Graph Machine Learning based Doubly Robust Estimator for Network Causal Effects

Baharan Khatami[1] **Harsh Parikh**[2] **Haowei Chen**[1] **Sudeepa Roy**[3] **Babak Salimi**[1]
[1]UC San Diego [2]Johns Hopkins University [3]Duke University

Abstract

Estimating causal effects in social network data presents unique challenges due to the presence of spillover effects and network-induced confounding. While much of the existing literature addresses causal inference in social networks, many methods rely on strong assumptions about the form of network-induced confounding. These assumptions often fail to hold in high-dimensional networks, limiting the applicability of such approaches. To address this, we propose a novel methodology that integrates graph machine learning techniques with the double machine learning framework, facilitating accurate and efficient estimation of both direct and peer effects in a single observational social network. Our estimator achieves semiparametric efficiency under mild regularity conditions, enabling consistent uncertainty quantification. Through extensive simulations, we demonstrate the accuracy, robustness, and scalability of our method. Finally, we apply the proposed approach to examine the impact of Self-Help Group participation on financial risk tolerance, highlighting its practical relevance.

1 Introduction

Our paper addresses the challenge of causal inference from social networks, a problem crucial for decision-making across various vital domains such as social media, healthcare, and economic networks (Jackson et al., 2008; Ogburn et al., 2022; Atanasov and Black, 2016; Gassen, 2014). For instance, causal inference helps understand the impact of recommendation algorithms on user engagement and preferences in social media, evaluate the effect of public health interventions like self-quarantine or school closures on the spread of infectious diseases within specific communities, and assess the influence of participation in self-help groups (SHG) on individuals' financial behaviors. Our paper uses the latter as a case study to investigate whether participation in SHG affects financial risk-taking behavior, observable through outstanding loans as a proxy measure.

Estimating causal effects from observational network data is challenging due to several factors: First, dependencies among individuals deviate from the traditional i.i.d. assumption, causing standard methods to fail. Second, network dependencies introduce interference between units, where their neighbors' treatments influence an individual's outcome, further complicating causal analysis (Hudgens and Halloran, 2008b; Ogburn and VanderWeele, 2014; Aronow and Samii, 2017; Halloran and Hudgens, 2012; VanderWeele and Tchetgen Tchetgen, 2011). Third, a unit's neighbors' covariates act as confounders that require adjusting for a complex set of potentially high-dimensional covariates with variable sizes for each unit, influenced by network structure and topology (VanderWeele and An, 2013; Ogburn and VanderWeele, 2014).

Existing techniques often use predefined aggregates to summarize network information and develop estimators with theoretical guarantees, assuming these aggregates are sufficient (Forastiere et al., 2021; Salimi et al., 2020; Ogburn et al., 2022; Forastiere et al., 2018). This approach can lead to a loss of essential network details and inaccurate conclusions. Recent methods using Graph Neural Networks (GNNs) aim to handle high-dimensional network covariates (Ma and Tresp, 2020; Jiang and Sun, 2022; Guo et al., 2020a), but they lack theoretical guarantees and valid confidence intervals in causal effect estimation. While graph machine learning techniques are powerful, their direct use in causal inference can lead to regularization bias, overfitting, and slow convergence rates (Chernozhukov et al., 2018). Our method provides a principled integration of graph machine learning for causal inference, offering theoretical guarantees that address these limitations.

In this paper, we extend the Double Machine Learning (DML) framework (Chernozhukov et al., 2018) to develop

Proceedings of the 28th International Conference on Artificial Intelligence and Statistics (AISTATS) 2025, Mai Khao, Thailand. PMLR: Volume 258.

a consistent and asymptotically normal estimator for causal effects in network data, using modern graph machine learning (ML) methods. DML refers to a methodology that combines machine learning models with cross-fitting and orthogonalization to debias causal effect estimates by mitigating the influence of nuisance parameter estimation errors. Our approach efficiently handles high-dimensional network covariates with provable guarantees, tackling the challenge of complex network confounding, where the confounding map, aggregating neighborhood information, is unknown. We focus on cases where the confounding variables are either too high-dimensional for classical methods or where their influence on treatment and outcome cannot be modeled well by parametric functions. Machine learning models are well-suited for these complex mappings. However, blindly using these models for inferring causal parameters can introduce regularization bias or overfitting, leading to biased estimators.

To address these challenges, we systematically apply Neyman-orthogonal moments (Neyman, 1959; Chernozhukov et al., 2018), a statistical technique designed to minimize the impact of errors in estimating nuisance parameters, such as the propensity score (probability of treatment given network covariates) and the outcome model (predicting the outcome given network covariates). This is achieved by constructing moment conditions that satisfy an orthogonality property, meaning that the derivative of the moment condition with respect to the nuisance parameter estimates is zero at the true values. Intuitively, this ensures that small errors in the estimation of nuisance parameters, such as the propensity score or the outcome model, do not substantially bias the causal effect estimate. Robustness against estimation errors is improved through the use of cross-fitting.

Our estimator integrates graph machine learning algorithms, such as Graph Neural Networks (GNNs), to estimate both nuisance parameters. By using "double" or "orthogonalized" ML techniques (Chernozhukov et al., 2018) and sample splitting, we construct high-quality point and interval estimates for causal parameters. This framework addresses common issues like slow convergence rates, model misspecification, and regularization bias. Furthermore, our estimator is doubly robust, meaning it remains consistent if either the propensity score or the outcome model is correctly specified. We also establish its consistency and asymptotic normality under specific assumptions, allowing for the establishment of valid confidence intervals, which is crucial for the practical application of causal inference methods in social networks.

Contributions. The key contributions of this research include:

1. We develop a method for causal inference from social network data that integrates graph machine learning techniques, leveraging "orthogonalized" ML and sample splitting to construct high-quality point and interval estimates, addressing issues like slow convergence rates, model misspecification, and regularization bias.
2. We demonstrate the theoretical properties of our estimator, including consistency and asymptotic normality under certain assumptions, which enable the construction of valid confidence intervals.
3. We evaluate our framework on three semi-synthetic datasets and compare its performance against six leading methods, demonstrating superior performance.
4. We conduct a case study on real-world data to examine the impact of Self-Help Group participation on financial risk tolerance, showcasing the practical applicability of our approach.

Related Work. Research in causal inference under network interference divides into experimental design and inference from observational data. Experimental design strategies, such as cluster-based randomization (Bland, 8 13; Sobel, 2006; Hudgens and Halloran, 2008a) and graph cluster randomization (Ugander et al., 2013; Eckles et al., 2016; Ugander and Yin, 2023; Pouget-Abadie et al., 2019; Karrer et al., 2021), aim to minimize interference using network information. On the observational front, Inverse Probability Weighting (IPW) estimators for network interference are prominent (Hudgens and Halloran, 2008b; Forastiere et al., 2021; Tchetgen and VanderWeele, 2012; Liu et al., 2016). Our research introduces a doubly robust estimator, diverging from these traditional methods by not relying solely on propensity scores, making them less robust to model misspecification.

Recent progress in nonparametric modeling for sparse networks includes frameworks that evaluate treatment effects using local network configurations (Auerbach and Tabord-Meehan, 2021). These approaches define similarly configured agents using distance-based isomorphism metrics for every pair of nodes, akin to what GNNs systematically do, and then pool outcome data across these agents to evaluate the impact of a policy or treatment assignment. Studies addressing unobserved confounding use networks as proxies for latent confounders and they learn representations of hidden confounders through mapping both network structure and features into a shared space, then inferring potential outcomes based on these representations (Veitch et al., 2019; Guo, 2019; Cristali and Veitch, 2022). Representation balancing using domain-adapted learning and Graph Neural Networks (GNNs) is also explored to mimic randomized trials (Johansson et al., 2016; Shalit et al., 2017; Yao et al., 2018; Ma and Tresp, 2020; Jiang and Sun, 2022; Guo et al., 2020a). Cai et al. (2023) combines IPW-based reweighting and representation balancing to address residual confounding from imperfect propensity score

estimation. It introduces an IPM-based term to model dependencies between reweighted features and treatment pairs, deriving generalization bounds and proposing an architecture that minimizes estimation error. However, these methods often lack theoretical guarantees, robustness to model misspecification, and valid approaches for establishing confidence intervals.

Recently, there has been significant work on developing doubly robust estimators for causal inference from network data, enhancing robustness and efficiency, and addressing challenges like latent treatment homophily in causal effect identification (McNealis et al., 2023). Other key contributions include the TMLE estimator for treatment and spillover effects (Laan, 2014; Ogburn et al., 2022), which use predefined aggregates. Liu et al. (2023); Chen et al. (2024) require the treatment variable to be categorical, while our approach supports both categorical and continuous treatments. While both methods assume a well-defined exposure map, they restrict exposure variation to a finite range for nonparametric regression, whereas our semi-parametric approach accommodates broader exposure values. Additionally, Liu et al. (2023) relies on kernel regression with nonparametric moment conditions and predefined covariate aggregation, while ours uses OLS with a partially linear moment condition and learnable graph aggregation tools like GNNs. The most relevant work to us is by Leung and Loupos (2022) and Emmenegger et al. (2022), which provide a doubly robust non-parametric estimator integrated with machine learning models (GNNs in case of Leung and Loupos (2022)). Non-parametric methods make minimal assumptions about the underlying data distribution, allowing for greater flexibility but often at the cost of higher variance and computational complexity. However, their method requires discrete treatments and neighborhood exposures, which can be unrealistic in network scenarios, and does not decompose the treatment effect into direct and indirect effects. In contrast, our semi-parametric approach combines the flexibility of non-parametric models with the efficiency of parametric models, supporting continuous exposures and treatments—crucial for real-world applications like our case study on self-help groups, microinsurance, and risk appetite. Furthermore, our theoretical contributions establish guarantees for a semi-parametric estimator using different proof techniques than these methods.

2 Causal Inference and Networks

In this section, we introduce the required notations and the setup of our problem, including causal estimands and necessary assumptions for identification. As a convention in our paper, we represent random variables with capital letters (e.g., A), scalars with lowercase letters (e.g., a), matrices with script letters (e.g., $\mathcal{A}$), vectors with boldface symbols (e.g., $\mathbf{A}$), and sets with blackboard bold symbols (e.g., $\mathbb{A}$). Further, we also denote the shape of the matrix or vector as a subscript when and where necessary e.g. $\mathcal{A}_{m\times p}$ and $\mathbf{A}_{m\times 1}$.

2.1 Formal Setup and Assumptions

Consider a social network $\mathcal{G} = (\mathbb{V}^n, \mathcal{A}^n, \mathcal{Z}^n)$, where $\mathbb{V}^n = \{1 \dots n\}$ denotes the set of n units, $\mathcal{A}^n \in \{0,1\}^{n\times n}$ is the adjacency matrix representing the connectivity structure of the network across the n units. If $\mathcal{A}^n_{i,j} = 1$ for $i, j \in \mathbb{V}^n$, then units i and j are connected. The feature matrix $\mathcal{Z}^n$ can be decomposed as $\mathcal{Z}^n = (\mathcal{X}^n, \mathbf{T}^n, \mathbf{Y}^n)$, where $\mathcal{X}^n \in \mathbb{R}^{n\times d}$ is the matrix of pretreatment covariates, $\mathbf{T}^n = \{T_1, T_2, \dots, T_n\}$ is the vector of treatments for all units in $\mathbb{V}^n$, and $\mathbf{Y}^n = \{Y_1, Y_2, \dots, Y_n\}$ is the vector of outcomes for all units in the network. Our framework accommodates both binary and continuous treatments. The potential outcome of unit i under treatment vector $\mathbf{t}^n$ is denoted by $Y_i(\mathbf{t}^n)$. We drop the superscript n indicating the sample size for parsimony in the rest of the paper.

Let $\mathbb{N}_i = \{j : \mathcal{A}_{i,j} = 1\}$ be the set of nodes sharing ties with node i (i.e., the neighborhood of node i). Having '$-i$' in the subscript denotes everything but i, hence $\mathbb{N}_{-i} = \mathbb{V} \setminus (\mathbb{N}_i \cup \{i\})$ is the set of non-neighbors of unit i. The vectors of treatments and outcomes for all nodes except node i are denoted as $\mathbf{T}_{-\mathbf{i}}$ and $\mathbf{Y}_{-\mathbf{i}}$ respectively, and the matrix of covariates for all nodes except for node i as $\mathcal{X}_{-i}$. Similarly, for the neighbors of i , we denote the vectors of their treatments and outcomes as $\mathbf{T}_{\mathbb{N}_\mathbf{i}}$ and $\mathbf{Y}_{\mathbb{N}_\mathbf{i}}$, respectively, and the matrix of their covariates as $\mathcal{X}_{\mathbb{N}_i}$. We assume that our network data is generated via the mechanism defined by the following structural equations:

$$\begin{aligned} \mathbf{T} &= m_0(\mathcal{X}, \mathcal{A}) + \epsilon^{\mathbf{T}} && \mathbb{E}[\epsilon^{\mathbf{T}} \mid \mathcal{X}] = 0 \\ \mathbf{Y} &= \theta_0 \mathbf{T} + \alpha_0 \phi_{YT}(\mathbf{T}, \mathcal{A}) && \mathbb{E}[\epsilon^{\mathbf{Y}} \mid \mathcal{X}, \mathbf{T}] = 0 \\ &\quad + g_0(\mathcal{X}, \mathcal{A}) + \epsilon^{\mathbf{Y}} \end{aligned} \tag{1}$$

where $\{\epsilon\} = \{\epsilon^T_i\}_i \cup \{\epsilon^Y_i\}_i$ is a set of unobserved exogenous variables affecting random variables $\mathbf{X_i}$, T_i and Y_i, and (ϕ_{YT}, m_0, g_0) are a set of functional mappings that describe the causal dependence of the observed variables. $m_0(\mathcal{X}, \mathcal{A})$ and $g_0(\mathcal{X}, \mathcal{A})$ summarizes the covariates of the unit and its peers, i.e. $W_i = g_0(\mathcal{X}, \mathcal{A})$. Akin to the effective treatment function in Manski (2013), ϕ_{YT} is an exposure map that, for any unit i, summarizes the vector of treatments of all units $\mathbf{T}$ to an effective treatment exposure (Aronow and Samii, 2013), i.e., $Z_i = \phi_{YT}(\mathbf{T}, \mathcal{A})$. In other words, an exposure map is supposed to capture the full nature of interference of a unit from all other units. Given Z_i, the outcome Y_i can be determined, rendering it independent of the treatments of the remaining network: $Y_i(\mathbf{T}) = Y_i(T_i, Z_i)$. We do not consider feedback self-loops in the causal DAG, as addressing them is beyond the scope of this project.

Estimand: Our objective is to estimate two key causal esti-

mands: the average direct effect (ADE), denoted τ_{ADE}, and the average peer effect (APE), denoted τ_{APE}. ADE aims to capture the direct impact of treatment on the outcomes within individual units, whereas APE assesses the effect of treatments on a unit through its connections within a network. To illustrate the practical implications of these concepts, consider a friendship network where the treatment is the recommendation of a product in an advertisement to users, and the outcome is the purchasing of the product. This scenario prompts two pertinent questions: How does showing an advertisement to a user influence their likelihood of purchase? And, how does showing an advertisement to a user affect their friends' likelihood of purchase, considering potential discussions about the product? These questions correspond to the ADE and APE, respectively, which are well-established causal estimands in the literature (Hu et al., 2022; Jiang and Sun, 2022; Halloran and Struchiner, 1995; Blattman et al., 2021; Hudgens and Halloran, 2008a; Sobel, 2006). The estimands are formally defined as follows:

$$\tau_{\text{ADE}} = \mathbb{E}_{(\mathcal{X},\mathbf{T},\mathbf{Y})|\mathcal{G}}\left[\frac{1}{n}\sum_{i\in\mathbb{V}}\tau_{i,\text{DE}}\right],$$
$$\text{where if } t\in\mathbb{R}:\tau_{i,\text{DE}} = \frac{\partial Y_i(t,\mathbf{T}_{-i})}{\partial t},$$
$$\text{if } t\in\{0,1\}:\tau_{i,\text{DE}} = Y_i(1,\mathbf{T}_{-i}) - Y_i(0,\mathbf{T}_{-i}). \tag{2}$$

$$\tau_{\text{APE}} = \mathbb{E}_{(\mathcal{X},\mathbf{T},\mathbf{Y})|\mathcal{G}}\left[\frac{1}{n}\sum_{i\in\mathbb{V}}\tau_{i,\text{PE}}\right],$$
$$\text{where if } z\in\mathbb{R}:\tau_{i,\text{PE}} = \frac{\partial Y_i(T_i,z)}{\partial z},$$
$$\text{if } z\in\{0,1\}:\tau_{i,\text{PE}} = Y_i(T_i,\mathbf{1}_{-i}) - Y_i(T_i,\mathbf{0}_{-i}). \tag{3}$$

Here, $\tau_{i,\text{DE}}$ and $\tau_{i,\text{PE}}$ respectively denote the direct and peer effects on individual unit i. In the context of the structural equations presented earlier, these correspond to the parameters $\theta_0(\mathbf{X_i})$ and $\alpha_0(\mathbf{X_i})$. These effects are functions of the pre-treatment variables $\mathbf{X_i}$. Figure 1 illustrates a three-node causal graph, demonstrating network dynamics, causal interactions, and the alignment of $\tau_{i,\text{DE}}$ and $\tau_{i,\text{PE}}$ within the network structure.

Assumptions: We introduce the assumptions required for the identification of ADE and APE, which are standard in the causal inference literature from social networks (Bhattacharya et al., 2019; Guo et al., 2020b; Jiang and Sun, 2022; Ogburn et al., 2022). The network structure, defined by the adjacency matrix $\mathcal{A}$, is considered fixed and not treated as a random variable. It serves as an information pathway, where connected units can influence each other's treatments and outcomes.

A.1 **Exogeneity:** Unobserved exogenous variables are assumed to be independent. Formally, for any $i, j \in \mathbb{V}$, we assume:

$$\epsilon_{\mathbf{i}}^{\mathcal{X}} \perp \epsilon_{\mathbf{j}}^{\mathcal{X}}, \epsilon_i^T \perp \epsilon_j^T \mid \mathbf{X_i}, \mathbf{X_j}, \epsilon_i^Y \perp \epsilon_j^Y \mid \mathbf{X_i}, \mathbf{X_j}, T_i, T_j \tag{4}$$

A.2 **Partial Interference:** Each unit's potential outcome is influenced only by its own and its k-hop away neighbors' treatments. In this paper, we consider $k = 1$:

$$Y_i(T_i = t, \mathbf{T}_{\mathbb{N}_\mathbf{i}}, \mathbf{T}_{\mathbb{N}_{-\mathbf{i}}}) = Y_i(T_i = t, \mathbf{T}_{\mathbb{N}_\mathbf{i}}, \mathbf{T}'_{\mathbb{N}_{-\mathbf{i}}}) \tag{5}$$

A.3 **Known Exposure Map:** The exposure map ϕ_{YT} is well-defined and known a priori such that $Z_i = \phi_{YT}(\mathbf{T})$

A.4 **Positivity:** For all values of W_i present in the population of interest, i.e. $f(W_i) > 0$, all possible values of treatments and exposures have non-zero probabilities:

$$\forall(i,t,z),\ 0 < f(T_i = t, Z_i = z \mid W_i) \tag{6}$$

where f is the probability density function.

A.5 **Consistency:** The observed outcomes match potential outcomes under the observed treatment assignments:

$$Y_i(T_i = t, Z_i = z) = Y_i \quad \text{if} \quad T_i = t, Z_i = z. \tag{7}$$

A.6 **Strong Ignorability:** Conditional on the features $\mathcal{X}$, the potential outcome is independent of treatment and peer exposure:

$$Y_i(T_i = t_i, Z_i = z) \perp T_i, Z_i \mid \mathcal{X} \tag{8}$$

Proposition 1. *Under the assumptions of A.1-6, the average direct effect (ADE) and the average peer effect (APE) are identifiable.*

The proof can be found in the appendix 8.

Remark 2.1. Assumption A.3 implies knowledge of exposure map ϕ_{YT} to ensure consistent definition of the estimand. This assumption is common across the literature (see e.g, (Ogburn and VanderWeele, 2014; Jiang and Sun, 2022; Toulis and Kao, 2013; Zigler and Papadogeorgou, 2021; Papadogeorgou and Samanta, 2023; Forastiere et al., 2021)). On the other hand, g_0 and m_0, the functions moderating network-induced confounding are ***un****known* in our setup. This distinction is crucial.

Remark 2.2. Assumption A.6 implies that there are no unobserved confounders, a common practice in the literature (Ogburn et al., 2022; Jiang and Sun, 2022; Ma and Tresp, 2020; Liu et al., 2023). Sensitivity analysis and partial identification approaches address challenges due to hidden confounders when estimands are not point identified (Rosenbaum, 1987; Imbens, 2003; Frauen et al., 2023; VanderWeele, 2011). While these unobserved confounding remains an important challenge, these concerns are beyond the scope of our paper.

3 Method

In this section, we address the limitations of traditional Double Machine Learning (DML), which was originally

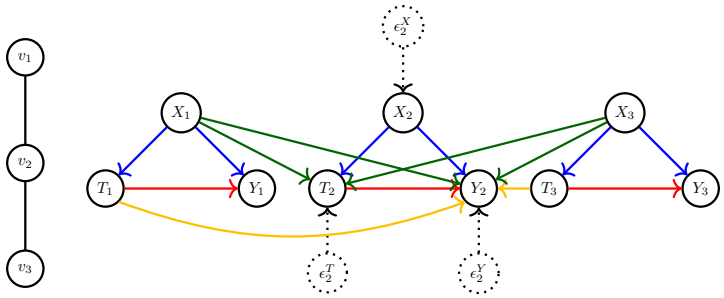

Figure 1: Partial causal graph of a network with three nodes. The left side shows the network topology, and the right side depicts the causal graph for each node with $\mathcal{X}$, $\mathbf{T}$, and $\mathbf{Y}$ as confounder, treatment, and outcome, respectively. Solid circles represent endogenous variables; dotted circles, exogenous. Blue edges indicate within-unit confounding, green edges show neighbor confounding, red edges represent direct effects, and yellow edges denote treatment interference.

developed for i.i.d. data and is not directly applicable to non-i.i.d. settings due to unit interdependence, such as interference. Unlike the standard DML framework, where confounding maps rely solely on a unit's covariates, our approach considers the covariates of neighboring units as well. To tackle these challenges, we introduce **Graph Double Machine Learning (GDML)**, which extends the DML framework to effectively handle non-i.i.d. data by incorporating graph machine learning techniques to efficiently estimate ADE and APE by adjusting for complex network confounders.

We illustrate our approach for exposure map $\phi_{YT}(\mathbf{T}) = \sum_{j\in\mathbb{N}_i} T_j$ for unit i. Recall that,

$$\mathbf{Y} = \theta_0\mathbf{T} + \alpha_0\mathcal{A}\mathbf{T} + g_0(\mathcal{X},\mathcal{A}) + \epsilon^{\mathbf{Y}}. \tag{9}$$

Taking expectations with respect to $\mathcal{X}$ on both sides, noting that $\mathcal{A}$ is constant, and subtracting it from Equation 9 yields:

$$\mathbf{Y} - \ell_0(\mathcal{X},\mathcal{A}) = (\theta_0 + \alpha_0\mathcal{A})\cdot(\mathbf{T} - m_0(\mathcal{X},\mathcal{A})) + \epsilon^{\mathbf{Y}} \tag{10}$$

where $m_0(\mathcal{X},\mathcal{A}) := \mathbb{E}[\mathbf{T} \mid \mathcal{X},\mathcal{A}]$ and $\ell_0(\mathcal{X},\mathcal{A}) := \mathbb{E}[\mathbf{Y} \mid \mathcal{X},\mathcal{A}]$.

Let $\zeta = (\theta,\alpha)$ and $\eta = (m,\ell)$ to be the unknown target and nuisance parameters with $\zeta_0 := (\theta_0,\alpha_0)$ and $\eta_0 := (m_0,\ell_0)$ as the true values of these parameters of our interest that satisfies equation 10. Now, let $W := (\mathcal{X},\mathbf{T},\mathbf{Y})$ be a random element taking values in a measurable space $(\mathcal{W},\mathcal{A}_{\mathcal{W}})$ with law determined by a probability measure $P \in \mathcal{P}_N$ with $(W_i)_{i=1}^n$ random samples available for estimation and inference. Then, consider a squared loss derived $\mathcal{L}(W,\mathcal{A};\zeta,\eta) := \frac{\mathbf{B}_{1\times n}^{\intercal}\mathbf{B}_{n\times 1}}{2}$ where $\mathbf{B}_{n\times 1} := [\mathbf{Y} - \ell(\mathcal{X},\mathcal{A}) - (\theta+\alpha\mathcal{A})(\mathbf{T} - \mathbf{m}(\mathcal{X},\mathcal{A}))]$ such that the partial derivatives of the loss function with respect to target parameters and nuisance parameters, evaluated at ζ_0 and η_0 yields zero:

$$\begin{aligned}\mathbb{E}_P\left[\partial_\zeta\mathcal{L}(W,\mathcal{A};\zeta,\eta)|_{\zeta_0,\eta_0}\right] = 0,\\ \mathbb{E}_P\left[\partial_\eta\mathcal{L}(W,\mathcal{A};\zeta,\eta)|_{\zeta_0,\eta_0}\right] = 0\end{aligned} \tag{11}$$

Thus, the target parameters can be identified by minimizing the following squared loss:

$$\zeta_0,\eta_0 \in \arg\min_{\zeta,\eta}\mathbb{E}_P\left[\mathcal{L}(W,\mathcal{A};\zeta,\eta)\right], \tag{12}$$

Now, we construct an efficient score function, ψ that enables doubly robust estimation of target parameters, similar to Chernozhukov et al. (2018) and Morucci et al. (2023):
$\psi(W,\mathcal{A};\zeta,\eta) = \partial_\zeta\mathcal{L}(W,\mathcal{A};\zeta,\eta) - \mu\partial_\eta\mathcal{L}(W,\mathcal{A};\zeta,\eta)$, where μ is an orthogonalization parameter matrix such that its optimal value solves the equation: $J_{\zeta\eta} - \mu J_{\eta\eta} = 0$ where,

$$\begin{pmatrix} J_{\zeta\zeta} & J_{\zeta\eta} \\ J_{\eta\zeta} & J_{\eta\eta}\end{pmatrix} = \partial_{(\zeta',\eta')}\mathbb{E}_P\left[\partial_{(\zeta',\eta')'}\mathcal{L}(W;\zeta,\eta)\right]\big|_{\zeta_0;\eta_0}$$

The detailed derivation of the score function is provided in the appendix 9 for further reference. The score function is identified as follows:

$$\psi(W,\mathcal{A};\zeta,\eta) = \begin{pmatrix}(\mathbf{Y} - \ell(\mathcal{X},\mathcal{A}) - (\theta+\alpha\mathcal{A})(\mathbf{T} - \mathbf{m}(\mathcal{X},\mathcal{A})))^{\intercal} \\ \times(\mathbf{T} - \mathbf{m}(\mathcal{X},\mathcal{A})) \\ (\mathbf{Y} - \ell(\mathcal{X},\mathcal{A}) - (\theta+\alpha\mathcal{A})(\mathbf{T} - \mathbf{m}(\mathcal{X},\mathcal{A})))^{\intercal} \\ \times\mathcal{A}(\mathbf{T} - \mathbf{m}(\mathcal{X},\mathcal{A}))\end{pmatrix} \tag{13}$$

We can now use the score function to construct an estimator for ζ such that $\psi\left(W,\mathcal{A};\widehat{\zeta},\widehat{\eta}\right) = 0$ where $\widehat{\eta} = \left(\widehat{\ell},\widehat{m}\right)$ are the estimates of nuisance parameters. Thus,

$$\begin{aligned}&\left[\left(\mathbf{Y} - \widehat{\ell}(\mathcal{X},\mathcal{A})\right)^{\intercal}(\mathbf{T} - \widehat{\mathbf{m}}(\mathcal{X},\mathcal{A}))\right] \\ &= \widehat{\theta}\left[(\mathbf{T} - \widehat{\mathbf{m}}(\mathcal{X},\mathcal{A}))^{\intercal}(\mathbf{T} - \widehat{\mathbf{m}}(\mathcal{X},\mathcal{A}))\right] + \\ &\quad\widehat{\alpha}\left[(\mathbf{T} - \widehat{\mathbf{m}}(\mathcal{X},\mathcal{A}))^{\intercal}\mathcal{A}^{\intercal}(\mathbf{T} - \widehat{\mathbf{m}}(\mathcal{X},\mathcal{A}))\right]\end{aligned}$$

and

$$\begin{aligned}&\left[\left(\mathbf{Y} - \widehat{\ell}(\mathcal{X},\mathcal{A})\right)^{\intercal}\mathcal{A}(\mathbf{T} - \widehat{\mathbf{m}}(\mathcal{X},\mathcal{A}))\right] \\ &= \widehat{\theta}\left[(\mathbf{T} - \widehat{\mathbf{m}}(\mathcal{X},\mathcal{A}))^{\intercal}\mathcal{A}(\mathbf{T} - \widehat{\mathbf{m}}(\mathcal{X},\mathcal{A}))\right] + \\ &\quad\widehat{\alpha}\left[(\mathbf{T} - \widehat{\mathbf{m}}(\mathcal{X},\mathcal{A}))^{\intercal}\mathcal{A}^{\intercal}\mathcal{A}(\mathbf{T} - \widehat{\mathbf{m}}(\mathcal{X},\mathcal{A}))\right].\end{aligned}$$

For accurate estimation of nuisance parameters η_0, we use GNNs, which aggregate individual and neighborhood covariate information, handling complex dependencies in network data (Xu et al., 2018; Kipf and Welling, 2016; Veličković et al., 2017; Hamilton et al., 2017). We em-

ploy the Graph Isomorphism Network (GIN) (Xu et al., 2018) due to its superior performance over other GNN architectures like GCN Kipf and Welling (2016), GAT Veličković et al. (2017), and GraphSAGE Hamilton et al. (2017). For consistent estimation of nuisance parameters and to address non-i.i.d. data, we use a focal set approach similar to Athey et al. (2015). Our algorithm first constructs a set of units that are conditionally independent of each other given the ego-centric network features, referred to as the focal set, defined formally below, and then performs cross-fitting to train the graph machine learning model for modeling the nuisance parameters on the focal set. This conditional independence across units aids in consistent estimation of uncertainty around the estimated target parameters by avoiding bias due to dependence between the units. Below, we formally define the focal set:

Definition 3.1. Focal set $\mathbb{S}^* \subseteq \mathbb{V}$ is a maximal set of nodes, in which $\forall u, v \in \mathbb{S}^*$, $\mathbb{N}_u \cap \mathbb{N}_v = \emptyset$. We denote the size of the focal set $\mathbb{S}^*$ as n_f, i.e. $|\mathbb{S}^*| = n_f$

According to the partial interference assumption, since neighborhoods of nodes in the focal set do not overlap, $(\mathcal{X}, \mathbf{T}, \mathbf{Y})$ of these nodes will not be correlated, ensuring that the samples are conditionally independent of each other. Specifically, for any two nodes i and j in the focal set, we assert the following:

$$T_i \perp T_j | X_i, X_{\mathbb{N}_i} \quad Y_i \perp Y_j | X_i, X_{\mathbb{N}_i}, T_i, T_{\mathbb{N}_j} \tag{14}$$

Intuitively, these assumptions reflect local dependencies: the treatment T_i or outcome Y_i of a node depends only on its features X_i and those of its k-hop neighbors $X_{\mathbb{N}_i}$, with independence beyond this neighborhood. This is practical and realistic, as individuals are primarily influenced by their immediate social circles, with distant influences diminishing significantly (Christakis and Fowler, 2007; Granovetter, 1973).

Further, as discussed by Chernozhukov et al. (2018); Zivich and Breskin (2021); Parikh et al. (2022), for the error term of the estimator to vanish, to overcome overfitting, and to gain full efficiency, we cross-fit our estimator. Consider a $K-$fold random partition $(I_k)_{k=1}^K$ of our data $\{1, ..., n_f\}$, such that each fold I_k will be of size $\frac{n_f}{K}$. Let $I_{-k} = \{1, ..., n_f\} \backslash I_k$. For each k, let I_{-k} be the train split and I_k be the estimation split. We construct a ML estimator $\hat{\eta}_k$ of the nuisance function η_0 using the train split:

$$\hat{\eta}_k = \hat{\eta}\left((W_i)_{i \in I_{-k}}\right). \tag{15}$$

Then, for each $k \in \{1 \dots K\}$, we plugin the estimated nuisance parameters $\hat{\eta}_k$ to estimate $\hat{\zeta}_k$ as the solution to $\frac{K}{n_f} \sum_{i \in I_k} \psi\left(W_i, \mathcal{A}; \zeta, \hat{\eta}_k\right) = 0$

Our final estimation would be an aggregation of the estimators: $\widehat{\zeta} = \frac{1}{K} \sum_{k=1}^K \hat{\zeta}_k$. Note that the choice of K may have a significant impact in small sample sizes. Intuitively, selecting larger values of K yields more observations in I_{-k}, which can be advantageous for estimating high-dimensional nuisance functions, which seems to be the more difficult part of the problem. Empirical evidence and simulations indicate that moderate values of K, such as 4 or 5, yield more reliable estimations than using $K = 2$. This underscores the importance of carefully selecting K based on the sample size and the complexity of the functions being estimated.

Putting Everything Together: To estimate ADE and APE, our method begins by constructing a 'focal set' of conditionally independent units, which is the core of our analysis to ensure asymptotic normality. Using graph machine learning, which allows the confounding maps to depend on the covariates of neighboring units, we train models on this focal set to accurately model the nuisance functions. Partial interference assumption assists in accurately defining the graph machine learning model to aggregate information from K-hop away neighbors. To enhance accuracy and robustness, we employ cross-fitting by partitioning the data into multiple folds. For each fold, we estimate the parameters θ_0 (direct effect) and α_0 (peer effect) using linear regression, incorporating the predefined exposure map in the peer effect regression. These estimations are then aggregated across all folds to construct our comprehensive model of network dynamics. This integrative approach, combining the precision of graph machine leaning with the robustness of cross-fitting and the targeted analysis of the focal set, enables a nuanced and precise understanding of causal relationships in social networks. Figure 2 illustrates the proposed framework.

4 Theory

Now, we establish the theoretical results on the consistency and asymptotic normality of the proposed estimator. Detailed proofs are provided in Appendix 10. We consider a nested sequence of networks with an increasing number of units, $\{\mathbb{V}^n, \mathcal{A}^n, \mathcal{Z}^n\}_{n=1}^{\infty}$, such that key features of the network topology, e.g. degree distribution and clustering, are preserved. We assume that the maximum degree of the units in $\mathcal{A}^n$ is $d_n \leq \sqrt{n} - 1$. This growth rate of the maximum degree of the network is a common trait in many real-world social networks where most units possess a low degree, and a smaller proportion of units have a high degree, with the maximum degree dependent on the size of the network (Newman and Park, 2003). This characteristic ensures broad applicability of our model to real-world social networks.

To prove the theoretical results, we need some regularity conditions from the DML framework (Chernozhukov et al., 2018) and adapted to social networks. Intuitively, these conditions ensure enough variability in the treatment and outcome models, prevent the alignment of error terms that could distort causal effect estimation, and ensure accurate

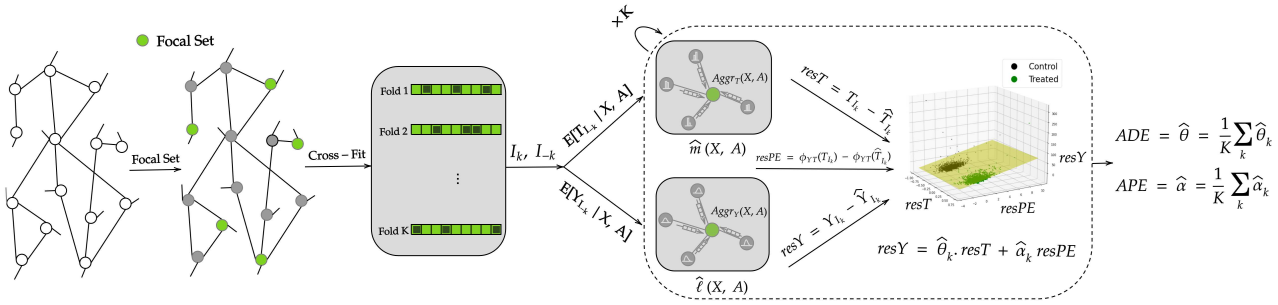

Figure 2: Framework schema. The focal set is partitioned into train and estimation folds I_{-k} and I_k for cross-fitting. Propensity score and outcome models are learned over I_{-k} using graph machine learning. Estimations of $\mathbb{E}[\mathbf{T} \mid \mathcal{X}, \mathcal{A}]$ and $\mathbb{E}[\mathbf{Y} \mid \mathcal{X}, \mathcal{A}]$ for I_k are computed to derive residuals $resT$, $resPE$, and $resY$. Finally, $resY$ is regressed on $resT$ and $resPE$ to obtain $\hat{\theta}_k$ and $\hat{\alpha}_k$ for ADE and APE. This process is repeated across folds, and results are aggregated for final estimations of θ and α.

and reliable estimators for nuisance parameters. While maintaining the positivity of treatment assignment as in Chernozhukov et al. (2018), we introduce an additional regularity condition to ensure the strict positivity of neighborhood exposure. Additionally, they guarantee that the nuisance parameter estimators converge to their true values as the sample size increases, which is crucial for the consistency and asymptotic normality of the causal estimators, allowing for valid statistical inference. The formal statement of these assumptions, along with a comparison to the original DML paper highlighting both similarities and differences, is provided in the Appendix. 10.1.

Theorem 4.1. Under regularity conditions 10.1, the estimator $\tilde{\zeta}_0$ concentrates in a $\sigma/\sqrt{n_f}$-neighborhood of ζ_0 and the sampling error $\sqrt{n_f}\left(\tilde{\zeta}_0 - \zeta_0\right)$ is asymptotically normal

$$\sqrt{n_f}\left(\tilde{\zeta}_0 - \zeta_0\right) \rightsquigarrow N\left(\mathbf{0}_{2\times 1}, \sigma^2_{2\times 2}\right)$$

with mean zero and variance given by
$\sigma^2 := (\mathbf{J_0})^{-1}\mathbb{E}\left[\psi\left(W;\zeta_0,\eta_0\right)\psi\left(W;\zeta_0,\eta_0\right)^\intercal\right]((\mathbf{J_0})^{-1})^\intercal$
where $\mathbf{J_0} = \mathbb{E}\left(\psi_a\left(W;\eta_0\right)\right)$, if the score function is linear in the parameters ζ. For these score functions, estimates of the variance, $\hat{\sigma}^2$, are obtained by

$$(\hat{\mathbf{J}}_\mathbf{0})^{-1}\frac{1}{n_f}\sum_{k=1}^{K}\sum_{i\in I_k}\left[\psi\left(W_i;\tilde{\zeta},\hat{\eta}_k\right)\psi\left(W_i;\tilde{\zeta},\hat{\eta}_k\right)^\intercal\right]((\hat{\mathbf{J}}_\mathbf{0})^{-1})^\intercal$$

$$\text{where } \hat{\mathbf{J}}_\mathbf{0} = \frac{1}{n_f}\sum_{k=1}^{K}\sum_{i\in I_k}\psi_a\left(W_i;\hat{\eta}_k\right)$$

$$\psi_a = \begin{pmatrix} -(\hat{\epsilon}^T)^\intercal(\epsilon^T) & -(\hat{\epsilon}^T)^\intercal\mathcal{A}^\intercal(\hat{\epsilon}^T) \\ -(\hat{\epsilon}^T)^\intercal\mathcal{A}(\hat{\epsilon}^T) & -(\hat{\epsilon}^T)^\intercal\mathcal{A}^\intercal\mathcal{A}(\hat{\epsilon}^T) \end{pmatrix},$$

$$\text{and } \hat{\epsilon}^T = (\mathbf{T} - \hat{\mathbf{m}}(\mathcal{X},\mathcal{A}))$$

The confidence interval is given by $\left[\tilde{\zeta}_0 \pm \hat{\sigma}/\sqrt{n_f}Z^{-1}(1-\alpha/2)\right]$.

The result of Theorem 4.1 guarantees that our estimator is consistent, asymptotically normal, and statistically efficient in the size of the focal set such that the standard deviation shrinks at the rate of $\sqrt{n_f}$. The proof of the theorem is in Appendix 10

5 Empirical Analysis and Results

This section details the empirical evaluation of our framework via semi-synthetic and real-data case studies. These experiments aim to examine our framework's effectiveness and compare its performance with state-of-the-art baseline methods. In Appendix 12, we present additional empirical experiments to investigate the performance of our approach under varying levels of graph density and corresponding effective sample sizes. We evaluate the coverage probability of estimated 95% confidence intervals and explore the performance of an alternative graph aggregation tool combined with our framework, demonstrating the framework's generality beyond GNN models.

Setup: We use real-world networks from the Cora (McCallum et al., 2000), Pubmed (Sen et al., 2008), and Flickr (Guo et al., 2020b) datasets, generating semi-synthetic data with synthetic covariates, treatments, and outcomes to ensure ground truth availability (see Appendix 11.1 and 11.2 for details). Our method employs the GIN (Xu et al., 2018) for propensity score and outcome models, which outperforms other methods we tested, using a single layer of GINConv, followed by two fully connected layers and a softmax layer, with ReLU activation and dropout ($p = 0.5$). GIN models are trained over 300 epochs with a batch size of 16, using the Adam optimizer with a 0.01 learning rate, and $K = 3$ folds for cross-fitting. Our implementation is publicly available[1]. Experiments on Cora and Pubmed ran on a MacBook Pro with an M1 Pro chip, while Flickr was tested on an NVIDIA RTX-3090

[1] https://github.com/BaharanKh/GDML

Table 1: Mean squared error comparison of GDML with baselines. Peer effect estimation is inapplicable for T-Learner and Net TMLE. L&L's framework concentrates on total effect and does not calculate ADE and APE separately.
*: results for Net TMLE, MaTresp, and TNet on Flickr are not reported because it ran out of system memory.

	Cora			Pubmed			Flickr		
	ADE	APE	ATE	ADE	APE	ATE	ADE	APE	ATE
PA	$0.31_{\pm 0.83}$	$1.02_{\pm 2.90}$	$1.41_{\pm 4.05}$	$0.35_{\pm 0.69}$	$10.69_{\pm 6.52}$	$14.30_{\pm 9.66}$	$1133_{\pm 4700}$	$37719_{\pm 143300}$	$51790_{\pm 199836}$
T-learner(Künzel et al., 2019)	$9.84_{\pm 51.32}$	N/A	N/A	$1.67_{\pm 4.58}$	N/A	N/A	$2380_{\pm 5495}$	N/A	N/A
NetEst(Jiang and Sun, 2022)	$174.66_{\pm 1.07}$	$9.48_{\pm 1.75}$	$71.96_{\pm 4.35}$	$1655.8_{\pm 30.94}$	$0.44_{\pm 0.39}$	$1603.53_{\pm 45.65}$	$53827_{\pm 921}$	$103_{\pm 105}$	$58503_{\pm 3444}$
Net TMLE(Ogburn et al., 2022)	$13.67_{\pm 6.47}$	N/A	N/A	$1.24_{\pm 1.36}$	N/A	N/A	N/A*	N/A	N/A
L&L(Leung and Loupos, 2022)	N/A	N/A	$120.20_{\pm 34.31}$	N/A	N/A	$42.76_{\pm 2.92}$	N/A	N/A	$25.32_{\pm 6.58}$
Ma & Tresp(Ma and Tresp, 2020)	$3.87_{\pm 42.98}$	$0.02_{\pm 0.14}$	$4.26_{\pm 47.37}$	$0.02_{\pm 0.03}$	$0.01_{\pm 0.01}$	$0.04_{\pm 0.06}$	N/A*	N/A*	N/A*
GDML w/o FS	$0.26_{\pm 0.83}$	$0.99_{\pm 2.79}$	$1.37_{\pm 3.72}$	$0.04_{\pm 0.13}$	$0.30_{\pm 0.73}$	$0.45_{\pm 1.21}$	$4.92_{\pm 10.91}$	$121_{\pm 308}$	$95_{\pm 276}$
GDML	$0.33_{\pm 0.79}$	$0.29_{\pm 0.80}$	$0.88_{\pm 2.21}$	0.03 ± 0.11	$0.28_{\pm 0.84}$	$0.30_{\pm 0.87}$	$76_{\pm 211}$	$26.01_{\pm 26.07}$	$84_{\pm 272}$

Table 2: Comparison of runtime in seconds for different methods on each dataset. N/A indicates that the method did not return any results within a 12-hour runtime limit.

	PA	T-learner (Künzel et al., 2019)	NetEst (Jiang and Sun, 2022)	Net TMLE (Ogburn et al., 2022)	L&L (Leung and Loupos, 2022)	Ma & Tresp (Ma and Tresp, 2020)	GDML w/o FS	GDML
Cora	2	3.32	19020	4	5	26	9	5
Pubmed	55	48	22560	2464	133	8604	104	64
Flickr	1666	1712	31118	N/A	2909	N/A	2296	1832

GPU.

Baselines: We compare our method against six baselines. NetEst Jiang and Sun (2022) uses GNNs with adversarial learning for confounder representations. Net-TMLE Ogburn et al. (2022) derives a doubly robust estimator using an efficient influence function. The T-Learner Künzel et al. (2019) models each treatment arm separately with GNNs. DML with predefined aggregates applies DML in the i.i.d. setting using max, min, and mean neighbor aggregates. Ma & Tresp Ma and Tresp (2020) maps covariate representations with HSIC regularization, using GNNs for neighbor aggregation. The L&L method Leung and Loupos (2022) combines a doubly robust estimator with GNNs, requiring binary exposure conversion. Lastly, Guo et al. (2020b) models hidden confounders but reduces to the T-learner under our assumptions. For more details on the baselines, please refer to Appendix 11.3.

Results: We compare two versions of our method: GDML w/o FS, which does not use a focal set and encompasses the entire dataset, and GDML, which operationalize our method using focal set to evaluate the effect of this strategy on the quality of the results. Table 1 demonstrates the results, and Table 2 reports the running time. We report the mean squared error (MSE) calculated over 100 simulations for each evaluated method (The decomposition of the reported MSE into bias and variance is presented in the table7 and the relative errors are reported in Figure 3 in the appendix).

Across all three semi-synthetic datasets, our GDML approach performs on par with or better than the state-of-the-art baseline methods (see Table 1) and scales significantly better (see Table 2). This performance enhancement can be attributed to the use of graph ML methods (such as GIN) to adjust for network confounders. Further, the DML framework guarantees consistency and efficiency while using complex ML methods with regularization. Upon closer examination of Table 1, our method demonstrates consistently strong results, being either the best or second-best across all cases, and significant advantages in scalability and ability for uncertainty quantification compared to existing baselines, which is crucial in causal inference to assess the reliability of estimated effects.

Comparing GDML with the GIN-based T-learner shows that the GIN-based T-learner has higher MSE due to regularization-induced bias. Methods that employ predefined aggregation functions, such as Net-TMLE and DML methods using predefined aggregates (min, max, sum, and average), fall short as these aggregates do not capture complex network functions as effectively as GNNs. The regularization bias introduced by using GNNs in non-doubly robust estimators, such as T-Learner and NetEst, or by employing simple predefined aggregates in PA, is further exacerbated as the network size and complexity increase, as demonstrated by experiments with the Flickr dataset. Specifically:

Performance Comparison: Our method consistently outperforms four baselines (PA, T-Learner, NetEst, and Net-TMLE). Compared to L&L, our method achieves better results in two out of three cases. While L&L occasionally shows slightly lower MSE, it is limited by its inability to de-

compose treatment effects into direct and peer effects and its restriction to binary treatments and exposures. In contrast, our framework handles more general settings. Similarly, against Ma & Tresp, our method outperforms in two cases, matches their performance in one, and is less effective in two; however, Ma & Tresp fails to terminate on the larger Flickr dataset, highlighting scalability issues.

Scalability and Efficiency: Tresp & Ma, while slightly better in certain cases, fails to scale well, taking more than two and a half hours on PubMed and failing to terminate on Flickr within a 12-hour time limit.In contrast, our GDML method completes PubMed in just 64 seconds and Flickr in 31 minutes (Table 2), demonstrating unparalleled efficiency and scalability. This makes GDML the superior choice for handling larger and more complex datasets.

Statistical Interpretability: Importantly, only our method, L&L, and Net-TMLE allow for computing confidence intervals, which are crucial for statistical inference.

In summary, while certain baselines may occasionally achieve marginally lower MSEs, our GDML method consistently delivers competitive or superior performance, scales effectively to large networks, and offers statistical interpretability. These advantages make GDML a compelling and well-rounded solution for diverse applications.

A pivotal aspect of our methodology is the emphasis on focal sets analysis, exploiting the conditional independence between units to enhance performance. While restricting to focal sets reduces the size of training data for nuisance parameter estimation, this focus has demonstrably outperformed variants of our method that omit focal sets, highlighting the strategic value of considering focal sets in the analysis of networked data. This ensures robust and accurate estimations, validating our approach in dealing with networked data.

Analysis of Real Data For a case study, we used the Indian Village dataset from Karnataka, India, encompassing 16,995 individuals across 77 villages, with 15 features and 12 social networks (Banerjee et al., 2014; Jackson et al., 2012). This dataset's rich social structure provides insights into economic and social behaviors, such as borrowing, lending, and advice networks, making it invaluable for understanding the impacts of social networks on individual and collective outcomes. We operationalize our approach to estimating the causal effect of participation in self-help groups (SHGs) on financial risk tolerance, measured by the existence of an outstanding credit/loan. Specifically, we are interested in estimating the direct and peer effects of SHG participation. We construct the focal set that consists of 1766 individuals.

Our analysis indicates that the point estimate for the average direct effect (ADE) is 0.315 with a 95% confidence interval ranging from -1.570 to 2.200. The positive point estimate suggests a potential positive effect of SHG participation. However, given the limited effective sample size in the social network setting (which is equal to the size of the focal set in our case), these estimates are not statistically significant, which is expected as the potential effect size in such social interventions is typically small. Additionally, our results show that the APE is approximately zero (0.050), indicating minimal to no benefit from peers' participation in SHGs. We provide point estimates from the baselines in Appendix12.2 – our estimates agree with these point estimates as well as with the literature.

We also conducted a qualitative assessment, aligning our findings with established econometric research. Studies show that SHG participation boosts financial risk tolerance (Güner and Yildiz, 2019; Srivastava et al., 2024; Kumar and Singh, 2023; Patil and Patil, 2024) through mechanisms like financial literacy, economic empowerment, and social support. SHGs provide financial education, improving risk management, and offer credit and savings opportunities, enhancing financial confidence. The collaborative nature of SHGs builds social networks that reduce perceived financial risks through shared knowledge and access to formal services like credit and insurance, empowering members to take calculated financial actions.

6 Conclusions

Our work presents a double-machine learning framework combined with graph representation learning to adjust for complex network confounders and efficiently estimate treatment effects. Simulations and real data studies demonstrate its effectiveness. Limitations include reliance on GNN convergence assumptions, which remain an active research area, and the need to observe all confounders, despite handling high-dimensional covariates in propensity scores and outcome models. Its performance declines with densely connected graphs due to reduced effective sample size. Future work involves adapting the framework for relational data with heterogeneous graphs, exploring higher-order graphs to surpass message-passing GNN expressive power limitations, and assessing the impact of missing network ties on estimation in partially observable graphs.

7 Acknowledgments

This research was supported by NSF awards IIS-2340124 and IIS-2147061, NIH grant U54HG012510, and NIH NIDA R01DA056407-01. The views, opinions, and findings presented are those of the authors and do not necessarily represent those of the NSF or NIH.

References

Aronow, P. M. and Samii, C. (2013). Estimating average causal effects under interference between units. arXiv: Statistics Theory.

Aronow, P. M. and Samii, C. (2017). Estimating average causal effects under general interference, with application to a social network experiment.

Atanasov, V. A. and Black, B. S. (2016). Shock-based causal inference in corporate finance and accounting research. Critical Finance Review, 5:207–304.

Athey, S., Eckles, D., and Imbens, G. W. (2015). Exact P-values for Network Interference. NBER Working Papers 21313, National Bureau of Economic Research, Inc.

Auerbach, E. and Tabord-Meehan, M. (2021). The local approach to causal inference under network interference. arXiv preprint arXiv:2105.03810.

Banerjee, A., Chandrasekhar, A. G., Duflo, E., and Jackson, M. O. (2014). Gossip: Identifying central individuals in a social network. Technical report, National Bureau of Economic Research.

Bhattacharya, R., Malinsky, D., and Shpitser, I. (2019). Causal inference under interference and network uncertainty. Uncertain Artificial Intelligence, 2019:372.

Bland, J. M. (2004-08-13). Cluster randomised trials in the medical literature: two bibliometric surveys. 4(1):21.

Blattman, C., Green, D. P., Ortega, D., and Tobón, S. (2021). Place-Based Interventions at Scale: The Direct and Spillover Effects of Policing and City Services on Crime [Clustering as a Design Problem]. Journal of the European Economic Association, 19(4):2022–2051.

Cai, R., Yang, Z., Chen, W., Yan, Y., and Hao, Z. (2023). Generalization bound for estimating causal effects from observational network data. In Proceedings of the 32nd ACM International Conference on Information and Knowledge Management, CIKM '23, page 163–172, New York, NY, USA. Association for Computing Machinery.

Chen, W., Cai, R., Yang, Z., Qiao, J., Yan, Y., Li, Z., and Hao, Z. (2024). Doubly robust causal effect estimation under networked interference via targeted learning. arXiv preprint arXiv:2405.03342.

Chernozhukov, V., Chetverikov, D., Demirer, M., Duflo, E., Hansen, C., Newey, W., and Robins, J. (2018). Double/debiased machine learning for treatment and structural parameters.

Christakis, N. A. and Fowler, J. H. (2007). The spread of obesity in a large social network over 32 years. New England Journal of Medicine, 357(4):370–379.

Cristali, I. and Veitch, V. (2022). Using embeddings for causal estimation of peer influence in social networks. In Oh, A. H., Agarwal, A., Belgrave, D., and Cho, K., editors, Advances in Neural Information Processing Systems.

Eckles, D., Karrer, B., and Ugander, J. (2016). Design and analysis of experiments in networks: Reducing bias from interference. Journal of Causal Inference, 5(1):20150021.

Emmenegger, C., Spohn, M.-L., Elmer, T., and Bühlmann, P. (2022). Treatment effect estimation from observational network data using augmented inverse probability weighting and machine learning. arXiv preprint arXiv:2206.14591.

Farrell, M. H., Liang, T., and Misra, S. (2021). Deep neural networks for estimation and inference. Econometrica, 89(1):181–213.

Favaro, S., Fortini, S., and Peluchetti, S. (2023). Deep stable neural networks: large-width asymptotics and convergence rates. Bernoulli, 29(3):2574–2597.

Forastiere, L., Airoldi, E. M., and Mealli, F. (2021). Identification and estimation of treatment and interference effects in observational studies on networks. Journal of the American Statistical Association, 116(534):901–918.

Forastiere, L., Mealli, F., Wu, A., and Airoldi, E. M. (2018). Estimating causal effects on social networks. In 2018 IEEE 5th International Conference on Data Science and Advanced Analytics (DSAA), pages 60–69.

Frauen, D., Imrie, F., Curth, A., Melnychuk, V., Feuerriegel, S., and van der Schaar, M. (2023). A neural framework for generalized causal sensitivity analysis. arXiv preprint arXiv:2311.16026.

Gallier, J. and Quaintance, J. (2023). Algebra, Topology, Differential Calculus, and Optimization Theory For Computer Science and Machine Learning, chapter 9, page 336. University of Pennsylvania. Proposition 9.8.

Gassen, J. (2014). Causal inference in empirical archival financial accounting research. Accounting, Organizations and Society, 39(7):535–544.

Gilad, A., Parikh, H., Roy, S., and Salimi, B. (2021). Heterogeneous treatment effects in social networks. arXiv preprint arXiv:2105.10591.

Granovetter, M. S. (1973). The strength of weak ties. American Journal of Sociology, 78(6):1360–1380.

Guo, R. (2019). Learning individual causal effects from networked observational data. Proceedings of the 13th International Conference on Web Search and Data Mining.

Guo, R., Li, J., and Liu, H. (2020a). Ignite: A minimax game toward learning individual treatment effects from networked observational data. In International Joint Conference on Artificial Intelligence.

Guo, R., Li, J., and Liu, H. (2020b). Learning individual causal effects from networked observational data. In

Proceedings of the 13th international conference on web search and data mining, pages 232–240.

Güner, Z. N. and Yildiz, B. (2019). Effect of financial literacy and risk perception on individual investment choices. In European Proceedings of Social and Behavioural Sciences, volume 68, pages 563–570.

Halloran, M. E. and Hudgens, M. G. (2012). Causal inference for vaccine effects on infectiousness. The International Journal of Biostatistics, 8(2):1–40.

Halloran, M. E. and Struchiner, C. J. (1995). Causal inference in infectious diseases. Epidemiology, 6(2):142–151.

Hamilton, W., Ying, Z., and Leskovec, J. (2017). Inductive representation learning on large graphs. Advances in neural information processing systems, 30.

Holland, P. W., Laskey, K. B., and Leinhardt, S. (1983). Stochastic blockmodels: First steps. Social Networks, 5(2):109–137.

Hu, Y., Li, S., and Wager, S. (2022). Average direct and indirect causal effects under interference. Biometrika, 109(4):1165–1172.

Hudgens, M. and Halloran, M. (2008a). Toward causal inference with interference. Journal of the American Statistical Association, 103:832–842.

Hudgens, M. G. and Halloran, E. (2008b). Toward causal inference with interference. Journal of the American Statistical Association, 103:832 – 842.

Imbens, G. W. (2003). Sensitivity to exogeneity assumptions in program evaluation. American Economic Review, 93(2):126–132.

Jackson, M. O. et al. (2008). Social and economic networks, volume 3. Princeton university press Princeton.

Jackson, M. O., Rodriguez-Barraquer, T., and Tan, X. (2012). Social capital and social quilts: Network patterns of favor exchange. American Economic Review, 102(5):1857–1897.

Jegelka, S. (2022). Theory of graph neural networks: Representation and learning. In The International Congress of Mathematicians.

Jiang, S. and Sun, Y. (2022). Estimating causal effects on networked observational data via representation learning. In Proceedings of the 31st ACM International Conference on Information & Knowledge Management, pages 852–861.

Johansson, F., Shalit, U., and Sontag, D. (2016). Learning representations for counterfactual inference. In Balcan, M. F. and Weinberger, K. Q., editors, Proceedings of The 33rd International Conference on Machine Learning, volume 48 of Proceedings of Machine Learning Research, pages 3020–3029, New York, New York, USA. PMLR.

Karrer, B., Shi, L., Bhole, M., Goldman, M., Palmer, T., Gelman, C., Konutgan, M., and Sun, F. (2021). Network experimentation at scale. In Proceedings of the 27th acm sigkdd conference on knowledge discovery & data mining, pages 3106–3116.

Kipf, T. N. and Welling, M. (2016). Semi-supervised classification with graph convolutional networks. arXiv preprint arXiv:1609.02907.

Kohler, M. and Langer, S. (2019). On the rate of convergence of fully connected very deep neural network regression estimates. arXiv preprint arXiv:1908.11133.

Kumar, A. and Singh, R. (2023). Financial inclusion and risk-taking behavior: Evidence from self-help groups in india. Journal of Development Economics, 150:102635.

Künzel, S. R., Sekhon, J. S., Bickel, P. J., and Yu, B. (2019). Metalearners for estimating heterogeneous treatment effects using machine learning. Proceedings of the National Academy of Sciences, 116(10):4156–4165.

Laan, M. (2014). Causal inference for a population of causally connected units. Journal of Causal Inference, 2.

Leung, M. P. and Loupos, P. (2022). Unconfoundedness with network interference. arXiv preprint arXiv:2211.07823.

Liu, J., Ye, F., and Yang, Y. (2023). Nonparametric doubly robust estimation of causal effect on networks in observational studies. Stat, 12(1):e549.

Liu, L., Hudgens, M. G., and Becker-Dreps, S. (2016). On inverse probability-weighted estimators in the presence of interference. Biometrika, 103(4):829–842.

Ma, Y. and Tresp, V. (2020). Causal inference under networked interference and intervention policy enhancement. In International Conference on Artificial Intelligence and Statistics.

Manski, C. F. (2013). Identification of treatment response with social interactions. The Econometrics Journal, 16(1):S1–S23.

McCallum, A. K., Nigam, K., Rennie, J., and Seymore, K. (2000). Automating the construction of internet portals with machine learning. Information Retrieval, 3(2):127–163.

McNealis, V., Moodie, E. E., and Dean, N. (2023). Doubly robust estimation of causal effects in network-based observational studies. arXiv preprint arXiv:2302.00230.

Morris, C., Ritzert, M., Fey, M., Hamilton, W. L., Lenssen, J. E., Rattan, G., and Grohe, M. (2019). Weisfeiler and leman go neural: Higher-order graph neural networks. In Proceedings of the Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, Honolulu, Hawaii, USA, January 27 - February 1, 2019, pages 4602–4609. AAAI Press.

Morucci, M., Orlandi, V., Parikh, H., Roy, S., Rudin, C., and Volfovsky, A. (2023). A double machine learning approach to combining experimental and observational data. arXiv preprint arXiv:2307.01449.

Newman, M. E. and Park, J. (2003). Why social networks are different from other types of networks. Physical review E, 68(3):036122.

Neyman, J. (1959). Optimal asymptotic tests of composite statistical hypotheses. In Grenander, U., editor, Probability and Statistics, pages 416–444. John Wiley, New York.

Ogburn, E. L., Sofrygin, O., Diaz, I., and Van der Laan, M. J. (2022). Causal inference for social network data. Journal of the American Statistical Association, pages 1–15.

Ogburn, E. L. and VanderWeele, T. J. (2014). Causal diagrams for interference.

Papadogeorgou, G. and Samanta, S. (2023). Spatial causal inference in the presence of unmeasured confounding and interference. arXiv preprint arXiv:2303.08218.

Parikh, H., Rudin, C., and Volfovsky, A. (2022). Malts: Matching after learning to stretch. The Journal of Machine Learning Research, 23(1):10952–10993.

Patil, S. V. and Patil, R. G. (2024). Empowering self-help groups: The impact of financial inclusion on social well-being in rural maharashtra, india. Journal of Risk and Financial Management, 17(6):217.

Pouget-Abadie, J., Saint-Jacques, G., Saveski, M., Duan, W., Ghosh, S., Xu, Y., and Airoldi, E. M. (2019). Testing for arbitrary interference on experimentation platforms. Biometrika, 106(4):929–940.

Rosenbaum, P. R. (1987). Sensitivity analysis for certain permutation inferences in matched observational studies. Biometrika, 74(1):13–26.

Salimi, B., Parikh, H., Kayali, M., Getoor, L., Roy, S., and Suciu, D. (2020). Causal relational learning. In Proceedings of the 2020 ACM SIGMOD international conference on management of data, pages 241–256.

Sen, P., Namata, G., Bilgic, M., Getoor, L., Galligher, B., and Eliassi-Rad, T. (2008). Collective classification in network data. AI Magazine, 29(3):93.

Shalit, U., Johansson, F. D., and Sontag, D. (2017). Estimating individual treatment effect: generalization bounds and algorithms. In Precup, D. and Teh, Y. W., editors, Proceedings of the 34th International Conference on Machine Learning, volume 70 of Proceedings of Machine Learning Research, pages 3076–3085. PMLR.

Smucler, E., Rotnitzky, A., and Robins, J. M. (2019). A unifying approach for doubly-robust ℓ_1 regularized estimation of causal contrasts. arXiv preprint arXiv:1904.03737.

Sobel, M. E. (2006). What do randomized studies of housing mobility demonstrate? Journal of the American Statistical Association, 101(476):1398–1407.

Srivastava, B., Kandpal, V., and Jain, A. K. (2024). Financial well-being of women self-help group members: A qualitative study. Environment, Development and Sustainability, 26(4):3456–3472.

Tang, T., Levina, L., and Zhu, J. (2024). Interpretable network-assisted prediction via random forests. Working Paper (Unpublished).

Tchetgen, E. J. T. and VanderWeele, T. J. (2012). On causal inference in the presence of interference. Statistical Methods in Medical Research, 21(1):55–75. PMID: 21068053.

Toulis, P. and Kao, E. (2013). Estimation of causal peer influence effects. In Dasgupta, S. and McAllester, D., editors, Proceedings of the 30th International Conference on Machine Learning, volume 28 of Proceedings of Machine Learning Research, pages 1489–1497, Atlanta, Georgia, USA. PMLR.

Ugander, J., Karrer, B., Backstrom, L., and Kleinberg, J. (2013). Graph cluster randomization: network exposure to multiple universes. In Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '13, page 329–337, New York, NY, USA. Association for Computing Machinery.

Ugander, J. and Yin, H. (2023). Randomized graph cluster randomization. Journal of Causal Inference, 11(1):20220014.

VanderWeele, T. and Tchetgen Tchetgen, E. (2011). Bounding the infectiousness effect in vaccine trials. Epidemiology, 22(5):686–693.

VanderWeele, T. J. (2011). Sensitivity analysis for contagion effects in social networks. Sociological Methods & Research, 40(2):240–255.

VanderWeele, T. J. and An, W. (2013). Social networks and causal inference. Handbook of causal analysis for social research, pages 353–374.

Veitch, V., Wang, Y., and Blei, D. (2019). Using embeddings to correct for unobserved confounding in networks. In Wallach, H., Larochelle, H., Beygelzimer, A., Alché-Buc, F. d., Fox, E., and Garnett, R., editors, Advances in Neural Information Processing Systems, volume 32. Curran Associates, Inc.

Veličković, P., Cucurull, G., Casanova, A., Romero, A., Lio, P., and Bengio, Y. (2017). Graph attention networks. arXiv preprint arXiv:1710.10903.

Xu, K., Hu, W., Leskovec, J., and Jegelka, S. (2018). How powerful are graph neural networks? arXiv preprint arXiv:1810.00826.

Yao, L., Li, S., Li, Y., Huai, M., Gao, J., and Zhang, A. (2018). Representation learning for treatment effect estimation from observational data. In Bengio, S., Wallach, H., Larochelle, H., Grauman, K., Cesa-Bianchi, N., and Garnett, R., editors, Advances in Neural Information Processing Systems, volume 31. Curran Associates, Inc.

Zigler, C. M. and Papadogeorgou, G. (2021). Bipartite causal inference with interference. Statistical science: a review journal of the Institute of Mathematical Statistics, 36(1):109.

Zivich, P. N. and Breskin, A. (2021). Machine learning for causal inference: on the use of cross-fit estimators. Epidemiology (Cambridge, Mass.), 32(3):393.

Checklist

1. For all models and algorithms presented, check if you include:
   (a) A clear description of the mathematical setting, assumptions, algorithm, and/or model. [Yes, the setting and assumptions are clearly discussed in section 2.1 and the algorithm is discussed in section 3.]
   (b) An analysis of the properties and complexity (time, space, sample size) of any algorithm. [Yes. In terms of theoretical properties, they are discussed in section 4 and for time, the complexity is not discussed in terms of input size, but a comparison of runtime of the algorithm with other baselines is presented in table 2.]
   (c) (Optional) Anonymized source code, with specification of all dependencies, including external libraries. [No. We aim to publicize the code upon acceptance.]

2. For any theoretical claim, check if you include:
   (a) Statements of the full set of assumptions of all theoretical results. [Yes, in the appendix 10.1]
   (b) Complete proofs of all theoretical results. [Yes, the proof for identifiability is presented in the appendix 8 and the complete proof for the main theorem is discussed in the appendix 4.1.]
   (c) Clear explanations of any assumptions. [Yes, after listing the assumptions in the appendix 10.1, we discuss the assumptions and their implications.]

3. For all figures and tables that present empirical results, check if you include:
   (a) The code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL). [No, we aim to publicize the git repository upon acceptence. The dat]
   (b) All the training details (e.g., data splits, hyperparameters, how they were chosen). [Yes, in paragraph 5 in section 5.]
   (c) A clear definition of the specific measure or statistics and error bars (e.g., with respect to the random seed after running experiments multiple times). [Yes, in paragraph 5.]
   (d) A description of the computing infrastructure used. (e.g., type of GPUs, internal cluster, or cloud provider). [Yes, in paragraph 5 in section 5.]

4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets, check if you include:
   (a) Citations of the creator If your work uses existing assets. [Yes, we properly cited all the works we are referring to.]
   (b) The license information of the assets, if applicable. [Not Applicable]
   (c) New assets either in the supplemental material or as a URL, if applicable. [Not Applicable]
   (d) Information about consent from data providers/curators. [Not Applicable, the data we used for our experiments are open-sourced.]
   (e) Discussion of sensible content if applicable, e.g., personally identifiable information or offensive content. [Not Applicable]

5. If you used crowdsourcing or conducted research with human subjects, check if you include:
   (a) The full text of instructions given to participants and screenshots. [Not Applicable]
   (b) Descriptions of potential participant risks, with links to Institutional Review Board (IRB) approvals if applicable. [Not Applicable]
   (c) The estimated hourly wage paid to participants and the total amount spent on participant compensation. [Not Applicable]

Graph Machine Learning based Doubly Robust Estimator for Network Causal Effects: Supplementary Materials

8 Proof of Identifiability

In this section, we present a detailed, step-by-step proof of the identifiability of the Average Direct Effect (ADE) and the Average Partial Effect (APE), based on the assumptions outlined in Section 2.1.

$$\tau_{ADE} = \mathbb{E}\left[\frac{1}{n}\sum_{i\in\mathcal{V}}\tau_{i,DE}\right] \tag{16}$$

$$= \mathbb{E}\left[\frac{1}{n}\sum_{i\in\mathcal{V}}Y_i(1,\mathbf{T}_{-\mathbf{i}}) - Y_i(0,\mathbf{T}_{-\mathbf{i}})\right] = \tag{17}$$

$$= \mathbb{E}\left[\frac{1}{n}\sum_{i\in\mathcal{V}}Y_i(1,\mathbf{T}_{\mathbb{N}_\mathbf{i}}) - Y_i(0,\mathbf{T}_{\mathbb{N}_\mathbf{i}})\right] = \tag{18}$$

$$= \mathbb{E}\left[\frac{1}{n}\sum_{i\in\mathcal{V}}Y_i(1,z_i) - Y_i(0,z_i)\right] = \tag{19}$$

$$= \mathbb{E}_X\left[\mathbb{E}\left[\frac{1}{n}\sum_{i\in\mathcal{V}}Y_i(1,z_i) - Y_i(0,z_i) \mid \mathcal{X}\right]\right] = \tag{20}$$

$$= \mathbb{E}_X\left[\mathbb{E}\left[\frac{1}{n}\sum_{i\in\mathcal{V}}Y_i(1,z_i) - Y_i(0,z_i) \mid \mathbf{X_i}, \mathcal{X}_{\mathbb{N}_i}\right]\right] = \tag{21}$$

$$= \mathbb{E}_X\Bigg[\mathbb{E}\Bigg[\frac{1}{n}\sum_{i\in\mathcal{V}}Y_i(1,z_i) \mid \mathbf{X_i}, \mathcal{X}_{\mathbb{N}_i}, t_i, z_i\Bigg] - \mathbb{E}\Bigg[\frac{1}{n}\sum_{i\in\mathcal{V}}Y_i(0,z_i) \mid \mathbf{X_i}, \mathcal{X}_{\mathbb{N}_i}, t_i, z_i\Bigg]\Bigg] \tag{22}$$

$$= \mathbb{E}_X\Bigg[\mathbb{E}\Bigg[\frac{1}{n}\sum_{i\in\mathcal{V}}Y_i \mid \mathbf{X_i}, \mathcal{X}_{\mathbb{N}_i}, t_i = 1, z_i\Bigg] - \mathbb{E}\Bigg[\frac{1}{n}\sum_{i\in\mathcal{V}}Y_i \mid \mathbf{X_i}, \mathcal{X}_{\mathbb{N}_i}, t_i = 0, z_i\Bigg]\Bigg] \tag{23}$$

Equation 18 uses partial interference assumption, 19 uses the assumption that the exposure map is well-defined and known, 20 uses law of total expectation, 21 uses partial interference assumption, 22 uses strong ignorability and 23 uses consistency assumption.

$$\tau_{APE} = \mathbb{E}\left[\frac{1}{n}\sum_{i\in\mathcal{V}}\tau_{i,PE}\right] \tag{24}$$

$$= \mathbb{E}\left[\frac{1}{n}\sum_{i\in\mathcal{V}} Y_i(T_i, \mathbf{1}_{-i}) - Y_i(T_i, \mathbf{0}_{-i})\right] = \tag{25}$$

$$= \mathbb{E}\left[\frac{1}{n}\sum_{i\in\mathcal{V}} Y_i(T_i, \mathbf{T}_{\mathbb{N}_\mathbf{i}} = \mathbf{1}) - Y_i(T_i, \mathbf{T}_{\mathbb{N}_\mathbf{i}} = \mathbf{0})\right] = \tag{26}$$

$$= \mathbb{E}\left[\frac{1}{n}\sum_{i\in\mathcal{V}} Y_i(T_i, z_i') - Y_i(T_i, z_i'')\right] = \tag{27}$$

$$= \mathbb{E}_X\left[\mathbb{E}\left[\frac{1}{n}\sum_{i\in\mathcal{V}} Y_i(T_i, z_i') - Y_i(T_i, z_i'') \mid \mathcal{X}\right]\right] = \tag{28}$$

$$= \mathbb{E}_X\left[\mathbb{E}\left[\frac{1}{n}\sum_{i\in\mathcal{V}} Y_i(T_i, z_i') - Y_i(T_i, z_i'') \mid \mathbf{X_i}, \mathcal{X}_{\mathbb{N}_i}\right]\right] = \tag{29}$$

$$\begin{aligned} &= \mathbb{E}_X\Bigg[\mathbb{E}\Bigg[\frac{1}{n}\sum_{i\in\mathcal{V}} Y_i(T_i, z_i') \mid \mathbf{X_i}, \mathcal{X}_{\mathbb{N}_i}, t_i, z_i\Bigg] - \\ &\quad \mathbb{E}\Bigg[\frac{1}{n}\sum_{i\in\mathcal{V}} Y_i(T_i, z_i'') \mid \mathbf{X_i}, \mathcal{X}_{\mathbb{N}_i}, t_i, z_i\Bigg]\Bigg] \end{aligned} \tag{30}$$

$$\begin{aligned} &= \mathbb{E}_X\Bigg[\mathbb{E}\Bigg[\frac{1}{n}\sum_{i\in\mathcal{V}} Y_i \mid \mathbf{X_i}, \mathcal{X}_{\mathbb{N}_i}, t_i, z_i = z_i'\Bigg] - \\ &\quad \mathbb{E}\Bigg[\frac{1}{n}\sum_{i\in\mathcal{V}} Y_i \mid \mathbf{X_i}, \mathcal{X}_{\mathbb{N}_i}, t_i, z_i = z_i''\Bigg]\Bigg] \end{aligned} \tag{31}$$

Equation 26 uses partial interference assumption, 27 uses the assumption that the exposure map is well-defined and known, 28 uses law of total expectation, 29 uses partial interference assumption, 30 uses strong ignorability and 31 uses consistency assumption.

9 Derivation of score function

In this section, we introduce the concept of the neyman orthogonal score function and proceed to derive the corresponding score function pertinent to our study. This derivation is structured around our specific set of structural equations and is guided by the methodology outlined in (Neyman, 1959; Chernozhukov et al., 2018).

Let $\zeta \in \mathcal{Z} \subset \mathbb{R}^{d_\zeta}$ and $\beta \in \mathcal{B} \subset \mathbb{R}^{d_\beta}$ be the target and nuisance parameters respectively. Suppose the true parameter values ζ_0 and β_0 that solves the following optimization problem

$$\max_{\zeta\in\mathcal{Z},\beta\in\mathcal{B}} \mathbb{E}_P[\mathcal{L}(W;\zeta,\beta)]$$

where W is a random element taking values in a measurable space $(\mathcal{W}, \mathcal{A}_\mathcal{W})$ with law determined by a probability measure $P \in \mathcal{P}_n$ and $\mathcal{L}(W;\zeta,\beta)$ is a known criterion function. ζ_0 and β_0 satisfy

$$\mathbb{E}_P\left[\partial_\zeta \mathcal{L}\left(W;\zeta_0,\beta_0\right)\right] = 0, \quad \mathbb{E}_P\left[\partial_\beta \mathcal{L}\left(W;\zeta_0,\beta_0\right)\right] = 0$$

Definition. (neyman orthogonality) The score $\psi = (\psi_1, \ldots, \psi_{d_\theta})'$ obeys the orthogonality condition at (θ_0, η_0) with respect to the nuisance realization set $\Gamma_n \subset T$ if

$$\mathbb{E}_P\left[\psi\left(W;\theta_0,\eta_0\right)\right] = 0$$

holds and the pathwise derivative map $\mathrm{D}_r\left[\eta - \eta_0\right]$ exists for all $r \in [0,1)$ and $\eta \in \Gamma_n$ and vanishes at $r = 0$; namely,

$$\partial_\eta \mathbb{E}_P \psi\left(W;\theta_0,\eta_0\right)\left[\eta - \eta_0\right] = 0, \quad \text{for all } \eta \in \Gamma_n.$$

We remark here that the condition holds with $\Gamma_n = T$ when η is a finite-dimensional vector as long as $\partial_\eta \mathbb{E}_P\left[\psi_j\left(W;\theta_0,\eta_0\right)\right] = 0$ for all $j = 1, \ldots, d_\theta$, where $\partial_\eta \mathbb{E}_P\left[\psi_j\left(W;\theta_0,\eta_0\right)\right]$ denotes the vector of partial derivatives of the function $\eta \mapsto \mathbb{E}_P\left[\psi_j\left(W;\theta_0,\eta\right)\right]$ for $\eta = \eta_0$.

The neyman orthogonal score function is

$$\psi(W, \mathcal{A}; \zeta, \eta) = \partial_\zeta \mathcal{L}(W;\zeta,\beta) - \mu \partial_\beta \mathcal{L}(W;\zeta,\beta)$$

where $\psi = (\psi_1, ..., \psi_{d_\zeta})'$ is a vector of known score functions,the nuisance parameter is

$$\eta = (\beta', \text{vec}(\mu)')' \in T = \mathcal{B} \times \mathbb{R}^{d_\zeta d_\beta} \subset \mathbb{R}^p, \quad p = d_\beta + d_\zeta d_\beta,$$

and μ is the $d_\zeta \times d_\beta$ orthogonalization parameter matrix whose true value μ_0 solves the equation

$$J_{\zeta\beta} - \mu J_{\beta\beta} = 0$$

for

$$J = \begin{pmatrix} J_{\zeta\zeta} & J_{\zeta\beta} \\ J_{\beta\zeta} & J_{\beta\beta} \end{pmatrix} = \partial_{(\zeta', \beta')} \mathbb{E}_P \left[\partial_{(\zeta', \beta')'} \mathcal{L}(W; \zeta, \beta) \right] \big|_{\zeta=\zeta_0; \beta=\beta_0}$$

The true value of the nuisance parameter η is

$$\eta_0 = \left(\beta_0', \text{vec}\left(\mu_0\right)' \right)'$$

and when $J_{\beta\beta}$ is invertible, it has the unique solution,

$$\mu_0 = J_{\zeta\beta} J_{\beta\beta}^{-1}$$

If $J_{\beta\beta}$ is not invertible, the equation typically has multiple solutions. In this case, it is convenient to focus on a minimal norm solution,

$$\mu_0 = \arg\min \|\mu\| \text{ such that } \|J_{\zeta\beta} - \mu J_{\beta\beta}\|_q = 0$$

for a suitably chosen norm $\|\cdot\|_q$ on the space of $d_\zeta \times d_\beta$ matrices.

In our case, we consider the following criterion function, which is the negative of standard squared loss:

$$\mathcal{L}(\mathbf{W}; \zeta, \beta)_{1\times 1} = -\frac{\mathbf{B}_{1\times n}^\intercal \mathbf{B}_{n\times 1}}{2}; \quad \mathbf{B}_{n\times 1} = [\mathbf{Y} - \ell(\mathcal{X}, \mathcal{A}) - \theta(\mathbf{T} - \mathbf{m}(\mathcal{X}, \mathcal{A})) - \alpha(\mathcal{A}(\mathbf{T} - \mathbf{m}(\mathcal{X}, \mathcal{A})))]$$

where $\zeta = (\theta, \alpha)$ are the target parameters and $\beta = (m, \ell)$ are nuisance parameters. m and ℓ are estimates of $m_0(\mathcal{X}, \mathcal{A})$ and $\ell_0(\mathcal{X}, \mathcal{A})$ where $m_0(\mathcal{X}, \mathcal{A}) = \mathbb{E}_P[T|\mathcal{X}, \mathcal{A}]$ and $\ell_0(\mathcal{X}, \mathcal{A}) = \mathbb{E}_P[Y|\mathcal{X}, \mathcal{A}]$. Thus, we want to solve the following maximization problem and find θ_0 and α_0 such that

$$\theta_0, \alpha_0 = \arg \max_{\theta\in\Theta, \alpha\in\triangle} \mathbb{E}_P[\mathcal{L}(\mathbf{W}; \zeta, \beta)_{1\times 1}]$$

We take the derivatives to build the score function

$$\begin{aligned}
\partial_\theta \mathcal{L}(\mathbf{W}; \zeta, \beta)_{1\times 1} &= \mathbf{B}_{1\times n}^\intercal (\mathbf{T} - \mathbf{m}(\mathcal{X}, \mathcal{A}))_{n\times 1} \\
\partial_\alpha \mathcal{L}(\mathbf{W}; \zeta, \beta)_{1\times 1} &= \mathbf{B}_{1\times n}^\intercal \mathcal{A}(\mathbf{T} - \mathbf{m}(\mathcal{X}, \mathcal{A}))_{n\times 1} \\
\partial_m \mathcal{L}(\mathbf{W}; \zeta, \beta)_{1\times n} &= -\mathbf{B}_{1\times n}^\intercal (\theta I_n + \alpha\mathcal{A})_{n\times n} \\
\partial_\ell \mathcal{L}(\mathbf{W}; \zeta, \beta)_{1\times n} &= \mathbf{B}_{1\times n}^\intercal I_n = \mathbf{B}_{1\times n}^\intercal
\end{aligned}$$

$\mathbf{I_n}$ is identity matrix with dimension $n \times n$.

Let $\mathbf{B_0}_{n\times 1} = \mathbf{Y} - \ell_\mathbf{0}(\mathcal{X}, \mathcal{A}) - \theta_0(\mathbf{T} - \mathbf{m_0}(\mathcal{X}, \mathcal{A})) - \alpha_0(\mathcal{A}(\mathbf{T} - \mathbf{m_0}(\mathcal{X}, \mathcal{A})))$

$$\begin{aligned}
J_{\beta\beta} &= \partial_{\beta'} \mathbb{E}_P \left[\partial_\beta \mathcal{L}(\mathbf{W}; \zeta, \beta) \right] \big|_{\zeta=\zeta_0; \beta=\beta_0} \\
&= \begin{pmatrix} -[(\theta_0 I_n + \alpha_0 \mathcal{A})^\intercal (\theta_0 I_n + \alpha_0 \mathcal{A})]_{n\times n} & [(\theta_0 I_n + \alpha_0 \mathcal{A})^\intercal]_{n\times n} \\ [(\theta_0 I_n + \alpha_0 \mathcal{A})]_{n\times n} & -[I_n]_{n\times n} \end{pmatrix}_{2n\times 2n} \Rightarrow \text{not invertible}
\end{aligned}$$

Since $J_{\beta\beta}$ is not invertible, we need to find the minimal norm solution

$$\mu_0 = \arg\min \|\mu\| \text{ such that } \|J_{\zeta\beta} - \mu J_{\beta\beta}\|_q = 0$$

Here μ_0 and $J_{\zeta\beta}$ are $2 \times 2n$ matrices and $J_{\beta\beta}$ is a $2n \times 2n$ matrix.

$$\begin{aligned}
&J_{\zeta\beta} = \partial_{\zeta'} \mathbb{E}_P \left[\partial_\beta \mathcal{L}(\mathbf{W}; \zeta, \beta) \right] \big|_{\zeta=\zeta_0; \beta=\beta_0} = \\
&\begin{pmatrix} -\mathbb{E}_P[\mathbf{B_0}^\intercal + (\mathbf{m_0}(\mathcal{X}, \mathcal{A}) - \mathbf{T})^\intercal (\theta_0 I_n + \alpha_0 \mathcal{A})]_{1\times n} & \mathbb{E}_P[(\mathbf{m_0}(\mathcal{X}, \mathcal{A}) - \mathbf{T})^\intercal]_{1\times n} \\ -\mathbb{E}_P[B_0^\intercal \mathcal{A} + (\mathbf{m_0}(\mathcal{X}, \mathcal{A}) - \mathbf{T})^\intercal \mathcal{A}^\intercal (\theta_0 I_n + \alpha_0 \mathcal{A})]_{1\times n} & \mathbb{E}_P[(\mathbf{m_0}(\mathcal{X}, \mathcal{A}) - \mathbf{T})^\intercal \mathcal{A}^\intercal]_{1\times n} \end{pmatrix}_{2\times 2n}
\end{aligned}$$

Since $m_0(\mathcal{X}, \mathcal{A}) = \mathbb{E}_P[T|\mathcal{X}, \mathcal{A}]$ and $\ell_0(\mathcal{X}, \mathcal{A}) = \mathbb{E}_P[Y|\mathcal{X}, \mathcal{A}]$, $\mathbb{E}_P[m_0(\mathcal{X}, \mathcal{A}) - T] = 0$ and $\mathbb{E}_P[\ell_0(\mathcal{X}, \mathcal{A}) - Y] = 0$. The expectation of multiplication of a fixed matrix in each of these vectors would also be zero because it would be a linear combination of elements with zero expectation. Thus, $J_{\zeta\beta} = 0$ and by inspection, due to the fact that $\|.\| \geq 0$, $\mu = \mathbf{0}$ would make this norm minimum.

Hence, the score function would be:

$$\psi(\mathcal{W}, \mathcal{A}; \zeta, \eta) = \begin{pmatrix} (\mathbf{Y} - \ell(\mathcal{X}, \mathcal{A}) - \theta(\mathbf{T} - \mathbf{m}(\mathcal{X}, \mathcal{A})) - \alpha(\mathcal{A}(\mathbf{T} - \mathbf{m}(\mathcal{X}, \mathcal{A})))^\intercal (\mathbf{T} - \mathbf{m}(\mathcal{X}, \mathcal{A})) \\ (\mathbf{Y} - \ell(\mathcal{X}, \mathcal{A}) - \theta(\mathbf{T} - \mathbf{m}(\mathcal{X}, \mathcal{A})) - \alpha(\mathcal{A}(\mathbf{T} - \mathbf{m}(\mathcal{X}, \mathcal{A})))^\intercal \mathcal{A}(\mathbf{T} - \mathbf{m}(\mathcal{X}, \mathcal{A})) \end{pmatrix}$$

10 Proof of Theorem 4.1 and Corresponding Conditions to Verify

In this section, we describe the essential regularity conditions and provide their respective proofs. These conditions form the foundational basis for proving Theorem 4.1. By demonstrating that the score function fulfills specific assumptions, we

can effectively invoke Theorems 3.1 and 3.2, along with Corollary 3.1 from Chernozhukov et al. (2018).
This application is crucial for establishing two key properties of our estimators: consistency and asymptotic normality. These 2 are the asymptotic properties of an estimator. Asymptotic refers to a mathematical property of a sequence of random variables or a statistical estimator as the sample size approaches infinity. More specifically, it refers to the behavior of the estimator as the sample size becomes larger and larger. An asymptotic result holds in the limit as the sample size grows infinitely large.
We say that an estimate $\hat{\theta}$ is consistent if $\hat{\theta} \to \theta_0$ in probability as $n \to \infty$, where θ_0 is the 'true' unknown parameter of the distribution of the sample.
We say that $\hat{\theta}$ is asymptotically normal if

$$\sqrt{n}\left(\hat{\theta} - \theta_0\right) \xrightarrow{\mathrm{d}} \mathcal{N}\left(0, \sigma_{\theta_0}^2\right)$$

where $\sigma_{\theta_0}^2$ is called the asymptotic variance of the estimate $\hat{\theta}$. Asymptotic normality says that the estimator not only converges to the unknown parameter, but it converges fast enough, at a rate $1/\sqrt{n}$, where n is the sample size.

These properties are fundamental in reinforcing the statistical robustness and reliability of our estimators in both finite sample and asymptotic contexts. Besides, they allow us to perform uncertainty quantification and build confidence intervals.
The invocation of Theorems 3.1 and 3.2 along with Corollary 3.1 from Chernozhukov et al. (2018) are sufficient for proving our Theorem 4.1.

Remark on convergence rate: The root-n rate represents the theoretical bound for the fastest convergence rate in regular parametric models under standard assumptions. This is the rate derived in the original DML paper for the i.i.d. setting, and we have also proven it for our estimator, which is based on its own score function under specific assumptions. However, non-doubly robust estimators, such as IPW-based methods, rely solely on the convergence of a single nuisance parameter (e.g., the propensity model), making them more sensitive to model misspecification. The use of machine learning models like random forests or neural networks for propensity or outcome modeling provides greater flexibility but often results in slower-than-$\sqrt{n}$ convergence rates. In complex settings such as social networks, models like Graph Convolutional Networks (GCNs) or Graph Isomorphism Networks (GINs) can enhance accuracy but typically converge more slowly. The convergence rates of these machine learning models, particularly Graph Neural Networks (GNNs), remain an active area of research and are beyond the scope of this work. For further details, we refer the reader to Jegelka (2022); Farrell et al. (2021); Favaro et al. (2023); Kohler and Langer (2019).

A key advantage of doubly robust estimators is that their convergence rate is determined by the product of the convergence rates of the propensity and outcome models, as detailed in Smucler et al. (2019) (see page 6, definitions 2 and 3). In our work, we employ GNNs and Network Random Forests as over-parameterized and nonparametric models, respectively, to estimate nuisance parameters. While these models generally converge slower than $\sqrt{n}$ Jegelka (2022), Theorem 1 shows that even with a convergence rate of $n^{-1/4}$, the overall estimator achieves $\sqrt{n}$-efficiency. Despite $n^{-1/4}$ being significantly slower than $\sqrt{n}$, our method remains robust and efficient under these conditions. In line with the DML framework, our doubly robust estimator relies on the product of the convergence rates of the nuisance parameters, ensuring that $\sqrt{n}$-consistency is still achievable even when the nuisances converge at slower rates. As demonstrated in our work, we achieve $\sqrt{n}$-efficiency even when the nuisances converge at a rate of $n^{-1/4}$, which is common in ML models like GNNs due to their flexibility and overparameterization.

In the following discussion, we delve into two distinct sets of conditions as outlined in Chernozhukov et al. (2018) that are necessary to invoke these theorems.
We use $\|.\|_{P,q}$ to denote the $L^q(P)$ norm, i.e. $\|f\|_{P,q} := \|f(W)\|_{P,q} := (\int \mid f(w) \mid^q dP(w))^{\frac{1}{q}}$.

Assumption 10.1. (Regularity Conditions) Let $c > 0, C > 0, c_1 \geqslant c_0 > 0, q > 4$ and $K \geqslant 2$ be some finite constants; and let $\{\delta_n\}_{n=1}^{\infty}$ and $\{\Delta_n\}_{n=1}^{\infty}$ be some sequences of positive constants converging to zero such that $\delta_n \geqslant n_f^{-1/2}$. For all probability laws $P \in \mathcal{P}$ for the triple $W = (\mathbf{T}, \mathbf{Y}, \mathcal{X})$ the following conditions hold:

1. Equation set 1 holds
2. $c \leqslant \|\epsilon^{\mathbf{T}}\|_{P,2}$, $\|\epsilon^{\mathbf{T}}\|_{P,q} \leqslant C, \leqslant \|\epsilon^{\mathbf{Y}}\|_{P,q} \leqslant C$
3. $c \leqslant \|\epsilon^{\mathbf{Y}\intercal}\epsilon^{\mathbf{T}}\|_{P,2}, c \leqslant \mathbb{E}_P\left[\epsilon^{\mathbf{T}\intercal}\epsilon^{\mathbf{T}}\right], c \leqslant \mathbb{E}_P\left[\epsilon^{\mathbf{T}\intercal}\mathcal{A}^{\intercal}\epsilon^{\mathbf{T}}\right]$
4. $\|\mathbf{Y}\|_{P,q} \leqslant C$
5. $\epsilon^{\mathbf{T}}$ and $\epsilon^{\mathbf{Y}}$ are not eigen vectors of $\mathcal{A}$.

6. Given a random subset I of $[n_f]$ of size $n' = n_f/K$, the nuisance parameter estimator $\widehat{\eta}_0 = \widehat{\eta}_0\left((W_i)_{i\in I^c}\right)$ belongs to the realization set $\mathcal{T}_n$ with probability at least $1-\Delta_n$, where $\eta_0 \in \mathcal{T}_n$.

7. Given a random subset I of $[n_f]$ of size $n' = n_f/K$, the nuisance parameter estimator $\widehat{\eta}_0 = \widehat{\eta}_0\left((W_i)_{i\in I^c}\right)$ obeys the following conditions: With P-probability no less than $1-\Delta_n$,

$$\|\widehat{\eta}_0 - \eta_0\|_{P,q} \leqslant C, \quad \|\widehat{\eta}_0 - \eta_0\|_{P,2} \leqslant \delta_n, \quad \text{and}$$

for the score ψ, where $\widehat{\eta}_0 = \left(\widehat{m}_0, \widehat{\ell}_0\right)$,

$$\|\widehat{m}_0 - m_0\|_{P,2} \times \left(\|\widehat{m}_0 - m_0\|_{P,2} + \left\|\widehat{\ell}_0 - \ell_0\right\|_{P,2}\right) \leqslant \delta_n n_f^{-1/2}.$$

Remark on how we diverge from Chernozhukov et al. (2018) in regularity conditions: Our work extends the DML framework to non-i.i.d. data by relaxing the i.i.d. assumption made in the original DML paper. However, to accommodate this relaxation, we introduce additional assumptions necessary for our approach:

- **Known exposure map:** We define our causal estimand using a predefined "exposure map" that aggregates neighborhood treatment information. Learning a data-driven estimand that learns the exposure map from the data is beyond the scope of this project as it introduces many auxiliary challenges.
- **Confounding map:** In the original DML paper, the nuisance parameters m, g are functions of units' own covariates. In our setup, these nuisance parameters are a function of the unit's covariates as well as neighbors' covariates. Our work leverages recent advancements in Graph ML to estimate these nuisance parameters.
- **Partial interference:** This assumption limits the interdependencies of units, so that each unit's outcome is affected by its at most k-hop away neighbors. The regularity condition 9.1 part 5 is a consequence of this assumption. It states that the noise variables are not fully correlated. If we are given the noise variable for a unit, and we are able to identify the noise variable for other units within the network, then the exogenous variables would have been an eigenvector for the adjacency matrix.
- **Positivity of treatment assignment and neighborhood exposure:** Regularity condition $\mathbb{E}_P[\epsilon^{\mathbf{T}\intercal}\epsilon^{\mathbf{T}}] > c$ such that $c > 0$ guarantees strict positivity of individual-level treatment assignment. Apart from this regularity conditin which is also in Chernozhukov et al. (2018), we also need an additional condition $\mathbb{E}_P[\epsilon^{\mathbf{T}\intercal}\mathcal{A}\epsilon^{\mathbf{T}}] > c$, which implies the strict positivity of the neighborhood exposure, as in regularity condition 9.1 part 3.

10.1 Condition Set 1: Linear Scores with Approximate Neyman Orthogonality

For all $n_f \geqslant 3$ and probability measures $P \in \mathcal{P}_n$ that determines the underlying law of W:

1. Moment condition vanishes at the true parameter ζ_0: $\mathbb{E}_P\left[\psi\left(W;\zeta_0,\eta_0\right)\right] = 0$
2. The score function is linear in the sense that: $\psi(W,\mathcal{A};\zeta,\eta) = \psi^a(W,\mathcal{A};\zeta,\eta)\theta + \psi^b(W;\zeta,\eta)\alpha + \psi^c(W;\zeta,\eta)$
3. The map $\eta \to \mathbb{E}_P\left[\psi\left(W;\zeta_0,\eta_0\right)\right]$ is twice continuously Gateaux-differentiable.
4. The score ψ is Neyman orthogonal or, more generally, it is Neyman λ_n near-orthogonal at (ζ_0,η_0) with respect to the nuisance realization set $\Gamma_n \subset T$ for

$$\lambda_n := \sup_{\eta\in\Gamma_n} \left\|\partial_\eta \mathbb{E}_P\psi\left(W;\zeta_0,\eta_0\right)\left[\eta-\eta_0\right]\right\| \leqslant \delta_n n_f^{-1/2}$$

5. The identification condition holds; namely, the singular values of the matrix

$$J_{0,a} := \mathbb{E}_P\left[\psi^a\left(W;\eta_0\right)\right]$$

are between c_0 and c_1.

10.2 Condition Set 2: Score Regularity and Quality of nuisance Parameter Estimators

For all $n_f \geqslant 3$ and $P \in \mathcal{P}_n$, the following conditions hold:

1. Given a random subset I of $[n_f]$ of size $n' = n_f/K$, the nuisance parameter estimator $\widehat{\eta}_0 = \widehat{\eta}_0\left((W_i)_{i\in I^c}\right)$ belongs to the realization set Γ_n with probability at least $1-\Delta_n$, where Γ_n contains η_0 and is constrained by the next conditions.

2. The moment conditions hold:
$$m_n := \sup_{\eta\in\Gamma_n}\left(\mathbb{E}_P\left[\|\psi\left(W;\zeta_0,\eta\right)\|^q\right]\right)^{1/q} \leqslant c_1,$$
$$m'_n := \sup_{\eta\in\Gamma_n}\left(\mathbb{E}_P\left[\|\psi^a(W;\eta)\|^q\right]\right)^{1/q} \leqslant c_1.$$

3. The following conditions on the statistical rates r_n, r'_n, and λ'_n hold:
$$r_n := \sup_{\eta\in\Gamma_n}\left\|\mathbb{E}_P\left[\psi^a(W;\eta)\right] - \mathbb{E}_P\left[\psi^a\left(W;\eta_0\right)\right]\right\| \leqslant \delta_n,$$
$$r'_n := \sup_{\eta\in\Gamma_n}\left(\mathbb{E}_P\left[\left\|\psi\left(W;\zeta_0,\eta\right) - \psi\left(W;\zeta_0,\eta_0\right)\right\|^2\right]\right)^{1/2} \leqslant \delta_n,$$
$$\lambda'_n := \sup_{r\in(0,1),\eta\in\Gamma_n}\left\|\partial_r^2\mathbb{E}_P\left[\psi\left(W;\zeta_0,\eta_0 + r\left(\eta-\eta_0\right)\right)\right]\right\| \leqslant \delta_n/\sqrt{n_f}.$$

4. The variance of the score ψ is non-degenerate: All eigenvalues of the matrix
$$\mathbb{E}_P\left[\psi\left(W;\zeta_0,\eta_0\right)\psi\left(W;\zeta_0,\eta_0\right)'\right]$$
are bounded from below by c_0.

In the rest of this section, we attempt to prove the condition sets 10.1 and 10.2 under regularity assumptions 10.1.

10.3 Proof of Condition Set 1

C.1.1 The true parameter values ζ_0 and β_0 solve the following optimization problem

$$\max_{\zeta\in\mathcal{Z},\beta\in\mathcal{B}}\mathbb{E}_P[\mathcal{L}(W;\zeta,\beta)]$$

where $\mathcal{L}(W;\zeta,\beta)$ is a known criterion function. ζ_0 and β_0 satisfy

$$\mathbb{E}_P\left[\partial_\zeta\mathcal{L}\left(W;\zeta_0,\beta_0\right)\right] = 0, \quad \mathbb{E}_P\left[\partial_\beta\mathcal{L}\left(W;\zeta_0,\beta_0\right)\right] = 0$$

The neyman orthogonal score function is

$$\psi(W,\mathcal{A};\zeta,\eta) = \partial_\zeta\mathcal{L}(W;\zeta,\beta) - \mu\partial_\beta\mathcal{L}(W;\zeta,\beta)$$

Thus, by definition of ζ_0 and η_0, we have:
$$\mathbb{E}_P\left[\psi\left(W;\zeta_0,\eta_0\right)\right] = 0$$

C.1.2 The score function is linear in the sense that:

$$\psi(W,\mathcal{A};\zeta,\eta) = \begin{pmatrix} (\mathbf{Y}-\ell(\mathcal{X},\mathcal{A}) - \theta(\mathbf{T}-\mathbf{m}(\mathcal{X},\mathcal{A})) - \alpha(\mathcal{A}(\mathbf{T}-\mathbf{m}(\mathcal{X},\mathcal{A})))^\intercal(\mathbf{T}-\mathbf{m}(\mathcal{X},\mathcal{A})) \\ (\mathbf{Y}-\ell(\mathcal{X},\mathcal{A}) - \theta(\mathbf{T}-\mathbf{m}(\mathcal{X},\mathcal{A})) - \alpha(\mathcal{A}(\mathbf{T}-\mathbf{m}(\mathcal{X},\mathcal{A})))^\intercal\mathcal{A}(\mathbf{T}-\mathbf{m}(\mathcal{X},\mathcal{A})) \end{pmatrix} =$$
$$\underbrace{\begin{pmatrix} -(\mathbf{T}-\mathbf{m}(\mathcal{X},\mathcal{A}))^\intercal(\mathbf{T}-\mathbf{m}(\mathcal{X},\mathcal{A})) & -(\mathbf{T}-\mathbf{m}(\mathcal{X},\mathcal{A}))^\intercal\mathcal{A}^\intercal(\mathbf{T}-\mathbf{m}(\mathcal{X},\mathcal{A})) \\ -(\mathbf{T}-\mathbf{m}(\mathcal{X},\mathcal{A}))^\intercal\mathcal{A}(\mathbf{T}-\mathbf{m}(\mathcal{X},\mathcal{A})) & -(\mathbf{T}-\mathbf{m}(\mathcal{X},\mathcal{A}))^\intercal\mathcal{A}^\intercal\mathcal{A}(\mathbf{T}-\mathbf{m}(\mathcal{X},\mathcal{A})) \end{pmatrix}}_{\psi^a}\begin{pmatrix}\theta\\ \alpha\end{pmatrix} + \underbrace{\begin{pmatrix} (\mathbf{Y}-\ell(\mathcal{X},\mathcal{A}))^\intercal(\mathbf{T}-\mathbf{m}(\mathcal{X},\mathcal{A})) \\ (\mathbf{Y}-\ell(\mathcal{X},\mathcal{A}))^\intercal\mathcal{A}(\mathbf{T}-\mathbf{m}(\mathcal{X},\mathcal{A})) \end{pmatrix}}_{\psi^b}$$

C.1.3 The score function can trivially be shown to be twice Gateaux differentiable.

C.1.4 To show neyman orthogonality, we need to show that Gateaux derivative vanishes in addition to the moment condition. The Gateaux derivative in the direction $\eta - \eta_0 = (\mathbf{m} - \mathbf{m_0}, \ell - \ell_\mathbf{0})$ is:

$$\partial_\eta \mathbb{E}_P \psi\left(W; \zeta_0, \eta_0\right)[\eta - \eta_0] =$$

$$\lim_{r\to 0} \frac{\mathbb{E}_P\begin{pmatrix} (\mathbf{Y} - (\ell_\mathbf{0} + r(\ell - \ell_\mathbf{0})) - \theta_0(\mathbf{T} - (\mathbf{m_0} + r(\mathbf{m} - \mathbf{m_0}))) - \alpha_0(\mathcal{A}(\mathbf{T} - (\mathbf{m_0} + r(\mathbf{m} - \mathbf{m_0}))))^\intercal(\mathbf{T} - (\mathbf{m_0} + r(\mathbf{m} - \mathbf{m_0}))) \\ (\mathbf{Y} - (\ell_\mathbf{0} + r(\ell - \ell_\mathbf{0})) - \theta_0(\mathbf{T} - (\mathbf{m_0} + r(\mathbf{m} - \mathbf{m_0}))) - \alpha_0(\mathcal{A}(\mathbf{T} - (\mathbf{m_0} + r(\mathbf{m} - \mathbf{m_0}))))^\intercal\mathcal{A}(\mathbf{T} - (\mathbf{m_0} + r(\mathbf{m} - \mathbf{m_0}))) \end{pmatrix} - \mathbb{E}_P\begin{pmatrix} (\mathbf{Y} - \ell_\mathbf{0} - \theta_0(\mathbf{T} - \mathbf{m_0}) - \alpha_0(\mathcal{A}(\mathbf{T} - \mathbf{m_0}))^\intercal(\mathbf{T} - \mathbf{m_0}) \\ (\mathbf{Y} - \ell_\mathbf{0} - \theta_0(\mathbf{T} - \mathbf{m_0}) - \alpha_0(\mathcal{A}(\mathbf{T} - \mathbf{m_0}))^\intercal\mathcal{A}(\mathbf{T} - \mathbf{m_0}) \end{pmatrix}}{r} =$$

$$\lim_{r\to 0} \frac{\mathbb{E}_P\begin{pmatrix} \overbrace{(\mathbf{Y} - \ell_\mathbf{0} - \theta_0(\mathbf{T} - \mathbf{m_0}) - \alpha_0(\mathcal{A}(\mathbf{T} - \mathbf{m_0} - r(\mathbf{m} - \mathbf{m_0}))) - \overbrace{r(\ell - \ell_\mathbf{0})}^{\mathbf{G}} + \overbrace{\theta_0 r(\mathbf{m} - \mathbf{m_0})}^{\theta_0\mathbf{D}} + \overbrace{\alpha_0\mathcal{A}r(\mathbf{m} - \mathbf{m_0})}^{\alpha_0\mathcal{A}\mathbf{D}})^\intercal(\overbrace{\mathbf{T} - \mathbf{m_0}}^{\epsilon^\mathbf{T}} - \overbrace{r(\mathbf{m} - \mathbf{m_0})}^{\mathbf{D}})}^{\epsilon^\mathbf{Y}} \\ (\mathbf{Y} - \ell_\mathbf{0} - \theta_0(\mathbf{T} - \mathbf{m_0}) - \alpha_0(\mathcal{A}(\mathbf{T} - \mathbf{m_0} - r(\mathbf{m} - \mathbf{m_0}))) - r(\ell - \ell_\mathbf{0}) + \theta_0 r(\mathbf{m} - \mathbf{m_0}) + \alpha_0\mathcal{A}r(\mathbf{m} - \mathbf{m_0}))^\intercal\mathcal{A}(\mathbf{T} - \mathbf{m_0} - r(\mathbf{m} - \mathbf{m_0})) \end{pmatrix} - \mathbb{E}_P\begin{pmatrix} \overbrace{(\mathbf{Y} - \ell_\mathbf{0} - \theta_0(\mathbf{T} - \mathbf{m_0}) - \alpha_0(\mathcal{A}(\mathbf{T} - \mathbf{m_0}))^\intercal}^{\epsilon^{\mathbf{Y}\intercal}}\overbrace{(\mathbf{T} - \mathbf{m_0})}^{\epsilon^T} \\ (\mathbf{Y} - \ell_\mathbf{0} - \theta_0(\mathbf{T} - \mathbf{m_0}) - \alpha_0(\mathcal{A}(\mathbf{T} - \mathbf{m_0}))^\intercal\mathcal{A}(\mathbf{T} - \mathbf{m_0}) \end{pmatrix}}{r}$$

$$= \mathbb{E}_P\begin{pmatrix} \cancel{\epsilon^{\mathbf{Y}\intercal}\epsilon^\mathbf{T}} - \epsilon^{\mathbf{Y}\intercal}\mathbf{D} - \mathbf{G}^\intercal\epsilon^\mathbf{T} + \theta_0\mathbf{D}^\intercal\epsilon^\mathbf{T} \overbrace{+\mathbf{G}\mathbf{D} - \theta_0\mathbf{D}^\intercal\mathbf{D} - \alpha_0\mathbf{D}^\intercal\mathcal{A}^\intercal\mathbf{D}}^{\text{goes to 0 includes } r^2} + \alpha_0\mathbf{D}^\intercal\mathcal{A}^\intercal\epsilon^\mathbf{T} - \cancel{\epsilon^{\mathbf{Y}\intercal}\epsilon^\mathbf{T}} \\ \cancel{\epsilon^{\mathbf{Y}\intercal}\mathcal{A}\epsilon^\mathbf{T}} - \epsilon^{\mathbf{Y}\intercal}\mathcal{A}\mathbf{D} - \mathbf{G}^\intercal\mathcal{A}\epsilon^\mathbf{T} + \theta_0\mathbf{D}^\intercal\mathcal{A}\epsilon^\mathbf{T} \overbrace{+\mathbf{G}\mathcal{A}\mathbf{D} - \theta_0\mathbf{D}^\intercal\mathcal{A}\mathbf{D} - \alpha_0\mathbf{D}^\intercal\mathcal{A}^\intercal\mathcal{A}\mathbf{D}}^{\text{goes to 0 includes } r^2} + \alpha_0\mathbf{D}^\intercal\mathcal{A}^\intercal\mathcal{A}\epsilon^\mathbf{T} - \cancel{\epsilon^{\mathbf{Y}\intercal}\mathcal{A}\epsilon^\mathbf{T}} \end{pmatrix}}{r}$$

$$= \lim_{r\to 0} \frac{\cancel{r}\mathbb{E}_P\begin{pmatrix} -\epsilon^{\mathbf{Y}\intercal}(\mathbf{m} - \mathbf{m_0}) - (\ell - \ell_\mathbf{0})^\intercal\epsilon^\mathbf{T} + \theta_0(\mathbf{m} - \mathbf{m_0})^\intercal\epsilon^\mathbf{T} + \alpha_0(\mathbf{m} - \mathbf{m_0})^\intercal\mathcal{A}^\intercal\epsilon^\mathbf{T} \\ -\epsilon^{\mathbf{Y}\intercal}\mathcal{A}(\mathbf{m} - \mathbf{m_0}) - (\ell - \ell_\mathbf{0})^\intercal\mathcal{A}\epsilon^\mathbf{T} + \theta_0(\mathbf{m} - \mathbf{m_0})^\intercal\mathcal{A}\epsilon^\mathbf{T} + \alpha_0(\mathbf{m} - \mathbf{m_0})^\intercal\mathcal{A}^\intercal\mathcal{A}\epsilon^\mathbf{T} \end{pmatrix}}{\cancel{r}}$$

$$= \mathbb{E}_P\begin{pmatrix} -\epsilon^{\mathbf{Y}\intercal}(\mathbf{m} - \mathbf{m_0}) - (\ell - \ell_\mathbf{0})^\intercal\epsilon^\mathbf{T} + \theta_0(\mathbf{m} - \mathbf{m_0})^\intercal\epsilon^\mathbf{T} + \alpha_0(\mathbf{m} - \mathbf{m_0})^\intercal\mathcal{A}^\intercal\epsilon^\mathbf{T} \\ -\epsilon^{\mathbf{Y}\intercal}\mathcal{A}(\mathbf{m} - \mathbf{m_0}) - (\ell - \ell_\mathbf{0})^\intercal\mathcal{A}\epsilon^\mathbf{T} + \theta_0(\mathbf{m} - \mathbf{m_0})^\intercal\mathcal{A}\epsilon^\mathbf{T} + \alpha_0(\mathbf{m} - \mathbf{m_0})^\intercal\mathcal{A}^\intercal\mathcal{A}\epsilon^\mathbf{T} \end{pmatrix}$$

Consider the first term in the above expectation. We use Law of Iterated Expectations:

$$\mathbb{E}_P[\epsilon^{\mathbf{Y}\intercal}(\mathbf{m} - \mathbf{m_0})] = \mathbb{E}_{X,D,Y}[\epsilon^{\mathbf{Y}\intercal}(\mathbf{m} - \mathbf{m_0})] = \mathbb{E}_X[\mathbb{E}_{YT|X}[\epsilon^{\mathbf{Y}\intercal}\overbrace{(\mathbf{m} - \mathbf{m_0})}^{\text{constant given X}} \mid X]] = \mathbb{E}_X[(\mathbf{m} - \mathbf{m_0})\mathbb{E}_{YT|X}[\overbrace{(\epsilon^{\mathbf{Y}\intercal})}^{0} \mid X]] = 0$$

A similar argument can be used to show that other expectation terms are 0.

C.1.5

$$\begin{aligned} J_{0,a} &:= \mathbb{E}_P\left[\psi^a\left(W; \eta_0\right)\right] \\ &= \mathbb{E}_P\left[\begin{pmatrix} -(\mathbf{T} - \mathbf{m_0}(\mathcal{X}, \mathcal{A}))^\intercal(\mathbf{T} - \mathbf{m_0}(\mathcal{X}, \mathcal{A})) & -(\mathbf{T} - \mathbf{m_0}(\mathcal{X}, \mathcal{A}))^\intercal\mathcal{A}^\intercal(\mathbf{T} - \mathbf{m_0}(\mathcal{X}, \mathcal{A})) \\ -(\mathbf{T} - \mathbf{m_0}(\mathcal{X}, \mathcal{A}))^\intercal\mathcal{A}(\mathbf{T} - \mathbf{m_0}(\mathcal{X}, \mathcal{A})) & -(\mathbf{T} - \mathbf{m_0}(\mathcal{X}, \mathcal{A}))^\intercal\mathcal{A}^\intercal\mathcal{A}(\mathbf{T} - \mathbf{m_0}(\mathcal{X}, \mathcal{A})) \end{pmatrix}\right] \\ &= \begin{pmatrix} \mathbf{A_1} & \mathbf{A_2} \\ \mathbf{A_3} & \mathbf{A_4} \end{pmatrix} \end{aligned}$$

$$J_{0,a}^\intercal J_{0,a} = \begin{pmatrix} \mathbf{A_1} & \mathbf{A_3} \\ \mathbf{A_2} & \mathbf{A_4} \end{pmatrix}\begin{pmatrix} \mathbf{A_1} & \mathbf{A_2} \\ \mathbf{A_3} & \mathbf{A_4} \end{pmatrix} = \begin{pmatrix} \mathbf{A_1}^2 + \mathbf{A_3}^2 & \mathbf{A_1}\mathbf{A_2} + \mathbf{A_3}\mathbf{A_4} \\ \mathbf{A_1}\mathbf{A_2} + \mathbf{A_3}\mathbf{A_4} & \mathbf{A_2}^2 + \mathbf{A_4}^2 \end{pmatrix}$$

The eigen values of this matrix are the roots of the following quadratic equation:

$$\lambda^2 - \lambda(\mathbf{A_1}^2 + \mathbf{A_2}^2 + \mathbf{A_3}^2 + \mathbf{A_4}^2) + (\mathbf{A_1}^2 + \mathbf{A_3}^2)(\mathbf{A_2}^2 + \mathbf{A_4}^2) - (\mathbf{A_1}\mathbf{A_2} + \mathbf{A_3}\mathbf{A_4})^2 = 0$$

We know that in a quadratic equation of form $a_2x^2 + a_1x + a_0 = 0$, the sum of the roots are $\frac{-a_1}{a_2}$ and the product of the roots are $\frac{a_0}{a_2}$. To ensure that all the eigen values are positive, we need to make sure both $\frac{-a_1}{a_2}$ and $\frac{a_0}{a_2}$ are positive:

$$\frac{-a_1}{a_2} = \mathbf{A_1}^2 + \mathbf{A_2}^2 + \mathbf{A_3}^2 + \mathbf{A_4}^2 \geqslant c > 0 \tag{32}$$

$$\frac{a_0}{a_2} = (\mathbf{A_1}^2 + \mathbf{A_3}^2)(\mathbf{A_2}^2 + \mathbf{A_4}^2) - (\mathbf{A_1}\mathbf{A_2} + \mathbf{A_3}\mathbf{A_4})^2 = (A_1A_4 - A_2A_3)^2 \tag{33}$$

32 holds since the summation of squared elements are non-negative and $A_1^2 = \mathbb{E}_P[(\mathbf{T}-\mathbf{m_0}(\mathcal{X},\mathcal{A}))^\intercal(\mathbf{T}-\mathbf{m_0}(\mathcal{X},\mathcal{A}))]^2 = \|\epsilon^{\mathbf{T}}\| \geqslant c$ by assumption 2. For 33 to hold, since the squared value is non-negative, we need to show it is not zero, i.e. $A_1A_4 \neq A_2A_3$. By Cauchy-Schwarz inequality, we know that:

$$\mathbb{E}_P\left[\epsilon^{\mathbf{T}\intercal}\epsilon^{\mathbf{T}}\right]\mathbb{E}_P\left[\epsilon^{\mathbf{T}\intercal}\mathcal{A}^\intercal\mathcal{A}\epsilon^{\mathbf{T}}\right] \geqslant \mathbb{E}_P\left[\epsilon^{\mathbf{T}\intercal}\mathcal{A}\epsilon^{\mathbf{T}}\right]^2$$

where the equality holds if $\|\epsilon^{\mathbf{T}}\| = 0$ or $\|\mathcal{A}\epsilon^{\mathbf{T}}\| = 0$, which does not hold by assumption 2 and the fact that $\mathcal{A}$ is a non-zero matrix. Also the equality can happen if $\exists r : \mathcal{A}\epsilon^{\mathbf{T}} = r\epsilon^{\mathbf{T}}$, which does not hold by assumption 5. Thus, summation and product of the eigen values are positive, leading to positivity of the singular values of $J_{0,a}$.

Following proposition is derived from Gallier and Quaintance (2023):

Proposition 2. *For every norm $\|\|$ on $\mathbb{C}^n$ (or $\mathbb{R}^n$), for every matrix $A \in \mathrm{M}_n(\mathbb{C})$ (or $A \in \mathrm{M}_n(\mathbb{R})$), there is a real constant $C_A \geqslant 0$, such that*

$$\|Au\| \leq C_A\|u\|,$$

for every vector $u \in \mathbb{C}^n$ (or $u \in \mathbb{R}^n$ if A is real).

2 states that every linear map on a finite-dimensional space is bounded.

10.4 Proof of Condition Set 2

C.2.1 Condition C.2.1 holds by the construction of the set Γ_n and Assumption 6.

C.2.2 We prove the boundedness of norms of these matrices by showing the bound on the norm of the elements considering the fact that if norm of each element is bounded, then the norm of the matrix is bounded. We first show the bound for the first elements of ψ_a and ψ.:

$$\psi^a(W;\eta) = \begin{pmatrix} -(\mathbf{T}-\mathbf{m}(\mathcal{X},\mathcal{A}))^\intercal(\mathbf{T}-\mathbf{m}(\mathcal{X},\mathcal{A})) & -(\mathbf{T}-\mathbf{m}(\mathcal{X},\mathcal{A}))^\intercal\mathcal{A}^\intercal(\mathbf{T}-\mathbf{m}(\mathcal{X},\mathcal{A})) \\ -(\mathbf{T}-\mathbf{m}(\mathcal{X},\mathcal{A}))^\intercal\mathcal{A}(\mathbf{T}-\mathbf{m}(\mathcal{X},\mathcal{A})) & -(\mathbf{T}-\mathbf{m}(\mathcal{X},\mathcal{A}))^\intercal\mathcal{A}^\intercal\mathcal{A}(\mathbf{T}-\mathbf{m}(\mathcal{X},\mathcal{A})) \end{pmatrix}$$

$$\psi(W;\zeta_0,\eta) = \begin{pmatrix} (\mathbf{Y}-\ell(\mathcal{X},\mathcal{A})-\theta_0(\mathbf{T}-\mathbf{m}(\mathcal{X},\mathcal{A}))-\alpha_0(\mathcal{A}(\mathbf{T}-\mathbf{m}(\mathcal{X},\mathcal{A})))^\intercal(\mathbf{T}-\mathbf{m}(\mathcal{X},\mathcal{A})) \\ (\mathbf{Y}-\ell(\mathcal{X},\mathcal{A})-\theta_0(\mathbf{T}-\mathbf{m}(\mathcal{X},\mathcal{A}))-\alpha_0(\mathcal{A}(\mathbf{T}-\mathbf{m}(\mathcal{X},\mathcal{A})))^\intercal\mathcal{A}(\mathbf{T}-\mathbf{m}(\mathcal{X},\mathcal{A})) \end{pmatrix}$$

$$\left(\mathbb{E}_P\left[\|\psi^a_{11}(W;\eta)\|^{q/2}\right]\right)^{2/q} = \|\psi^a_{11}(W;\eta)\|_{P,q/2} = \|-(\mathbf{T}-\mathbf{m}(\mathcal{X},\mathcal{A}))^\intercal(\mathbf{T}-\mathbf{m}(\mathcal{X},\mathcal{A})\|_{P,q/2} =$$
$$\|-((\mathbf{T}-\mathbf{m_0}(\mathcal{X},\mathcal{A}))-(\mathbf{m}(\mathcal{X},\mathcal{A})-\mathbf{m_0}(\mathcal{X},\mathcal{A})))^\intercal((\mathbf{T}-\mathbf{m_0}(\mathcal{X},\mathcal{A}))-(\mathbf{m}(\mathcal{X},\mathcal{A})-\mathbf{m_0}(\mathcal{X},\mathcal{A})))\|_{P,q/2} =$$
$$\left\|((\mathbf{m}(\mathcal{X},\mathcal{A})-\mathbf{m_0}(\mathcal{X},\mathcal{A}))-\epsilon^{\mathbf{T}})^\intercal(\epsilon^{\mathbf{T}}-(\mathbf{m}(\mathcal{X},\mathcal{A})-\mathbf{m_0}(\mathcal{X},\mathcal{A})))\right\|_{P,q/2} =$$
$$\Big\|(\mathbf{m}(\mathcal{X},\mathcal{A})-\mathbf{m_0}(\mathcal{X},\mathcal{A}))^\intercal\epsilon^{\mathbf{T}} - (\mathbf{m}(\mathcal{X},\mathcal{A})-\mathbf{m_0}(\mathcal{X},\mathcal{A}))^\intercal(\mathbf{m}(\mathcal{X},\mathcal{A})-\mathbf{m_0}(\mathcal{X},\mathcal{A})) - \epsilon^{\mathbf{T}\intercal}\epsilon^{\mathbf{T}} +$$
$$\epsilon^{\mathbf{T}\intercal}(\mathbf{m}(\mathcal{X},\mathcal{A})-\mathbf{m_0}(\mathcal{X},\mathcal{A}))\Big\|_{P,q/2} \leqslant \|\mathbf{m}(\mathcal{X},\mathcal{A})-\mathbf{m_0}(\mathcal{X},\mathcal{A})\|_{P,q}\left\|\epsilon^{\mathbf{T}}\right\|_{P,q} +$$
$$\|\mathbf{m}(\mathcal{X},\mathcal{A})-\mathbf{m_0}(\mathcal{X},\mathcal{A})\|_{P,q}\|\mathbf{m}(\mathcal{X},\mathcal{A})-\mathbf{m_0}(\mathcal{X},\mathcal{A})\|_{P,q} + \left\|\epsilon^{\mathbf{T}}\right\|_{P,q}\left\|\epsilon^{\mathbf{T}}\right\|_{P,q} +$$
$$\left\|\epsilon^{\mathbf{T}}\right\|_{P,q}\|\mathbf{m}(\mathcal{X},\mathcal{A})-\mathbf{m_0}(\mathcal{X},\mathcal{A})\|_{P,q} \leqslant 4C^2$$

by assumptions 2 and 7. Following the exact same approach along with proposition 2, we can derive an upperbound for other elements of $\|\psi^a(W;\eta)\|_{P,q/2}$, which gives the bound on m'_n in condition 2.

Next, we establish an upper-bound for the first element of $\left(\mathbb{E}_P\left[\|\psi(W;\zeta_0,\eta)\|^{q/2}\right]\right)^{2/q}$. First, we need an upper-bound on θ_0 and α_0, which will be used later.

$$\mathbb{E}_P\left[\psi(W;\zeta_0,\eta_0)\right] = \mathbb{E}_P\left[\psi^a(W;\eta_0)\right]\begin{pmatrix}\theta\\ \alpha\end{pmatrix} + \mathbb{E}_P\left[\psi^b(W;\eta_0)\right] = 0$$

$$\theta_0 = \frac{\mathbb{E}_P\left[(\mathbf{Y}-\ell_{\mathbf{0}}(\mathcal{X},\mathcal{A}))^\intercal(\mathbf{m_0}(\mathcal{X},\mathcal{A})-\mathbf{T})\right]}{\mathbb{E}_P\left[(\mathbf{T}-\mathbf{m_0}(\mathcal{X},\mathcal{A}))^\intercal(\mathbf{m_0}(\mathcal{X},\mathcal{A})-\mathbf{T})\right]} = \frac{\mathbb{E}_P\left[(\mathbf{Y}-\ell_{\mathbf{0}}(\mathcal{X},\mathcal{A}))^\intercal\epsilon^{\mathbf{T}}\right]}{\mathbb{E}_P\left[\epsilon^{\mathbf{T}\intercal}\epsilon^{\mathbf{T}}\right]}$$

$$\alpha_0 = \frac{\mathbb{E}_P\left[(\mathbf{Y}-\ell_{\mathbf{0}}(\mathcal{X},\mathcal{A}))^\intercal(\mathbf{m_0}(\mathcal{X},\mathcal{A})-\mathbf{T})\right]}{\mathbb{E}_P\left[(\mathbf{T}-\mathbf{m_0}(\mathcal{X},\mathcal{A}))^\intercal\mathcal{A}^\intercal(\mathbf{m_0}(\mathcal{X},\mathcal{A})-\mathbf{T})\right]} = \frac{\mathbb{E}_P\left[(\mathbf{Y}-\ell_{\mathbf{0}}(\mathcal{X},\mathcal{A}))^\intercal\epsilon^{\mathbf{T}}\right]}{\mathbb{E}_P\left[\epsilon^{\mathbf{T}\intercal}\mathcal{A}^\intercal\epsilon^{\mathbf{T}}\right]}$$

$$\mid \theta_0 \mid = \frac{\mid \mathbb{E}_P[(\mathbf{Y} - \ell_{\mathbf{0}}(\mathcal{X}, \mathcal{A}))^\intercal \epsilon^{\mathbf{T}}] \mid}{\mid \mathbb{E}_P \left[\epsilon^{\mathbf{T}^\intercal} \epsilon^{\mathbf{T}}\right] \mid} \leqslant c^{-1} C(\|Y\|_{P,q} + \|\ell_0(\mathcal{X}, \mathcal{A})\|_{P,q}) \leqslant 2c^{-1} C(\|Y\|_{P,q}) \leqslant 2C^2/c$$

$$\mid \alpha_0 \mid = \frac{\mid \mathbb{E}_P[(\mathbf{Y} - \ell_{\mathbf{0}}(\mathcal{X}, \mathcal{A}))^\intercal \epsilon^{\mathbf{T}}] \mid}{\mid \mathbb{E}_P \left[\epsilon^{\mathbf{T}^\intercal} \mathcal{A}^\intercal \epsilon^{\mathbf{T}}\right] \mid} \leqslant c^{-1} C(\|Y\|_{P,q} + \|\ell_0(\mathcal{X}, \mathcal{A})\|_{P,q}) \leqslant 2c^{-1} C(\|Y\|_{P,q}) \leqslant 2C^2/c$$

$$\begin{aligned}
&\left(\mathbb{E}_P \left[\|\psi_{11}(W; \zeta_0, \eta)\|^{q/2}\right]\right)^{2/q} = \|\psi_{11}(W; \zeta_0, \eta)\|_{P,q/2} = \\
&\|(\mathbf{Y} - \ell(\mathcal{X}, \mathcal{A}) - \theta_0(\mathbf{T} - \mathbf{m}(\mathcal{X}, \mathcal{A})) - \alpha_0(\mathcal{A}(\mathbf{T} - \mathbf{m}(\mathcal{X}, \mathcal{A})))^\intercal (\mathbf{T} - \mathbf{m}(\mathcal{X}, \mathcal{A}))\|_{P,q/2} = \\
&\big\|(\mathbf{Y} - \ell_{\mathbf{0}}(\mathcal{X}, \mathcal{A}) - (\ell(\mathcal{X}, \mathcal{A}) - \ell_{\mathbf{0}}(\mathcal{X}, \mathcal{A})) - \theta_0((\mathbf{T} - \mathbf{m}_{\mathbf{0}}(\mathcal{X}, \mathcal{A})) - (\mathbf{m}(\mathcal{X}, \mathcal{A}) - \mathbf{m}_{\mathbf{0}}(\mathcal{X}, \mathcal{A}))) - \\
&\alpha_0(\mathcal{A}((\mathbf{T} - \mathbf{m}_{\mathbf{0}}(\mathcal{X}, \mathcal{A})) - (\mathbf{m}(\mathcal{X}, \mathcal{A}) - \mathbf{m}_{\mathbf{0}}(\mathcal{X}, \mathcal{A}))))^\intercal ((\mathbf{T} - \mathbf{m}_{\mathbf{0}}(\mathcal{X}, \mathcal{A})) - (\mathbf{m}(\mathcal{X}, \mathcal{A}) - \mathbf{m}_{\mathbf{0}}(\mathcal{X}, \mathcal{A})))\big\|_{P,q/2} = \\
&\big\|(\epsilon^{\mathbf{Y}} - (\ell(\mathcal{X}, \mathcal{A}) - \ell_{\mathbf{0}}(\mathcal{X}, \mathcal{A})) + \theta_0(\mathbf{m}(\mathcal{X}, \mathcal{A}) - \mathbf{m}_{\mathbf{0}}(\mathcal{X}, \mathcal{A})) + \\
&\alpha_0 \mathcal{A}(\mathbf{m}(\mathcal{X}, \mathcal{A}) - \mathbf{m}_{\mathbf{0}}(\mathcal{X}, \mathcal{A})))^\intercal (\epsilon^{\mathbf{T}} - (\mathbf{m}(\mathcal{X}, \mathcal{A}) - \mathbf{m}_{\mathbf{0}}(\mathcal{X}, \mathcal{A})))\big\| \leqslant \\
&(2C + 2C^3/c + 2C_A C^3/c) 2C = 4C^2 + 4C^4/c + 4C_A C^4/c
\end{aligned}$$

where C_A is the constant term introduced in proposition 2, which gives the bound on m_n in condition 2. Following the exact same approach along with proposition 2, we can derive an upperbound for other elements of $\|\psi(W; \zeta_0, \eta)\|_{P,q/2}$.

C.2.3 Following the same argument in the previous section, we prove the boundedness of elements of these matrices:

$$\begin{aligned}
&\|\mathrm{E}_P\left[\psi_{11}^a(W; \eta)\right] - \mathrm{E}_P\left[\psi_{11}^a\left(W; \eta_0\right)\right]\| = |\mathrm{E}_P\left[\psi_{11}^a(W; \eta) - \psi_{11}^a\left(W; \eta_0\right)\right]| = \\
&|\mathrm{E}_P\left[-(\mathbf{T} - \mathbf{m}(\mathcal{X}, \mathcal{A}))^\intercal (\mathbf{T} - \mathbf{m}(\mathcal{X}, \mathcal{A})) + (\mathbf{T} - \mathbf{m}_{\mathbf{0}}(\mathcal{X}, \mathcal{A}))^\intercal (\mathbf{T} - \mathbf{m}_{\mathbf{0}}(\mathcal{X}, \mathcal{A}))\right]| = \\
&\big|\mathrm{E}_P\big[-((\mathbf{T} - \mathbf{m}_{\mathbf{0}}(\mathcal{X}, \mathcal{A})) - (\mathbf{m}(\mathcal{X}, \mathcal{A}) - \mathbf{m}_{\mathbf{0}}(\mathcal{X}, \mathcal{A})))^\intercal ((\mathbf{T} - \mathbf{m}_{\mathbf{0}}(\mathcal{X}, \mathcal{A})) - (\mathbf{m}(\mathcal{X}, \mathcal{A}) - \mathbf{m}_{\mathbf{0}}(\mathcal{X}, \mathcal{A}))) + \\
&(\mathbf{T} - \mathbf{m}_{\mathbf{0}}(\mathcal{X}, \mathcal{A}))^\intercal (\mathbf{T} - \mathbf{m}_{\mathbf{0}}(\mathcal{X}, \mathcal{A}))\big]\big| = \big|\mathrm{E}_P\big[((\mathbf{m}(\mathcal{X}, \mathcal{A}) - \mathbf{m}_{\mathbf{0}}(\mathcal{X}, \mathcal{A})) - \epsilon^{\mathbf{T}})^\intercal (\epsilon^{\mathbf{T}} - (\mathbf{m}(\mathcal{X}, \mathcal{A}) - \mathbf{m}_{\mathbf{0}}(\mathcal{X}, \mathcal{A}))) + \\
&\epsilon^{\mathbf{T}^\intercal} \epsilon^{\mathbf{T}}\big]\big| = \big|\mathrm{E}_P\big[(\mathbf{m}(\mathcal{X}, \mathcal{A}) - \mathbf{m}_{\mathbf{0}}(\mathcal{X}, \mathcal{A}))^\intercal \epsilon^{\mathbf{T}} + \epsilon^{\mathbf{T}^\intercal} (\mathbf{m}(\mathcal{X}, \mathcal{A}) - \mathbf{m}_{\mathbf{0}}(\mathcal{X}, \mathcal{A})) - \\
&(\mathbf{m}(\mathcal{X}, \mathcal{A}) - \mathbf{m}_{\mathbf{0}}(\mathcal{X}, \mathcal{A}))^\intercal (\mathbf{m}(\mathcal{X}, \mathcal{A}) - \mathbf{m}_{\mathbf{0}}(\mathcal{X}, \mathcal{A}))\big]\big| = 2\left\|\epsilon^{\mathbf{T}}\right\|_{P,2} \left\|\mathbf{m}(\mathcal{X}, \mathcal{A}) - \mathbf{m}_{\mathbf{0}}(\mathcal{X}, \mathcal{A})\right\|_{P,2} + \\
&\|\mathbf{m}(\mathcal{X}, \mathcal{A}) - \mathbf{m}_{\mathbf{0}}(\mathcal{X}, \mathcal{A})\|_{P,2}^2 \leqslant 2C\delta_n + \delta_n^2 \leqslant \delta_n'
\end{aligned}$$

by assumption 7, which gives the bound on r_n in condition 3. Further,

$$\begin{aligned}
&\left(\mathrm{E}_P\left[\left\|\psi_{11}\left(W; \zeta_0, \eta\right) - \psi_{11}\left(W; \zeta_0, \eta_0\right)\right\|^2\right]\right)^{1/2} = \left\|\psi_{11}\left(W; \theta_0, \eta\right) - \psi_{11}\left(W; \theta_0, \eta_0\right)\right\|_{P,2} = \\
&\big\|(\mathbf{Y} - \ell(\mathcal{X}, \mathcal{A}) - \theta_0(\mathbf{T} - \mathbf{m}(\mathcal{X}, \mathcal{A})) - \alpha_0(\mathcal{A}(\mathbf{T} - \mathbf{m}(\mathcal{X}, \mathcal{A})))^\intercal (\mathbf{T} - \mathbf{m}(\mathcal{X}, \mathcal{A}))) - ((\mathbf{Y} - \ell_{\mathbf{0}}(\mathcal{X}, \mathcal{A}) - \\
&\theta_0(\mathbf{T} - \mathbf{m}_{\mathbf{0}}(\mathcal{X}, \mathcal{A})) - \alpha_0(\mathcal{A}(\mathbf{T} - \mathbf{m}_{\mathbf{0}}(\mathcal{X}, \mathcal{A})))^\intercal (\mathbf{T} - \mathbf{m}_{\mathbf{0}}(\mathcal{X}, \mathcal{A}))))\big\|_{P,2} = \\
&\big\|-\epsilon^{\mathbf{Y}^\intercal} (\mathbf{m}(\mathcal{X}, \mathcal{A}) - \mathbf{m}_{\mathbf{0}}(\mathcal{X}, \mathcal{A})) + (\ell_{\mathbf{0}}(\mathcal{X}, \mathcal{A}) - \ell(\mathcal{X}, \mathcal{A}) + \theta_0(\mathbf{m}(\mathcal{X}, \mathcal{A}) - \mathbf{m}_{\mathbf{0}}(\mathcal{X}, \mathcal{A})) + \\
&\alpha_0 \mathcal{A}(\mathbf{m}(\mathcal{X}, \mathcal{A}) - \mathbf{m}_{\mathbf{0}}(\mathcal{X}, \mathcal{A})))^\intercal \epsilon^{\mathbf{T}} - (\ell_{\mathbf{0}}(\mathcal{X}, \mathcal{A}) - \ell(\mathcal{X}, \mathcal{A}) + \theta_0(\mathbf{m}(\mathcal{X}, \mathcal{A}) - \mathbf{m}_{\mathbf{0}}(\mathcal{X}, \mathcal{A})) + \\
&\alpha_0 \mathcal{A}(\mathbf{m}(\mathcal{X}, \mathcal{A}) - \mathbf{m}_{\mathbf{0}}(\mathcal{X}, \mathcal{A})))^\intercal (\mathbf{m}(\mathcal{X}, \mathcal{A}) - \mathbf{m}_{\mathbf{0}}(\mathcal{X}, \mathcal{A}))\big\|_{P,2} \leqslant \\
&(C + 2C^3/c + 2C_A C^3/c)\big\|\mathbf{m}(\mathcal{X}, \mathcal{A}) - \mathbf{m}_{\mathbf{0}}(\mathcal{X}, \mathcal{A})\big\| + C\big\|\ell(\mathcal{X}, \mathcal{A}) - \ell_{\mathbf{0}}(\mathcal{X}, \mathcal{A})\big\| + \\
&(\big\|\ell(\mathcal{X}, \mathcal{A}) - \ell_{\mathbf{0}}(\mathcal{X}, \mathcal{A})\big\| + (2C^2/c + 2C_A C^2/c)\big\|\mathbf{m}(\mathcal{X}, \mathcal{A}) - \mathbf{m}_{\mathbf{0}}(\mathcal{X}, \mathcal{A})\big\|)\big\|\mathbf{m}(\mathcal{X}, \mathcal{A}) - \mathbf{m}_{\mathbf{0}}(\mathcal{X}, \mathcal{A})\big\| \leqslant \\
&(1 + 2C^2/c + 2C_A C^2/c)\delta_n n_f^{-1/2} \leqslant (1 + 2C^2/c + 2C_A C^2/c)\delta_n \leqslant \delta_n'
\end{aligned}$$

by assumption 7. Following the same approach along with proposition 2, we can derive an upper bound for the other dimensions of ψ and ψ^a. This upper bound provides the bound on r_n in condition 3.
Lastly, let

$$f(r) := \mathbb{E}_P\left[\psi\left(W; \theta_0, \eta_0 + r\left(\eta - \eta_0\right)\right], \quad r \in (0, 1).\right.$$

Then for any $r \in (0,1)$, for the first dimension of the score function:

$$
\begin{aligned}
&f(r) = \mathbb{E}_P \begin{pmatrix} (\mathbf{Y} - (\ell + r(\ell - \ell_{\mathbf{0}})) - \theta_0(\mathbf{T} - (\mathbf{m} + r(\mathbf{m} - \mathbf{m_0}))) \\ -\alpha_0 \mathcal{A}(\mathbf{T} - (\mathbf{m} + r(\mathbf{m} - \mathbf{m_0}))))^\intercal (\mathbf{T} - (\mathbf{m} + r(\mathbf{m} - \mathbf{m_0}))) \end{pmatrix} \\
&\partial f(r) = \mathbb{E}_P \big[(\ell_{\mathbf{0}} - \ell + \theta_0(\mathbf{m} - \mathbf{m_0}) + \alpha_0 \mathcal{A}(\mathbf{m} - \mathbf{m_0}))^\intercal (\mathbf{T} - \mathbf{m} - r(\mathbf{m} - \mathbf{m_0})) + \\
&(\mathbf{Y} - \ell - r(\ell - \ell_{\mathbf{0}}) - \theta_0(\mathbf{T} - \mathbf{m} - r(\mathbf{m} - \mathbf{m_0})) - \alpha_0 \mathcal{A}(\mathbf{T} - \mathbf{m} - r(\mathbf{m} - \mathbf{m_0})))^\intercal (\mathbf{m_0} - \mathbf{m}) \big] \\
&\partial^2 f(r) = \mathbb{E}_P \begin{pmatrix} (\ell_{\mathbf{0}} - \ell + \theta_0(\mathbf{m} - \mathbf{m_0}) + \alpha_0 \mathcal{A}(\mathbf{m} - \mathbf{m_0}))^\intercal (\mathbf{m_0} - \mathbf{m}) \\ +(\ell_{\mathbf{0}} - \ell + \theta_0(\mathbf{m} - \mathbf{m_0}) + \alpha_0 \mathcal{A}(\mathbf{m} - \mathbf{m_0}))^\intercal (\mathbf{m_0} - \mathbf{m}) \end{pmatrix} \\
&= 2\mathbb{E}_P \big[(\ell_{\mathbf{0}} - \ell + \theta_0(\mathbf{m} - \mathbf{m_0}) + \alpha_0 \mathcal{A}(\mathbf{m} - \mathbf{m_0}))^\intercal (\mathbf{m_0} - \mathbf{m}) \big] \\
&\leqslant 2(\|\ell - \ell_0\| + 2C^2/c \|m - m_0\| + 2C_A C^2/c \|m - m_0\|) \|\mathbf{m} - \mathbf{m_0}\| \\
&\leqslant 2(1 + 2C^2/c + 2C_A C^2/c) \delta_n n_f^{-1/2} \leqslant \delta'_n n_f^{-1/2}
\end{aligned}
$$

which gives the bound on λ'_n in condition 3.

C.2.4

$$
\begin{aligned}
\psi(W; \zeta_0, \eta_0) &= \begin{pmatrix} (\mathbf{Y} - \ell_{\mathbf{0}}(\mathcal{X}, \mathcal{A}) - \theta_0(\mathbf{T} - \mathbf{m_0}(\mathcal{X}, \mathcal{A})) - \alpha_0(\mathcal{A}(\mathbf{T} - \mathbf{m_0}(\mathcal{X}, \mathcal{A})))^\intercal (\mathbf{T} - \mathbf{m_0}(\mathcal{X}, \mathcal{A})) \\ (\mathbf{Y} - \ell_{\mathbf{0}}(\mathcal{X}, \mathcal{A}) - \theta_0(\mathbf{T} - \mathbf{m_0}(\mathcal{X}, \mathcal{A})) - \alpha_0(\mathcal{A}(\mathbf{T} - \mathbf{m_0}(\mathcal{X}, \mathcal{A})))^\intercal \mathcal{A}(\mathbf{T} - \mathbf{m_0}(\mathcal{X}, \mathcal{A})) \end{pmatrix} \\
&= \begin{pmatrix} \epsilon^{\mathbf{Y}^\intercal} \epsilon^{\mathbf{T}} \\ \epsilon^{\mathbf{Y}^\intercal} \mathcal{A} \epsilon^{\mathbf{T}} \end{pmatrix}
\end{aligned}
$$

$$
\begin{aligned}
\mathbb{E}_P \left[\psi(W; \zeta_0, \eta_0) \, \psi(W; \theta_0, \eta_0)' \right] &= \mathbb{E}_P \left[\begin{pmatrix} \epsilon^{\mathbf{Y}^\intercal} \epsilon^{\mathbf{T}} \\ \epsilon^{\mathbf{Y}^\intercal} \mathcal{A} \epsilon^{\mathbf{T}} \end{pmatrix} \begin{pmatrix} \epsilon^{\mathbf{Y}^\intercal} \epsilon^{\mathbf{T}} & \epsilon^{\mathbf{Y}^\intercal} \mathcal{A} \epsilon^{\mathbf{T}} \end{pmatrix} \right] \\
&= \mathbb{E}_P \left[\begin{pmatrix} (\epsilon^{\mathbf{Y}^\intercal} \epsilon^{\mathbf{T}})^2 & (\epsilon^{\mathbf{Y}^\intercal} \epsilon^{\mathbf{T}})(\epsilon^{\mathbf{Y}^\intercal} \mathcal{A} \epsilon^{\mathbf{T}}) \\ (\epsilon^{\mathbf{Y}^\intercal} \mathcal{A} \epsilon^{\mathbf{T}})(\epsilon^{\mathbf{Y}^\intercal} \epsilon^{\mathbf{T}}) & (\epsilon^{\mathbf{Y}^\intercal} \mathcal{A} \epsilon^{\mathbf{T}})^2 \end{pmatrix} \right] \qquad (34)
\end{aligned}
$$

The eigen values of this matrix are the roots of the following quadratic equation:

$$
\lambda^2 - \lambda(\mathbb{E}_P[(\epsilon^{\mathbf{Y}^\intercal} \epsilon^{\mathbf{T}})^2 + (\epsilon^{\mathbf{Y}^\intercal} \mathcal{A} \epsilon^{\mathbf{T}})^2]) + \mathbb{E}_P[(\epsilon^{\mathbf{Y}^\intercal} \epsilon^{\mathbf{T}})^2] + \mathbb{E}_P[(\epsilon^{\mathbf{Y}^\intercal} \mathcal{A} \epsilon^{\mathbf{T}})^2] - \mathbb{E}_P[(\epsilon^{\mathbf{Y}^\intercal} \epsilon^{\mathbf{T}})(\epsilon^{\mathbf{Y}^\intercal} \mathcal{A} \epsilon^{\mathbf{T}})]^2 = 0 \qquad (35)
$$

We know that in a quadratic equation of form $a_2 x^2 + a_1 x + a_0 = 0$, the sum of the roots are $\frac{-a_1}{a_2}$ and the product of the roots are $\frac{a_0}{a_2}$. To ensure that all the eigen values are positive, we need to make sure both $\frac{-a_1}{a_2}$ and $\frac{a_0}{a_2}$ are positive:

$$
\frac{-a_1}{a_2} = \|\epsilon^{\mathbf{Y}^\intercal} \epsilon^{\mathbf{T}}\|_{P,2} + \|\epsilon^{\mathbf{Y}^\intercal} \mathcal{A} \epsilon^{\mathbf{T}}\|_{P,2} > 0 \qquad (36)
$$

$$
\frac{a_0}{a_2} = \mathbb{E}_P[(\epsilon^{\mathbf{Y}^\intercal} \epsilon^{\mathbf{T}})^2] + \mathbb{E}_P[(\epsilon^{\mathbf{Y}^\intercal} \mathcal{A} \epsilon^{\mathbf{T}})^2] - \mathbb{E}_P[(\epsilon^{\mathbf{Y}^\intercal} \epsilon^{\mathbf{T}})(\epsilon^{\mathbf{Y}^\intercal} \mathcal{A} \epsilon^{\mathbf{T}})]^2 > 0 \qquad (37)
$$

36 holds according to assumption 3. Equation 37 also holds according to Cauchy-Schwarz inequality. The equality in Cauchy-Schwarz inequality for two random variables X and Y happens when $\|X\| = 0$ or $\|Y\| = 0$ or $Y = rX$ for some $r \neq 0$. neither of these cases hold: $\|\epsilon^{\mathbf{Y}^\intercal} \epsilon^{\mathbf{T}}\|_{P,2} > 0$ based on 3. $\|\epsilon^{\mathbf{Y}^\intercal} \mathcal{A} \epsilon^{\mathbf{T}}\|_{P,2} > 0$ based on 3 and the fact that $\mathcal{A}$ is the adjacency matrix with non-negative elements and $\mathcal{A} \neq 0$. Also, $\nexists r \neq 0 : \epsilon^{\mathbf{Y}^\intercal} \mathcal{A} \epsilon^{\mathbf{T}} = r \epsilon^{\mathbf{Y}^\intercal} \epsilon^{\mathbf{T}}$ according to 5. Thus, the roots of equation 35, which are the eigen values of matrix 34 are bounded from below by some positive c_0.

Thus, all conditions 10.1 and 10.2 are verified. This completes the proof.

11 Complementary Experimental Setup

11.1 Datasets Details

We use the following network datasets for our evaluations:

- Real World Data

  - IndianVillage (Banerjee et al., 2014; Jackson et al., 2012): It is a 2010 survey data from villages in Karnataka, India. The survey gathered information from 16,995 individuals residing in 77 villages. It includes 15 features like age, occupation, gender, and more. Additionally, the dataset incorporated 12 distinct social networks involving 69,000 individuals, which included both the surveyed group of 16,995 individuals and others. These networks represented relationships like friendships, relatives, social visits, and financial exchanges. We treated all these connections uniformly, ensuring a consistent network where all edges carried the same meaning.

- Semi-Synthetic Data
  $(\mathcal{X}, \mathbf{T}, \mathbf{Y})$ are generated based on data generative processes 38 and 39 and the network comes from real-world network dataset below:
  - Cora (McCallum et al., 2000): It comprises academic research papers and their citation links, forming a graph structure. It consists of 2708 scientific publications classified into one of seven classes. The citation network consists of 5429 links.
  - Pubmed: Similar to Cora, it is a citation network, consists of 19717 scientific publications from PubMed database classified into one of three classes. The citation network consists of 44338 links.
  - Flickr: It is a network derived from Flickr, one of the largest platform for sharing photos. Each node in the graph represents an image, and if two images have shared characteristics like geographic location, gallery, or comments by the same user, there will be an edge connecting their respective nodes. It consists of 105938 nodes and 2316948 edges.

- Synthetic Data
  $(\mathcal{X}, \mathbf{T}, \mathbf{Y})$ are generated based on data generative process 38 and the network comes from the synthetic network generative process below:
  - Stochastic Block Model (SBM) (Holland et al., 1983): It is a generative model for networks. We also tried our method on a synthetic network produced by SBM, to have more control over the network parameters. In SBM, nodes are partitioned into multiple blocks or communities, and the probability of an edge existing between two nodes depends on their respective block assignments.

11.2 Data Generative Process

The covariates $(\mathcal{X})$, treatment assignments $(\mathbf{T})$, and outcomes $(\mathbf{Y})$ are synthetically generated following a specific data generative process outlined in Section 2.1. This section details one such data-generative process used in our experiments:

$$
\begin{aligned}
&\mathcal{X} \sim \mathcal{N}(0,1) \\
&\pi = \left(1 + \exp(\frac{\mathcal{X} + \gamma \mathcal{A}\mathcal{X})}{-10}\right)^{-1} \\
&\mathbf{T} \sim \text{Bin}(\pi) \\
&\mathbf{Y} = \mathcal{X} + \mathcal{A}\mathcal{X} + \mathbf{T} \times \theta_0 + \alpha \mathcal{A}\mathbf{T}
\end{aligned} \tag{38}
$$

Where $\mathcal{A}\mathbf{T}$ is the exposure map and would be the sum of treated neighbors for each node. In our setup, we assumed that this exposure map is known. In the experiments in which we compare our method against baselines, the target parameters are $\theta_0 = 10$ and $\alpha_0 = 5$. For the Pubmed and Flickr datasets, we adopt this data generative process to generate $(\mathcal{X}, \mathbf{T}, \mathbf{Y})$.

We also simulated another data generative process, which is more complex and involves non-linearity:

$$
\begin{aligned}
&\mathbf{X_i} \sim \mathcal{N}(0,1) \\
&\pi = \left(1 + \exp(\frac{\mathcal{X} + \gamma \mathcal{A}\mathcal{X})}{-10}\right)^{-1} \\
&T_i \sim \text{Bin}(\pi_i) \\
&Y_i = \sigma(\sum_j X_{ij} + MAX_j((A\mathcal{X})_{ij})) + T_i \times \theta_0 + \alpha \mathcal{A_i}\mathbf{T}
\end{aligned} \tag{39}
$$

where σ denotes the sigmoid function. The target parameters are $\theta_0 = 20$ and $\alpha_0 = 5$. For the Cora dataset, we adopt this data generative process to generate $(\mathcal{X}, \mathbf{T}, \mathbf{Y})$.

Table 3: The results of coverage study our approach across datasets over 100 trials with the 95% confidence interval

Dataset	θ	α
Cora	100%	100%
Pubmed	100%	100%
Flickr	92%	52%

11.3 Extended Baselines

This section provides a more thorough description of the baselines used.

1. **NetEst** (Jiang and Sun, 2022): Utilizes GNNs for learning representations of confounders for individual units and their neighbors, coupled with an adversarial learning process to align distributions for networked causal inference.

2. **Net-TMLE** (Ogburn et al., 2022): Employs an efficient influence function and moment condition to derive a doubly robust estimator. This approach leverages the efficiency of targeted maximum likelihood estimation (TMLE) to improve the robustness and accuracy of causal effect estimates in the presence of network interference.

3. **T-Learner** (Künzel et al., 2019): Creates two separate models to predict outcomes for each treatment arm based on unit and neighbor covariates, with estimations modeled using GNNs. This method provides a straightforward way to estimate treatment effects by splitting the problem into two learning tasks.

4. **DML with predefined aggregates**: Applies Double Machine Learning (DML) in the i.i.d. setting but uses predefined aggregates like max, min, and mean for neighbor information aggregation. This approach simplifies the network structure into summary statistics, facilitating the application of traditional DML techniques.

5. **Tresp & Ma** (Ma and Tresp, 2020): Maps the representation of covariates to a new space where treatment and covariates are disentangled, incorporating the Hilbert-Schmidt Independence Criterion (HSIC) as a regularization term. Subsequently, GNNs are employed to aggregate covariate information from neighboring nodes. Two separate models are then trained to estimate the outcome based on the output of the GNNs and a predefined exposure map for the treatment and control groups.

6. **L&L method** (Leung and Loupos, 2022).In our paper, we designate this approach as the "L&L" method in the experiment section. L&L is a working paper and the code is not publicly available and according to the authors, will become available after the work is published. We implemented a version of their method to the best of our understanding. Unlike methods that utilize continuous exposure measures, the L&L method requires the conversion of exposure data into a binary format for application to our dataset. Specifically, for each node, if more than half of its neighbors, including the node itself, are subject to treatment, we assign an exposure value of 1; if not, the exposure value is set to 0. Utilizes a standard doubly robust estimator combined with GNNs to estimate the total effect.

12 Complementary Experimental Results

12.1 Coverage Study

In our investigation, we performed a comprehensive coverage analysis leveraging the closed-form formula for calculating variance and confidence intervals as detailed in Section 4.1. This analysis involved applying our proposed methodology across multiple executions—specifically, 100 iterations—on each dataset under consideration. For each iteration, we computed confidence intervals and assessed the frequency at which the true value of the target parameter fell within these intervals. This measure of frequency serves as a critical indicator of the reliability and precision of our methodology in capturing the parameter of interest across varied datasets. The results are presented in table 3.

12.2 Case Study: SHG Participation

Below is the table presenting the results of six baseline methods and two versions of our framework on the Indian Village dataset:

Table 4: The results of six baseline methods and two versions of our framework on the Indian Village dataset

	GDML	w/o Focal Set	PA	T-learner	NetEst	Net TMLE	L&L	Ma & Tresp
ADE	0.315	0.390	0.209	0.469	N/A*	0.291	N/A	0.295
APE	0.050	-0.002	-0.004	N/A	N/A*	N/A	N/A	0.016
ATE	0.365	0.388	0.205	N/A	N/A*	N/A	0.113	0.311

NetEst was not stable during training on this data and resulted in NaN values. Note that the confidence intervals for ADE and APE generated by our framework are $[-1.570, 2.200]$ and $[-1.017, 1.116]$ respectively with a 95% confidence. In the literature, Gilad et al. (2021) also quantified the direct effect of SHG membership on the probability of possessing an outstanding loan as **0.30**.

12.3 Graph density

To assess the impact of network data sparsity on estimation performance, we utilize Stochastic Block Model (SBM) synthetic graphs, providing greater control over graph generation.
We fix the number of components as 200, the number of nodes as 3000, and the probability of existence of an edge between components as 0.0001. Subsequently, we vary the probability of edge existence within the component, denoted as P_{intra}, to modulate the sparsity of the graph. For each P_{intra}, we generate a single graph and for each graph, we generate 100 different datasets $\mathcal{X}, \mathbf{T}$ and $\mathbf{Y}$ and report the average of estimated direct effect. Table 5 presents the results. notably, as P_{intra} increases, the number of edges rises. Given the fixed number of nodes, this causes a reduction in the size of the focal set (sample size), resulting in an increased bias in the estimation process. This result showcases that our methodology exhibits enhanced performance in sparser networks. As the number of edges increases within a network with a fixed number of nodes, we observe a corresponding rise in both the relative error and the variance of our estimations. This trend suggests a direct relationship between network density and the performance of our method.

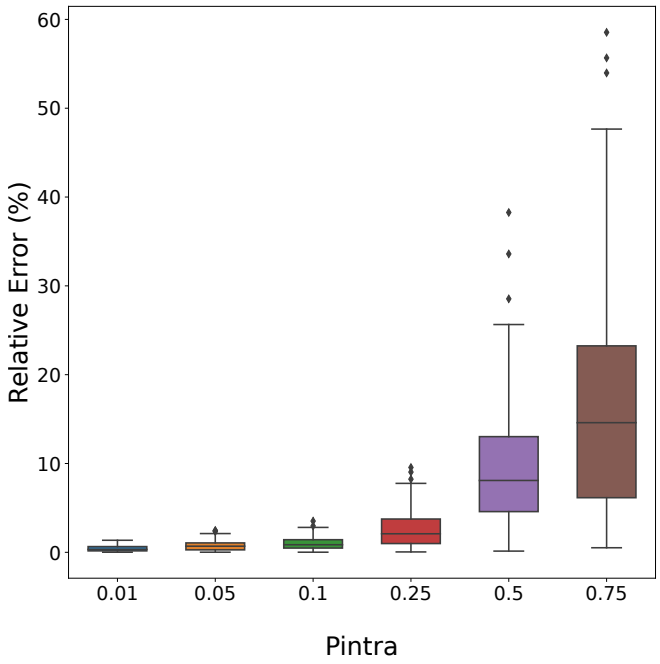

Figure 4: Relative error of the Average Direct Effect estimations over 100 trials on the synthetic graphs generated from Stochastic Block Model (SBM) using different values for the probability of edge existence within each component (P_{intra}), 3000 nodes, 200 components, and $P_{inter} = 0.0001$, where P_{inter} represents the probability of presence of edge between components.

Table 5: Mean squared error (MSE) for our GDML approach on graphs generated using a stochastic block model with 3000 nodes and 200 blocks for different values of intra-block tie probabilities, represented as P_{intra}.

P_{intra}	focal set size	# edges	MSE
0.01	2382	655	0.052
0.05	1788	1536	0.093
0.1	1349	2588	0.124
0.25	636	5673	0.351
0.5	271	10872	1.193
0.75	200	16157	2.120

Table 6: MSE of the estimations of the causal quantities over 100 trials from the application of various graph aggregation tools combined with the core of our framework

	ADE	**APE**	**ATE**
GDML + NeRF+	$1.61_{\pm 2.52}$	$87.25_{\pm 1057}$	$83.42_{\pm 980}$
GDML + GIN	$0.16_{\pm 0.43}$	$0.21_{\pm 0.73}$	$0.40_{\pm 1.48}$

12.4 Generality of GDML Framework: Choice of Nuisance Function Approximator

For accurate and consistent estimation of nuisance parameters, we leverage the flexible machine learning approach using GNNs. However, our framework can integrate with any graph aggregation tool to estimate propensity scores and outcome models. As the nuisance parameters are functions of both the covariates of an individual unit and those of their social neighbors, their estimation requires aggregating information across the neighborhood. In an effort to demonstrate the generality of our framework, we adopted Network Random Forests (NeRF+) (Tang et al., 2024), which is a family of network-assisted prediction models built upon a generalization of random forests. These models may lack the representational power of GNNs; however, they are interpretable and can be an ideal choice for certain applications. In our work, we employ the Graph Isomorphism Network (GIN)(Xu et al., 2018) due to its superior performance over other GNN architectures like GCN (Kipf and Welling, 2016), GAT (Veličković et al., 2017), and GraphSAGE (Hamilton et al., 2017). GIN's alignment with the representational capabilities of the Weisfeiler-Lehman test (Morris et al., 2019) makes it an ideal choice for effectively capturing the intricate dynamics inherent in social network structures. Table6 presents the result of two variations of our framework combined with GIN and NeRF+ on the Cora dataset with data generative process 38. The performance of GIN in this case is superior; however, the performance of NeRF+ is also close to the ground truth. As mentioned earlier, NeRF+ is interpretable, which may be necessary for some applications.

12.5 MSE Decomposition into Bias vs. Variance

Table 7: Comparison of mean squared error of our GDML approach with other baselines. The tables, from top to bottom, show the results from Cora, Pubmed, and Flickr. The error is decomposed into bias and variance. PA stands for DML combined with predefined aggregates. For T-Learner and Net TMLE methods, peer effect estimation is not applicable. L&L's framework concentrates on total effect and does not calculate ADE and APE separately.
*: results for Net TMLE and MaTresp on Flickr are not reported because it ran out of system memory.

	ADE			APE			ATE		
	MSE	Bias	Var	MSE	Bias	Var	MSE	Bias	Var
PA	$0.31_{\pm 0.83}$	-0.07	0.30	$1.02_{\pm 2.90}$	0.01	1.02	$1.41_{\pm 4.05}$	-0.06	1.40
T-learner(Künzel et al., 2019)	$9.84_{\pm 51.32}$	0.13	9.82	N/A	N/A	N/A	N/A	N/A	N/A
NetEst(Jiang and Sun, 2022)	$174.66_{\pm 1.07}$	13.22	0.00	$9.48_{\pm 1.75}$	-3.08	0.02	$71.96_{\pm 4.35}$	8.48	0.02
Net TMLE(Ogburn et al., 2022)	$13.67_{\pm 6.47}$	-3.67	0.20	N/A	N/A	N/A	N/A	N/A	N/A
L&L(Leung and Loupos, 2022)	N/A	N/A	N/A	N/A	N/A	N/A	$120.20_{\pm 34.31}$	-10.93	0.64
Ma & Tresp(Ma and Tresp, 2020)	$3.87_{\pm 42.98}$	-0.37	3.74	$0.02_{\pm 0.14}$	-0.03	0.02	$4.26_{\pm 47.37}$	-0.40	4.10
GDML w/o FS	$0.26_{\pm 0.83}$	-0.11	0.25	$0.99_{\pm 2.79}$	0.06	0.99	$1.37_{\pm 3.72}$	-0.05	1.36
GDML	$0.33_{\pm 0.79}$	-0.42	0.16	$0.29_{\pm 0.80}$	-0.23	0.23	$0.88_{\pm 2.21}$	-0.65	0.46

	ADE			APE			ATE		
	MSE	Bias	Var	MSE	Bias	Var	MSE	Bias	Var
PA	$0.35_{\pm 0.69}$	0.49	0.11	$10.69_{\pm 6.52}$	3.20	0.43	$14.30_{\pm 9.66}$	3.69	0.66
T-learner(Künzel et al., 2019)	$1.67_{\pm 4.58}$	0.55	1.37	N/A	N/A	N/A	N/A	N/A	N/A
NetEst(Jiang and Sun, 2022)	$1655.8_{\pm 30.94}$	40.69	0.04	$0.44_{\pm 0.39}$	-0.65	0.02	$1603.53_{\pm 45.65}$	40.04	0.08
Net TMLE(Ogburn et al., 2022)	$1.24_{\pm 1.36}$	-1.06	0.11	N/A	N/A	N/A	N/A	N/A	N/A
L&L(Leung and Loupos, 2022)	N/A	N/A	N/A	N/A	N/A	N/A	$42.76_{\pm 2.92}$	-6.54	0.01
Ma & Tresp(Ma and Tresp, 2020)	$0.02_{\pm 0.03}$	0.07	0.01	$0.01_{\pm 0.01}$	0.09	0.00	$0.04_{\pm 0.06}$	0.15	0.02
GDML w/o FS	$0.04_{\pm 0.13}$	-0.04	0.04	$0.30_{\pm 0.73}$	0.02	0.30	$0.45_{\pm 1.21}$	-0.02	0.45
GDML	0.03 ± 0.11	-0.05	0.03	$0.28_{\pm 0.84}$	-0.08	0.27	$0.30_{\pm 0.87}$	-0.12	0.29

	ADE			APE			ATE		
	MSE	Bias	Var	MSE	Bias	Var	MSE	Bias	Var
PA	$1133_{\pm 4700}$	24.38	539.19	$37719_{\pm 143300}$	136.77	19014.63	$51790_{\pm 199836}$	161.15	25820.42
T-learner(Künzel et al., 2019)	$2380_{\pm 5495}$	40.36	751.20	N/A	N/A	N/A	N/A	N/A	N/A
NetEst(Jiang and Sun, 2022)	$53827_{\pm 921}$	232.00	1.03	$103_{\pm 105}$	9.84	6.86	$58503_{\pm 3444}$	241.85	13.14
Net TMLE(Ogburn et al., 2022)	N/A*	N/A*	N/A*	N/A	N/A	N/A	N/A	N/A	N/A
L&L(Leung and Loupos, 2022)	N/A	N/A	N/A	N/A	N/A	N/A	$25.32_{\pm 6.58}$	-5.02	0.11
Ma & Tresp(Ma and Tresp, 2020)	N/A*	N/A*	N/A*	N/A*	N/A*	N/A*	N/A*	N/A*	N/A*
GDML w/o FS	$4.92_{\pm 10.91}$	-1.71	2.00	$121_{\pm 308}$	6.79	75.37	$95_{\pm 276}$	5.09	69.95
GDML	$76_{\pm 211}$	-3.95	60.43	$26.01_{\pm 26.07}$	1.73	23.02	$84_{\pm 272}$	-2.23	79.60

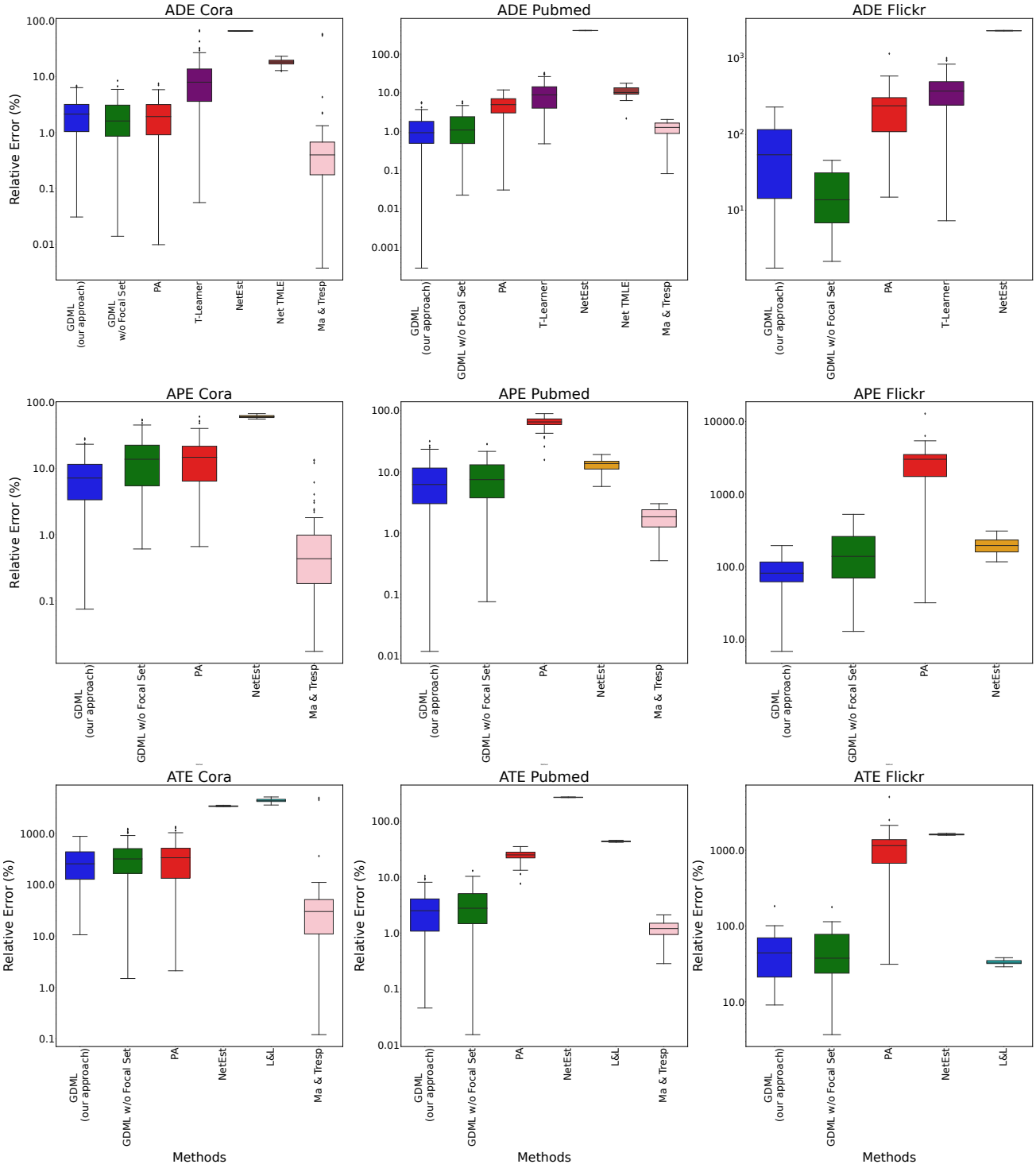

Figure 3: Relative Error of different methods for estimating causal effects across different datasets. Note that the y-axis is log-scaled. In the figure, two variants of our method are presented: one utilizing a focal set and another without a focal set, encompassing the entire dataset. 'PA' refers to Double Machine Learning combined with predefined aggregates