High-dimensional Mediation Analysis By Using GAN approach

Jiaming Zhang and Hanwen Ning*
Innovation and Talent Base for Digital Technology and Finance
Department of Financial Statistics, Zhongnan University of Economics and Law
Yiqi Lin, Xinyuan Song
Department of Statistics, The Chinese University of Hong Kong

June 27, 2025

Abstract

High-dimensional mediation analysis (HMA) investigates causal pathways in complex systems involving a large number of mediators, and plays a critical role in fields such as genomics, economics, and medicine. To address high dimensionality, existing HMA methods typically depend on restrictive parametric assumptions and random effect specifications, which limit their ability to capture nonlinearity, heterogeneity, and inter-mediator correlations. In this study, inspired by the flexibility of GANs in image generation problems, we propose a novel GAN-based mediation model termed Generative Adversarial High-Dimensional Mediation Network (GAHMN). GAHMN incorporates two specially designed generative networks with partially linear structures and multi-channel 1×1 convolutions, and formulates a novel min-max optimization problem tailored to the high-dimensional mediation setting. Thanks to its carefully constructed architecture, innovative optimization schemes, and the inherent advantages of the GAN framework, GAHMN effectively addresses heterogeneity and mediator correlations, while significantly reducing model complexity from traditional $O(p^2)$ to $O(p)$, where p denotes the number of mediators. By relaxing the stringent assumptions commonly required in existing approaches, GAHMN improves both modeling flexibility and estimation accuracy, especially for complex and high-dimensional data. This work highlights the potential of generative learning in tackling broader high-dimensional modeling problems. Theoretical results and extensive numerical experiments are presented to illustrate our method.

Keywords: Mediation analysis; High-dimensional mediators; Conditional GAN; Counterfactual; Regularization;

*Corresponding Author: ninghanwen@gmail.com

1 Introduction

Causal mediation analysis is a powerful statistical method used to examine the mechanisms through which an independent variable X affects a dependent variable Y via an intermediate variable, known as the mediator M. In a general mediation problem, the goal is to decompose the total effect of X on Y into two components: the direct effect (the effect of X on Y not mediated by M) and the indirect effect (the effect of X on Y that operates through M). Traditional mediation analysis is typically conducted in low-dimensional settings, where the number of variables is small (MacKinnon, 2012). In many modern scientific applications, such as genomics (Liu et al., 2022), neuroscience (Lindquist, 2012), and social network analysis (Liu et al., 2021), where understanding complex causal mechanisms is critical, the mediation problems often involve a large number of potential mediators, which gives rise to the high-dimensional mediation problem. In contrast to its low-dimensional counterpart, the primary difficulties in high-dimensional mediation problems stem from the curse of dimensionality and the associated computational challenges. Issues such as variable selection, model estimation, and statistical inference become more complex in high dimensions (Zeng et al., 2021; Yang et al., 2024).

Most existing HMA methods extend low-dimensional mediation structures by significantly increasing the number of variables, without fundamentally altering the model architecture. These methods typically rely on regularization techniques, such as LASSO, for variable selection and parameter estimation (Guo et al., 2022), and often adopt a stepwise estimation scheme to further derive mediation effects. While effective in many applications,, they still have critical limitations that hinder improvements in modeling performance. In high-dimensional mediation settings, the data are often collected at the micro level, reflecting individual specific traits, which inherently exhibit strong nonlinearity, complexity, and heterogeneity. Traditional methods, however, tend to impose strict assumptions, such as linearity, homogeneity, and normality, to facilitate parameter estimation, statistical inference, and theoretical analysis. Yet, there is little theoretical or empirical justification for these simplifying assumptions. On the contrary, these assumptions are likely to be violated in real-world settings. For instance, in fields such as financial econometrics (Carpena and Zia, 2020), neuroscience (Nath et al., 2023), and survival analysis (Zhou and Song, 2021), data frequently exhibit complex nonlinear relationships and significant heterogene-

ity. In such contexts, conventional methods may prove too rigid to capture these intricate patterns. In particular, individual-level variables are often influenced by shared but unobserved latent factors, leading to strong correlations among mediators-an issue that cannot be ignored. In low-dimensional settings, heterogeneity can be addressed by parameterizing its patterns, and correlations can be modeled by specifying covariance matrices for mediator variables, mitigating the risk of model misspecification. These setups allow for the application of Bayesian methods for estimation. However, such strategies become impractical in high-dimensional contexts. As the number of mediators increases, the number of parameters and the overall model complexity grow at least quadratically. Specifically, if the number of mediators is p, the model dimension must be at least $O(p^2)$ to adequately accommodate heterogeneity and correlations. This poses significant challenges for both modeling and estimation when p is very large.

GANs are among the most prominent models in generative learning. A typical GAN consists of two neural networks—a generator and a discriminator—that are trained in opposition through a minimax optimization process. The generator aims to produce synthetic data that mimic the real data distribution, while the discriminator attempts to distinguish between real and generated samples. This adversarial training enables the generator to learn complex data distributions with high fidelity (Gui et al., 2021). GANs are particularly effective at generating high-dimensional, diverse outputs through deep neural networks and have demonstrated strong performance in handling complex image data. Images can be viewed as high-dimensional vectors (or tensors) characterized by intricate distributions, correlations, and heterogeneity. Inspired by this, if high-dimensional mediators are reinterpreted as analogous to image pixels, then the mediator vectors can naturally be treated as either input or response variables within a learning framework. This perspective opens the door to investigating high-dimensional mediation problems using a GAN-based approach. In this way, the inherent flexibility of GANs in modeling complex, high-dimensional distributions can be leveraged to significantly enhance the performance of HMA.

In this study, we propose a GAN-based high-dimensional mediation model, termed GAHMN. Traditional GAN architectures and optimization strategies—originally developed for image generation tasks—cannot be directly applied to HMA. This is because HMA requires not only clear model interpretability, but also the integration of various prior structural characteristics inherent to mediation models and high-dimensional data. A

key characteristic of high-dimensional mediation models is that the mediators serve both as input and response variables within the mediation framework. Accordingly, GAHMN is composed of two main components, each implemented using a conditional generative adversarial network. When the mediators are treated as response variables, the corresponding component is referred to as the mediator block; when they serve as input variables, the corresponding component is termed the outcome block. To address the challenges, GAHMN is also designed with several tailored features. First, we incorporate partially high-dimensional linear structures, which significantly facilitate the estimation of mediation effects within a counterfactual framework, while ensuring interpretability. Second, GAHMN embeds a specially designed multi-channel 1×1 convolutional structure in mediator block, allowing different mediators—when treated as response variables—to share parameters within the model. This not only alleviates the curse of dimensionality but also substantially enhances computational efficiency within deep learning frameworks such as PyTorch. In contrast, a naive implementation based on traditional modeling paradigm would require constructing P separate GANs for mediator block, which is not only computationally infeasible, but also dramatically increases model complexity and degrades performance. Third, covariates are coupled with random noise through deep neural networks. This design reduces model complexity from $O(p^2)$—as seen in conventional approaches—to $O(p)$, while still effectively capturing correlations and heterogeneity among high-dimensional mediators. Moreover, we introduce two novel min-max optimization schemes that effectively handle sparse estimation problems, regardless of whether high-dimensional mediators are modeled as inputs or responses. With our carefully crafted architecture and the inherent advantages of the GAN framework, GAHMN can deliver more accurate mediation results and presents a promising solution for high-dimensional mediation analysis. This study also highlights the potential of generative learning frameworks in tackling a broader class of high-dimensional modeling challenges. Theoretical justifications and extensive numerical experiments are provided to demonstrate the efficacy of our new method.

The remainder of this paper is structured as follows. Section 2 provides a brief review of the existing benchmark methods and the motivations of this study. Section 3 presents our new model. Theoretical results are given in Section 4. Section 5 gives the numerical experiments, further illustrating the effectiveness and advantages of our method. Conclusions and discussions are given in Section 6.

2 Benchmark Mediation Models and Our Motivations

This section presents a review of the benchmark mediation models and GAN methods. Discussions on the limitations of the existing methods and motivations of this study are also presented.

2.1 Linear Mediation Model

In mediation analysis, the total effect of a treatment on the outcome is decomposed into two components: (I) the direct effect, which reflects the treatment's direct influence on the outcome, and (II) the indirect effect, which captures the treatment's impact on the outcome through the mediator (MacKinnon, 2012). The goal of mediation analysis is to identify and assess the mechanisms through which a treatment affects the outcome. Throughout the equations in this paper, vectors are denoted in boldface, while scalars are represented in standard font. Let T denote the treatment, $\boldsymbol{M}$ represent the p-dimensional mediator vector, $\boldsymbol{X}$ be the d-dimensional vector of covariate variables, and Y be the outcome, respectively. The traditional mediator analysis can be modeled by the following equations

$$\begin{cases} \boldsymbol{M} = \boldsymbol{\beta_1} + \boldsymbol{a}T + \boldsymbol{\delta_1}\boldsymbol{X} + \boldsymbol{\epsilon_1}, \\ Y = \beta_2 + c'T + \boldsymbol{b}^\top \boldsymbol{M} + \boldsymbol{\delta_2}^\top \boldsymbol{X} + \epsilon_2, \end{cases} \tag{1}$$

where $\boldsymbol{\epsilon_1}$ and ϵ_2 are normally distributed error terms. $\boldsymbol{\delta_1} = (\boldsymbol{\delta_{1,1}}, \boldsymbol{\delta_{1,2}}, \ldots, \boldsymbol{\delta_{1,p}})$ is a $p \times d$ matrix. Plugging the first equation of (1) into the second one yields

$$\begin{aligned} Y &= (\beta_2 + \boldsymbol{b}^\top \boldsymbol{\beta_1}) + (c' + \boldsymbol{b}^\top \boldsymbol{a})T + (\boldsymbol{b}^\top \boldsymbol{\delta_1} + \boldsymbol{\delta_2}^\top)\boldsymbol{X} + (\boldsymbol{b}^\top \boldsymbol{\epsilon_1} + \epsilon_2) \\ &= \beta_3 + \gamma T + \boldsymbol{\delta_3}^\top \boldsymbol{X} + \epsilon_3, \end{aligned} \tag{2}$$

where $\gamma = c' + \boldsymbol{b}^\top \boldsymbol{a}$, $\boldsymbol{b}^\top \boldsymbol{a}$ and c' give total effect, total indirect effect and direct effect, respectively. The element-wise product $\boldsymbol{a} \odot \boldsymbol{b} = [a_1 b_1, a_2 b_2, \ldots, a_p b_p]$ represents the indirect effects through the p different mediators. The parameters in (1) can be estimated using ordinary least squares (OLS) (Valente et al., 2020) or maximum likelihood estimates (MLE) (Wang, 2019). The standard errors of $a_k b_k$, $(k = 1, 2, \cdots, p)$ and their corresponding confidence intervals (CIs) are derived as detailed in Tofighi and MacKinnon (2011). Additionally, structural equation model (SEM) (Gunzler et al., 2013) is another popular approach, which involves an error covariance matrix to capture the correlations among

mediators. SEM can be estimated through generalized least squares (GLS) (Olsson et al., 1999), weighted least squares (WLS) (Olsson et al., 2000), or Bayesian methods (Lee, 2007).

2.2 High-dimensional Mediation Model

HMA explores causal pathways in complex systems with numerous mediators. A benchmark approach to constructing high-dimensional mediation models involves extending low-dimensional mediation structures by increasing their dimensionality. Specifically, in Equation (1), the parameters $\boldsymbol{a}$ and $\boldsymbol{b}$ are set as high-dimensional vectors without fundamentally modifying the underlying model structure.

Notably, in conventional settings, only a small subset of mediators are assumed to respond to treatment T and subsequently influence the outcome Y, this suggests that most of the indirect effects $a_k b_k$, $(k = 1, 2, \cdots, p)$ are likely to be zero. With suitable penalty, Guo et al. (2022) effectively identifies active mediators by shrinking irrelevant coefficients to zero, while preserving those with significant effects. The total effect γ is estimated based on the low-dimensional equation (2) using OLS. c' and $\boldsymbol{b}$ are estimated by minimizing:

$$\frac{1}{2n}\sum_{i=1}^{n}\{Y_i - \beta_2 - c'T_i - \boldsymbol{b}^\top \boldsymbol{M}_i - \boldsymbol{\delta_2}^\top \boldsymbol{X}_i\}^2 + \lambda\sum_{k=1}^{p}|b_k|, \tag{3}$$

where $\boldsymbol{X}_i, T_i, \boldsymbol{M}_i, Y_i$ represents the i-th sample, n is the sample size, and λ is the tuning parameter of the penalty function. LASSO can be applied to solve (3). Finally, the total indirect effect is derived as $\widehat{\boldsymbol{b}^\top \boldsymbol{a}} = \hat{\gamma} - \hat{c'}$. $\sum_{k=1}^{p}|b_k| = |\boldsymbol{b}|_1$ represents an L_1 penalty, and can be replaced with alternative ones, such as SCAD (Smoothly Clipped Absolute Deviation) (Kim et al., 2008), Adaptive LASSO (Zou, 2006), and Elastic Net (Zou and Hastie, 2005), according to the specific mediation tasks.

To relax the linear restraints on the covariates and further specify the indirect effects, a more flexible semi-parametric model is introduced in Cai et al. (2022):

$$\begin{cases} \boldsymbol{M} = \boldsymbol{\beta_1} + \boldsymbol{a}T + g_1(\boldsymbol{X}) + \boldsymbol{\epsilon_1}, \\ Y = \beta_2 + c'T + \boldsymbol{b}^\top \boldsymbol{M} + g_2(\boldsymbol{X}) + \epsilon_2, \end{cases} \tag{4}$$

where $g_1(\cdot)$ and $g_2(\cdot)$ are unknown nonlinear functions that are commonly estimated using kernel model (Hollander et al., 2013; Bingham and Kiesel, 2002). This semi-parametric setting is flexible to model confounders non-parametrically while preserving the linear structure for mediators and treatment, thus ensure the interpretability of treatment effects.

The direct and indirect treatment effects remain as c' and $\boldsymbol{a} \odot \boldsymbol{b}$. A two-step estimation procedure is also employed. In the first step, $\boldsymbol{b}$ and c' are estimated by minimizing:

$$\frac{1}{2n}\sum_{i=1}^{n}\{Y_i - \beta_2 - c'T_i - \boldsymbol{b}^\top \boldsymbol{M}_i - g_2(\boldsymbol{X}_i)\}^2 + \lambda_2 \sum_{k=1}^{p}|b_k|, \tag{5}$$

where n is the sample size. The nonzero components of $\boldsymbol{b}$ can be also identified by solving (5) using LASSO. Assume that q nonzero components are selected, and the set of their indices is denoted as Θ with $q = dim(\Theta)$. The second step is to estimate $\boldsymbol{a}$ by minimizing:

$$\frac{1}{2nq}\sum_{i=1}^{n}\sum_{k\in\Theta}\{M_{ik} - \beta_{1k} - a_k T_i - g_{1,k}(\boldsymbol{X}_i)\}^2 + \lambda_3 \sum_{k\in\Theta}|a_k|, \tag{6}$$

The selected mediators are used to construct a multivariate regression model, ultimately yielding refined estimates for c' and $\boldsymbol{a} \odot \boldsymbol{b}$. We remark that the semi-parametric model provides a flexible and explainable approach for HMA. However, the heterogeneity correlation issues are still unaddressed in high-dimensional setting.

The aforementioned approaches have been extended to other high-dimensional mediation studies. Huang et al. (2025) employ a semiparametric efficient influence function approach to select mediators and estimate natural indirect effects, ensuring robust post-selection inference through a stabilized one-step estimator. Wang et al. (2023) utilize penalized regression to identify high-dimensional mediators while addressing latent confounding, achieving precise effect estimation with false discovery rate control.

2.3 Counterfactual Framework

The counterfactual framework is a powerful approach for estimating treatment and mediation effects. Under the counterfactual framework, we denote $Y_i(t, \boldsymbol{m})$ as the potential outcome of the ith sample when T and $\boldsymbol{M}$ are set to be t and $\boldsymbol{m}$, respectively, and $\boldsymbol{M}_i(t)$ as the potential value of the mediator. To identify the direct and indirect effects through each causal pathway, the following sequential ignorability assumptions must be satisfied (Imai et al., 2010; Pearl, 2014):

(I) Conditional independence of the treatment: $Y^\star(t, m) \perp T \mid X$,

(II) Conditional independence of the mediator: $Y^\star(t, m) \perp M(t) \mid (T, X)$,

(III) Conditional independence of the treatment-mediator relationship: $M(t) \perp T \mid X$,

(IV) No mediator-outcome confounder affected by the treatment: $Y^\star(t, m) \perp M(t^*) \mid X$.

$A \perp B \mid C$ indicates the independence of A and B given C. As pointed out in Díaz and van der Laan (2013), these assumptions indicate that there are no unmeasured confounders, which can be ensured by including as many pre-treatment confounders $\boldsymbol{X}$ as possible.

Following the notations in Huang and Yang (2017), we denote $T = t_0$ versus t_1. For the ith individual, the treatment effects can be calculated based on potential outcomes corresponding to different pathways:

$$\triangle_{i,T\to Y} = Y_i(t_1, \boldsymbol{M}_i(t_0)) - Y_i(t_0, \boldsymbol{M}_i(t_0)), \tag{7}$$

$$\triangle_{i,T\to \boldsymbol{M}\to Y} = Y_i(t_1, \boldsymbol{M}_i(t_1)) - Y_i(t_1, \boldsymbol{M}_i(t_0)), \tag{8}$$

$$\triangle_{i,T\to \boldsymbol{M}^p\to Y} = Y_i(t_1, \boldsymbol{M}_i(t_1)) - Y_i(t_1, \boldsymbol{M}_i^{(-k)}(t_1)). \tag{9}$$

Here $\boldsymbol{M}_i(t_0) = [M_i^1(t_0), M_i^2(t_0), \cdots, M_i^p(t_0)]$ and $\boldsymbol{M}_i(t_1) = [M_i^1(t_1), M_i^2(t_1), \cdots, M_i^p(t_1)]$ represent the mediator vectors when all p elements are treated with t_0 and t_1, respectively. The vector $\boldsymbol{M}_i^{(-k)}(t_1)$ indicates that the kth mediator is treated with t_0 while the remaining $p-1$ mediators are treated with t_1. This can be expressed as:

$$\boldsymbol{M}_i^{(-k)}(t_0) = [M_i^1(t_1), M_i^2(t_1), \cdots, M_i^{k-1}(t_1), M_i^k(t_0), M_i^{k+1}(t_1), \cdots, M_i^p(t_1)]. \tag{10}$$

Once the individual treatment effects (ITEs) defined in (7), (8) and (9) are obtained, the average treatment effects (ATEs) can be naturally calculated as their expectations across the individuals. For any given individual, only $Y_i(t_0, \boldsymbol{M}_i(t_0))$ or $Y_i(t_1, \boldsymbol{M}_i(t_1))$ is observable, while the unobserved potential outcomes are referred to as counterfactuals, the inference of counterfactuals constitutes the foundation of the counterfactual framework. Notably, the estimations of total, direct, and indirect effects under the counterfactual framework align with the traditional mediation method when the underlying system follows a linear structure. For instance, with the linear model (1), the average direct and indirect treatment effects calculated under the conterfactual framework are equal to $c'(t_1 - t_0)$ and $\boldsymbol{a} \odot \boldsymbol{b}(t_1 - t_0)$, respectively. Nevertheless, the counterfactual framework retains a unique advantage: it allows for the calculation of various treatment effects without relying on strict linearity or random assumptions within the model. This presents a valuable opportunity to leverage powerful deep learning models and techniques to accommodate complex, non-linear relationships and interactions, enabling more precise mediation analysis.

2.4 A Brief Review of GAN and CGAN

Generative Adversarial Networks (GANs) (Goodfellow et al., 2014), along with their conditional variant (CGANs) (Mirza, 2014), represent an important class of generative models that have been widely applied in tasks such as image generation (Zhai et al., 2019), image-to-image translation (Lin et al., 2018), and video synthesis (Bansal et al., 2018). A GAN consists of two neural networks, a generator G and a discriminator D, which are trained simultaneously in a minimax game. The generator G aims to produce data that closely resembles real data, while the discriminator D strives to distinguish between real data and data generated by G. The training process is guided by the following objective function:

$$\min_G \max_D \mathbb{E}_{\boldsymbol{x}\sim P_{\text{data}}(\boldsymbol{x})}[\log D(\boldsymbol{x})] + \mathbb{E}_{\boldsymbol{z}\sim P_{\boldsymbol{z}}(\boldsymbol{z})}[\log(1 - D(G(\boldsymbol{z})))], \tag{11}$$

where $\boldsymbol{x} \sim P_{\text{data}}(\boldsymbol{x})$ represents real samples drawn from the true data distribution, and $\boldsymbol{z} \sim P_{\boldsymbol{z}}(\boldsymbol{z})$ is noise sampled from a prior distribution, such as Gaussian or uniform. The generator maps $\boldsymbol{z}$ to the data space, producing synthetic samples $G(\boldsymbol{z})$, while the discriminator evaluates whether a given sample $\boldsymbol{x}$ is real or generated. During training, G seeks to minimize the objective by "fooling" D, whereas D aims to maximize it by correctly identifying real and fake data. Through this adversarial process, the generator iteratively learns to produce realistic data.

CGANs extend the GAN framework by incorporating conditional information $\boldsymbol{y}$ (e.g., class labels, attributes, or auxiliary data) into both the generator and the discriminator. This extension enables controlled data generation conditioned on $\boldsymbol{y}$. The objective function for CGANs is as follows:

$$\min_G \max_D \mathbb{E}_{\boldsymbol{x},\boldsymbol{y}\sim P_{\text{data}}(\boldsymbol{x},\boldsymbol{y})}[\log D(\boldsymbol{x},\boldsymbol{y})] + \mathbb{E}_{\boldsymbol{z}\sim P_{\boldsymbol{z}}(\boldsymbol{z}),\boldsymbol{y}\sim P_{\boldsymbol{y}}(\boldsymbol{y})}[\log(1 - D(G(\boldsymbol{z},\boldsymbol{y}),\boldsymbol{y}))], \tag{12}$$

where $\boldsymbol{y}$ is the conditional input, $G(\boldsymbol{z},\boldsymbol{y})$ generates synthetic data conditioned on $\boldsymbol{y}$, and $D(\boldsymbol{x},\boldsymbol{y})$ evaluates both the realism of $\boldsymbol{x}$ and its consistency with the condition $\boldsymbol{y}$. In CGANs, G tries to create samples $G(\boldsymbol{z},\boldsymbol{y})$ that not only look realistic but also match the condition $\boldsymbol{y}$, while D aims to distinguish between real and fake samples while ensuring that the data corresponds to the given condition $\boldsymbol{y}$.

CGANs have attracted increasing attention from the statistical and financial research communities. For example, Aggarwal et al. (2019) demonstrated that CGANs outperform Gaussian Processes in regression tasks with low-dimensional output spaces. Zhang et al.

(2023) applied CGANs within a counterfactual framework to address mediation and causal inference problems in low-dimensional settings. Their use of deep neural networks effectively captured nonlinear relationships among variables. QuantGANs (Wiese et al., 2020) integrate GAN-based architectures with temporal convolutional networks to model financial time series. These models successfully capture complex noise structures and outperform traditional GARCH models in terms of both distributional accuracy and dependence measures. Notably, however, these pioneering studies are primarily limited to low-dimensional settings. The potential of GANs to model high-dimensional data remains largely underexplored in mediation and causal inference contexts.

2.5 Limitations of the Benchmarks and Our Motivations

Despite their effectiveness, the existing benchmarks still have several critical limitations. First, these methods are often constrained by rigid assumptions of model structure. The noise terms $\boldsymbol{\epsilon_1}$ and ϵ_2 in (1) and (4), are generally assumed to be normally distributed and homoscedastic to facilitate estimation and inference. However, mediation analysis usually focuses on the underlying patterns within micro data collected from individuals, such as healthcare patients (Hayes, 2017), market consumers (Preacher and Hayes, 2008), or businesses (Xie et al., 2018). In these cases, the data are fundamentally generated by individual micro-decisions and usually exhibit high complexity and heterogeneity, especially in the context of high-dimensional setting. Second, the correlations among mediators is another issue that needs to be considered. From a statistical perspective, the elements of $\boldsymbol{\epsilon_1}$ represent unobserved random factors that influence the mediators. These elements are often correlated, implying that the mediators themselves are also interdependent. Assuming that the mediators are conditionally independent oversimplifies the model and may fail to capture their complex interactions, potentially resulting in biased estimates (Wang, 2019; Zeng et al., 2021). Some benchmark SEM methods address this by incorporating a covariance matrix for $\boldsymbol{\epsilon_1}$ (Ullman and Bentler, 2012) to account for these correlations. However, this approach is only feasible in low-dimensional settings. When dealing with high-dimensional mediation problems, the number of mediators p becomes very large. Estimating a $p \times p$ high-dimensional covariance matrix introduces a substantial number of additional parameters into the model, significantly reducing the degrees of freedom and often rendering the estimation infeasible. Third, benchmark methods encounter inher-

ent challenges in model estimation within semi-parametric settings. Kernel methods are commonly used to handle nonlinear components in semi-parametric models such as (4). However, their performance heavily depends on hyperparameters—such as the choice of kernel function and bandwidth—which require careful tuning. In particular, when $\boldsymbol{X}$ contains a large number of features, using a single bandwidth across all components fails to capture the varying nonlinear relationships associated with $\boldsymbol{X}$, making kernel-based models less suitable for high-dimensional scenarios. Thus, the benchmark methods may be too rigid to accommodate these prevalent intricate patterns and inadequate for many real-world applications.

Within the counterfactual framework, it is crucial to estimate unobserved potential outcomes or generate counterfactuals along different causal pathways in order to infer treatment effects. This, in turn, requires accurately characterizing the conditional distribution of the outcome. Consider the second equation in (4), one popular approach is likelihood-based methods (Yuan and MacKinnon, 2009; Sun and Song, 2024). Specifically, they focus on constructing and optimizing the conditional probability density function $P(Y|C;\theta)$, where θ denotes the parameters to be estimated, and the conditioning set C includes information from T, $\boldsymbol{X}$ and $\boldsymbol{M}$. Likelihood-based methods, such as maximum likelihood estimation (MLE) and Baysian method, offer rigorous estimations of treatment effects, especially under correct model specification. In ideal conditions, they yield estimators that are asymptotically efficient and achieve the Cramér-Rao lower bound, providing the most precise unbiased estimates as sample size grows. However, these methods still come with notable limitations. Their performance is heavily dependent on strong parametric and random assumptions, and can be highly sensitive to model misspecification. These limitations underscore the need for alternative HMA approaches that are more flexible and impose fewer assumptions on the underlying data-generating process.

Fortunately, CGANs offer a promising approach to relax those rigid assumptions while also reducing the misspecification error in mediation models. Instead of assuming a parametric form for $P(Y|C)$, GANs can leverage adversarial training to learn this conditional distribution. During the training of CGANs, we first maximize (12) by finding the point where the partial derivative with respect to D equals zero, yielding the optimal discrimi-

nator D^*, which can be expressed as:

$$D^*(Y|C) = \frac{P_{\text{data}}(Y|C)}{P_{\text{data}}(Y|C) + P_{\text{g}}(Y|C)}, \tag{13}$$

where $P_{\text{g}}(Y|C)$ is the distribution of the generated data. Next, we substitute D^* back into (11) and rewrite it as:

$$\begin{aligned} C(G) &= \mathbb{E}_{Y\sim P_{\text{data}}(Y|C)}[\log D^*(Y|C)] + \mathbb{E}_{Y\sim P_{\boldsymbol{g}}(Y|C)}[\log(1 - D^*(\boldsymbol{x}))] \\ &= \mathbb{E}_{Y\sim P_{\text{data}}(Y|C)}\Big[\frac{P_{\text{data}}(Y|C)}{P_{\text{data}}(Y|C) + P_{\text{g}}(Y|C)}\Big] + \mathbb{E}_{Y\sim P_{\boldsymbol{g}}(Y|C)}\Big[\frac{P_{\text{g}}(Y|C)}{P_{\text{data}}(Y|C) + P_{\text{g}}(Y|C)}\Big] \\ &= -\log(4) + D_{KL}\Big(P_{\text{data}}\Big\|\frac{P_{\text{data}} + P_{\text{g}}}{2}\Big) + D_{KL}\Big(P_{\text{g}}\Big\|\frac{P_{\text{data}} + P_{\text{g}}}{2}\Big) \\ &= -\log(4) + 2\cdot D_{JS}(P_{\text{data}}\|P_{\text{g}}), \end{aligned} \tag{14}$$

where D_{KL} is Kullback-Leibler Divergence, and D_{JS} is Jensen-Shannon Divergence, both of which measure the similarity between two distributions.

Minimizing $C(G)$ is equivalent to minimizing the JS Divergence between the true data distribution and the generated data distribution. When G reaches its optimal state, the conditional distributions align, i.e., $P_{\text{data}}(Y|C) = P_{\text{g}}(Y|C)$. It is worth noting that directly minimizing the JS divergence typically requires assumptions about the form of the generated distribution $P_{\text{g}}(\boldsymbol{x})$. In contrast, the adversarial process employed by CGANs allows optimization to be driven by a separately trained discriminator rather than relying on predefined likelihood functions. This setup enables GANs to leverage highly flexible deep neural networks to implicitly model the conditional distribution of interest. As a result, CGANs can sidestep the rigid assumptions that are often imposed in traditional mediation models, offering a more adaptive and data-driven solution to HMA. Furthermore, CGANs can leverage their deep neural network architecture to effectively accommodate the covariates $\boldsymbol{X}$, offering greater flexibility and adaptability than kernel-based methods, particularly when modeling high-dimensional covariates. Therefore, we are motivated to adpot CGANs to reformulate the high-dimensional mediation models, providing more flexible and promising modeling approach from generative learning perspective. Nonetheless, developing GAN-based high-dimensional mediation model faces several key challenges:

- Dimensionality: We need to design an effective high-dimensional mediation network to address high dimensionality. The model should accommodate nonlinearity, het-

erogeneity, and correlations while ensuring that the number of parameters does not grow in a superlinear manner, keeping the model's complexity at an acceptable level.

- Overparameterization: In high-dimensional mediation analysis and high-dimensional causal inference problems, we often encounter situations where the number of variables p is large relative to the number of observations n, and in some cases, p even exceeds n. In such overparameterized settings, it becomes crucial to carefully design the network architecture and the corresponding optimization scheme to achieve reliable estimation of mediation effects and maintain model stability.

- Interpretability: GANs are primarily designed to generate images, where interpretability is less considered. In mediation analysis, the focus is on identifying which mediation pathways influence the outcome. Thus, designing interpretable network architectures becomes crucial.

- Generalization: In image-related tasks, the conditional variables used in CGANs are typically discrete, and the generated samples (i.e., predictions) are produced based on observed values of these discrete variables. In contrast, in mediation analysis, the conditional variables are often continuous, but their observed values are discrete in practice. This distinction necessitates a tailored approach: we must account for this characteristic in the design of the loss function and introduce novel techniques to ensure that the model can generate reliable counterfactual predictions for unobserved values of the conditional variables. This is essential for enhancing the model's generalization ability.

- Sparsity: High-dimensional mediation analysis necessitates identifying a small subset of influential mediators from a large pool of potential mediators. This requires a careful design of loss functions and network structures to enforce the sparsity associated with potential mediators.

3 Methodology

In this section, we first introduce the problem setting and notations used throughout this paper. Next, we present our new method and the particularly designed training schemes,

showing how to estimate mediation effects under generative learning framework.

3.1 Problem Setting and Notations

We adopt the notation used in (4), where $\boldsymbol{X}$, $\boldsymbol{M}$, T and Y represent the covariates, mediators, treatment, and outcome, respectively. $(\boldsymbol{X}_i, \boldsymbol{M}_i, T_i, Y_i)$, $(i = 1, 2, \cdots, n)$ denotes the sample of $(\boldsymbol{X}, \boldsymbol{M}, T, Y)$ for the ith individual in the dataset. We remind the reader to pay attention to the dimensions and ranges of the variables involved. Specifically, for $\forall i$, we have the following: $\boldsymbol{X}_i \in \mathbb{R}^d$, $\boldsymbol{M}_i \in \mathbb{R}^p$, $Y_i \in \mathbb{R}$, and T_i is a binary treatment such that $T_i \in \{t_0, t_1\}$. We emphasize that $\boldsymbol{M}$ is a high-dimensional variable characterized by inherent sparsity, implying that T primarily influences Y through a limited subset of specific dimensions within $\boldsymbol{M}$. Following the counterfactual framework given by (7), (8) and (9), for the ith individual $(\boldsymbol{X}_i, \boldsymbol{M}_i, T_i, Y_i)$, for simplicity, we assume the treatment corresponds to $T_i = t_1$. In this case, $Y_i(t_1, \boldsymbol{M}_i(t_1))$ and $\boldsymbol{M}_i(t_1)$ can be observed and are referred to as factuals. In contrast, the counterfactuals are the unobserved mediators and outcomes, such as $\boldsymbol{M}_i(t_0)$, $Y_i(t_1, \boldsymbol{M}_i(t_0))$, $Y_i(t_0, \boldsymbol{M}_i(t_0))$, and $Y_i(t_1, \boldsymbol{M}_i^{(-k)}(t_1))$. The reverse applies when $T_i = t_0$. The most important task in this problem is to learn the conditional distribution of $\boldsymbol{M}$ and Y given $\boldsymbol{X}$ and T.

3.2 The Network Structure of GAHMN

As the counterpart of mediation models (1) and (4), GAHMN consists of two distinct components: a mediation block and an outcome block, both constructed using CGANs.

3.2.1 The mediator block of GAHMN

Network structure Corresponding to the first equation of (1) and (4), the generator of the mediator block is given by

$$\hat{\boldsymbol{M}} = G_M(\boldsymbol{Z}_M, T, \boldsymbol{X}; \theta_{G_M}), \tag{15}$$

where $\hat{\boldsymbol{M}}$ represents the generated p-dimensional mediator, $\boldsymbol{Z}_M \sim N(\boldsymbol{0}, \boldsymbol{I}_{d_M})$ is a d_M-dimensional i.i.d. noise sampled from a multivariate standard normal distribution. θ_{G_M} represents the network parameters. The network structure is shown in Figure 1. The input layer of G_M consists of $(\boldsymbol{Z}_M, T, \boldsymbol{X})$, where $\boldsymbol{Z}_M$ and $\boldsymbol{X}$ constitute a fully-connected

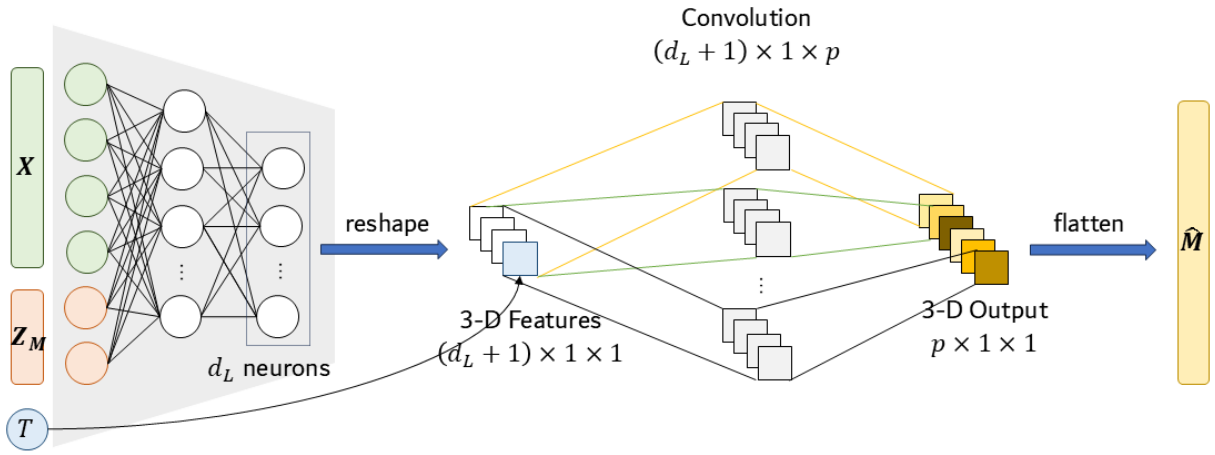


Figure 1: The Network Structure of G_M

neural network (FNN). The FNN maps $\boldsymbol{Z}_M$ and $\boldsymbol{X}$ into a latent feature representation by a hidden layer with d_L neurons. We reshape the extracted latent features into a tensor of size $(1, 1, d_L)$, then, T is appended as the $(d_L + 1)$th channel, augmenting the tensor to dimension $(1, 1, d_L + 1)$. A 1×1 convolutional kernel is applied to this tensor. This operation produces an output tensor with p channels, where p corresponds to the number of mediators. Each output channel involves $d_L + 1$ weight parameters. Specifically, the $(d_L + 1)$th weight parameter, associated with T, quantifies the direct influence of the treatment on the corresponding mediator, providing a clear interpretation of treatment effects within the learned representation.

The observed sample set is denoted as $S_M = \{\boldsymbol{M}_i, T_i, \boldsymbol{X}_i\}_{i=1}^n$, while $\widehat{S_M} = \{\hat{\boldsymbol{M}}_i, T_i, \boldsymbol{X}_i\}_{i=1}^n$ represents the data set generated by G_M. The discriminator in the mediator block, denoted as $D_M(\boldsymbol{M}, T, \boldsymbol{X}; \theta_{D_M})$, is used to evaluate the similarity between S_M and $\widehat{S_M}$, Here, D_M is implemented by a FNN with parameters θ_{D_M}. Each dimension of $\boldsymbol{M}$ must be accurately generated by G_M to match the observed data. To address this, the discriminator D_M is designed as an FNN with its the output layer consisting of p neurons. Each neuron corresponds to one dimension of $\boldsymbol{M}$, and is activated by sigmoid function.

Minmax Loss Function Given that $\boldsymbol{M}$ is high dimensional, it is understood that not every component of $\boldsymbol{M}$ is affected by the treatment T. In other words, the influence of of T on $\boldsymbol{M}$ is sparse. The expected change in $\boldsymbol{M}$ when T changes from t_0 to t_1 is $\mathbb{E}(\boldsymbol{M}(t_1) - \boldsymbol{M}(t_0))$, and only a few elements of this difference are expected to deviate significantly from zero. By G_M, for any given individual i, we can generate its factual and

counterfactual under both treatments: $G_M(\boldsymbol{Z}_M, t_1, \boldsymbol{X}_i; \theta_{G_M})$ and $G_M(\boldsymbol{Z}_M, t_0, \boldsymbol{X}_i; \theta_{G_M})$. The minmax optimization for the mediation block is given by

$$\begin{aligned}
&\min_{\theta_{G_M}} \max_{\theta_{D_M}} \mathbb{E}_{M\sim P_{data}}[\log D_M(\boldsymbol{M}, T, \boldsymbol{X}; \theta_{D_M})] \\
&+\mathbb{E}_{\boldsymbol{Z}_M\sim N(\boldsymbol{0},\boldsymbol{I}_{d_M})}[\log(1 - D_M(G_M(\boldsymbol{Z}_M, T, \boldsymbol{X}; \theta_{G_M}), T, \boldsymbol{X}; \theta_{D_M}))] \\
&+\lambda_1 \mathbb{E}_{\boldsymbol{Z}_M\sim N(\boldsymbol{0},\boldsymbol{I}_{d_M})} ||G_M(\boldsymbol{Z}_M, t_1, \boldsymbol{X}; \theta_{G_M}) - G_M(\boldsymbol{Z}_M, t_0, \boldsymbol{X}; \theta_{G_M})||_1. \qquad (16)
\end{aligned}$$

where λ_1 is a hyperparameter use for balancing. The first two terms drive the model to learn the conditional distribution associated with $\boldsymbol{M}$, T and $\boldsymbol{X}$. The third term of (16), an L_1 regularization term, is introduced to identify the sparsity associated with T.

3.2.2 The outcome block of GAHMN

Network structure Corresponding to the second equation of (1) and (4), the generator of the outcome block is given by

$$\hat{Y} = G_Y(\boldsymbol{Z}_Y, T, \boldsymbol{X}, \boldsymbol{M}; \theta_{G_Y}), \qquad (17)$$

where $\hat{Y}$ is the generated outcome, $\boldsymbol{Z}_Y \sim N(\boldsymbol{0}, \boldsymbol{I}_{d_Y})$ is a d_Y-dimensional i.i.d. noise sampled from a multivariate standard normal distribution. θ_{G_Y} represents the network parameters. The network structure is shown in Figure 2. The input layer of G_Y consists of $(\boldsymbol{Z}_Y, T, \boldsymbol{X}, \boldsymbol{M})$, where $\boldsymbol{Z}_Y$ and $\boldsymbol{X}$ constitute a multiple-layer FNN, while T and $\boldsymbol{M}$ are in the last hidden layer, forming a partially linear structure. The observed sample set is denoted as $S_Y = \{Y_i, \boldsymbol{M}_i, T_i, \boldsymbol{X}_i, \}_{i=1}^n$, while $\widehat{S_Y} = \{\hat{Y}_i, \boldsymbol{M}_i, T_i, \boldsymbol{X}_i\}_{i=1}^n$ represents the data set generated by G_Y. The discriminator in the mediator block, denoted as $D_Y(Y, \boldsymbol{M}, T, \boldsymbol{X}; \theta_{D_Y})$, is used to evaluate the similarity between S_Y and $\widehat{S_Y}$, Here, D_Y is implemented as a FNN with parameters θ_{D_Y}.

Minmax Loss Function In both the mediator block and the outcome block, we encounter challenges related to high dimensionality. The key difference is that in the mediator block, $\boldsymbol{M}$ appears as a high-dimensional output variable, whereas in the outcome block, $\boldsymbol{M}$ serves as an input to the model. $\boldsymbol{M}$ is continuous, with each component $\boldsymbol{M}_i$ varying substantially across individuals. As the dimensionality of $\boldsymbol{M}$ increases, the observed set $\{M_i\}_{i=1}^n$ becomes more diverse, and more gaps between different samples associated with $\boldsymbol{M}_i$s arise, there may be few or even no samples corresponding to certain conditions. This can lead to poor generalization when the generator encounters a new condition that is not observed

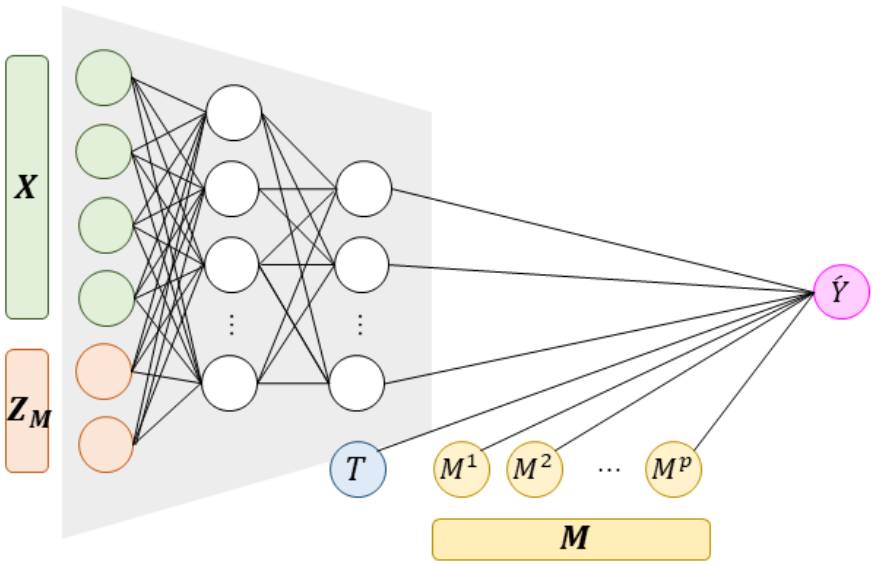


Figure 2: The Network Structure of G_Y

in the training set. It is important to note that the generator $\boldsymbol{G}_Y$ is used to describe the conditional distribution $P(Y|\boldsymbol{T}, \boldsymbol{X}, \boldsymbol{M})$. For continuous-valued conditioning variables, we want a minor perturbation to the condition to only slightly disturb the conditional distribution, meaning that the distribution obtained by $\boldsymbol{G}_Y$ should shift smoothly as we change $\boldsymbol{M}$ gradually. This approach makes the model smoother and helps avoid the risk of overfitting.

Therefore, on one hand, to improve the generalization of the model, we aim for the model to capture the true underlying patterns of the data as accurately as possible. On the other hand, since only a subset of the variables in $\boldsymbol{M}$ affects Y in the outcome block, we also need to identify the mediators that truly influence the outcome during the optimization process, imposing sparsity on $\boldsymbol{M}$. For these issues, we introduce the following partially L_1 regularization term

$$L_{RY}(G) = \mathbb{E}_{\substack{\boldsymbol{Z}_Y \sim N(\boldsymbol{0}, \boldsymbol{I}_{d_Y}) \\ T, \boldsymbol{M}, \boldsymbol{X} \sim P_{\text{data}}}} \| \bigtriangledown_{\boldsymbol{M}} G_Y(\boldsymbol{Z}_Y, T, \boldsymbol{X}, \boldsymbol{M}; \theta_{G_Y}) \|_1, \tag{18}$$

where $\bigtriangledown_{\boldsymbol{M}} G_Y(\boldsymbol{Z}_Y, T, \boldsymbol{X}, \boldsymbol{M}; \theta_{G_Y})$ denotes the vector of partial derivatives of the outcome generator with respect to the mediators, given by

$$\bigtriangledown_{\boldsymbol{M}} G_Y(\boldsymbol{Z}_Y, T, \boldsymbol{X}, \boldsymbol{M}; \theta_{G_Y}) = \left[\frac{\partial G_Y(\boldsymbol{Z}_Y, T, \boldsymbol{X}, \boldsymbol{M}; \theta_{G_Y})}{\partial M^1} \quad \cdots \quad \frac{\partial G_Y(\boldsymbol{Z}_Y, T, \boldsymbol{X}, \boldsymbol{M}; \theta_{G_Y})}{\partial M^p} \right]. \tag{19}$$

(18) takes the form of a Lipschitz penalty. It is important to note that both $\boldsymbol{M}$ and T are

included in the last hidden layer of G_Y, and each element of (19) corresponds to the weight parameters associated with $\boldsymbol{M}$. By integrating this regularization term into the classical minimax optimization of conditional GAN, we have following loss function for the outcome block

$$\begin{aligned}
&\min_{\theta_{G_Y}} \max_{\theta_{D_Y}} \mathbb{E}_{Y\sim P_{data}}[\log D_Y(Y, \boldsymbol{M}, T, \boldsymbol{X}; \theta_{D_Y})] \\
&+\mathbb{E}_{\boldsymbol{Z}_Y\sim N(\boldsymbol{0},\boldsymbol{I}_{d_Y})}[\log(1 - D_Y(G_Y(\boldsymbol{Z}_Y, T, \boldsymbol{X}, \boldsymbol{M}; \theta_{G_Y}), \boldsymbol{M}, T, \boldsymbol{X}; \theta_{D_Y}))] \\
&+\lambda_2 \mathbb{E}_{\substack{\boldsymbol{Z}_Y\sim N(\boldsymbol{0},\boldsymbol{I}_{d_Y}) \\ T,\boldsymbol{M},\boldsymbol{X}\sim P_{\text{data}}}} \| \bigtriangledown_{\boldsymbol{M}} G_Y(\boldsymbol{Z}_Y, T, \boldsymbol{X}, \boldsymbol{M}; \theta_{G_Y})\|_1. \qquad (20)
\end{aligned}$$

where λ_2 is a hyperparameter used to balance the similarity between the generated and true data distributions and the degree of regularization. An appropriate λ_1 can bring both accurate out-of-sample predictions and identification of sparsity.

3.3 Training Process of GAHMN

The optimization problems of GAHMN involve additional L_1 regularization terms, making them difficult to solve via standard optimization schemes for conventional GANs. The optimization problems in GAHMN involve additional L_1 regularization terms, which make them difficult to solve using standard optimization methods commonly applied to conventional GANs. Following the Alternating Direction Method of Multipliers (ADMM), we adopt a stepwise estimation approach to train the two GANs.

For each iteration of (16), we decompose the optimization into three steps. The first step is to optimize θ_{D_M} by maximizing $\mathbb{E}_{M\sim P_{data}}[\log D_M(\boldsymbol{M}, T, \boldsymbol{X}; \theta_{D_M})] + \mathbb{E}_{\boldsymbol{Z}_M\sim N(\boldsymbol{0},\boldsymbol{I}_{d_M})}[\log(1 - D_M(G_M(\boldsymbol{Z}_M, T, \boldsymbol{X}; \theta_{G_M}), T, \boldsymbol{X}; \theta_{D_M}))]$ using gradient-based methods such as Adam or SGD. The second step is to optimize θ_{G_M} by minimizing $\mathbb{E}_{\boldsymbol{Z}_M\sim N(\boldsymbol{0},\boldsymbol{I}_{d_M})}[\log(1 - D_M(G_M(\boldsymbol{Z}_M, T, \boldsymbol{X}; \theta_{G_M}), T, \boldsymbol{X}; \theta_{D_M}))]$ with fixed θ_{D_M}, again using gradient-based method. The third step is to further update θ_{G_M} by minimizing the regularization term $\mathbb{E}_{\boldsymbol{Z}_M\sim N(\boldsymbol{0},\boldsymbol{I}_{d_M})}||G_M(\boldsymbol{Z}_M, t_1, \boldsymbol{X}; \theta_{G_M}) - G_M(\boldsymbol{Z}_M, t_0, \boldsymbol{X}; \theta_{G_M})||_1$, using the Proximal Algorithm (PROX) (Parikh et al., 2014). The results are then fed into the next iteration.

The optimization for (20) is decomposed into three steps in each iteration. The first and second steps involve min-max optimization to update θ_{D_Y} and θ_{G_Y} using gradient-based methods, following the standard GAN training procedure. The third step further updates θ_{G_Y} by minimizing the L_1 regularization term $\mathbb{E}_{\substack{\boldsymbol{Z}_Y\sim N(\boldsymbol{0},\boldsymbol{I}_{d_Y}) \\ T,\boldsymbol{M},\boldsymbol{X}\sim P_{\text{data}}}} \| \bigtriangledown_{\boldsymbol{M}} G_Y(\boldsymbol{Z}_Y, T, \boldsymbol{X}, \boldsymbol{M}; \theta_{G_Y})\|_1$

Algorithm 1 Training schemes of GAHMN

Initialization: θ_{G_M}, θ_{D_M}, θ_{G_Y}, θ_{D_Y}, regularization parameter λ_1, λ_2 and learning rate η.

while training loss V_1 and V_2 has not converged **do**

Receiving $\{(Y_i, \boldsymbol{M}_i, T_i, \boldsymbol{X}_i)\}_{i=1}^n$; Drawing $\{\boldsymbol{Z}_{M_i}\}_{i=1}^n$ and $\{\boldsymbol{Z}_{Y_i}\}_{i=1}^n$

for $i = 1, 2, \ldots, n$ **do**

$\hat{\boldsymbol{M}}_i \leftarrow G_M(\boldsymbol{Z}_{M_i}, T_i, \boldsymbol{X}_i; \theta_{G_M})$; $\hat{Y}_i \leftarrow G_Y(\boldsymbol{Z}_{Y_i}, \boldsymbol{M}_i, T_i, \boldsymbol{X}_i; \theta_{G_Y})$

end for

Discriminator optimization

Fixed θ_{G_M} and θ_{G_Y}

Maximize $V_1 = \frac{1}{n}\sum_{i=1}^n [\log D_M(\boldsymbol{M}_i, T_i, \boldsymbol{X}_i; \theta_{D_M}) + \log D_Y(Y_i, \boldsymbol{M}_i, T_i, \boldsymbol{X}_i; \theta_{D_Y})]$
$+\frac{1}{n}\sum_{i=1}^n \Big[\log(1 - D_M(\hat{\boldsymbol{M}}_i, T_i, \boldsymbol{X}_i; \theta_{D_M})) + \log(1 - D_Y(\hat{Y}_i, \boldsymbol{M}_i, T_i, \boldsymbol{X}_i; \theta_{D_Y}))\Big]$

Update θ_{D_M} and θ_{D_Y} by Adam

Generator optimization

Fixed θ_{D_M} and θ_{D_Y}

Minimize $V_2 = \frac{1}{n}\sum_{i=1}^n \big[\log(1 - D_M(G_M(\boldsymbol{Z}_{M_i}, T_i, \boldsymbol{X}_i; \theta_{G_M}), T_i, \boldsymbol{X}_i; \theta_{D_M}))$
$+\log(1 - D_Y(G_Y(\boldsymbol{Z}_{Y,i}, \boldsymbol{M}_i, T_i, \boldsymbol{X}_i; \theta_{G_Y}), \boldsymbol{M}_i, T_i, \boldsymbol{X}_i; \theta_{D_Y}))$

Update θ_{G_M} and θ_{G_Y} by Adam

Generator regulization

Update θ_{G_M} by minimizing $\frac{1}{n}\sum_{i=1}^n ||G_M(\boldsymbol{Z}_{M_i}, t_1, \boldsymbol{X}_i; \theta_{G_M}) - G_M(\boldsymbol{Z}_{M_i}, t_0, \boldsymbol{X}_i; \theta_{G_M})||_1$ using PROX;

Update $\frac{1}{n}\sum_{i=1}^n \| \bigtriangledown_{\boldsymbol{M}} G_Y(\boldsymbol{Z}_{Y_i}, T_i, \boldsymbol{X}_i, \boldsymbol{M}_i; \theta_{G_Y})\|_1$ using PROX;

end while

Output: $\theta_{G_M}, \theta_{D_M}, \theta_{G_Y}$ and θ_{D_Y}

using the PROX method. We use the Adam algorithm as the optimizer for the first and second steps. The optimization scheme for our GAHMN is presented in Algorithm 1.

It should be noticed that the specially designed G_M and G_Y ensure the computational feasibility of the regularization terms. For the mediator block, as illustrated in Figure 1, d_L features extracted by the FNN are reshaped into a $d_L \times 1 \times 1$ tensor, with each feature represented as a separate channel. The treatment is then concatenated as an additional $(d_L + 1)$-th channel. Subsequently, p convolutional kernels of size 1×1 are applied to this 3-D feature to generate the mediator vector $\hat{\boldsymbol{M}}$. According to the definition of the 1×1

convolution operation, for the k-th element of $\hat{\boldsymbol{M}}$, we have

$$\hat{M}^k = f_{k,0} + f_{k,1} neuronM_1 + f_{k,2} neuronM_2 + \cdots + f_{k,d_L} neuronM_{d_L} + f_{k,d_L+1} T, \quad (21)$$

where $f_{k,0}$ represents the bias of the convolution kernel, $neuronM_1, \ldots, neuronM_{d_L}$ are the neurons in the last hidden layer of G_M, and $f_{k,1}, f_{k,2}, \ldots, f_{k,d_L}$ are the corresponding coefficients. The term f_{k,d_L+1} indicates the direct effect of the treatment on the k-th mediator. Thus $||G_M(\boldsymbol{Z}_{M_i}, t_1, \boldsymbol{X}_i; \theta_{G_M}) - G_M(\boldsymbol{Z}_{M_i}, t_0, \boldsymbol{X}_i; \theta_{G_M})||_1$ can be rewritten as $\sum_{k=1}^{p} |f_{k,d_L+1}|$ and be solved by PROX algorithm. For the outcome block, as shown in Figure 2, the influence of mediators on the outcome is incorporated into the last layer. The generated outcome can be expressed as:

$$\hat{Y} = b_0 + b_1 M^1 + b_2 M^2 + \cdots + b_p M^p + cT + \sum_{l=1}^{d_K} f_l neuronY_l, \quad (22)$$

where $neuronY_1, \ldots, neuronY_{d_L}$ are the neurons in the last hidden layer of G_Y. The nonlinear component is approximated by $\sum_{l=1}^{d_K} f_l neuronY_l$. According to (21) and (22), our GAHMN can be further formulated as follows

$$\begin{cases} \hat{\boldsymbol{M}} = \boldsymbol{f_0} + \boldsymbol{f_{d_L+1}} T + \boldsymbol{F}\overrightarrow{\boldsymbol{nM}}, \\ \hat{Y} = b_0 + cT + \boldsymbol{b}^\top \boldsymbol{M} + \boldsymbol{f}\overrightarrow{\boldsymbol{nY}}, \end{cases} \quad (23)$$

where $\boldsymbol{f_0} = (f_{1,0}, \ldots, f_{k,0})^T$, $\boldsymbol{F} = [f_{i,j}]_{i=1,\ldots,k; j=1,\ldots,d_L}$, $\boldsymbol{f_{d_L+1}} = (f_{1,d_L+1}, \ldots, f_{k,d_L+1})^T$, $\boldsymbol{b} = (b_1, \ldots, b_p)^T$, $\overrightarrow{\boldsymbol{nM}} = (neuronM_1, \ldots, neuronM_{d_L})$ and $\overrightarrow{\boldsymbol{nY}} = (neuronY_l, \ldots, neuronY_{d_K})$. Consequently, $\| \bigtriangledown_{\boldsymbol{M}} G_Y(\boldsymbol{Z}_{Y_i}, T_i, \boldsymbol{X}_i, \boldsymbol{M}_i; \theta_{G_Y})\|_1$ can be simplified as $|\boldsymbol{b}|$ $(\sum_{k=1}^{p} |b_k|)$ and solved using PROX algorithm. Specifically, the L_1 regulization term can be efficiently optimized using a proximal method, given by

$$\text{PROX}(v) = \text{sign}(v) \cdot \max(|v| - \lambda, 0), \quad (24)$$

which is commonly referred to as a shrinkage operator, and widely used in L_1 optimization problems. PROX does not rely on gradient computations. We leverage this algorithm to adjust specific parameters of θ_{G_M} and θ_{G_Y} obtained by gradient updates, to impose sparsity.

3.4 Estimating direct and indirect effects

As G_M and G_Y are well-trained, they can be used to generate counterfactuals for mediation analysis. As shown in Section 2.3, the counterfactual framework facilitates the estimation

of direct and indirect effects through each mediator. A trivial approach is to generate the outcome for each counterfactual path separately for a given individual as proposed in the traditional methods, which is computationally infeasible, particularly in the context of high-dimensional mediation problems. Instead of processing each counterfactual path separately, our method organizes different counterfactual values of $\boldsymbol{M}$ into a single batch, which is then simultaneously fed into G_Y to generate the corresponding outcomes. This parallel computation strategy significantly reduces the computational costs. Moreover, it is well-supported by widely used deep learning frameworks such as PyTorch and TensorFlow, which enable efficient GPU acceleration for further speed improvements.

Specifically, the first step involves generating high-dimensional mediators under different treatments for a given sample $\boldsymbol{X}$ and noise $\boldsymbol{Z}_M$. To achieve this, we duplicate $\boldsymbol{X}$ into a tensor of size $d \times 2$, while $\boldsymbol{Z}_M$ is randomly sampled noise as a tensor of size $d_M \times 2$, and treatment T is a tensor of size 1×2 filled with t_0 and t_1, respectively. Since the second dimension of all these tensors is 2, we can efficiently parallelize the generation of $\hat{\boldsymbol{M}}(t_0)$ and $\hat{\boldsymbol{M}}(t_1)$ using well-trained generator G_M, and then padding them to obtain $\hat{\boldsymbol{M}}^{-k}(t_1)$, $(k = 1, 2, \ldots, p)$. In the second step, we replicate $\boldsymbol{X}$ into a tensor of size $d \times (p+3)$, while $\boldsymbol{Z}_Y$ is randomly sampled noise as a tensor of size $d_Y \times (p+3)$, The treatment T is constructed as a tensor of size $1 \times (p+3)$, as show in Figure 3. Since the second dimension of all these tensors is now $p+3$, we can generate all $p+3$ counterfactual paths simultaneously using the generator G_Y. The final step involves calculating the individual treatment effects based on the previously generated counterfactual outcomes, which allows for the estimation of the average treatment effect. To provide a clearer explanation, the estimation process will be detailed below using formal equations and notation.

After the algorithm in Algorithm 1 has converged, we can use the well-trained generators to obtain counterfactuals and estimate the treatment effects. Let the two generators be denoted as $G_M(\cdot, \hat{\theta}_{G_M})$ and $G_Y(\cdot, \hat{\theta}_{G_Y})$. For a given individual i $(i = 1, 2, \ldots, n)$, to distinguish between the mediators generated under different treatment conditions, we sample two independent random noise sets, $\boldsymbol{Z}_{M_i}(j_0)$ and $\boldsymbol{Z}_{M_i}(j_1)$, which are drawn from normal distribution $N(\boldsymbol{0}, \boldsymbol{I}_{d_M})$, where N_g is a large integer. The generated $\boldsymbol{M}_i$s under different

treatment can be written as:

$$\hat{\boldsymbol{M}}_i(t_0, \boldsymbol{Z}_{M_i}(j_0)) = G_M(\boldsymbol{Z}_{M_i}(j_0), t_0, \boldsymbol{X}_i; \hat{\theta}_{G_M}), \tag{25}$$

$$\hat{\boldsymbol{M}}_i(t_1, \boldsymbol{Z}_{M_i}(j_1)) = G_M(\boldsymbol{Z}_{M_i}(j_1), t_1, \boldsymbol{X}_i; \hat{\theta}_{G_M}). \tag{26}$$

Both $\hat{\boldsymbol{M}}_i(t_0, \boldsymbol{Z}_{M_i}(j_0))$ and $\hat{\boldsymbol{M}}_i(t_1, \boldsymbol{Z}_{M_i}(j_1))$ are p-dimensional vectors, which represent predictions for the p mediators based on the empirical distribution. By averaging these generated mediator values, we can compute the filtered factual for $\boldsymbol{M}_i$.

$$\hat{\boldsymbol{M}}_i(t_0) = [\hat{M}_i^1(t_0), \hat{M}_i^2(t_0), \ldots, \hat{M}_i^p(t_0)] = \frac{1}{N_g}\sum_{j_0=1}^{N_g} \hat{\boldsymbol{M}}_i(t_0, \boldsymbol{Z}_{M_i}(j_0)), \tag{27}$$

$$\hat{\boldsymbol{M}}_i(t_1) = [\hat{M}_i^1(t_1), \hat{M}_i^2(t_1), \ldots, \hat{M}_i^p(t_1)] = \frac{1}{N_g}\sum_{j_1=1}^{N_g} \hat{\boldsymbol{M}}_i(t_1, \boldsymbol{Z}_{M_i}(j_1)), \tag{28}$$

Based on the expressions in (25) and (26), we can derive the counterfactual associated with $\boldsymbol{M}_i^{(-k)}$ by

$$\begin{aligned} \hat{\boldsymbol{M}}_i^{(-k)}(t_1) &= [\hat{M}_i^1(t_1), \ldots, \hat{M}_i^{k-1}(t_1), \hat{M}_i^k(t_0), \hat{M}_i^{k+1}(t_1) \ldots, \hat{M}_i^p(t_1)] \\ &= \hat{\boldsymbol{M}}_i(t_1) \odot \boldsymbol{T}_k + \hat{\boldsymbol{M}}_i(t_0) \odot (\boldsymbol{1} - \boldsymbol{T}_k), \end{aligned} \tag{29}$$

where $\boldsymbol{T}_k$ is a p-dimensional vector, the p-th element of which is 0, and the other $p-1$ elements are set to 1. $\boldsymbol{1}$ represents a p-dimensional vector with all elements equal to 1. The symbol $\odot$ denotes the Hadamard product (element-wise product). Figure 3 also helps illustrate our design. The yellow blocks in varying shades represent the generated mediators, with $\hat{\boldsymbol{M}}_i(t_0)$ marked with a hatching pattern, while $\hat{\boldsymbol{M}}_i(t_1)$ remains without it for distinction. For each $k = 1, 2, \ldots, p$, the k-th element of $\hat{\boldsymbol{M}}_i(t_1)$ is replaced by the k-th element of $\hat{\boldsymbol{M}}_i(t_0)$ to obtain $\hat{\boldsymbol{M}}_i^{(-k)}(t_1)$. This is visually represented in the figure by the hatching pattern along the diagonal line.

The direct effect. According to (7), the direct effects for the i-th individual can be generated by

$$\triangle_{i,T\to Y}(j) = \hat{Y}_i(t_1, \hat{\boldsymbol{M}}_i(t_0), \boldsymbol{Z}_{Y_i}^1(j)) - \hat{Y}_i(t_0, \hat{\boldsymbol{M}}_i(t_0), \boldsymbol{Z}_{Y_i}^2(j)), \tag{30}$$

where $\boldsymbol{Z}_{Y_i}^1(j)$ and $\boldsymbol{Z}_{Y_i}^2(j)$ $(j = 1, 2, \ldots N_g)$ are independently sampled from the multivariate normal distribution $N(\boldsymbol{0}, \boldsymbol{I}_{d_Y})$. $\triangle_{i,T\to Y}(j)$ represents the individual direct effect corresponding to $\boldsymbol{Z}_{Y_i}^1(j)$ and $\boldsymbol{Z}_{Y_i}^2(j)$. The counterfactual outcomes $\hat{Y}_i(t_1, \hat{\boldsymbol{M}}_i(t_0), \boldsymbol{Z}_{Y_i}^1(j))$s

and $\hat{Y}_i(t_0, \hat{\boldsymbol{M}}_i(t_0), \boldsymbol{Z}^2_{Y_i}(j))$s are used to represent the generated values of Y_i under varying treatment conditions using G_Y, and can be computed as follows:

$$\hat{Y}_i(t_1, \hat{\boldsymbol{M}}_i(t_0), \boldsymbol{Z}^1_{Y_i}(j)) = G_Y(\boldsymbol{Z}^1_{Y_i}(j), \hat{\boldsymbol{M}}_i(t_0), t_1, \boldsymbol{X}(i); \hat{\theta}_{G_Y}), \tag{31}$$

$$\hat{Y}_i(t_0, \hat{\boldsymbol{M}}_i(t_0), \boldsymbol{Z}^2_{Y_i}(j)) = G_Y(\boldsymbol{Z}^2_{Y_i}(j), \hat{\boldsymbol{M}}_i(t_0), t_0, \boldsymbol{X}(i); \hat{\theta}_{G_Y}). \tag{32}$$

Based on (30), the average direct effects corresponding to different noises can be calculated as

$$\triangle_{T\to Y}(j) = \frac{1}{n}\sum_{i=1}^{n}\left(\hat{Y}_i(t_1, \hat{\boldsymbol{M}}_i(t_0), \boldsymbol{Z}^1_{Y_i}(j)) - \hat{Y}_i(t_0, \hat{\boldsymbol{M}}_i(t_0), \boldsymbol{Z}^2_{Y_i}(j))\right). \tag{33}$$

Hence, the direct effect can be estimated as

$$\begin{aligned}\triangle_{T\to Y} &= \frac{1}{N_g}\sum_{j=1}^{N_g}\triangle_{T\to Y}(j) \\ &= \frac{1}{N_g}\frac{1}{n}\sum_{j=1}^{N_g}\sum_{i=1}^{n}\left(\hat{Y}_i(t_1, \hat{\boldsymbol{M}}_i(t_0), \boldsymbol{Z}^1_{Y_i}(j)) - \hat{Y}_i(t_0, \hat{\boldsymbol{M}}_i(t_0), \boldsymbol{Z}^2_{Y_i}(j))\right).\end{aligned} \tag{34}$$

The total indirect effect. According to (8), using $\hat{\boldsymbol{M}}_i(t_1)$, the total indirect effects for the ith individual can be generated by:

$$\triangle_{i,T\to \boldsymbol{M}\to Y}(j) = \hat{Y}_i(t_1, \hat{\boldsymbol{M}}_i(t_1), \boldsymbol{Z}^3_{Y_i}(j)) - \hat{Y}_i(t_1, \hat{\boldsymbol{M}}_i(t_0), \boldsymbol{Z}^4_{Y_i}(j)), \tag{35}$$

where $\boldsymbol{Z}^3_{Y_i}(j)$ and $\boldsymbol{Z}^4_{Y_i}(j)$ $(j = 1, 2, \ldots N_g)$ are independently sampled from the multivariate normal distribution $N(\boldsymbol{0}, \boldsymbol{I}_{d_Y})$. $\triangle_{i,T\to \boldsymbol{M}\to Y}(j)$ denotes the total individual indirect effect associated with $\boldsymbol{Z}^3_{Y_i}(j)$ and $\boldsymbol{Z}^4_{Y_i}(j)$. The counterfactual outcomes, $\hat{Y}_i(t_1, \hat{\boldsymbol{M}}_i(t_1), \boldsymbol{Z}^3_{Y_i}(j))$s and $\hat{Y}_i(t_0, \hat{\boldsymbol{M}}_i(t_0), \boldsymbol{Z}^4_{Y_i}(j))$s, are computed as follows:

$$\hat{Y}_i(t_1, \hat{\boldsymbol{M}}_i(t_1), \boldsymbol{Z}^3_{Y_i}(j)) = G_Y(\boldsymbol{Z}^3_{Y_i}(j), \hat{\boldsymbol{M}}_i(t_1), t_1, \boldsymbol{X}(i); \hat{\theta}_{G_Y}), \tag{36}$$

$$\hat{Y}_i(t_1, \hat{\boldsymbol{M}}_i(t_0), \boldsymbol{Z}^4_{Y_i}(j)) = G_Y(\boldsymbol{Z}^4_{Y_i}(j), \hat{\boldsymbol{M}}_i(t_0), t_1, \boldsymbol{X}(i); \hat{\theta}_{G_Y}). \tag{37}$$

Based on (35), the average total indirect effects corresponding to different noises can be calculated as

$$\triangle_{T\to \boldsymbol{M}\to Y}(j) = \frac{1}{n}\sum_{i=1}^{n}\left(\hat{Y}_i(t_1, \hat{\boldsymbol{M}}_i(t_1), \boldsymbol{Z}^3_{Y_i}(j)) - \hat{Y}_i(t_1, \hat{\boldsymbol{M}}_i(t_0), \boldsymbol{Z}^4_{Y_i}(j))\right). \tag{38}$$

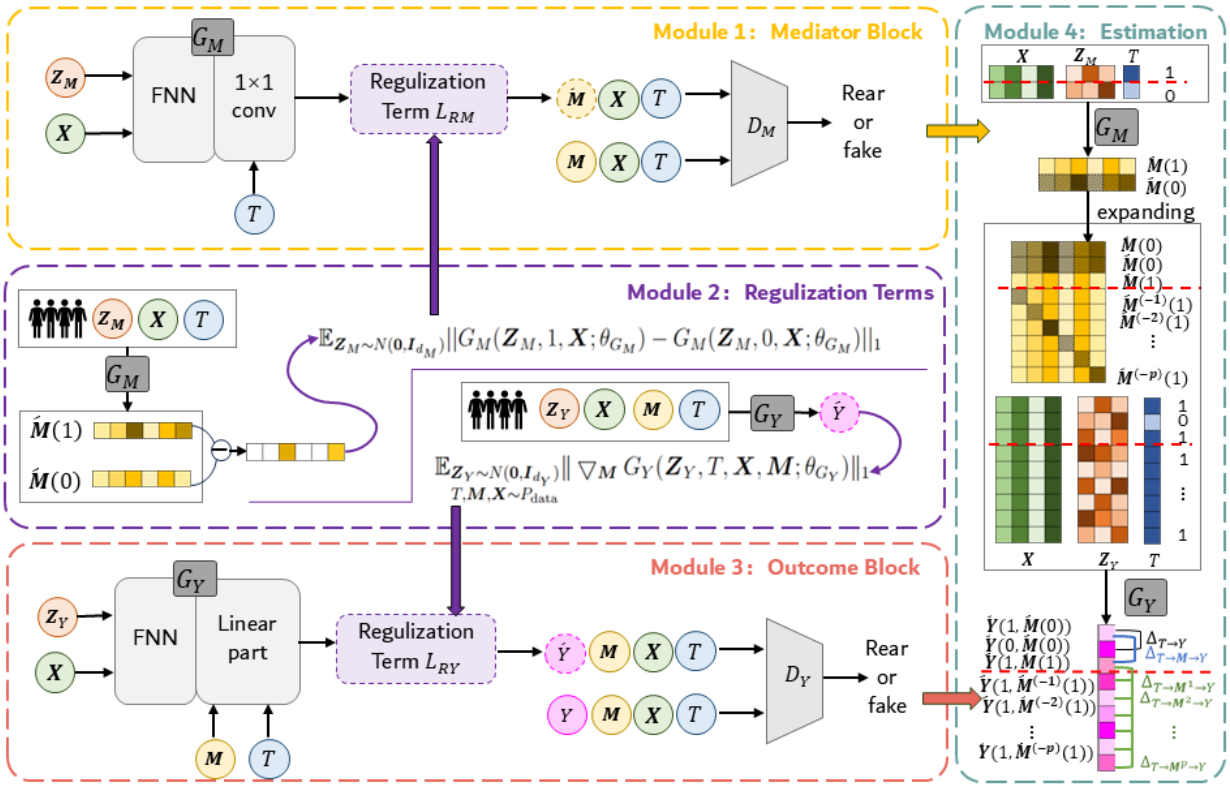


Figure 3: Block Diagram of GAHMN

Then, the total indirect effect can be calculated as

$$\begin{aligned}\triangle_{T\to \boldsymbol{M}\to Y} &= \frac{1}{N_g}\sum_{j=1}^{N_g}\triangle_{T\to \boldsymbol{M}\to Y}(j)\\ &= \frac{1}{N_g}\frac{1}{n}\sum_{j=1}^{N_g}\sum_{i=1}^{n}\left(\hat{Y}_i(t_1,\hat{\boldsymbol{M}}_i(t_1),\boldsymbol{Z}^3_{Y_i}(j))-\hat{Y}_i(t_1,\hat{\boldsymbol{M}}_i(t_0),\boldsymbol{Z}^4_{Y_i}(j))\right).\end{aligned}\quad(39)$$

The indirect effects through a given mediator. For $\forall k$ $(k=1,2,\ldots,p)$, with $\hat{\boldsymbol{M}}_i(t_1)$ and $\hat{\boldsymbol{M}}_i^{(-k)}(t_1)$, according to (9), the individual indirect effects implemented through the k-th mediator M^k can be generated by

$$\triangle_{i,T\to M^p\to Y}(j)=\hat{Y}_i(t_1,\hat{\boldsymbol{M}}_i(t_1),\boldsymbol{Z}^{k(1)}_{Y_i}(j))-\hat{Y}_i(t_1,\hat{\boldsymbol{M}}_i^{(-k)}(t_1),\boldsymbol{Z}^{k(2)}_{Y_i}(j)).\quad(40)$$

Here, $\boldsymbol{Z}^{k(1)}_{Y_i}(j)$ and $\boldsymbol{Z}^{k(2)}_{Y_i}(j)$ $(j=1,2,\ldots,N_g)$ represent two independent sets of noise sampled from $N(\boldsymbol{0},\boldsymbol{I}_{d_Y})$. $\triangle_{i,T\to M^k\to Y}(j)$ denotes the individual indirect effects associated with $\boldsymbol{Z}^{k(1)}_{Y_i}(j)$ and $\boldsymbol{Z}^{k(2)}_{Y_i}(j)$. The counterfactual outcomes, $\hat{Y}_i(t_1,\hat{\boldsymbol{M}}_i(t_1),\boldsymbol{Z}^{k(1)}_{Y_i}(j))$s and

$\hat{Y}_i(t_1, \hat{\boldsymbol{M}}_i^{(-k)}(t_1), \boldsymbol{Z}_{Y_i}^{k(2)}(j))$s, are computed as follows:

$$\hat{Y}_i(t_1, \hat{\boldsymbol{M}}_i(t_1), \boldsymbol{Z}_{Y_i}^{k(1)}(j)) = G_Y(\boldsymbol{Z}_{Y_i}^{k(1)}(j), \hat{\boldsymbol{M}}_i(t_1), t_1, \boldsymbol{X}(i); \hat{\theta}_{G_Y}), \tag{41}$$

$$\hat{Y}_i(t_1, \hat{\boldsymbol{M}}_i^{(-k)}(t_1), \boldsymbol{Z}_{Y_i}^{k(2)}(j)) = G_Y(\boldsymbol{Z}_{Y_i}^{k(1)}(j), \hat{\boldsymbol{M}}_i^{(-k)}(t_1), t_1, \boldsymbol{X}(i); \hat{\theta}_{G_Y}). \tag{42}$$

The average indirect effects through the kth mediator corresponding to different noises can be calculated as

$$\triangle_{T\to M^k\to Y}(j) = \frac{1}{n}\sum_{i=1}^{n}\left(\hat{Y}_i(t_1, \hat{\boldsymbol{M}}_i(t_1), \boldsymbol{Z}_{Y_i}^{k(1)}(j)) - \hat{Y}_i(t_1, \hat{\boldsymbol{M}}_i^{(-k)}(t_1), \boldsymbol{Z}_{Y_i}^{k(2)}(j))\right). \tag{43}$$

Then, the indirect effect through the kth mediator can be estimated as

$$\begin{aligned}\triangle_{T\to M^k\to Y} &= \frac{1}{N_g}\sum_{j=1}^{N_g}\triangle_{T\to M^k\to Y}(j)\\ &= \frac{1}{N_g}\frac{1}{n}\left(\hat{Y}_i(t_1, \hat{\boldsymbol{M}}_i(t_1), \boldsymbol{Z}_{Y_i}^{k(1)}(j)) - \hat{Y}_i(t_1, \hat{\boldsymbol{M}}_i^{(-k)}(t_1), \boldsymbol{Z}_{Y_i}^{k(2)}(j))\right).\end{aligned} \tag{44}$$

Notably, our architecture also supports the processing of multiple samples in parallel. By adding an additional sample dimension to the tensor, we can handle large-scale data efficiently. Furthermore, it is well-suited for end-to-end tasks, allowing for integrated causal inference within deep learning models. The block diagram of GAHMN is summarized and presented in Figure 3 .

3.5 Further discussions

The existing high-dimensional mediation methods often impose rigid structural and distributional assumptions, which limits them to handle complex characteristics datasets. The proposed GAHMN can effectively overcome these limitations.

Modeling correlations and heterogeneity. As shown in Section 2.1 and 2.2, the traditional methods often impose strict distributional assumptions on random terms to facilitate model estimation. For instance, likelihood-based approaches typically assume that the error terms $\boldsymbol{\epsilon_1}$ and ϵ_2 in equations (1) and (4) follow a normal distribution and exhibit homoscedasticity to ensure valid statistical inference. In contrast to the linear and additive formulation of $\boldsymbol{\epsilon_1}$ and ϵ_2 in conventional models, our proposed GAHMN introduces multi-dimensional random vectors $\boldsymbol{Z}_M$ and $\boldsymbol{Z}_Y$ to represent random components. Leveraging the expressive power of deep neural networks, $\boldsymbol{Z}_M$ can be transformed to approximate

an arbitrary joint distribution of $\boldsymbol{\epsilon_1}$, while $\boldsymbol{Z}_Y$ can approximate any distribution of ϵ_2. As a result, our model does not require $\boldsymbol{\epsilon_1}$ and ϵ_2 to follow a normal distribution, and the correlations among the elements of $\boldsymbol{\epsilon_1}$ can be naturally captured within GAHMN. By comparison, traditional approaches must explicitly specify a $p \times p$ covariance matrix to account for correlations, which significantly increases model complexity and the number of parameters. This unique structure of GAN-based models offers substantial advantages in modeling correlation structures. Furthermore, as illustrated in Figure 1 and Figure 2, the inputs $\boldsymbol{Z}_M$, $\boldsymbol{Z}_Y$, and $\boldsymbol{X}$ are all located in the input layer of the network, allowing for interactions between random components and covariates. This enables GAHMN to flexibly capture complex heterogeneous patterns and significantly relaxes the rigid assumptions required by conventional methods.

Dealing with a large number of mediators flexibly. In equations (1) and (4), each mediator M_i $(i = 1, 2, \ldots, p)$ corresponds to an independent sub-equation. From a network design perspective, this would suggest constructing a separate CGAN for each mediator to mirror these equations. However, building $p + 1$ individual GANs would greatly increase model complexity and require solving $p + 1$ min-max optimization problems simultaneously—making the estimation process computationally impractical. To address this challenge, the generator G_M in GAHMN incorporates a 1×1 convolutional layer, enabling all mediators to share the same hidden layers and network parameters related to the input variables $\boldsymbol{Z}_M$ and $\boldsymbol{X}$. In the final hidden layer, shared features are extracted and then routed into p distinct output channels, each with its own set of parameters. This design allows us to model all mediators using a single CGAN rather than p separate ones, while still maintaining the independence of treatment effects across mediators and enabling the generation of mediator-specific counterfactuals. Furthermore, this architecture allows all mediators to share the same random noise input $\boldsymbol{Z}_M$, which helps capture potential correlations among components of the error term $\boldsymbol{\epsilon_1}$. By sharing parameters in the early stages of the network, the model substantially reduces computational complexity and enhances both scalability and estimation feasibility.

Enhancing the generalization ability of network. For G_Y and a given treatment $T \in \{t_0^\star, t_1^\star\}$, we can only observe samples of the form $(Y, \boldsymbol{X}, \boldsymbol{M}(t_0^\star), t_0^\star)$ or $(Y, \boldsymbol{X}, \boldsymbol{M}(t_1^\star), t_1^\star)$. The counterfactuals $(Y, \boldsymbol{X}, \boldsymbol{M}(t_1^\star), t_0^\star)$ and $(Y, \boldsymbol{X}, \boldsymbol{M}(t_0^\star), t_1^\star)$ are unobservable. As a continuous neural network, G_Y is expected to infer these counterfactuals by extrapolating

from its learned mappings over $(\boldsymbol{X}, \boldsymbol{M}(t_1^\star), t_0^\star)$ and $(\boldsymbol{X}, \boldsymbol{M}(t_0^\star), t_1^\star)$. However, the distances between observed samples and their corresponding counterfactual configurations, such as $||(\boldsymbol{M}(t_1^\star), t_0^\star) - (\boldsymbol{M}(t_0^\star), t_0^\star)||$, $||(\boldsymbol{M}(t_1^\star), t_0^\star) - (\boldsymbol{M}(t_1^\star), t_1^\star)||$, $||(\boldsymbol{M}(t_0^\star), t_1^\star) - (\boldsymbol{M}(t_0^\star), t_0^\star)||$ and $||(\boldsymbol{M}(t_0^\star), t_1^\star) - (\boldsymbol{M}(t_0^\star), t_0^\star)||$, can be quite large, especially when $\boldsymbol{M}$ is high-dimensional. This indicates that the observed samples used to train G_Y may be significantly different from the counterfactual configurations in terms of $(\boldsymbol{M}, T)$, making extrapolation by G_Y along $(\boldsymbol{M}, T)$ direction both challenging and potentially inaccurate. To alleviate this issue and encourage smoother continuation, we adopt two key design strategies. First, following the well-known Occam's Razor principle in machine learning, we favor a simpler architecture for $\boldsymbol{M}$ and T. Specifically, we place $\boldsymbol{Z}_Y$ and $\boldsymbol{X}$ in the deeper layers of G_Y, while placing $\boldsymbol{M}$ and T in the shallow layers as high-dimensional linear components. This design reduces unnecessary complexity in the model's representation. Second, we introduce a regularization term on the partial derivatives of G_Y with respect to $\boldsymbol{M}$, as shown in equations (18) and (19). This regularization encourages smoothness of the model along the $\boldsymbol{M}$ direction, which helps improve the accuracy of counterfactual predictions. These two carefully designed strategies jointly enhance the generalization ability of our model.

Handling the sparcity of mediators. The outcome block of GAHMN is optimized according to the loss function in equation (20). The first two terms of (20) correspond to the standard loss used in CGAN, which ultimately minimizes the conditional JS divergence between the generated and real outcome data. The third term serves as a regularization component that penalizes the partial derivative of the generator G_Y with respect to the mediator variable $\boldsymbol{M}$. On the one hand, under our partially linear assumption for the high-dimensional component of G_Y, the term $|\nabla_{\boldsymbol{M}} G_Y(\boldsymbol{Z}_{Y_i}, T_i, \boldsymbol{X}_i, \boldsymbol{M}i; \theta G_Y)|_1$ simplifies to $\sum_{k=1}^{p} |b_k|$, where b_k denotes the linear coefficients corresponding to each mediator. This directly mirrors the ℓ_1-norm penalty used in LASSO regression, which is known to promote sparsity in high-dimensional settings. On the other hand, it can be shown that when the error term $\boldsymbol{\epsilon}_2$ follows a Gaussian distribution, the conditional JS divergence induced by the first two terms of the loss function becomes convex with respect to the high-dimensional coefficients of the mediators $\boldsymbol{M}$. By appropriately tuning the regularization hyperparameter, the overall objective function in equation (20) effectively encourages sparsity in the coefficients of the mediators, similar to how LASSO operates. Specifically, mediators with negligible contributions to the outcome are shrunk toward zero, enabling automatic variable

selection in the presence of high-dimensional mediators. The partially high-dimensional network structure and regularization thus play crucial roles: stabilizing the adversarial training and yielding sparse, interpretable mediation pathways.

4 Theoretical view

By comparing (1) with its GAN-based counterpart (23), it follows that the estimation of mediation effects within GAHMN is fundamentally governed by the estimation of its linear parameters. Consequently, if the linear parameters converge to their true values, the mediation effect estimates will also converge. Since both the mediator block and the outcome block can be represented within a general partially linear framework, we propose a unified approach to establish their convergence results. Without loss of generality, let $\boldsymbol{V}$ and $\boldsymbol{X}$ denote the conditioning variables, Y denote the outcome variable. The generating processes for both blocks can be expressed in the following partially linear form.

$$Y = \boldsymbol{\beta}^T \boldsymbol{V} + g(\boldsymbol{X}, \varepsilon), \tag{45}$$

where ε is an unknown random term, and g is an unknown nonlinear term. For fitting the process by our proposed approach, we have generating function as follows

$$\hat{Y} = G(\boldsymbol{V}, \boldsymbol{X}, \boldsymbol{Z}) = \hat{\boldsymbol{\beta}}^T \boldsymbol{V} + G_F(\boldsymbol{X}, \boldsymbol{Z}), \tag{46}$$

where $\boldsymbol{Z}$ is standard normally or uniformly distributed, and G_F is an FNN that is flexible to accommodate the nonlinear components of mediator and outcome blocks. For simplicity, Y is set be one-dimensional.

Let $P_{\boldsymbol{V},\boldsymbol{X},\hat{Y}}$ and $P_{\boldsymbol{V},\boldsymbol{X},Y}$ be the densities of $(\boldsymbol{V}, \boldsymbol{X}, \hat{Y})$ and $(\boldsymbol{V}, \boldsymbol{X}, Y)$, respectively. At population level, we attempt to obtain an optimal generator $G^\star$ and its associated $\boldsymbol{\beta}^\star$ that mimimize the JS-divergence $\mathcal{D}_{JS}(P_{\boldsymbol{V},\boldsymbol{X},\hat{Y}} || P_{\boldsymbol{V},\boldsymbol{X},Y})$.

Lemma 1. Let $G^\star$ be the minimimizer of the JS-divergence $\mathcal{D}_{JS}(P_{\boldsymbol{V},\boldsymbol{X},\hat{Y}} || P_{\boldsymbol{V},\boldsymbol{X},Y})$.

$$G^\star \in \arg\min_G \mathcal{D}_{JS}(P_{\boldsymbol{V},\boldsymbol{X},\hat{Y}} || P_{\boldsymbol{V},\boldsymbol{X},Y})$$

if and only if $P_{\boldsymbol{V},\boldsymbol{X},\hat{Y}} = P_{\boldsymbol{V},\boldsymbol{X},Y}$, which also implies $\boldsymbol{\beta}^\star = \boldsymbol{\beta}$.

Proof: The total variation norm of $P_{\boldsymbol{V},\boldsymbol{X},\hat{Y}} - P_{\boldsymbol{V},\boldsymbol{X},Y}$ is given by

$$\begin{aligned} D_{TV}(P_{\boldsymbol{V},\boldsymbol{X},\hat{Y}}, P_{\boldsymbol{V},\boldsymbol{X},Y}) &= \frac{1}{2} || P_{\boldsymbol{V},\boldsymbol{X},\hat{Y}} - P_{\boldsymbol{V},\boldsymbol{X},Y} ||_1 \\ &= \frac{1}{2} \int_{\mathcal{V},\mathcal{X},\mathcal{Y}} \left| P_{\boldsymbol{V},\boldsymbol{X},\hat{Y}}(v,x,y) - P_{\boldsymbol{V},\boldsymbol{X},Y}(v,x,y) \right| dz dx dy, \end{aligned} \tag{47}$$

where $||\cdot||_1$ denotes the ℓ_1-norm. We can bound the total variation (47) by its JS divergence, which gives

$$\begin{aligned}
& \mathcal{D}_{JS}(P_{\boldsymbol{V},\boldsymbol{X},\hat{Y}}||P_{\boldsymbol{V},\boldsymbol{X},Y}) \\
= \; & \frac{1}{2}D_{KL}\left(P_{\boldsymbol{V},\boldsymbol{X},\hat{Y}}\|\frac{P_{\boldsymbol{V},\boldsymbol{X},Y}+P_{\boldsymbol{V},\boldsymbol{X},\hat{Y}}}{2}\right)+\frac{1}{2}D_{KL}\left(P_{\boldsymbol{V},\boldsymbol{X},Y}\|\frac{P_{\boldsymbol{V},\boldsymbol{X},Y}+P_{\boldsymbol{V},\boldsymbol{X},\hat{Y}}}{2}\right) \\
\geq \; & \frac{1}{4}D^2_{TV}\left(P_{\boldsymbol{V},\boldsymbol{X},\hat{Y}},\frac{P_{\boldsymbol{V},\boldsymbol{X},Y}+P_{\boldsymbol{V},\boldsymbol{X},\hat{Y}}}{2}\right)+\frac{1}{4}D^2_{TV}\left(P_{\boldsymbol{V},\boldsymbol{X},Y},\frac{P_{\boldsymbol{V},\boldsymbol{X},Y}+P_{\boldsymbol{V},\boldsymbol{X},\hat{Y}}}{2}\right) \\
= \; & \frac{1}{8}\left\|\frac{P_{\boldsymbol{V},\boldsymbol{X},\hat{Y}}-P_{\boldsymbol{V},\boldsymbol{X},Y}}{2}\right\|_1^2 \\
= \; & \frac{1}{8}D^2_{TV}(P_{\boldsymbol{V},\boldsymbol{X},\hat{Y}},P_{\boldsymbol{V},\boldsymbol{X},Y}), \qquad (48)
\end{aligned}$$

where the inequality follows Pinsker's inequalities. Then, if $\mathcal{D}_{JS}(P_{\boldsymbol{V},\boldsymbol{X},\hat{Y}}||P_{\boldsymbol{V},\boldsymbol{X},Y})=0$, $P_{\boldsymbol{V},\boldsymbol{X},\hat{Y}}=P_{\boldsymbol{V},\boldsymbol{X},Y}$ almost everywhere. For any given $\boldsymbol{V}=v$, the conditional density of Y and $\hat{Y}$ are denoted by $P_{Y|\boldsymbol{V}=v}(u)$ and $P_{\hat{Y}|\boldsymbol{V}=v}(u)$, respectively. It is noticed that

$$Y=\boldsymbol{\beta}^T\boldsymbol{V}+g(\boldsymbol{X},\varepsilon), \;\text{ and }\; \hat{Y}=\hat{\boldsymbol{\beta}}^T\boldsymbol{V}+G_F(\boldsymbol{X},\eta), \qquad (49)$$

which implies

$$P_{Y|\boldsymbol{V}=v}(u)=P_{g(\boldsymbol{X},\varepsilon)}(u-\boldsymbol{\beta}^Tv) \;\text{ and }\; P_{\hat{Y}|\boldsymbol{V}=v}(u)=P_{G(\boldsymbol{X},\eta)}(u-\hat{\boldsymbol{\beta}}^Tv). \qquad (50)$$

Since $p_{\boldsymbol{V},\boldsymbol{X},\hat{Y}}=p_{\boldsymbol{V},\boldsymbol{X},Y}$, it follows

$$P_{Y|\boldsymbol{V}=v}(u)=P_{\hat{Y}|\boldsymbol{V}=v}(u) \;\text{ and }\; P_{g(\boldsymbol{X},\varepsilon)}(u-\boldsymbol{\beta}^Tv)=P_{G(\boldsymbol{X},\eta)}(u-\hat{\boldsymbol{\beta}}^Tv). \qquad (51)$$

Let $m=u-\boldsymbol{\beta}^Tv$, we have

$$P_{g(\boldsymbol{X},\varepsilon)}(m)=P_{G(\boldsymbol{X},\eta)}(m+(\boldsymbol{\beta}^T-\hat{\boldsymbol{\beta}}^T)v). \qquad (52)$$

Since for $\forall\boldsymbol{V}=v$, (52) holds. By taking partial derivatives with respect to $\boldsymbol{V}$ on both sides of (52), we have

$$\begin{aligned}
\frac{\partial P_{g(\boldsymbol{X},\varepsilon)}(m)}{\partial\boldsymbol{V}} &= \frac{\partial P_{G(\boldsymbol{X},\eta)}(m+(\boldsymbol{\beta}^T-\hat{\boldsymbol{\beta}}^T)\boldsymbol{V})}{\partial\boldsymbol{V}}\bigg|_{\boldsymbol{V}=v} \\
&= (\boldsymbol{\beta}^T-\hat{\boldsymbol{\beta}}^T)P'_{G(\boldsymbol{X},\eta)}(m+(\boldsymbol{\beta}^T-\hat{\boldsymbol{\beta}}^T)\boldsymbol{V})\bigg|_{\boldsymbol{V}=v}. \qquad (53)
\end{aligned}$$

Since (52) and (53) hold for $\forall\boldsymbol{V}=v$, $\frac{\partial P_{g(X,\varepsilon)}(m)}{\partial\boldsymbol{V}}=0$, we further have $\boldsymbol{\beta}^T-\hat{\boldsymbol{\beta}}^T=0$, which yields $\boldsymbol{\beta}^\star=\boldsymbol{\beta}$. Then, by (52), we also have $P_{g(\boldsymbol{X},\varepsilon)}=P_{G(\boldsymbol{X},\eta)}$. The proof is completed. In the Appendix, we further show the convergence results of $P_{\boldsymbol{V},\boldsymbol{X},\hat{Y}}\to P_{\boldsymbol{V},\boldsymbol{X},Y}$ as $n\to\infty$.

5 Numerical Experiments

5.1 Simulation studies

To evaluate our proposed method in terms of variable selection, estimation and inference performance in finite samples, we conducted several simulation studies with a sample size of $n = 1000$ and mediator dimension $p = 500$. The data-generating process is defined as:

$$\begin{cases} M_{ik} = a_k T_i + g_k(X_{i1}, X_{i2}) + \epsilon_{ik}, \\ Y_i = 2 - 0.5T_i + \sum_{k=1}^{p} b_k M_{ik} - 2\sqrt{X_{i1} + 5} + \sin(1.5X_{i2}) + \varepsilon_i, \end{cases} \tag{54}$$

where $k = 1, 2, \cdots, p$ and $i = 1, 2, \cdots, n$. The binary treatment T_i is assigned values 0 or 1 with equal probability. The two covariates, X_{i1} and X_{i2}, are independently drawn from the standard normal distribution $N(0, 1)$. We denote the vector of noise terms in the first equation of (54) as $\boldsymbol{\epsilon}_i = [\epsilon_{i1}, \epsilon_{i2}, \cdots, \epsilon_{ip}]$, where $\boldsymbol{\epsilon_i} \sim N(0, \Sigma^*)$. The covariance matrix Σ^* combines autoregressive (AR) correlation and heteroskedasticity. Specifically, its (i_0, j_0)-th element is given by $\rho^{|i_0 - j_0|} * (0.1 + 0.3|X_{i1}| + 0.3T_i)$. While the noise term of the second equation of (54) ε_i is independently drawn from $N(0, 1)$. To evaluate the capability of our method in modeling nonlinear relationships, we define the function $g_k(X_{i1}, X_{i2})$ as

$$g_k(X_{i1}, X_{i2}) = \delta_{k1} X_{i1} + \delta_{k2} X_{i2} + \delta_{k3} X_{i1}^2 + \delta_{k4} X_{i2}^2 + \delta_{k5} X_{i1} X_{i2}, \tag{55}$$

where the coefficients δ_{k1}, δ_{k2}, δ_{k3}, δ_{k4} and δ_{k5} are independently drawn from a uniform distribution $U[-1, 1]$. The other parameters, a_ks and b_ks, are configured according to the following two settings:

A. $\boldsymbol{a}_{1:p} = [1.6, 1.2, 0.8, 0.4, 1.6, 0, 0, 0, 0, \ldots, 0]$,$\boldsymbol{b}_{1:p} = [0, 0, 1, 1, 1, 1, 1, 0, 0, \ldots, 0]$.

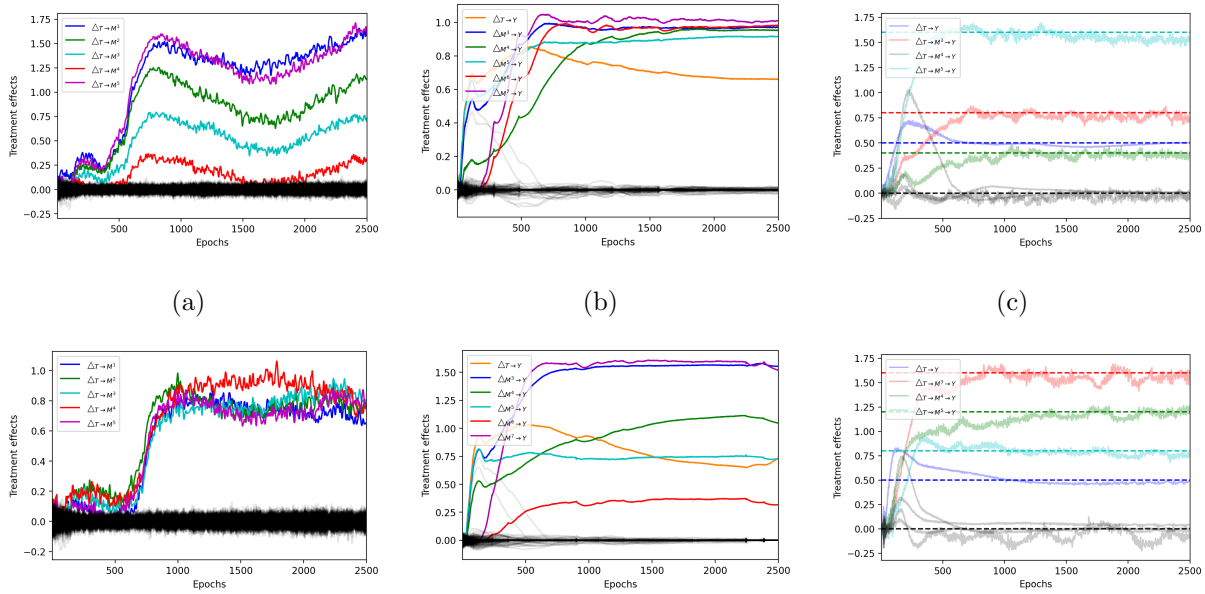
B. $\boldsymbol{a}_{1:p} = [1, 1, 1, 1, 1, 0, 0, 0, 0, \ldots, 0]$, $\boldsymbol{b}_{1:p} = [0, 0, 1.6, 1.2, 0.8, 0.4, 1.6, 0, 0, \ldots, 0]$.

In both settings, the first five elements of $\boldsymbol{a}_{1:p}$ are nonzero, and the 3rd to 7th elements of $\boldsymbol{b}_{1:p}$ are nonezero, which means only the indirect pathways through M^3, M^4 and M^5 contribute to the mediation effect.

Based on the architectures depicted in Figures 1 and 2, our HMA for (54) is designed as follows. In G_M, the network structure associated with $\boldsymbol{Z}_M$ and $\boldsymbol{X}$ is designed as a 4-layer neural network. The second and third layers are fully connected, comprising 32 and 128

neurons, respectively. The 128-neuron hidden layer is then reshaped into a 3D tensor of shape $128 \times 1 \times 1$, where 128 represents the number of channels. The auxiliary variable T is appended as an additional channel, forming a $129 \times 1 \times 1$ tensor. The final layer consists of a convolutional layer with a $129 \times 1 \times 500$ kernel, mapping the $129 \times 1 \times 1$ tensor to a 3D $500 \times 1 \times 1$ tensor representing the generated mediators $\boldsymbol{M}$. In G_Y, the linear components of the network are constructed by $\boldsymbol{M}$ and T as illustrated in Figure 2. The module corresponding to $\boldsymbol{Z}_Y$ and $\boldsymbol{X}$ is designed as a 4-layer neural network, where the second and third hidden layers contain 32 and 128 neurons, respectively. Both $\boldsymbol{Z}_M$ and $\boldsymbol{Z}_Y$ are independently sampled from the standard bivariate normal distribution $N((0,1)^2)$. To produce continuous outcomes, the output layers of G_M and G_Y adopt the identity activation function. The discriminators D_M and D_Y share the same architecture except for their input variables. Specifically, D_M receives $\boldsymbol{X}$, T and $\boldsymbol{M}$ as inputs, while D_Y takes $\boldsymbol{X}$, T, Y and $\boldsymbol{M}$. Both D_M and D_Y contain 128 and 32 neurons in their second and third hidden layers, respectively. The sigmoid function is employed at the output layer to distinguish real samples from generated ones. The only distinction lies in the output layer: D_M has 500 output units to assess each component of $\boldsymbol{M}$, individually, whereas D_Y has a single output unit. All hidden layers in G_M, G_Y, D_M and D_Y use the Leaky ReLU activation function with a negative slope coefficient of 0.2. In the first experiment, we randomly split the data into a training set of 1,600 samples (80%) and a testing set of 400 samples (20%). The HMA model is trained using the training data and then applied to estimate treatment effects on the testing set. To achieve satisfactory performance, the models are optimized using the Adam algorithm with a learning rate of $\eta = 0.0005$, and trained for 2,500 epochs.

For each simulation setting, we first perform a preliminary screening of potential mediators, followed by a re-estimation of the mediation effects based on the selected candidates. During the screening part, the regularization parameter λ is set to 0.0005, while in the re-estimation part it is set to 0 to ensure unbiased estimation. Figure 4 illustrates the learning dynamics of our model. The top row of subfigures corresponds to the results under Setting A, while the bottom row presents those under Setting B. Figure 4(a) and Figure 4(d) show how G_M identifies T the components of $\boldsymbol{M}$ that are affected by T. Figure 4(b) and Figure 4(e) depict how G_Y selects the mediators within $\boldsymbol{M}$ that have an impact on Y. As designed in our simulation settings A and B, the active mediators $\boldsymbol{M}$ are success-



(a) (b) (c)

(d) (e) (f)

Figure 4: The leaning performance on testing set for each epoch.

fully identified in both cases. Subsequently, we re-estimate the mediation effects using the selected seven candidate mediators $(M^1, M^2, \cdots, M^7)$. Figure 4(c) and Figure 4(f) depict the convergence curves of estimated direct effect $\triangle_{T\to Y}$ given by (34) and indirect effects $\triangle_{T\to M^k\to Y}$ $(k = 1, 2, \cdots, 7)$ given by (44) on the testing set. Here we set $N_g = 1$ to reduce computational cost. The true target values are indicated by dotted lines. As shown, the estimates produced by our model converge to the true values after approximately 1000 epochs, demonstrating the effectiveness and stability of our proposed method.

To evaluate the performance of our method under varying levels of sparsity in $\boldsymbol{M}$, we consider four additional settings, labeled C, D, E, and F:

C. $\boldsymbol{a}_{1:p} = [1.6, 1.2, 0.8, 0, 0, 0, 0, 0, 0, 0, \ldots, 0]$,$\boldsymbol{b}_{1:p} = [0, 1, 1, 1, 0, 0, 0, 0, 0, 0, \ldots, 0]$.

D. $\boldsymbol{a}_{1:p} = [1, 1, 1, 0, 0, 0, 0, 0, 0, 0, \ldots, 0]$, $\boldsymbol{b}_{1:p} = [0, 1.6, 1.2, 0.8, 0, 0, 0, 0, 0, 0, \ldots, 0]$.

E. $\boldsymbol{a}_{1:p} = [1.6, 1.2, 0.8, 0.4, 1.6, 1.2, 0.8, 0.4, 1.6, 1.2, 0, 0, \ldots, 0]$,
$\boldsymbol{b}_{1:p} = [0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 0, 0, \ldots, 0]$.

F. $\boldsymbol{a}_{1:p} = [1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 0, 0, \ldots, 0]$,
$\boldsymbol{b}_{1:p} = [0, 0, 0, 0, 0, 0, 0, 1.6, 1.2, 0.8, 0.4, 1.6, 1.2, 0.8, 0.4, 1.6, 1.2, 0, 0, \ldots, 0]$.

In Settings C and D, the first three elements of $\boldsymbol{a}_{1:p}$ are nonzero, and the 2nd to 4th

elements of $\boldsymbol{b}_{1:p}$ are nonzero. This implies that only the indirect pathways through M^2 and M^3 contribute to the mediation effect. In contrast, Settings E and F feature ten nonzero elements in $\boldsymbol{a}_{1:p}$ and ten nonzero elements in $\boldsymbol{b}_{1:p}$ (from the 7th to the 16th positions), indicating that only the indirect pathways through M^7, M^8, M^9, and M^{10} contribute to the mediation effect. The direct effects $\triangle_{T\to Y}(j)$s, indirect effects via the kth mediator $\triangle_{T\to M^k\to Y}(j)$s and total indirect effects $\triangle_{T\to \boldsymbol{M}\to Y}(j)$ for $(j = 1, 2, \cdots, N_g)$ are calculated on the entire dataset with $N_g = 1000$. Subsequently, $\triangle_{T\to Y}$, $\triangle_{T\to M^k\to Y}$, and $\triangle_{T\to \boldsymbol{M}\to Y}$, along with their 95% confidence intervals, are obtained using the method described in Section 3.4. To enhance estimation accuracy and stability, the final results are obtained by averaging the estimates over the last 100 training epochs.

Table 1: Treatment effects by HMA under Setting A-D

	Setting A	Setting B	Setting C	Setting D
$\triangle_{T\to Y}$	123	123	123	123
	[456, 789]	[456, 789]	[456, 789]	[456, 789]
$\triangle_{T\to \boldsymbol{M}\to Y}$	123	123	123	123
	[456, 789]	[456, 789]	[456, 789]	[456, 789]
$\triangle_{T\to M^1\to Y}$	123	123	123	123
	[456, 789]	[456, 789]	[456, 789]	[456, 789]
$\triangle_{T\to M^2\to Y}$	123	123	123	123
	[456, 789]	[456, 789]	[456, 789]	[456, 789]
$\triangle_{T\to M^3\to Y}$	123	123	123	123
	[456, 789]	[456, 789]	[456, 789]	[456, 789]
$\triangle_{T\to M^4\to Y}$	123	123	123	123
	[456, 789]	[456, 789]	[456, 789]	[456, 789]
$\triangle_{T\to M^5\to Y}$	–	–	123	123
			[456, 789]	[456, 789]
$\triangle_{T\to M^6\to Y}$	–	–	123	123
			[456, 789]	[456, 789]
$\triangle_{T\to M^7\to Y}$	–	–	123	123
			[456, 789]	[456, 789]

Table 2: Treatment effects by HMA under Setting E and F

	Setting E	Setting F		Setting E	Setting F
$\triangle_{T\to Y}$	123 [456, 789]	123 [456, 789]	$\triangle_{T\to \boldsymbol{M}\to Y}$	123 [456, 789]	123 [456, 789]
$\triangle_{T\to M^1\to Y}$	123 [456, 789]	123 [456, 789]	$\triangle_{T\to M^9\to Y}$	123 [456, 789]	123 [456, 789]
$\triangle_{T\to M^2\to Y}$	123 [456, 789]	123 [456, 789]	$\triangle_{T\to M^{10}\to Y}$	123 [456, 789]	123 [456, 789]
$\triangle_{T\to M^3\to Y}$	123 [456, 789]	123 [456, 789]	$\triangle_{T\to M^{11}\to Y}$	123 [456, 789]	123 [456, 789]
$\triangle_{T\to M^4\to Y}$	123 [456, 789]	123 [456, 789]	$\triangle_{T\to M^{12}\to Y}$	123 [456, 789]	123 [456, 789]
$\triangle_{T\to M^5\to Y}$	123 [456, 789]	123 [456, 789]	$\triangle_{T\to M^{13}\to Y}$	123 [456, 789]	123 [456, 789]
$\triangle_{T\to M^6\to Y}$	123 [456, 789]	123 [456, 789]	$\triangle_{T\to M^{14}\to Y}$	123 [456, 789]	123 [456, 789]
$\triangle_{T\to M^7\to Y}$	123 [456, 789]	123 [456, 789]	$\triangle_{T\to M^{15}\to Y}$	123 [456, 789]	123 [456, 789]
$\triangle_{T\to M^8\to Y}$	123 [456, 789]	123 [456, 789]	$\triangle_{T\to M^{16}\to Y}$	123 [456, 789]	123 [456, 789]

6 Conclusion

In this study, we propose a novel GAN-based model, GAHMN, for high-dimensional mediation analysis. Unlike existing HMA methods that rely heavily on restrictive linearity and distributional assumptions, GAHMN harnesses the expressive power of conditional GANs to capture complex nonlinear, heterogeneous, and correlated relationships among mediators and outcomes. By incorporating partially linear structures, multi-channel convolutions, and targeted regularization strategies, GAHMN effectively integrates domain knowledge and structural priors into the GAN architecture. This enables the model to control complexity and maintain computational feasibility, while preserving interpretability and scalability. Both theoretical analysis and extensive numerical experiments confirm the robustness and superior performance of the proposed method. The success of GAHMN in mediation analysis opens the door to broader applications of GAN-based approaches in other high-dimensional statistical learning problems. In particular, fields such as structural equation modeling, multi-task regression, and high-dimensional financial analysis—especially those involving treatment effect heterogeneity—can benefit from generative frameworks capable of flexibly modeling joint distributions and counterfactual structures. This work marks an important step toward the development of more promising and flexible data-driven tools for high-dimensional modeling.

Appendix

We now provide theoretical justification for the convergence of (46). Specifically, the learning objective can be viewed as distribution matching tasks, i.e., we aim to learn conditional generators G that produce synthetic samples matching the conditional distribution of given observed variables. Our theoretical results is inspired by the deep generative conditional sampling framework developed in (Zhou et al., 2023), where the authors showed that if the generator is trained to match the joint distribution with the data distribution, then under mild conditions, the learned conditional generator converges in distribution to the true conditional distribution. For (45) and (46), we have following assumptions.

- **(A1)**: The target generative model G^* exists, and its linear form is $G^*(v, x, z) = \boldsymbol{\beta}^{*\top} v + G_F^*(x, z)$, where $\|\beta^*\|_1 < \infty$, and G_F^* is continuous and its L_∞ norm is upper bounded by constant B.
- **(A2)**: $\frac{P_{\boldsymbol{V},\boldsymbol{X},Y}}{P_{\boldsymbol{V},\boldsymbol{X},Y}+P_{\boldsymbol{V},\boldsymbol{X},\hat{Y}}}$ is lower and upper bounded.
- **(A3)**: G_F is a ReLU-based FNN with the whole model size S, depth H, width W. As sample size n goes to infinity, $HW \to \infty$ and $BSH \log S \log n/n \to 0$.
- **(A4)**: The discriminator D is also implemented using a ReLU-based FNN, satisfying the same parameter constraints as assumed for G_F.

Theorem For the objective function (45) and generator (46), it aims to minimize the following expected risk:

$$\mathcal{L}_\alpha(G) = \sup_D \{\mathcal{L}(G, D)\} + \alpha\|\boldsymbol{\beta}\|_1, \tag{56}$$

where $\mathcal{L}(G, D) = \mathbb{E}_{(v,x,y)\sim p_{V,X,Y}} [\log D(v, x, y)] + \mathbb{E}_{(v,x,z)\sim p_{V,X,Z}} [\log (1 - D(v, x, G(v, x, z)))]$. The empirical version of (56) is given by

$$\widehat{\mathcal{L}}_\alpha(G) = \sup_{D\in\mathcal{D}} \{\hat{\mathcal{L}}(G, D)\} + \alpha \|\boldsymbol{\beta}\|_1 , \tag{57}$$

where $\hat{\mathcal{L}}(G, D) = \frac{1}{n}\sum_{i=1}^n \log D(v_i, x_i, y_i) + \frac{1}{n}\sum_{i=1}^n \log (1 - D(v_i, x_i, G(v_i, x_i, z_i)))$. Under assumptions **(A1)**-**(A4)**, when $\alpha = \alpha_n = O(\frac{1}{n})$, there exists a unique conditional generator $\hat{G}$ such that

$$\mathbb{E}_{(v_i,x_i,z_i,Y_i)_{i=1}^n} \left[\|P_{\boldsymbol{V},\boldsymbol{X},\hat{G}(\boldsymbol{V},\boldsymbol{X},\boldsymbol{Z})} - P_{\boldsymbol{V},\boldsymbol{X},Y}\|_1^2\right] \to 0, \quad as \;\; n \to \infty. \tag{58}$$

Proof Denote the final trained generator as $\hat{G}$ with $\hat{\beta}$, the theoretically optimal generator as G^* with $\beta^\star$, and empirical optimal generator as $\bar{G}$ with $\bar{\beta}$. Following the techniques in (Zhou et al., 2023), we have

$$\begin{aligned}
\mathcal{L}_\alpha(\hat{G}) - \mathcal{L}_\alpha(G^*) &= \sup_D \mathcal{L}(\hat{G}, D) - \sup_{D\in\mathcal{D}} \mathcal{L}(\hat{G}, D) \\
&\quad + \sup_{D\in\mathcal{D}} \mathcal{L}(\hat{G}, D) - \sup_{D\in\mathcal{D}} \hat{\mathcal{L}}(\hat{G}, D) \\
&\quad + \sup_{D\in\mathcal{D}} \hat{\mathcal{L}}(\hat{G}, D) - \sup_{D\in\mathcal{D}} \hat{\mathcal{L}}(\bar{G}, D) \\
&\quad + \sup_{D\in\mathcal{D}} \hat{\mathcal{L}}(\bar{G}, D) - \sup_{D\in\mathcal{D}} \mathcal{L}(\bar{G}, D) \\
&\quad + \sup_{D\in\mathcal{D}} \mathcal{L}(\bar{G}, D) - \sup_{D\in\mathcal{D}} \mathcal{L}(\bar{G}, D) \\
&\quad + \sup_{D\in\mathcal{D}} \mathcal{L}(\bar{G}, D) - \sup_{D\in\mathcal{D}} \mathcal{L}(G^*, D) + \alpha(|\hat{\boldsymbol{\beta}}| - |\boldsymbol{\beta}^\star|). \qquad (59)
\end{aligned}$$

The third and fifth terms on the right-hand side of (59) are non-positive. The second and fourth terms are smaller than $\sup_{D\in\mathcal{D},G\in\mathcal{G}} |\mathcal{L}(G, D) - \hat{\mathcal{L}}(G, D)|$. Thus, we have

$$\begin{aligned}
\mathcal{L}_\alpha(\hat{G}) - \mathcal{L}_\alpha(G^*) &\leq \sup_D \mathcal{L}(\hat{G}, D) - \sup_{D\in\mathcal{D}} \mathcal{L}(\hat{G}, D) + 2 \sup_{D\in\mathcal{D},G\in\mathcal{G}} |\mathcal{L}(G, D) - \hat{\mathcal{L}}(G, D)| \\
&\quad + \sup_{D\in\mathcal{D}} \mathcal{L}(\bar{G}, D) - \sup_{D\in\mathcal{D}} \mathcal{L}(G^*, D) + \alpha(|\hat{\boldsymbol{\beta}}| - |\boldsymbol{\beta}^\star|). \qquad (60)
\end{aligned}$$

Then, it follows that

$$\mathcal{L}_\alpha(\hat{G}) - \mathcal{L}_\alpha(G^*) \leq \Delta_1 + \Delta_2 + \Delta_3, \qquad (61)$$

where $\Delta_1 = \sup_D \mathcal{L}(\hat{G}, D) - \sup_{D\in\mathcal{D}} \mathcal{L}(\hat{G}, D)$, $\Delta_2 = 2\sup_{D\in\mathcal{D},G\in\mathcal{G}} |\mathcal{L}(G, D) - \hat{\mathcal{L}}(G, D)|$, and $\Delta_3 = \inf_{\bar{G}\in\mathcal{G}}[\sup_{D\in\mathcal{D}} \mathcal{L}(\bar{G}, D) - \sup_{D\in\mathcal{D}} \mathcal{L}(G^*, D)] + \alpha(|\hat{\boldsymbol{\beta}}| - |\boldsymbol{\beta}^\star|)$. The standard empirical process argument in (Zhou et al., 2023) (Lemma B.2) provides the convergence rate of estimation error Δ_2, which gives $\Delta_2 \lesssim \mathcal{O}\left(n^{-\frac{2}{2+\bar{d}+\bar{m}}} + n^{-\frac{2}{2+\bar{d}+\bar{q}}}\right) = \mathcal{O}\left(n^{-\frac{2}{3+\bar{d}}}\right)$, where $\bar{d}$ is the dimension of $\boldsymbol{V}$, and $\bar{m}$ is the dimension of noise $\boldsymbol{V}$, and $\bar{q}$ is the dimension of Y ($\bar{q} = 1$). Followed by the Lemma B.3 in (Zhou et al., 2023), we have $\mathbb{E}_{(v_i,x_i,z_i,Y_i)_{i=1}^n} \Delta_1 \to 0$ as n goes to infinity. We turn to control the approximation error Δ_3. For the fixed $\bar{G}$, using the optimality in D (Mirza, 2014), we have the following result:

$$\tilde{D}(v, x, y) = \arg\max_D \mathcal{L}(\bar{G}, D) = \frac{P_{\boldsymbol{V},\boldsymbol{X},Y}(v, x, y)}{P_{\boldsymbol{V},\boldsymbol{X},\bar{G}}(v, x, \hat{y}) + P_{\boldsymbol{V},\boldsymbol{X},Y}(v, x, y)}, \qquad (62)$$

for $\forall v, x, y$. Then, we can derive

$$\begin{aligned}
\Delta_3 &= \inf_{\bar{G}\in\mathcal{G}} [\sup_{D\in\mathcal{D}} \mathcal{L}(\bar{G}, D) - \sup_{D\in\mathcal{D}} \mathcal{L}(G^*, D)] + \alpha(|\hat{\boldsymbol{\beta}}| - |\boldsymbol{\beta}^\star|) \\
&= \inf_{\bar{G}\in\mathcal{G}} \left[\mathcal{L}(\bar{G}, \frac{P_{\boldsymbol{V},\boldsymbol{X},Y}}{P_{\boldsymbol{V},\boldsymbol{X},\bar{G}} + P_{\boldsymbol{V},\boldsymbol{X},Y}}) - \mathcal{L}(G^*, \frac{P_{\boldsymbol{V},\boldsymbol{X},Y}}{P_{\boldsymbol{V},\boldsymbol{X},G^\star} + P_{\boldsymbol{V},\boldsymbol{X},Y}})\right] + \alpha(|\hat{\boldsymbol{\beta}}| - |\boldsymbol{\beta}^\star|). \qquad (63)
\end{aligned}$$

Followed by the Lemma B.1 in (Zhou et al., 2023), and let $\alpha = \alpha_n = O(\frac{1}{n})$, we have $\Delta_3 \to 0$, as $n \to \infty$. Thus, $\mathcal{L}_\alpha(\hat{G}) - \mathcal{L}_\alpha(G^*) \to 0$, as $n \to \infty$. It is noticed that $\sup_D L(G^*, D) = 0$. By Pinsker's inequality (Tsybakov and Tsybakov, 2009), we have

$$\begin{aligned}\left[\|P_{\boldsymbol{V},\boldsymbol{X},\hat{G}(V,\boldsymbol{X},\boldsymbol{Z})} - P_{\boldsymbol{V},\boldsymbol{X},Y}\|_1^2\right] &\leq 2(\sup_D L(\hat{G}, D) - \sup_D L(G^*, D)) \\ &\leq 2(\mathcal{L}_\alpha(\hat{G}) - \mathcal{L}_\alpha(G^*)) + 2\alpha_n||\hat{\boldsymbol{\beta}}| - |\boldsymbol{\beta}^\star||. \qquad (64)\end{aligned}$$

Let $n \to \infty$, (58) holds. The proof is completed.

Acknowledgments

This work was supported by Innovation and Talent Base for Digital Technology and Finance (B21038), Fundamental Research Funds for the Central Universities under Project (Zhongnan University of Economics and Law) 2722024EJ011 and 2722022BY020, and General Research Fund grants 14303622 from Research Grant Council of the Hong Kong.

References

Aggarwal, K., M. Kirchmeyer, P. Yadav, S. S. Keerthi, and P. Gallinari (2019). Conditional generative adversarial networks for regression. *ArXiv190512868 Cs Stat.(10) 133*, 142–146.

Bansal, A., S. Ma, D. Ramanan, and Y. Sheikh (2018). Recycle-gan: Unsupervised video retargeting. In *Proceedings of the European conference on computer vision (ECCV)*, pp. 119–135.

Bingham, N. H. and R. Kiesel (2002). Semi-parametric modelling in finance: theoretical-foundations. *Quantitative Finance 2*(4), 241.

Cai, X., Y. Zhu, Y. Huang, and D. Ghosh (2022). High-dimensional causal mediation analysis based on partial linear structural equation models. *Computational Statistics & Data Analysis 174*, 107501.

Carpena, F. and B. Zia (2020). The causal mechanism of financial education: Evidence from mediation analysis. *Journal of Economic Behavior & Organization 177*, 143–184.

Díaz, I. and M. J. van der Laan (2013). Sensitivity analysis for causal inference under unmeasured confounding and measurement error problems. *The international journal of biostatistics 9*(2), 149–160.

Goodfellow, I., J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio (2014). Generative adversarial nets. *Advances in neural information processing systems 27*.

Gui, J., Z. Sun, Y. Wen, D. Tao, and J. Ye (2021). A review on generative adversarial networks: Algorithms, theory, and applications. *IEEE transactions on knowledge and data engineering 35*(4), 3313–3332.

Gunzler, D., T. Chen, P. Wu, and H. Zhang (2013). Introduction to mediation analysis with structural equation modeling. *Shanghai archives of psychiatry 25*(6), 390.

Guo, X., R. Li, J. Liu, and M. Zeng (2022). High-dimensional mediation analysis for selecting dna methylation loci mediating childhood trauma and cortisol stress reactivity. *Journal of the American Statistical Association 117*(539), 1110–1121.

Hayes, A. F. (2017). *Introduction to mediation, moderation, and conditional process analysis: A regression-based approach*. Guilford publications.

Hollander, M., D. A. Wolfe, and E. Chicken (2013). *Nonparametric statistical methods*. John Wiley & Sons.

Huang, T.-J., Z. Liu, and I. W. McKeague (2025). Post-selection inference for high-dimensional mediation analysis with survival outcomes. *Scandinavian Journal of Statistics*.

Huang, Y.-T. and H.-I. Yang (2017). Causal mediation analysis of survival outcome with multiple mediators. *Epidemiology 28*(3), 370–378.

Imai, K., L. Keele, and D. Tingley (2010). A general approach to causal mediation analysis. *Psychological methods 15*(4), 309.

Kim, Y., H. Choi, and H.-S. Oh (2008). Smoothly clipped absolute deviation on high dimensions. *Journal of the American Statistical Association 103*(484), 1665–1673.

Lee, S.-Y. (2007). *Structural equation modeling: A Bayesian approach*. John Wiley & Sons.

Lin, J., Y. Xia, T. Qin, Z. Chen, and T.-Y. Liu (2018). Conditional image-to-image translation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 5524–5532.

Lindquist, M. A. (2012). Functional causal mediation analysis with an application to brain connectivity. *Journal of the American Statistical Association 107*(500), 1297–1309.

Liu, H., I. H. Jin, Z. Zhang, and Y. Yuan (2021). Social network mediation analysis: A latent space approach. *Psychometrika 86*(1), 272–298.

Liu, Z., J. Shen, R. Barfield, J. Schwartz, A. A. Baccarelli, and X. Lin (2022). Large-scale hypothesis testing for causal mediation effects with applications in genome-wide epigenetic studies. *Journal of the American Statistical Association 117*(537), 67–81.

MacKinnon, D. (2012). *Introduction to statistical mediation analysis*. Routledge.

Mirza, M. (2014). Conditional generative adversarial nets. *arXiv preprint arXiv:1411.1784*.

Nath, T., B. Caffo, T. Wager, and M. A. Lindquist (2023). A machine learning based approach towards high-dimensional mediation analysis. *NeuroImage 268*, 119843.

Olsson, U. H., T. Foss, S. V. Troye, and R. D. Howell (2000). The performance of ml, gls, and wls estimation in structural equation modeling under conditions of misspecification and nonnormality. *Structural equation modeling 7*(4), 557–595.

Olsson, U. H., S. V. Troye, and R. D. Howell (1999). Theoretic fit and empirical fit: The performance of maximum likelihood versus generalized least squares estimation in structural equation models. *Multivariate behavioral research 34*(1), 31–58.

Parikh, N., S. Boyd, et al. (2014). Proximal algorithms. *Foundations and trends® in Optimization 1*(3), 127–239.

Pearl, J. (2014). Interpretation and identification of causal mediation. *Psychological methods 19*(4), 459.

Preacher, K. J. and A. F. Hayes (2008). Asymptotic and resampling strategies for assessing and comparing indirect effects in multiple mediator models. *Behavior research methods 40*(3), 879–891.

Sun, R. and X. Song (2024). Heterogeneous mediation analysis for cox proportional hazards model with multiple mediators. *Statistics in Medicine 43*(29), 5497–5512.

Tofighi, D. and D. P. MacKinnon (2011). Rmediation: An r package for mediation analysis confidence intervals. *Behavior research methods 43*, 692–700.

Tsybakov, A. B. and A. B. Tsybakov (2009). Nonparametric estimators. *Introduction to Nonparametric Estimation*, 1–76.

Ullman, J. B. and P. M. Bentler (2012). Structural equation modeling. *Handbook of Psychology, Second Edition 2*.

Valente, M. J., J. J. Rijnhart, H. L. Smyth, F. B. Muniz, and D. P. MacKinnon (2020). Causal mediation programs in r, m plus, sas, spss, and stata. *Structural equation modeling: a multidisciplinary journal 27*(6), 975–984.

Wang, K. (2019). Maximum likelihood analysis of linear mediation models with treatment–mediator interaction. *psychometrika 84*(3), 719–748.

Wang, X., J. Liu, S. S. Hu, Z. Liu, H. Lu, L. Liu, and A. D. N. Initiative (2023). Hilama: High-dimensional multi-omic mediation analysis with latent confounding. *bioRxiv*, 2023–09.

Wiese, M., R. Knobloch, R. Korn, and P. Kretschmer (2020). Quant gans: deep generation of financial time series. *Quantitative Finance 20*(9), 1419–1440.

Xie, X., H. Zou, and G. Qi (2018). Knowledge absorptive capacity and innovation performance in high-tech companies: A multi-mediating analysis. *Journal of business research 88*, 289–297.

Yang, H., Z. Liu, R. Wang, E.-Y. Lai, J. Schwartz, A. A. Baccarelli, Y.-T. Huang, and X. Lin (2024). Causal mediation analysis for integrating exposure, genomic, and phenotype data. *Annual Review of Statistics and Its Application 12*.

Yuan, Y. and D. P. MacKinnon (2009). Bayesian mediation analysis. *Psychological methods 14*(4), 301.

Zeng, P., Z. Shao, and X. Zhou (2021). Statistical methods for mediation analysis in the era of high-throughput genomics: current successes and future challenges. *Computational and structural biotechnology journal 19*, 3209–3224.

Zhai, M., L. Chen, F. Tung, J. He, M. Nawhal, and G. Mori (2019). Lifelong gan: Continual learning for conditional image generation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 2759–2768.

Zhang, J., Y. Lin, X. Song, and H. Ning (2023). Generative adversarial mediation network: A novel generative learning approach to causal mediation analysis. *Knowledge-Based Systems 282*, 111117.

Zhou, X., Y. Jiao, J. Liu, and J. Huang (2023). A deep generative approach to conditional sampling. *Journal of the American Statistical Association 118*(543), 1837–1848.

Zhou, X. and X. Song (2021). Mediation analysis for mixture cox proportional hazards cure models. *Statistical Methods in Medical Research 30*(6), 1554–1572.

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

Zou, H. and T. Hastie (2005). Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society Series B: Statistical Methodology 67*(2), 301–320.