, vol. abs/1507.05379, 2015.

[228] S. Ruder, "An overview of gradient descent optimization algorithms," *arXiv preprint arXiv:1609.04747*, 2016.

[229] L. Rosasco, S. Villa, S. Mosci, M. Santoro, and A. Verri, "Nonparametric sparsity and regularization," *ArXiv*, vol. abs/1208.2572, 2012.

[230] M. G. Kendall, A. L. Stuart, and J. K. Ord, "Kendall's advanced theory of statistics," *Journal of the American Statistical Association*, vol. 90, p. 398, 1995.

[231] S. Löwe, D. Madras, R. S. Zemel, and M. Welling, "Amortized causal discovery: Learning to infer causal graphs from time-series data," *ArXiv*, vol. abs/2006.10833, 2020.

[232] Y. Wang, V. Menkovski, H. Wang, X. Du, and M. Pechenizkiy, "Causal discovery from incomplete data: A deep learning approach," *ArXiv*, vol. abs/2001.05343, 2020.

[233] C.-P. Chou and P. Bentler, "Estimates and tests in structural equation modeling." *Structural equation modeling: Concepts, issues, and applications*, p. 37–55, 1995.

[234] A. Cheng, "Pac-gan: Packet generation of network traffic using generative adversarial networks," *2019 IEEE 10th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON)*, pp. 0728–0734, 2019.

[235] N. Srivastava, G. E. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting," *J. Mach. Learn. Res.*, vol. 15, pp. 1929–1958, 2014.

[236] D. A. Nix and A. S. Weigend, "Estimating the mean and variance of the target probability distribution," *Proceedings of 1994 IEEE International Conference on Neural Networks (ICNN'94)*, vol. 1, pp. 55–60, 1994.

[237] I. Higgins, L. Matthey, A. Pal, C. P. Burgess, X. Glorot, M. M. Botvinick, S. Mohamed, and A. Lerchner, "beta-vae: Learning basic visual concepts with a constrained variational framework," in *International Conference on Learning Representations*, 2016.

[238] G. V. Cybenko, D. P. O'Leary, J. Rissanen, and I. P. on Mathematics in High-Performance Computing, *The mathematics of information coding, extraction, and distribution*. Springer, 1999.

[239] R. A. Horn and C. R. Johnson, *Matrix analysis*. Cambridge, New York: Cambridge University Press, 1985.

[240] J. Peters, J. M. Mooij, D. Janzing, and B. Schölkopf, "Causal discovery with continuous additive noise models," *J. Mach. Learn. Res.*, vol. 15, pp. 2009–2053, 2013.

[241] M. W. Hirsch and S. Smale, "Differential equations, dynamical systems, and linear algebra," in *Journal of Mathematics*, 1974.

[242] A. Torfi and E. A. Fox, "Corgan: Correlation-capturing convolutional neural networks for generating synthetic healthcare records," *arXiv preprint arXiv:2001.09346*, 2020.

[243] P. Erdös and A. Rényi, "On random graphs, i," *Publicationes Mathematicae*, p. 290–297, 1959.

[244] A. E. W. Johnson, T. J. Pollard, L. Shen, L. wei H. Lehman, M. Feng, M. M. Ghassemi, B. Moody, P. Szolovits, L. A. Celi, and R. G. Mark, "Mimic-iii, a freely accessible critical care database," *Scientific Data*, vol. 3, 2016.

[245] R. Salakhutdinov and G. E. Hinton, "Replicated softmax: an undirected topic model," in *NIPS*, 2009, pp. 1607–1614.

[246] C. Louizos, U. Shalit, J. M. Mooij, D. A. Sontag, R. S. Zemel, and M. Welling, "Causal effect inference with deep latent-variable models," *ArXiv*, vol. abs/1705.08821, 2017.

[247] Y. Wang and D. M. Blei, "The blessings of multiple causes," *Journal of the American Statistical Association*, vol. 114, pp. 1574 – 1596, 2019.

[248] S. Zhao, J. Song, and S. Ermon, "Infovae: Information maximizing variational autoencoders," *ArXiv*, vol. abs/1706.02262, 2017.

[249] S. Zhao and J. Song, "The information-autoencoding family: A lagrangian perspective on latent variable generative modeling," *arXiv: Machine Learning*, 2018.

[250] A. Gretton, K. M. Borgwardt, M. J. Rasch, B. Scholkopf, and A. Smola, "A kernel method for the two-sample-problem," in *Neural Information Processing Systems*, 2006.

[251] K. Fukushima, "Visual feature extraction by a multilayered network of analog threshold elements," *IEEE Trans. Syst. Sci. Cybern.*, vol. 5, pp. 322–333, 1969.

[252] S. Arora and B. Barak, "Computational complexity: A modern approach," in *Cambridge University Press*, 2009.

[253] C. S. Calude, "Theories of computational complexity," in *Elsevier*, 1988.

[254] I. Khemakhem, R. Monti, R. Leech, and A. Hyvarinen, "Causal autoregressive flows," in *International conference on artificial intelligence and statistics*. PMLR, 2021, pp. 3520–3528.

[255] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778, 2015.

[256] B. Neal, "Introduction to causal inference," *Course Lecture Notes (draft)*, 2020.

[257] A. Wehenkel and G. Louppe, "Graphical normalizing flows," in *International Conference on Artificial Intelligence and Statistics*, 2020.

[258] A. M. K. Mamaghan, A. Dittadi, S. Bauer, K. H. Johansson, and F. Quinzan, "Diffusion-based causal representation learning," *Entropy*, vol. 26, 2024.

[259] I. T. Jolliffe and J. Cadima, "Principal component analysis: a review and recent developments," *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 374, 2016.

[260] S. Grof and M. D. Transpersonal, "On the mathematical foundations of theoretical statistics," *Philosophical Transactions of the Royal Society A*, vol. 222, pp. 309–368, 1921.

[261] G. Park, "Identifiability of additive noise models using conditional variances," *Journal of Machine Learning Research*, vol. 21, no. 75, pp. 1–34, 2020. [Online]. Available: http://jmlr.org/papers/v21/19-664.html

[262] M. Stone, "Cross-validatory choice and assessment of statistical predictions," *Journal of the royal statistical society series b-methodological*, vol. 36, pp. 111–133, 1976.

[263] R. J. Simard and P. L'Ecuyer, "Computing the two-sided kolmogorov-smirnov distribution," *Journal of Statistical Software*, vol. 39, pp. 1–18, 2011.

[264] C. A. Williams, "The choice of the number and width of classes for the chi-square test of goodness of fit," *Journal of the American Statistical Association*, vol. 45, pp. 77–86, 1950.

[265] R. Sánchez-Romero, J. Ramsey, M. Glymour, B. Huang, F. Eberhardt, and C. Glymour, "Estimating causal networks in high-dimensional settings," *Nature Communications*, vol. 10, no. 1, p. 5344, 2019.

[266] J. Runge, P. Nowack, M. Kretschmer, S. Flaxman, and D. Sejdinovic, "Inferring causation from time series in earth system sciences," *Nature Communications*, vol. 10, no. 1, p. 2553, 2019.

[267] Y. Huang, W. Cai, and M. Xu, "Causal modeling across multimodal data for alzheimer's diagnosis," *IEEE Transactions on Medical Imaging*, vol. 39, no. 9, pp. 2885–2897, 2020.

[268] S. Shimizu, K. Lee, and A. Hyvärinen, "Causal inference under missing data mechanisms in real-world ehrs," *Journal of the American Medical Informatics Association (JAMIA)*, vol. 29, no. 4, pp. 700–712, 2022.

[269] N. Haug, L. Geyrhofer, A. Londei, E. Dervic, A. Desvars-Larrive, V. Loreto, and et al., "Ranking the effectiveness of worldwide covid-19 government interventions," *Nature Human Behaviour*, vol. 4, no. 12, pp. 1303–1312, 2020.

[270] M. J. Kusner, J. Loftus, C. Russell, and R. Silva, "Counterfactual fairness," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2017, pp. 4066–4076.

[271] F. Villaescusa-Navarro, M. Aragon-Calvo, and et al., "The camels project: Cosmology and astrophysics with machine learning simulations," *The Astrophysical Journal*, vol. 915, no. 1, p. 71, 2021.

[272] M. Cranmer, R. Xu, P. Battaglia, S. Ho, D. Spergel, and Y. LeCun, "Discovering symbolic models from deep learning with inductive biases," *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

[273] M. Binz and E. Schulz, "Do large language models show biases in causal learning?" *arXiv preprint arXiv:2312.10509*, 2023.

[274] D. J. Drucker, "Glp-1 physiology informs the pharmacotherapy of obesity," *Molecular Metabolism*, vol. 57, 2021.