

# GRAPH NEURAL NETWORKS FOR CAUSAL INFERENCE UNDER NETWORK CONFOUNDING\*

Michael P. Leung<sup>†</sup>      Pantelis Loupos<sup>‡</sup>

March 21, 2024

**ABSTRACT.** This paper studies causal inference with observational network data. A challenging aspect of this setting is the possibility of interference in both potential outcomes and selection into treatment, for example due to peer effects in either stage. We therefore consider a nonparametric setup in which both stages are reduced forms of simultaneous-equations models. This results in high-dimensional network confounding, where the network and covariates of all units constitute sources of selection bias. The literature predominantly assumes that confounding can be summarized by a known, low-dimensional function of these objects, and it is unclear what selection models justify common choices of functions. We show that graph neural networks (GNNs) are well suited to adjust for high-dimensional network confounding. We establish a network analog of approximate sparsity under primitive conditions on interference. This demonstrates that the model has low-dimensional structure that makes estimation feasible and justifies the use of shallow GNN architectures.

JEL CODES: C14, C31, C45

KEYWORDS: causal inference, unconfoundedness, network interference, graph neural networks, approximate sparsity

---

\*This paper was previously circulated under the title “Unconfoundedness with Network Interference.” We thank Ruonan Xu and seminar audiences at Duke, UCSD, GraphEx2023, and the 2023 North American Summer Meetings for helpful comments on this paper.

<sup>†</sup>Department of Economics, University of California, Santa Cruz. E-mail: leungm@ucsc.edu.

<sup>‡</sup>Graduate School of Management, University of California, Davis. E-mail: ploupos@ucdavis.edu.

# 1 Introduction

Treatment assignment is said to be unconfounded if it is as good as random within subpopulations of observationally equivalent units. In settings where the stable unit treatment value assumption (SUTVA) is plausible, units with identical covariates are naturally considered observationally equivalent. However, when units are connected through a network, they may differ on other observed dimensions that may confound causal inference if SUTVA is violated and interference is mediated by the network. These dimensions include, for example, the number of type- $x$  neighbors, the number of type- $x$  neighbors with  $m$  neighbors of type  $y$ , and so on through higher-order neighbors.

Existing formulations of unconfoundedness only utilize a small subset of these dimensions. For example, a common set of controls used in the literature is the vector consisting of own covariates, number of neighbors, and average covariates of neighbors. This choice may be difficult to justify in practice due to a lack of behavioral models of selection. Neighbor covariates may influence selection into treatment in more complex ways not adequately captured by the mean. Furthermore, this choice of controls implies no confounding from higher-order neighbors, which we show rules out economically interesting sources of interference in treatment selection, such as endogenous peer effects.

In this paper, we study estimation and inference for treatment and spillover effects under a fully nonparametric formulation of unconfoundedness motivated by a model of selection. To allow for peer effects, selection is governed by the reduced form of a simultaneous-equations model, which is a function of the entirety of  $\mathbf{X}$ , the matrix of all units' covariates, and  $\mathbf{A}$ , the network adjacency matrix. As a result, it is not generally possible to summarize confounding by a simple low-dimensional function of these objects. Our unconfoundedness condition therefore considers units observationally equivalent if they occupy identical positions in the network, meaning that they match on all observed neighborhood and higher-order neighborhood dimensions.

Existing methods that rule out complex forms of interference in selection may result in biased estimates of treatment and spillover effects. For example, consider the causal effect of vaccine adoption on illness. With peer effects in vaccine adoption, vaccinated individuals tend to have more vaccinated direct and indirect social contacts, and a simple comparison of adopters and nonadopters may overstate vaccine

efficacy, even after controlling for neighbors’ covariates.

Because peer effects in selection induce high-dimensional network confounding, they are challenging to accommodate in a nonparametric setting. Most of the literature on interference in observational settings does not allow for such effects, including recent work using instrumental variables rather than unconfoundedness conditions (DiTraglia et al., 2023; Hoshino and Yanagi, 2023a; Kang and Imbens, 2016). Most papers that do allow for selection peer effects rely on semiparametric, game-theoretic models of selection, which substantially reduce the dimensionality of the problem but may be subject to model misspecification (Hoshino and Yanagi, 2023b; Jackson et al., 2020; Kim, 2020; Lin and Vella, 2021).<sup>1</sup>

To account for network confounding, an initial idea might be to apply double machine learning using the lasso for high-dimensional estimation (Chernozhukov et al., 2018). In the SUTVA setting, implementation of the lasso requires the specification of a basis  $\{P_k(X_i)\}_{k=1}^d$  for the unit-level covariates  $X_i$ . In our setting, however, a unit  $i$ ’s “covariates” correspond to its network position  $(i, \mathbf{X}, \mathbf{A})$  and it is unclear how to choose a basis  $\{P_k(i, \mathbf{X}, \mathbf{A})\}_{k=1}^d$  for such an object. Nothing in the standard toolbox for high-dimensional estimation suggests that it is possible to nonparametrically adjust for graph-structured confounders, which is presumably why this has been explicitly avoided in the prior literature.

## 1.1 Contributions

Our first insight is that graph neural networks (GNNs) can be used to construct a flexible basis for graph-structured confounders. The advantage of a neural network architecture is that the basis functions are “learnable,” meaning they are parameterized and estimated from data. Our practical proposal is to estimate treatment and spillover effects using a doubly robust estimator with first-stage nuisance functions approximated by GNNs. Whereas a variety of conventional machine learners can be employed for the first stage in the standard SUTVA setting, none are well suited to our setting, and the observation that GNNs can fill this gap is novel.

Our primary contribution is to provide theoretical justification for the proposed estimation strategy under a fully nonparametric behavioral model that allows for endogenous peer effects in both the outcome and treatment selection stages. We

---

<sup>1</sup>An exception is Balat and Han (2023) who study partial identification of a nonparametric model with strategic complementarities.

utilize the model of approximate neighborhood interference (ANI) proposed by Leung (2022a), which posits that interference in the outcome stage decays with network path distance. Leung shows that ANI allows for endogenous peer effects but focuses on a setting with randomized assignment. In observational settings, it stands to reason that peer effects in selection may be a possibility. We therefore relax his assumption of independent treatment assignment to allow for ANI in treatment selection. Since both stages are simultaneous-equations models, this creates significant new complications in the form of high-dimensional network confounding.

Because the first stage of doubly robust estimation involves high-dimensional nonparametric estimation, theoretical feasibility requires a form of low-dimensional structure. At the same time, GNNs have been found empirically to perform best with shallow architectures, which correspond to relatively low-dimensional parameterizations, in contrast to the deep architectures popular with convolutional neural networks (Alon and Yahav, 2021; Li et al., 2018). Here our key insight is to draw a novel connection between ANI and approximate sparsity conditions in the lasso literature. We argue that this connection justifies the use of shallow GNN architectures in our setting.

To understand the idea, let  $\mathcal{N}(i, K)$  denote  $i$ 's  $K$ -neighborhood, the set of units whose path distance from  $i$  is at most  $K$ , and  $(\mathbf{X}_{\mathcal{N}(i, K)}, \mathbf{A}_{\mathcal{N}(i, K)})$  denote the restriction of  $(\mathbf{X}, \mathbf{A})$  to  $\mathcal{N}(i, K)$ . The key parameter of a GNN is its *depth* or number of layers  $L$ , which determines the *receptive field*  $(\mathbf{X}_{\mathcal{N}(i, L)}, \mathbf{A}_{\mathcal{N}(i, L)})$  used to predict  $i$ 's outcome. For example, a one-layer GNN only uses  $i$ 's 1-neighborhood  $(\mathbf{X}_{\mathcal{N}(i, 1)}, \mathbf{A}_{\mathcal{N}(i, 1)})$  to predict its outcome, rather than the entirety of  $(\mathbf{X}, \mathbf{A})$ . Accordingly, the choice of  $L$  depends on prior information about the function being estimated. If it only depends on the 1-neighborhood of the ego, then  $L = 1$  suffices, whereas if it depends nontrivially on the entirety of  $(\mathbf{X}, \mathbf{A})$ , then this requires a larger choice of  $L$ .

ANI posits that interference decays with distance, so outcomes and treatments are less affected by distant units and primarily determined by  $(\mathbf{X}_{\mathcal{N}(i, L)}, \mathbf{A}_{\mathcal{N}(i, L)})$  for relatively small  $L$ . This is analogous to approximate sparsity, under which the regression function primarily depends on a small subset of regressors. As a result, our first-stage nuisance functions are well approximated by lower-dimensional analogs that only depend on the  $L$ -neighborhood, and these can be directly estimated with shallow  $L$ -layer GNNs. Our formal result provides primitive conditions on interference that rationalize small choices of  $L$  of order  $\log n$ .

We provide conditions under which the doubly robust estimator is approximately normally distributed as the network size grows large. This type of result is well known for i.i.d. data (e.g. [Farrell, 2018](#)), but it is nontrivial to extend to our setting since we allow for a complex form of network dependence. For example, asymptotically linearizing the doubly robust estimator requires a new argument due to dependence, and application of an appropriate CLT requires verification of a high-level weak dependence condition under a nonparametric model with outcome and selection stages both governed by simultaneous-equations models. For inference, we utilize a network HAC estimator due to [Kojevnikov et al. \(2021\)](#) and propose a new bandwidth that adjusts for estimation error in the first-stage machine learners.

We substantiate the theory in a simulation study and empirical application to microfinance diffusion. The simulations demonstrate that the use of GNNs can substantially reduce bias relative to conventional choices of network controls even with shallow architectures. The empirical illustration revisits the microfinance diffusion application of [He and Song \(2024\)](#). We show how our estimands can capture complementary aspects of diffusion relative to their “average diffusion at the margin” measure. Our theoretical framework allows for more complex diffusion processes without requiring the econometrician to prespecify the maximum number of within-period rounds of diffusion. Finally, by including richer controls that account for network confounding, we find more attenuated diffusion effects.

## 1.2 Related Literature

There is a large literature on interference, much of which focuses on randomized control trials (e.g. [Athey et al., 2018](#); [Li and Wager, 2022](#); [Toulis and Kao, 2013](#)). We contribute to a growing recent literature on unconfoundedness, much of which operates in a partial interference setting where units are partitioned into disjoint groups with no interference across groups (e.g. [Liu et al., 2019](#); [Qu et al., 2022](#)).

Studying a network interference setting, [Veitch et al. \(2019\)](#) propose to use “node embeddings” as network controls, which are learned functions of the graph. Since node embeddings can be obtained from a variety of methods, there remains the issue of justifying a particular choice of network controls. GNNs can be interpreted as a method of estimating node embeddings (see §3), and our behavioral model provides justification for their use. We defer to §2.1 a more detailed review of the literature

on network interference and unconfoundedness.

Auerbach (2022) studies identification conditions distinct from unconfoundedness but proposes a related strategy of “matching” on certain network statistics. He provides conditions under which pairwise differencing using unit pairs matched on a novel codegree statistic eliminates selection bias.

Prior to the GNN literature, graph kernels were the dominant method for graph learning tasks (Morris et al., 2021). These are to kernel regression as GNNs are to sieve estimation, so graph kernels require a measure of similarity between regressors, in this case, between two graphs. Auerbach and Tabord-Meehan (2023) propose a graph kernel estimator using a novel similarity measure based on graph isomorphism. Since there is no known algorithm for isomorphism testing with polynomial runtime in the network size (§A.1 discusses GNNs’ relationship to this problem), many graph kernel approaches amount to specifying an “embedding,” a mapping from networks to Euclidean space (Kriege et al., 2020). As noted by Wu et al. (2020), embeddings are predetermined functions of the network, whereas GNNs produce learnable embeddings.

Finally, our paper contributes to recent literature applying neural networks to econometric problems (Athey et al., 2021; Farrell et al., 2021; Kaji et al., 2020). These papers employ neural networks as nonparametric sieve estimators for regression functions. Whereas other machine learners can theoretically work well in their settings, our problem cannot be solved with standard methods. In this respect, our work relates to Pollmann (2023) which studies the problem of constructing counterfactuals for spatial treatments using convolutional neural networks.

The next section presents the model and its relation to the prior literature and defines the estimators. We introduce GNNs in §3 and characterize the asymptotic properties of our estimators in §4. Next, we establish a network analog of approximate sparsity in §5. We report results from a simulation study in §6, and §7 presents an empirical application to microfinance diffusion. Finally, §8 concludes.

We represent an undirected network  $\mathbf{A}$  as an  $n \times n$  binary adjacency matrix with  $ij$ th entry  $A_{ij} \in \{0, 1\}$  representing a link between units  $i$  and  $j$ . We assume no self-links, so  $A_{ii} = 0$ . Let  $\ell_{\mathbf{A}}(i, j)$  denote the *path distance* between  $i, j$  in  $\mathbf{A}$ , defined as the length of the shortest path between them, if one exists, and  $\infty$  otherwise. The  $K$ -neighborhood of a unit  $i$  in  $\mathbf{A}$  is denoted by  $\mathcal{N}(i, K) = \{j \in \mathcal{N}_n : \ell_{\mathbf{A}}(i, j) \leq K\}$  and its size by  $n(i, K) = |\mathcal{N}(i, K)|$ . We refer to the elements of  $\mathcal{N}(i, K) \setminus \{i\}$  for  $K = 1$  as

$i$ 's *neighbors* and the elements of the same set of  $K > 1$  as  $i$ 's *higher-order* neighbors. A unit  $i$ 's *degree* is  $n(i, 1)$ , the number of neighbors.

## 2 Setup

Let  $\mathcal{N}_n = \{1, \dots, n\}$  be the set of units connected through the network  $\mathbf{A}$ . Each unit  $i \in \mathcal{N}_n$  is endowed with unobservables  $(\varepsilon_i, \nu_i) \in \mathbb{R}^{d_\varepsilon} \times \mathbb{R}^{d_\nu}$  and observables  $X_i \in \mathbb{R}^{d_x}$ . The model primitives determine outcomes and treatments according to

$$Y_i = g_n(i, \mathbf{D}, \mathbf{X}, \mathbf{A}, \varepsilon) \quad \text{and} \quad D_i = h_n(i, \mathbf{X}, \mathbf{A}, \nu), \quad (1)$$

respectively, where  $\mathbf{X} = (X_i)_{i=1}^n$  is the matrix with  $i$ th row equal to  $X_i'$ ;  $\mathbf{Y}$ ,  $\mathbf{D}$ ,  $\varepsilon$ , and  $\nu$  are similarly defined; and  $\{(g_n, h_n)\}_{n \in \mathbb{N}}$  is a sequence of function pairs such that each  $g_n(\cdot)$  has range  $\mathbb{R}$  and  $h_n(\cdot)$  has range  $\{0, 1\}$ . The econometrician observes  $(\mathbf{Y}, \mathbf{D}, \mathbf{X}, \mathbf{A})$ . Our analysis treats  $(\mathbf{A}, \mathbf{X}, \varepsilon, \nu)$  as random, but the asymptotic theory in §4 conditions on  $(\mathbf{X}, \mathbf{A})$  to avoid imposing additional assumptions on its dependence structure.<sup>2</sup>

We view the timing of the model as follows. First, nature draws the primitives  $(\mathbf{A}, \mathbf{X}, \varepsilon, \nu)$ . Next, units select into treatment, potentially based on other units' decisions, and  $h_n(\cdot)$  is the reduced-form outcome of that process. Finally,  $g_n(\cdot)$  is the reduced form of the subsequent process that generates outcomes. Because  $g_n(\cdot)$  and  $h_n(\cdot)$  may depend on the primitives of all units, the setup allows  $Y_i$  and  $D_i$  to be outcomes of simultaneous-equations models with endogenous peer effects, as shown in the next examples.

**Example 1** (Linear-in-Means). Consider the outcome model

$$Y_i = \alpha + \beta \frac{\sum_{j=1}^n A_{ij} Y_j}{\sum_{j=1}^n A_{ij}} + \frac{\sum_{j=1}^n A_{ij} Z_j'}{\sum_{j=1}^n A_{ij}} \gamma + Z_i' \delta + \varepsilon_i,$$

where  $Z_i = (D_i, X_i)'$  (Manski, 1993). The coefficient  $\beta$  captures endogenous peer effects, the influence of neighbors' outcomes on own outcomes, while  $\gamma$  captures exogenous peer effects, the influence of neighbors' treatments and covariates. Letting  $\tilde{\mathbf{A}}$

---

<sup>2</sup>A design-based analysis would additionally condition on  $\varepsilon$ . This would generally preclude consistent estimation of the nonparametric functions in the doubly robust estimator defined in §2.3.

denote the row-normalized adjacency matrix and  $\mathbf{1}$  the  $n$ -dimensional vector of ones, if  $\mathbf{A}$  is connected, the reduced form of the model can be written in matrix form as

$$\mathbf{Y} = \frac{\alpha}{1 - \beta} \mathbf{1} + \mathbf{Z}\delta + \gamma\beta \sum_{k=0}^{\infty} \beta^k \tilde{\mathbf{A}}^{k+1} \mathbf{Z} + \sum_{k=0}^{\infty} \beta^k \tilde{\mathbf{A}}^k \boldsymbol{\varepsilon}.$$

This characterizes  $Y_i$  as a function  $g_n(i, \mathbf{D}, \mathbf{X}, \mathbf{A}, \boldsymbol{\varepsilon})$ .

**Example 2** (Binary Game). Consider the binary analog of Example 1 but for selection into treatment:

$$D_i = \mathbf{1} \left\{ \alpha + \beta \frac{\sum_{j=1}^n A_{ij} D_j}{\sum_{j=1}^n A_{ij}} + \frac{\sum_{j=1}^n A_{ij} Z'_j}{\sum_{j=1}^n A_{ij}} \gamma + Z'_i \delta + \nu_i > 0 \right\}. \quad (2)$$

Unlike Example 1, there may exist multiple equilibria. The equilibrium selection mechanism is a reduced-form mapping from the primitives  $(\mathbf{X}, \mathbf{A}, \boldsymbol{\nu})$  to outcomes  $\mathbf{D}$  and therefore characterizes  $D_i$  as a function  $h_n(i, \mathbf{X}, \mathbf{A}, \boldsymbol{\nu})$ . This formulation corresponds to a game of complete information. In a game of incomplete information, as modeled by Xu (2018) for instance, a unit  $i$ 's information set is  $(\nu_i, \mathbf{X}, \mathbf{A})$ . Here an analog of (2) holds with each  $D_j$  replaced with  $\sigma_j(\mathbf{X}, \mathbf{A})$ , the equilibrium belief that  $D_j = 1$ . This characterizes  $D_i$  as a function  $h_n(i, \mathbf{X}, \mathbf{A}, \nu_i)$ .

**Example 3** (Diffusion). He and Song (2024) study the following two-period diffusion model. Let  $D_i$  denote  $i$ 's decision to adopt microfinance in period 0 and  $Y_i$  its decision in period 1. Their equations (2.4) and (3.6) posit that

$$Y_i = g_n(\mathbf{D}_{\mathcal{N}(i,K)}, \varepsilon_i) \quad \text{and} \quad D_i = \mathbf{1}\{W'_i \gamma > \nu_i\},$$

where  $W_i$  is a known function of  $(\mathbf{X}, \mathbf{A})$  and  $K$  is the maximum distance that adoption decisions can diffuse through the network between periods 0 and 1. We provide a more detailed comparison of our models in §7.

Given specification (1), we define potential outcomes as

$$Y_i(\mathbf{d}) = g_n(i, \mathbf{d}, \mathbf{X}, \mathbf{A}, \boldsymbol{\varepsilon}).$$

Confounding may arise first because  $Y_i(\mathbf{d})$  is potentially correlated with  $D_i$  due to



the high-dimensional observables  $(\mathbf{X}, \mathbf{A})$  and second because of dependence between unobservables that drive outcomes  $\varepsilon$  and those that drive selection  $\nu$ . We restrict the second source of confounding.

**Assumption 1** (Unconfoundedness). *For any  $n \in \mathbb{N}$ ,  $\varepsilon \perp\!\!\!\perp \nu \mid \mathbf{X}, \mathbf{A}$ .*

As discussed below, unconfoundedness conditions used in the existing literature additionally limit the first source of confounding to known summary statistics of  $(\mathbf{X}, \mathbf{A})$ . Ours is analogous to standard formulations of unconfoundedness under SUTVA ( $\varepsilon_i \perp\!\!\!\perp \nu_i \mid X_i$ ) since we do not impose further restrictions on the nature of observed confounding.

Because the econometrician only observes a single network, a large-sample theory requires restrictions on interference in order to obtain some form of weak dependence. We next specify a nonparametric model of decaying interference that accommodates the previous examples. For any  $S \subseteq \mathcal{N}_n$ , let  $\mathbf{D}_S = (D_i)_{i \in S}$ , and similarly define  $\mathbf{X}_S$  and other such submatrices. Let  $\mathbf{A}_S = (A_{ij})_{i,j \in S}$  denote the subnetwork of  $\mathbf{A}$  on  $S$ , formally the submatrix of  $\mathbf{A}$  restricted to  $S$ . Recall that  $\mathcal{N}(i, s)$  is the  $s$ -neighborhood of  $i$  in  $\mathbf{A}$ .

**Assumption 2** (ANI). *There exists a sequence of functions  $\{(\gamma_n(\cdot), \eta_n(\cdot))\}_{n \in \mathbb{N}}$  with  $\gamma_n, \eta_n: \mathbb{R}_+ \rightarrow \mathbb{R}_+$  such that  $\sup_{n \in \mathbb{N}} \max\{\gamma_n(s), \eta_n(s)\} \xrightarrow{s \rightarrow \infty} 0$  and, for any  $n \in \mathbb{N}$ ,*

$$\begin{aligned} \max_{i \in \mathcal{N}_n} \mathbf{E} & \left[ |g_n(i, \mathbf{D}, \mathbf{X}, \mathbf{A}, \varepsilon) \right. \\ & \left. - g_{n(i,s)}(i, \mathbf{D}_{\mathcal{N}(i,s)}, \mathbf{X}_{\mathcal{N}(i,s)}, \mathbf{A}_{\mathcal{N}(i,s)}, \varepsilon_{\mathcal{N}(i,s)}) \mid \mathbf{D}, \mathbf{X}, \mathbf{A} \right] \leq \gamma_n(s) \quad a.s. \end{aligned} \quad (3)$$

and

$$\max_{i \in \mathcal{N}_n} \mathbf{E} \left[ |h_n(i, \mathbf{X}, \mathbf{A}, \nu) - h_{n(i,s)}(i, \mathbf{X}_{\mathcal{N}(i,s)}, \mathbf{A}_{\mathcal{N}(i,s)}, \nu_{\mathcal{N}(i,s)}) \mid \mathbf{X}, \mathbf{A} \right] \leq \eta_n(s) \quad a.s. \quad (4)$$

This is analogous to the model of approximate neighborhood interference proposed by [Leung \(2022a\)](#) but imposed on both the outcome and selection models. Whereas  $g_n(i, \dots)$  is unit  $i$ 's realized outcome,  $g_{n(i,s)}(i, \dots)$  is its outcome under a counterfactual “ $s$ -neighborhood model.” In the latter case, we fix all model primitives and treatments at their realized values, drop units outside of  $\mathcal{N}(i, s)$  from the model, and direct the remaining units to interact according to the process  $g_{n(i,s)}(\cdot)$  to produce

counterfactual  $s$ -neighborhood outcomes.<sup>3</sup> The error from approximating the observed outcome with the  $s$ -neighborhood counterfactual is bounded by  $\gamma_n(s)$ , which decays with the neighborhood radius  $s$ . This formalizes the idea that  $Y_i$  is primarily determined by units relatively proximate to  $i$ , so that the further a unit is from  $i$ , the less it influences  $i$ 's outcome. The second equation imposes the analogous requirement on  $D_i$ .

**Example 4.** For the linear-in-means model in Example 1, an argument similar to Proposition 1 of Leung (2022a) shows that (3) holds with  $\sup_n \gamma_n(s) \leq C|\beta|^s$  for some  $C > 0$ . For the binary game in Example 2, an argument similar to Proposition 2 of Leung (2022a) establishes (4) with  $\sup_n \eta_n(s)$  decaying at an exponential rate with  $s$ . Finally, for the He and Song (2024) diffusion model in Example 3,  $Y_i$  only depends on  $\mathbf{D}$  through  $\mathbf{D}_{\mathcal{N}(i,K)}$ , so (3) holds with  $\gamma_n(s) = c \mathbf{1}\{s < K\}$  for some universal constant  $c$ . In their empirical application, they use own covariates as controls, so  $W_i = X_i$ , in which case (4) holds with  $\eta_n(s) = 0$  for all  $s$ .

## 2.1 Related Literature

The standard SUTVA model and unconfoundedness condition correspond to

$$Y_i = g(D_i, X_i, \varepsilon_i) \quad \text{and} \quad \varepsilon_i \perp\!\!\!\perp D_i \mid X_i. \quad (5)$$

To generalize this setup to allow for network interference, the typical approach in the existing literature is as follows. Define

$$T_i = f_n(i, \mathbf{D}, \mathbf{A}) \quad \text{and} \quad W_i = q_n(i, \mathbf{X}, \mathbf{A}) \quad (6)$$

where  $f_n(\cdot)$  and  $q_n(\cdot)$  are known vector-valued functions. The *effective treatment* (Manski, 2013) or *exposure mapping* (Aronow and Samii, 2017)  $T_i$  is a low-dimensional function of the treatment assignment vector. The *network controls*  $W_i$  are low-dimensional functions of the covariates. The literature commonly employs the *neigh-*

---

<sup>3</sup>This formulation of ANI is related to an estimation strategy proposed by Xu (2018) for binary games on networks with incomplete information. His idea is to approximate an agent  $i$ 's strategy in the  $n$ -agent game with its strategy in the counterfactual game restricted to  $i$ 's  $s$ -neighborhood.

*borhood interference* model and unconfoundedness condition

$$Y_i = g(T_i, W_i, \varepsilon_i) \quad \text{and} \quad \varepsilon_i \perp\!\!\!\perp T_i \mid W_i, \quad (7)$$

which is a direct generalization of (5) (Emmenegger et al., 2022; Forastiere et al., 2021; Ogburn et al., 2022). Here  $T_i$  entirely summarizes interference while  $W_i$  summarizes confounding.

Common examples of  $T_i$  and  $W_i$  are

$$T_i = \left( D_i, \sum_{j=1}^n A_{ij} D_j \right) \quad \text{and} \quad W_i = \left( X_i, \sum_{j=1}^n A_{ij}, \frac{\sum_{j=1}^n A_{ij} X_j}{\sum_{j=1}^n A_{ij}} \right). \quad (8)$$

This choice of  $T_i$  implies that  $Y_i$  depends on  $\mathbf{D}$  only through two statistics: own treatment and the number of treated neighbors. Variation in the first component identifies a direct treatment effect and variation in the second a spillover effect. Like most exposure mappings used in the literature, this only depends on  $\mathbf{D}_{\mathcal{N}(i,1)}$ , so the outcome model (7) implies no interference beyond the 1-neighborhood. Likewise, this choice of  $W_i$  implies no confounding beyond 1-neighborhood covariates.

More generally, one could restrict the outcome model to depend only on the  $K$ -neighborhood treatments  $\mathbf{D}_{\mathcal{N}(i,K)}$  for some fixed threshold  $K$ . As shown by Leung (2022a), this rules out economically interesting forms of interference such as endogenous peer effects, which motivates the ANI condition (3). Furthermore, (7) assumes the econometrician can correctly specify the summary statistic  $T_i$  in the outcome model, which may be difficult to justify (Sävje, 2024).

Whereas Leung (2022a) and Sävje (2024) focus on randomized experiments, we study observational data on economic agents that choose to select into treatment. It then becomes important to specify a behavioral model rationalizing the choice of controls  $W_i$ . Sánchez-Becerra (2022) is the first to provide such a model. Under neighborhood interference (7) and an exposure mapping similar to (8), he shows that it is sufficient to set  $W_i = X_i$ , that is, to solely control for own covariates. Since much of the literature utilizes controls such as (8), this raises the question of what model of selection justifies their use or more broadly the use of “network controls” that depend more generally on  $\mathbf{X}$  and  $\mathbf{A}$ .

Our model (1) provides an answer. The presence of complex interference in both the outcome and treatment stages induces selection on  $(\mathbf{X}, \mathbf{A})$ , so that it is generally

insufficient to control only for a simple summary statistic such as (8). Our outcome model is considerably weaker than (7) because we do not require correct specification of a low-dimensional function  $T_i$  of  $(\mathbf{D}, \mathbf{A})$  to restrict interference. Our unconfound- edness condition (Assumption 1) is likewise considerably weaker than (7) because we do not require correct specification of a low-dimensional function  $W_i$  of  $(\mathbf{X}, \mathbf{A})$  to summarize confounding.

## 2.2 Estimand

Following much of the literature, we focus on estimands defined by exposure map- pings, though the core idea of accounting for high-dimensional network confounding using GNNs may potentially be applied to other estimands. Recall from the previous subsection the definition of the exposure mapping  $T_i = f_n(i, \mathbf{D}, \mathbf{A})$ , where  $\{f_n\}_{n \in \mathbb{N}}$  is a sequence of functions each with range  $\mathcal{T}$ , a discrete subset of  $\mathbb{R}^{d_t}$ . Let  $\mathcal{M}_n \subseteq \mathcal{N}_n$  be a subset of the units and  $m_n = |\mathcal{M}_n|$ . We study the estimand

$$\tau(t, t') = \frac{1}{m_n} \sum_{i \in \mathcal{M}_n} (\mathbf{E}[Y_i \mid T_i = t, \mathbf{X}, \mathbf{A}] - \mathbf{E}[Y_i \mid T_i = t', \mathbf{X}, \mathbf{A}])$$

for  $t, t' \in \mathcal{T}$ . This compares average outcomes of units under two different values of the exposure mapping while adjusting for high-dimensional network confounders. The comparison is restricted to a subpopulation  $\mathcal{M}_n$ , the choice of which can be important for ensuring overlap, as discussed below. Depending on the choice of  $f_n(\cdot)$ ,  $t$ , and  $t'$ ,  $\tau(t, t')$  may capture an average treatment or spillover effect, as illustrated in the examples below.

**Example 5.** Let  $T_i = D_i$  and  $\mathcal{M}_n = \mathcal{N}_n$ . Then  $\tau(1, 0)$  compares average outcomes of treated and untreated units using the full network, which measures the direct effect of the treatment.

**Example 6.** Consider the exposure mapping

$$T_i = \left( D_i, \mathbf{1} \left\{ \sum_{j=1}^n A_{ij} D_j > 0 \right\} \right).$$

For  $t = (0, 1)$  and  $t' = (0, 0)$ ,  $\tau(t, t')$  compares the average outcomes of untreated units

with and without at least one treated neighbor, which captures a spillover effect. For  $t = (1, 0)$  and  $t' = (0, 0)$ , it compares the average outcomes of treated and untreated units with no treated neighbors, which captures a treatment effect. For overlap, we need to exclude units with zero degree since a treated neighbor occurs with probability zero for such units. This is accomplished by choosing  $\mathcal{M}_n$  to be the subset of units whose degree  $n(i, 1) \equiv |\mathcal{N}(i, 1)|$  lies in some desired set excluding zero. That is, choose some  $\Gamma \subseteq \mathbb{R}_+ \setminus \{0\}$  and

$$\mathcal{M}_n = \{i \in \mathcal{N}_n : n(i, 1) \in \Gamma\}. \quad (9)$$

**Example 7.** We obtain a more granular version of Example 6 by setting

$$T_i = \left( D_i, \sum_{j=1}^n A_{ij} D_j \right) \quad (10)$$

and  $\mathcal{M}_n$  according to (9) with  $\Gamma = \{\gamma\}$  for some  $\gamma \in \mathbb{N}$ . If we choose  $\gamma = 3$ ,  $t = (0, 2)$  and  $t' = (0, 0)$ , then  $\tau(t, t')$  takes the subpopulation of untreated units with degree three units and compares those with two versus zero treated neighbors. For this choice of  $t, t'$ , it is important to choose  $\gamma \geq 2$  for overlap since otherwise  $T_i = t$  would be a zero-probability event.

Our large-sample results pertain to the following subpopulations and exposure mappings, which include the previous examples.

**Assumption 3** (Exposure Mappings). *Let  $\mathcal{M}_n$  be given by (9) for some possibly unbounded interval  $\Gamma \subseteq \mathbb{R}_+$ . For any  $t \in \mathcal{T}$ , there exist  $d \in \{0, 1\}$  and a possibly unbounded interval  $\Delta \subseteq \Gamma$  such that*

$$\mathbf{1}\{T_i = t\} = \mathbf{1} \left\{ D_i = d, \sum_{j=1}^n A_{ij} D_j \in \Delta \right\}.$$

In Example 6 for  $t = (0, 1)$ , this holds for  $d = 0$ ,  $\Delta = (0, \infty)$ , and  $\Gamma$  given in the example. In Example 7 with  $t = (0, 2)$ , this holds for  $d = 0$ ,  $\Delta = [1.5, 2.5]$ , and  $\Gamma = [2.5, 3.5]$ .

We restrict to this class of mappings for two reasons. First, it includes the most widely used examples in the literature, which are those presented above. Second,  $D_i$

can be a complex function of  $(\mathbf{X}, \mathbf{A}, \boldsymbol{\nu})$ , and both  $T_i$  and  $Y_i$  can be complex functions of  $\mathbf{D}$ , which makes it difficult to characterize the dependence structure necessary for the application of a central limit theorem without additional structure.

Identification results in [Leung \(2024\)](#) provide conditions under which  $\tau(t, t')$  has a causal interpretation. The focus of our paper is estimation, so we provide only a brief discussion here. Under the neighborhood interference model (7),  $\tau(t, t')$  has a transparent causal interpretation. In settings where (7) fails to hold, [Leung \(2024\)](#) shows that  $\tau(t, t')$  retains a causal interpretation under restrictions on interference either in potential outcomes or selection into treatment. For example, suppose treatment adoption follows a nonparametric game of incomplete information where  $\nu_i$  is unit  $i$ 's private information, so that  $D_i = h_n(i, \mathbf{X}, \mathbf{A}, \nu_i)$  (see Example 2). If private information is independent across units conditional on  $(\mathbf{X}, \mathbf{A})$ , as is typically assumed in structural analyses of the model (e.g. [Lin and Vella, 2021](#); [Xu, 2018](#)), then by Theorem 1 of [Leung \(2024\)](#)  $\tau(t, t')$  can be written as a non-negatively weighted average of certain unit-level effects.

Returning to the vaccine adoption example in §1, recall that, due to peer effects, adoption decisions are potentially correlated across units even after adjusting for low-dimensional network controls (6). However, under the model above, they are independent if we fully control for  $(\mathbf{X}, \mathbf{A})$ , and as a result,  $\tau(t, t')$  can identify a causal effect. The remainder of this paper develops an estimation theory for  $\tau(t, t')$  with high-dimensional network controls.

### 2.3 Estimator

Define the generalized propensity score ([Imbens, 2000](#)) and outcome regression, respectively, as

$$p_t(i, \mathbf{X}, \mathbf{A}) = \mathbf{P}(T_i = t \mid \mathbf{X}, \mathbf{A}) \quad \text{and} \quad \mu_t(i, \mathbf{X}, \mathbf{A}) = \mathbf{E}[Y_i \mid T_i = t, \mathbf{X}, \mathbf{A}]. \quad (11)$$

Let  $\hat{p}_t(i, \mathbf{X}, \mathbf{A})$  and  $\hat{\mu}_t(i, \mathbf{X}, \mathbf{A})$  denote their respective GNN estimators, defined at the end of §3.2. We use a standard doubly robust estimator for multi-valued treatments

$$\hat{\tau}(t, t') = \frac{1}{m_n} \sum_{i \in \mathcal{M}_n} \hat{\tau}_i(t, t'),$$

where

$$\hat{\tau}_i(t, t') = \frac{\mathbf{1}\{T_i = t\}(Y_i - \hat{\mu}_t(i, \mathbf{X}, \mathbf{A}))}{\hat{p}_t(i, \mathbf{X}, \mathbf{A})} + \hat{\mu}_t(i, \mathbf{X}, \mathbf{A}) - \frac{\mathbf{1}\{T_i = t'\}(Y_i - \hat{\mu}_{t'}(i, \mathbf{X}, \mathbf{A}))}{\hat{p}_{t'}(i, \mathbf{X}, \mathbf{A})} - \hat{\mu}_{t'}(i, \mathbf{X}, \mathbf{A}).$$

To estimate the asymptotic variance, we use the network HAC estimator

$$\hat{\sigma}^2 = \frac{1}{m_n} \sum_{i \in \mathcal{M}_n} \sum_{j \in \mathcal{M}_n} (\hat{\tau}_i(t, t') - \hat{\tau}(t, t'))(\hat{\tau}_j(t, t') - \hat{\tau}(t, t')) \mathbf{1}\{\ell_{\mathbf{A}}(i, j) \leq b_n\}$$

(Kojevnikov et al., 2021). The particular choice of a uniform kernel is discussed after Theorem 2 in §4. We propose the bandwidth

$$b_n = \lceil \tilde{b}_n \rceil \quad \text{for} \quad \tilde{b}_n = \begin{cases} \frac{1}{4} \mathcal{L}(\mathbf{A}) & \text{if } \mathcal{L}(\mathbf{A}) < 2 \frac{\log n}{\log \delta(\mathbf{A})}, \\ \mathcal{L}(\mathbf{A})^{1/4} & \text{otherwise,} \end{cases} \quad (12)$$

where  $\lceil \cdot \rceil$  rounds up to the nearest integer,  $\delta(\mathbf{A}) = n^{-1} \sum_{i,j} A_{ij}$  is the average degree, and  $\mathcal{L}(\mathbf{A})$  is the average path length.<sup>4</sup> This is similar to the proposal of Leung (2022a) but with constants adjusted to account for the first-stage estimates. In §B.2 of the appendix, we verify high-level assumptions needed to characterize the asymptotic properties of  $\hat{\sigma}^2$  (see Assumption 7) under the choice of bandwidth (12).

### 3 Graph Neural Networks

Consider the problem of estimating a nonparametric function of network position  $f(i, \mathbf{X}, \mathbf{A})$ . A GNN estimator for  $f(\cdot)$  is a parameterized function that maps  $(\mathbf{X}, \mathbf{A})$  to a vector of estimates,  $(\hat{f}(i, \mathbf{X}, \mathbf{A}))_{i=1}^n$ . In §3.1 we define the standard GNN architecture. In §3.2, we define GNN estimators for the nuisance functions in (11). As we will discuss, GNN architectures impose a nonparametric shape restriction on  $\hat{f}$  called *permutation invariance* whose economic content we study in §3.3.

---

<sup>4</sup>We assume  $\delta(\mathbf{A}) > 1$ , as is typical in practice. By the average path length, we mean the average over all unit pairs in the largest component of  $\mathbf{A}$ . A component is a connected subnetwork such that all units in the subnetwork have infinite path distance to non-members of the subnetwork.

### 3.1 Architecture

The standard GNN architecture consists of nested, parameterized, vector-valued functions called *neurons* that are arranged in  $L$  layers with  $n$  neurons per layer. Let  $h_i^{(l)}$  denote the  $i$ th neuron in layer  $l$ , which may be interpreted as unit  $i$ 's *node embedding*, a Euclidean representation of its network position. As we progress to higher-order layers, say  $h_i^{(l)}$  to  $h_i^{(l+1)}$ ,  $i$ 's embedding becomes richer in a sense discussed below. This idea is common to most modern neural network architectures, that of modeling some complex object as a Euclidean vector with learnable parameters, be it a node's network position (GNNs), a subregion of an image (CNNs), or the meaning of a word or sentence (transformers).

Connections between neurons in different layers are determined by  $\mathbf{A}$  through the following "message-passing" architecture. For layers  $l = 1, \dots, L$ ,

$$h_i^{(l)} = \Phi_{0l} \left( h_i^{(l-1)}, \Phi_{1l} \left( h_i^{(l-1)}, \{h_j^{(l-1)} : A_{ij} = 1\} \right) \right), \quad (13)$$

where  $\Phi_{0l}(\cdot), \Phi_{1l}(\cdot)$  are parameterized, vector-valued functions, examples of which are provided below. We initialize the process with  $h_i^{(0)} = X_i$  at the input layer, meaning that the initial node embedding incorporates no network information. In subsequent layers, unit  $i$ 's node embedding is a function of its 1-neighborhood's embeddings in the previous layer and therefore incorporates increasingly more network information as  $l$  increases. Neurons in the "hidden layers"  $l = 1, \dots, L - 1$  typically have the same dimension, and the final output is  $(h_i^{(L)})_{i=1}^n \in \mathbb{R}^n$ . We next highlight some important properties.

1. The second argument of the *aggregation function*  $\Phi_{1l}(\cdot)$  is the "multiset" (a set with possibly repeating elements) consisting of the node embeddings of the ego's 1-neighborhood. Because multisets are by definition unordered, the labels of the units are immaterial, so the output of each layer is *permutation invariant* in the sense that the output remains the same under any labeling of the units.
2. The *depth*  $L$  of a GNN determines the  $L$ -neighborhood  $(\mathbf{X}_{\mathcal{N}(i,L)}, \mathbf{A}_{\mathcal{N}(i,L)})$  used to predict  $Y_i$ . To see this, it helps to understand why a GNN layer (13) is often referred to as a "round of message passing." In this metaphor,  $h_j^{(l)}$  is the information, or message, held by unit  $j$  at step  $l$  of the process. Messages are successively diffused to neighbors of  $j$  at the next step  $l + 1$  and neighbors of



neighbors at step  $l+2$ , etc., since at each step, each unit aggregates the messages of its neighbors. This is reminiscent of DeGroot learning (DeGroot, 1974) but with more general aggregation functions. Since we initialize the process at  $h_i^{(0)} = X_i$ , the final output  $h_i^{(L)}$  is only a function of  $(\mathbf{X}_{\mathcal{N}(i,L)}, \mathbf{A}_{\mathcal{N}(i,L)})$ .

3. Both  $\Phi_{0l}(\cdot)$  and  $\Phi_{1l}(\cdot)$  depend on the dimension of covariates  $d_x$  but not the number of units. Accordingly, for any fixed architecture, a GNN can take as input a network of any size, provided unit-level covariates are of the same dimension  $d_x$ . The dimensionality of the parameter space is determined not by the size of the input network but rather by  $L$  and the number of parameters in each of the parameterized functions  $\Phi_{0l}(\cdot), \Phi_{1l}(\cdot)$ .

The choices of  $\Phi_{0l}(\cdot), \Phi_{1l}(\cdot)$  define different GNN architectures, two of which we discuss next.

**Example 8 (GIN).** Theoretical results on GNNs commonly pertain to the “graph isomorphism network” architecture

$$h_i^{(l)} = \phi_{0l} \left( h_i^{(l-1)}, \sum_{j=1}^n A_{ij} \phi_{1l}(h_j^{(l-1)}) \right),$$

where  $\phi_{0l}(\cdot), \phi_{1l}(\cdot)$  are nonparametric sieves such as multilayer perceptrons (MLPs). The use of sum aggregation in the second argument is motivated by the key insight that any injective function  $\Phi_{1l}(S)$  of a multiset  $S$  can be written as  $g(\sum_{s \in S} f(s))$  for some functions  $f, g$  when  $X_i$  has countable support (Xu et al., 2018). By approximating the unknown  $f$  and  $g$  with sieves, this architecture can approximate a large nonparametric function class, as discussed in §A.1.

**Example 9 (PNA).** Our simulations and empirical application utilize the “principal neighborhood aggregation” architecture due to Corso et al. (2020), which generalizes many available architectures by using multiple aggregation functions:

$$h_i^{(l)} = \phi_{0l} \left( h_i^{(l-1)}, \Gamma(\{\phi_{1l}(h_i^{(l-1)}), h_j^{(l-1)} : A_{ij} = 1\}) \right),$$

where  $\phi_{0l}(\cdot), \phi_{1l}(\cdot)$  are sieves such as MLPs and  $\Gamma(\cdot)$  is a possibly vector-valued function. The theoretical motivation is that the representation in Example 8 using sum

aggregation no longer holds when the support of  $X_i$  is uncountable, so using multiple aggregators can result in more powerful architectures (Corso et al., 2020).

For an example of  $\Gamma(\cdot)$ , let  $\mu(\cdot)$ ,  $\sigma(\cdot)$ ,  $\Sigma(\cdot)$ ,  $\min(\cdot)$ , and  $\max(\cdot)$  be respectively the mean, standard deviation, sum, min, and max functions, defined component-wise for multisets of vectors. Then setting  $\Gamma(\cdot) = \Gamma_1(\cdot)$  for

$$\Gamma_1(\cdot) = \left( \mu(\cdot) \quad \sigma(\cdot) \quad \Sigma(\cdot) \quad \min(\cdot) \quad \max(\cdot) \right)$$

results in an architecture utilizing five aggregation functions.

The authors combine multiple aggregators with “degree scalars” that multiply each aggregation function by a function of the size of the multiset input  $n(i, 1)$ . The simplest example is the identity scalar, which maps any multiset to unity. This trivially multiplies each aggregation function in  $\Gamma_1(\cdot)$  above, but it is useful to consider non-identity scalars. Let  $|\cdot|$  be the function that takes as input a multiset and outputs its size. Corso et al. (2020) define logarithmic amplification and attenuation scalars

$$S(\cdot, \alpha) = \left( \frac{\log(|\cdot| + 1)}{\delta} \right)^\alpha, \quad \delta = \frac{1}{n} \sum_{i=1}^n \log \left( \sum_{j=1}^n A_{ij} + 1 \right), \quad \alpha \in [-1, 1].$$

The choice of  $\alpha$  defines whether the scalar “amplifies” ( $\alpha = 1$ ) or “attenuates” ( $\alpha = -1$ ) the aggregation function, and  $\alpha = 0$  is the identity scalar. The purpose of the logarithm is to prevent small changes in degree from amplifying gradients in an exponential manner with each successive GNN layer. Thus, an aggregation function that augments  $\Gamma_1(\cdot)$  with logarithmic amplification and attenuation is

$$\Gamma_2(\cdot) = \left( S(\cdot, 0) \quad S(\cdot, 1) \quad S(\cdot, -1) \right) \otimes \Gamma_1(\cdot),$$

where  $\otimes$  denotes the tensor product, resulting in 15 aggregation functions.

### 3.2 Estimator

Let  $\mathcal{F}_{\text{GNN}}(L)$  denote the set of all GNNs with  $L$  layers ranging over all possible functions  $\Phi_{0l}(\cdot)$ ,  $\Phi_{1l}(\cdot)$  for  $l = 1, \dots, L$  within some function class (see Examples 8 and 9). For any  $f \in \mathcal{F}_{\text{GNN}}(L)$ , we let  $f(i, \mathbf{X}, \mathbf{A})$  denote its  $i$ th component, which corresponds to  $h_i^{(L)}$ . A GNN estimator is a function in this set that minimizes a loss

function  $\ell(\cdot)$ :

$$\hat{f}_{\text{GNN}} \in \operatorname{argmin}_{f \in \mathcal{F}_{\text{GNN}}(L)} \sum_{i \in \mathcal{M}_n} \ell(Y_i, f(i, \mathbf{X}, \mathbf{A})). \quad (14)$$

For  $Y_i \in \mathbb{R}$ , we may use squared-error loss

$$\ell(Y_i, f(i, \mathbf{X}, \mathbf{A})) = 0.5(Y_i - f(i, \mathbf{X}, \mathbf{A}))^2,$$

in which case  $\hat{f}_{\text{GNN}}(\mathbf{X}, \mathbf{A})$  estimates  $f^*(\mathbf{X}, \mathbf{A}) = (\mathbf{E}[Y_i | \mathbf{X}, \mathbf{A}])_{i=1}^n$ . For  $Y_i \in \{0, 1\}$ , we may use the logistic loss

$$\ell(Y_i, f(i, \mathbf{X}, \mathbf{A})) = -Y_i f(i, \mathbf{X}, \mathbf{A}) + \log(1 + \exp(f(i, \mathbf{X}, \mathbf{A}))),$$

in which case  $\hat{f}_{\text{GNN}}(\mathbf{X}, \mathbf{A})$  estimates the log odds  $f^*(\mathbf{X}, \mathbf{A}) = (\log(\mathbf{E}[Y_i | \mathbf{X}, \mathbf{A}] / (1 - \mathbf{E}[Y_i | \mathbf{X}, \mathbf{A}])))_{i=1}^n$ .

Returning to the doubly robust estimator in §2.3, to estimate the outcome regression with  $\mathbb{R}$ -valued outcomes, we restrict the sum in (14) to the set of units  $i$  for which  $T_i = t$  and use squared-error loss to obtain  $\hat{\mu}_t(i, \mathbf{X}, \mathbf{A}) = \hat{f}_{\text{GNN}}(i, \mathbf{X}, \mathbf{A})$ . To estimate the generalized propensity score, we replace  $Y_i$  in (14) with  $\mathbf{1}\{T_i = t\}$  and use logistic loss to obtain

$$\hat{p}_t(i, \mathbf{X}, \mathbf{A}) = \frac{\exp(\hat{f}_{\text{GNN}}(i, \mathbf{X}, \mathbf{A}))}{1 + \exp(\hat{f}_{\text{GNN}}(i, \mathbf{X}, \mathbf{A}))}.$$

### 3.3 Invariance

Modern neural network architectures often incorporate prior information in the form of input symmetries to reduce the dimensionality of the parameter space (Bronstein et al., 2021). Convolutional neural networks (CNNs), widely used in image recognition, process grid-structured inputs and impose translation invariance. GNNs process graph-structured inputs and impose permutation invariance.

Define a *permutation* as a bijection  $\pi: \mathcal{N}_n \rightarrow \mathcal{N}_n$ . Abusing notation, we write  $\pi(\mathbf{X}) = (X_{\pi(i)})_{i=1}^n$ , which permutes the rows of matrix  $\mathbf{X}$  according to  $\pi$ , and similarly define  $\pi(\mathbf{D})$  and permutations of other such arrays. Likewise, we write  $\pi(\mathbf{A}) = (A_{\pi(i)\pi(j)})_{i,j \in \mathcal{N}_n}$ , which permutes the rows and columns of the matrix  $\mathbf{A}$ . We say a function  $f(i, \mathbf{X}, \mathbf{A})$  is (*permutation-*)*invariant* if for any bijection  $\pi: \mathcal{N}_n \rightarrow \mathcal{N}_n$  and input  $(i, \mathbf{X}, \mathbf{A})$ , we have  $f(i, \mathbf{X}, \mathbf{A}) = f(\pi(i), \pi(\mathbf{X}), \pi(\mathbf{A}))$ .

As discussed in §3.1, any  $f \in \mathcal{F}_{\text{GNN}}(L)$  is invariant, so by using GNNs to estimate the nuisance functions (11), we are implicitly requiring that they are invariant functions. We next argue that this is an extremely weak and natural requirement.

First consider that we have thus far imposed few restrictions on the distribution of primitives, so the nuisance functions in (11) may depend arbitrarily on the unit label  $i$ . Computing  $\hat{\tau}(t, t')$  would then require estimating, for example,  $n$  distinct propensity score functions  $(p_t(i, \mathbf{X}, \mathbf{A}))_{i=1}^n$ , which is infeasible. The literature on neighborhood interference (7) avoids this problem by imposing the additional restriction that  $p_t(i, \mathbf{X}, \mathbf{A}) = p(W_i)$  for some function  $p(\cdot)$  that does not depend on  $i$ . That is, units are observationally equivalent if they have identical controls  $W_i$ , and observationally equivalent units have identical probabilities of being assigned to an exposure mapping realization of  $t$ .

In our setting,  $W_i$  is insufficient to account for confounding. The natural generalization is to require assignment probabilities to be equivalent for units that have *isomorphic network positions*. That is,  $p_t(i, \mathbf{X}, \mathbf{A}) = p_t(j, \mathbf{X}, \mathbf{A})$  if for some permutation  $\pi$ ,  $(j, \mathbf{X}, \mathbf{A}) = (\pi(i), \pi(\mathbf{X}), \pi(\mathbf{A}))$ . This is exactly what it means for the propensity score to be invariant.

Invariance is substantially weaker than the restriction employed under neighborhood interference. To see this, consider Figure 1, where each unit  $i$  has a binary covariate  $X_i$  that is an indicator for its color being gray, and let  $W_i$  be given as in (8). Then  $W_4 = W_5$ , but units 4 and 5 are not isomorphic (they would have been had units 2 and 3 been unlinked). Whereas the literature requires units 4 and 5 to have identical propensity scores, our invariance condition does not.

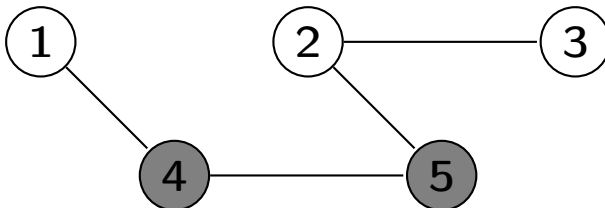


Figure 1: Units 4 and 5 are not isomorphic.

If the propensity score is invariant, this reduces our problem from estimating  $n$  separate scores to estimating only one because, for any  $i$ , there exists a permutation  $\pi_i$  (in particular the one that only permutes units 1 and  $i$ ) such that  $p_t(i, \mathbf{X}, \mathbf{A}) = p_t(1, \pi_i(\mathbf{X}), \pi_i(\mathbf{A}))$  and similarly for  $\mu_t(\cdot)$ . The right-hand side is a function  $p_t(1, \cdot, \cdot)$

that does not depend on  $i$ , so evaluating  $i$ 's propensity score is now only a matter of supplying the correct  $i$ -specific inputs  $(\pi_i(\mathbf{X}), \pi_i(\mathbf{A}))$ .

We close this section with a result demonstrating that invariance is an extremely weak requirement. In particular, it holds under minimal exchangeability conditions on the structural primitives.

**Proposition 1.** *Suppose for any  $n \in \mathbb{N}$  and permutation  $\pi$ ,*

$$\begin{aligned} f_n(i, \mathbf{D}, \mathbf{A}) &= f_n(\pi(i), \pi(\mathbf{D}), \pi(\mathbf{A})), \\ g_n(i, \mathbf{D}, \mathbf{X}, \mathbf{A}, \boldsymbol{\varepsilon}) &= g_n(\pi(i), \pi(\mathbf{D}), \pi(\mathbf{X}), \pi(\mathbf{A}), \pi(\boldsymbol{\varepsilon})), \quad \text{and} \\ h_n(i, \mathbf{X}, \mathbf{A}, \boldsymbol{\nu}) &= h_n(\pi(i), \pi(\mathbf{X}), \pi(\mathbf{A}), \pi(\boldsymbol{\nu})) \end{aligned}$$

*a.s., and  $(\mathbf{A}, \mathbf{X}, \boldsymbol{\varepsilon}, \boldsymbol{\nu}) \stackrel{d}{=} (\pi(\mathbf{A}), \pi(\mathbf{X}), \pi(\boldsymbol{\varepsilon}), \pi(\boldsymbol{\nu}))$ . Then for any  $t \in \mathcal{T}$ ,  $p_t(\cdot)$  and  $\mu_t(\cdot)$  in (11) are invariant functions.*

PROOF. See §D. ■

The first three equations impose invariance on  $(f_n, g_n, h_n)$ . Applied to  $f_n$ , this is a restriction on the choice of exposure mapping. It is satisfied by most, if not all, such mappings used in the literature, including those satisfying Assumption 3. Applied to  $(g_n, h_n)$ , invariance only says that unit identities do not influence behavior beyond the model primitives  $(\mathbf{A}, \mathbf{X}, \boldsymbol{\varepsilon}, \boldsymbol{\nu})$ , which is the case for virtually all models used in the literature. The final requirement says that these model primitives are themselves distributionally invariant, which is a weak condition in superpopulation settings since unit labels carry no intrinsic meaning.

## 4 Asymptotic Theory

We next characterize the asymptotic properties of  $\hat{\tau}(t, t')$  and  $\hat{\sigma}^2$  under a sequence of models sending  $n \rightarrow \infty$ . Along this sequence, the functions  $(f_n, g_n, h_n)$  may obviously vary, as may the distribution of the model primitives  $(\mathbf{A}, \mathbf{X}, \boldsymbol{\varepsilon}, \boldsymbol{\nu})$ , subject to the

conditions imposed below. Define

$$\begin{aligned} \varphi_{t,t'}(i) = & \frac{\mathbf{1}\{T_i = t\}(Y_i - \mu_t(i, \mathbf{X}, \mathbf{A}))}{p_t(i, \mathbf{X}, \mathbf{A})} + \mu_t(i, \mathbf{X}, \mathbf{A}) \\ & - \frac{\mathbf{1}\{T_i = t'\}(Y_i - \mu_{t'}(i, \mathbf{X}, \mathbf{A}))}{p_{t'}(i, \mathbf{X}, \mathbf{A})} - \mu_{t'}(i, \mathbf{X}, \mathbf{A}) - \tau(t, t'), \end{aligned}$$

whose average over  $i \in \mathcal{M}_n$  is the doubly robust moment condition, and let

$$\sigma_n^2 = \text{Var} \left( \frac{1}{\sqrt{m_n}} \sum_{i \in \mathcal{M}_n} \varphi_{t,t'}(i) \middle| \mathbf{X}, \mathbf{A} \right).$$

**Assumption 4** (Moments). (a) *There exists  $M < \infty$  and  $p > 4$  such that for any  $n \in \mathbb{N}$ ,  $i \in \mathcal{N}_n$ , and  $\mathbf{d} \in \{0, 1\}^n$ ,  $\mathbf{E}[|Y_i(\mathbf{d})|^p \mid \mathbf{X}, \mathbf{A}] < M$  a.s.* (b) *There exists  $[\underline{\pi}, \bar{\pi}] \subset (0, 1)$  such that  $\hat{p}_t(i, \mathbf{X}, \mathbf{A}), p_t(i, \mathbf{X}, \mathbf{A}) \in [\underline{\pi}, \bar{\pi}]$  and  $m_n/n \geq \underline{\pi}$  a.s. for all  $n \in \mathbb{N}$ ,  $i \in \mathcal{M}_n$ ,  $t \in \mathcal{T}$ .* (c)  *$\liminf_{n \rightarrow \infty} \sigma_n^2 > 0$  a.s.*

Part (b) requires sufficient overlap for the propensity scores. Under Assumption 3, this holds if  $\Gamma$  is a bounded set. Part (b) further imposes overlap on the propensity score estimator, which is common in the double machine learning literature (e.g. Chernozhukov et al., 2018; Farrell, 2018; Farrell et al., 2021). It also requires that  $\mathcal{M}_n$  is a nontrivial subset of  $\mathcal{N}_n$ . Part (c) is a standard non-degeneracy condition.

**Assumption 5** (GNN Rates). *For any  $t \in \mathcal{T}$ , both  $m_n^{-1} \sum_{i \in \mathcal{M}_n} (\hat{p}_t(i, \mathbf{X}, \mathbf{A}) - p_t(i, \mathbf{X}, \mathbf{A}))^2$  and  $m_n^{-1} \sum_{i \in \mathcal{M}_n} (\hat{\mu}_t(i, \mathbf{X}, \mathbf{A}) - \mu_t(i, \mathbf{X}, \mathbf{A}))^2$  are  $o_p(1)$ , their product is  $o_p(n^{-1})$ , and  $m_n^{-1} \sum_{i \in \mathcal{M}_n} (\hat{\mu}_t(i, \mathbf{X}, \mathbf{A}) - \mu_t(i, \mathbf{X}, \mathbf{A}))(1 - \mathbf{1}\{T_i = t\}p_t(i, \mathbf{X}, \mathbf{A}))^{-1} = o_p(n^{-1/2})$ .*

These are standard conditions (e.g. Assumption 3 of Farrell, 2018) for machine learners. Farrell et al. (2021) provide primitive conditions for MLPs under i.i.d. data. Theoretical properties of GNNs are the subject of a very recent field of research, and to our knowledge, the literature lacks several key intermediate results required for deriving primitive conditions, especially under network dependence. In §5, we obtain primitive conditions for a network analog of approximate sparsity, which shows that the effective dimension of the estimation problem is low. In §A.1 of the appendix, we reframe and combine several theoretical results in the GNN literature to show that GNNs can approximate functions in a large nonparametric class.

The next assumption is used to show that  $\{\varphi_{t,t'}(i)\}_{i=1}^n$  is  $\psi$ -dependent (see Definition C.1, which is due to [Kojevnikov et al., 2021](#)) to apply a central limit theorem. It imposes restrictions on the rate at which a certain dependence measure decays relative to the growth rate of network neighborhoods. Define

$$\mathcal{N}^\partial(i, s) = \{j \in \mathcal{N}_n : \ell(i, j) = s\} \quad \text{and} \quad \delta_n^\partial(s; k) = \frac{1}{n} \sum_{i=1}^n |\mathcal{N}^\partial(i, s)|^k,$$

respectively  $i$ 's  $s$ -neighborhood boundary and the  $k$ th moment of the  $s$ -neighborhood boundary size. Let

$$\begin{aligned} \Delta_n(s, m; k) &= \frac{1}{n} \sum_{i=1}^n \max_{j \in \mathcal{N}^\partial(i, s)} |\mathcal{N}(i, m) \setminus \mathcal{N}(j, s-1)|^k, \\ c_n(s, m; k) &= \inf_{\alpha > 1} \Delta_n(s, m; k\alpha)^{1/\alpha} \delta_n^\partial(s; \alpha/(\alpha-1))^{1-1/\alpha}, \quad \text{and} \\ \psi_n(s) &= \max_{i \in \mathcal{N}_n} (\gamma_n(s/4) + \eta_n(s/4) (1 + n(i, 1) + \Lambda_n(i, s/4) n(i, s/4))), \end{aligned} \quad (15)$$

where  $\Lambda_n(i, s/4)$  is a constant defined in the next assumption. The second quantity measures network density. The third bounds the covariance between  $\varphi_{t,t'}(i)$  and  $\varphi_{t,t'}(j)$  when  $\ell_{\mathbf{A}}(i, j) \leq s$ . Lastly, define

$$G_n(i, \mathbf{d}_{\mathcal{N}(i,s)}) = \mathbf{E}[g_n(i,s)(i, \mathbf{d}_{\mathcal{N}(i,s)}, \mathbf{X}_{\mathcal{N}(i,s)}, \mathbf{A}_{\mathcal{N}(i,s)}, \boldsymbol{\varepsilon}_{\mathcal{N}(i,s)}) \mid \mathbf{X}, \mathbf{A}].$$

**Assumption 6** (Weak Dependence). (a)  $\{(\varepsilon_i, \nu_i)\}_{i=1}^n$  is independently distributed conditional on  $(\mathbf{X}, \mathbf{A})$ . (b) For any  $n \in \mathbb{N}$ ,  $i \in \mathcal{N}_n$ ,  $s \geq 0$ , and  $\mathbf{d}, \mathbf{d}' \in \{0, 1\}^n$ ,

$$|G_n(i, \mathbf{d}_{\mathcal{N}(i,s)}) - G_n(i, \mathbf{d}'_{\mathcal{N}(i,s)})| \leq \Lambda_n(i, s) \sum_{j \in \mathcal{N}(i,s)} |d_j - d'_j| \quad a.s.$$

for some constant  $\Lambda_n(i, s)$  that may depend on  $(\mathbf{X}, \mathbf{A})$ . (c)  $\sup_{n \in \mathbb{N}} \max_{s \geq 1} \psi_n(s) < \infty$  a.s. (d) For  $p$  in Assumption 4(a), some positive sequence  $v_n \rightarrow \infty$  and any  $k \in \{1, 2\}$ ,

$$\begin{aligned} \frac{1}{n^{k/2}} \sum_{s=0}^{\infty} c_n(s, v_n; k) \psi_n(s)^{1-(2+k)/p} \rightarrow 0, \quad n^{3/2} \psi_n(v_n)^{1-1/p} \rightarrow 0, \quad \text{and} \\ \limsup_{n \rightarrow \infty} \sum_{s=0}^{\infty} \delta_n^\partial(s; 2)^{1/2} \gamma_n(s/2)^{1-2/p} < \infty \quad a.s. \end{aligned} \quad (16)$$

Parts (a) and (b) are used to establish that  $\{\varphi_{t,t'}(i)\}_{i=1}^n$  is  $\psi$ -dependent. Part (b) is a Lipschitz condition that holds if potential outcomes are uniformly bounded. In particular we can take  $\Lambda_n(s) = 2M$  where  $M$  is the uniform bound on the ranges of  $\{g_n\}_{n \in \mathbb{N}}$ . Part (a) facilitates the task of verifying  $\psi$ -dependence given that treatments are complex functions of unobservables and  $Y_i$  and  $T_i$  are complex functions of treatments. It can be relaxed to  $\psi$ -dependence under additional smoothness conditions on  $g_n, h_n$  (Kojevnikov et al., 2021, Proposition 2.5). The simulation study in §6 provides evidence that our methods can perform well when unobservables exhibit network dependence.

Perhaps the most substantive requirement is (d), which regulates the asymptotic behavior of three quantities in (16). The first two correspond to Condition ND of Kojevnikov et al. (2021), which they utilize to establish a CLT. The third is similar and used to asymptotically linearize the doubly robust estimator under network dependence. We illustrate how to verify (16) in §B.1.

**Theorem 1.** *Under Assumptions 1–6,*

$$\sigma_n^{-1/2} \sqrt{m_n} (\hat{\tau}(t, t') - \tau(t, t')) \xrightarrow{d} \mathcal{N}(0, 1).$$

PROOF. See §D. ■

The next result characterizes the asymptotic properties of  $\hat{\sigma}^2$ . Similar to the design-based setting of Leung (2022a), it is not guaranteed to be consistent due to conditioning on  $(\mathbf{X}, \mathbf{A})$ . However, as in that setting, we can make the case that it is typically asymptotically conservative. Define

$$\mathcal{J}_n(s, m) = \{(i, j, k, l) \in \mathcal{N}_n^4 : k \in \mathcal{N}(i, m), l \in \mathcal{N}(j, m), \ell_{\mathbf{A}}(i, j) = s\}.$$

**Assumption 7** (HAC). (a) For some  $M > 0$  and all  $n \in \mathbb{N}$ ,  $i \in \mathcal{N}_n$ , and  $t \in \mathcal{T}$ ,  $|\max\{Y_i, \hat{\mu}_t(i, \mathbf{X}, \mathbf{A})\}| < M$  a.s. (b)  $m_n^{-1} \sum_{i \in \mathcal{M}_n} (\hat{p}_t(i, \mathbf{X}, \mathbf{A}) - p_t(i, \mathbf{X}, \mathbf{A}))^2$  and  $m_n^{-1} \sum_{i \in \mathcal{M}_n} (\hat{\mu}_t(i, \mathbf{X}, \mathbf{A}) - \mu_t(i, \mathbf{X}, \mathbf{A}))^2$  are  $o_p(n^{-1/2})$ . (c) For some  $\epsilon \in (0, 1)$  and  $b_n \rightarrow \infty$ ,  $\lim_{n \rightarrow \infty} n^{-1} \sum_{s=0}^{\infty} c_n(s, b_n; 2) \psi_n(s)^{1-\epsilon} = 0$  a.s. (d)  $n^{-1} \sum_{i=1}^n n(i, b_n) = o_p(\sqrt{n})$ . (e)  $n^{-1} \sum_{i=1}^n n(i, b_n)^2 = O_p(\sqrt{n})$ . (f)  $\sum_{s=0}^n \mathcal{J}_n(s, b_n) |\psi_n(s)| = o(n^2)$ .

Part (a) strengthens Assumption 4(a) to uniformly bounded outcomes. Part (b)



strengthens Assumption 5(b) but only mildly since we nonparametrically estimate both nuisance functions. Since it does not require uniform convergence, it is more readily verifiable for machine learning estimators. Part (c) is Assumption 4.1(iii) of [Kojevnikov et al. \(2021\)](#). Parts (d)–(f) correspond to Assumptions 7(b)–(d) of [Leung \(2022a\)](#), which are used to characterize the bias of the variance estimator. We discuss verification of (c)–(f) in §B.2; the derivations there show that (f) is closely related to (c).

**Theorem 2.** *Define  $\tilde{\varphi}_{t,t'}(i)$  by replacing  $\tau(t, t')$  in the definition of  $\varphi_{t,t'}(i)$  with  $\tau_i(t, t') = \mathbf{E}[Y_i \mid T_i = t, \mathbf{X}, \mathbf{A}] - \mathbf{E}[Y_i \mid T_i = t', \mathbf{X}, \mathbf{A}]$ . Let*

$$\begin{aligned} \hat{\sigma}_*^2 &= \frac{1}{m_n} \sum_{i \in \mathcal{M}_n} \sum_{j \in \mathcal{M}_n} \tilde{\varphi}_{t,t'}(i) \tilde{\varphi}_{t,t'}(j) \mathbf{1}\{\ell_{\mathbf{A}}(i, j) \leq b_n\} \quad \text{and} \\ R_n &= \frac{1}{m_n} \sum_{i \in \mathcal{M}_n} \sum_{j \in \mathcal{M}_n} (\tau_i(t, t') - \tau(t, t')) (\tau_j(t, t') - \tau(t, t')) \mathbf{1}\{\ell_{\mathbf{A}}(i, j) \leq b_n\}. \end{aligned}$$

*Under Assumption 7 and the assumptions of Theorem 1,*

$$\hat{\sigma}^2 = \hat{\sigma}_*^2 + R_n + o_p(1) \quad \text{and} \quad |\hat{\sigma}_*^2 - \sigma_n^2| \xrightarrow{p} 0.$$

PROOF. See §D. ■

This extends Proposition 4.1 of [Kojevnikov et al. \(2021\)](#) and Theorem 4 of [Leung \(2022a\)](#) to accommodate first-stage estimators. As in the latter theorem,  $\hat{\sigma}^2$  is asymptotically biased by  $R_n$ .

Note that the HAC estimator  $\hat{\sigma}^2$  uses the uniform kernel  $\mathbf{1}\{\ell_{\mathbf{A}}(i, j) \leq b_n\}$ . [Leung \(2019\)](#) first observed that if a positive semidefinite kernel were used instead, then  $R_n \geq 0$ , so that  $\hat{\sigma}^2$  would be asymptotically conservative without any further conditions. However, all available positive semidefinite kernels are sloped, and in simulations in past work, we have found that the uniform kernel controls size substantially better than all sloped alternatives because it does not downweight the covariances of units near the ego. This is why we recommend its use even though it does not guarantee positive semidefiniteness without further conditions.

Observe that  $R_n$  is a HAC estimate of the variance of the unit-level contrasts  $\tau_i(t, t')$ . It should therefore well approximate  $\text{Var}(m_n^{-1/2} \sum_{i \in \mathcal{M}_n} \tau_i(t, t')) \geq 0$ , in which

case  $\hat{\sigma}^2$  would be asymptotically conservative. This can be formalized under additional weak dependence conditions on the superpopulation as in §A of [Leung \(2022b\)](#).

## 5 Approximate Sparsity

As discussed in §3, the number of layers  $L$  in a GNN determines its *receptive field*, the neighborhood  $(\mathbf{X}_{\mathcal{N}(i,L)}, \mathbf{A}_{\mathcal{N}(i,L)})$  used to construct  $i$ 's estimate. The choice of  $L$  depends on prior information about the function being estimated, in our case assumptions about interference. In practice, it is common to choose a small value, which results in a receptive field that excludes most of the network. This has been found to achieve better predictive performance than deep architectures with large  $L$  (see §6 and §A.2 for further discussion). In this section, we establish a network analog of approximate sparsity, which provides low-dimensional structure that justifies the use of shallow architectures and makes estimation of the nuisance functions feasible.

Recalling the architecture definition in (13), let  $d_{kl}$  be the number of parameters in  $\Phi_{kl}(\cdot)$  for  $k \in \{0, 1\}$ , so that  $\sum_{l=1}^L (d_{0l} + d_{1l})$  is the number of parameters in any element of  $\mathcal{F}_{\text{GNN}}(L)$  used to approximate a target nuisance function.

**Definition 1.** Let  $L = L_n$  be a possibly diverging sequence of GNN depths. *Network approximate sparsity* holds if the following conditions are satisfied for any  $t \in \mathcal{T}$ .

(a) The error from approximating the high-dimensional propensity score with its  $L$ -neighborhood analog is small:

$$\frac{1}{m_n} \sum_{i \in \mathcal{M}_n} (p_t(i, \mathbf{X}, \mathbf{A}) - p_t(i, \mathbf{X}_{\mathcal{N}(i,L)}, \mathbf{A}_{\mathcal{N}(i,L)}))^2 = o_p(n^{-1/2}) \quad (17)$$

and similarly for the outcome regression  $\mu_t(\cdot)$ . (b)  $\mathcal{F}_{\text{GNN}}(L)$  is low-dimensional in that  $\sum_{l=1}^L (d_{0l} + d_{1l}) = o(\sqrt{n})$ .

### 5.1 Lasso Analogy

To motivate the definition, we draw an analogy to approximate sparsity conditions in the lasso literature. Let  $h(X_i) = \mathbf{E}[Y_i | X_i]$ , and consider a lasso regression of  $Y_i$  on a vector of basis functions  $P(X_i)$ . For the lasso prediction  $P(X_i)' \hat{\beta}$  to be a good

estimate of  $h(X_i)$ , we require

$$\frac{1}{n} \sum_{i=1}^n (P(X_i)' \hat{\beta} - h(X_i))^2 = o_p(n^{-1/2}). \quad (18)$$

To verify this, it is common to impose approximate sparsity, which consists of two conditions (e.g. Belloni et al., 2014).

(a) There exists  $\beta$  such that  $n^{-1} \sum_{i=1}^n (P(X_i)' \beta - h(X_i))^2 = o_p(n^{-1/2})$ .

(b)  $\|\beta\|_0 = o(\sqrt{n})$ .

Definition 1 mirrors these conditions, which posit that  $h(\cdot)$  has an approximation  $P(\cdot)' \beta$  that can be estimated with a relatively low-dimensional regression.

**Example 10.** Suppose  $h(X_i) = \sum_{j=1}^{\infty} P_j(X_i) \theta_j$  with  $|\theta_j| \xrightarrow{j \rightarrow \infty} 0$ . That is, one can order the regressors  $P_1(X_i), \dots, P_m(X_i)$  such that their corresponding true regression coefficients decay to zero. Then the outcome depends primarily on the first few regressors despite  $m$  being potentially high-dimensional. This satisfies (a) and (b) above given a sufficiently quick rate of decay (Belloni et al., 2014, §4.1.2).

The main idea in our setting is that the dependence of  $Y_i$  and  $D_i$  on other units decays with network distance under ANI (Assumption 2). That is, these quantities primarily depend on  $(\mathbf{X}_{\mathcal{N}(i,L)}, \mathbf{A}_{\mathcal{N}(i,L)})$  for some small radius  $L$ , which is analogous to Example 10. We may then approximate  $p_t(i, \mathbf{X}, \mathbf{A})$  with the lower-dimensional estimand  $p_t(i, \mathbf{X}_{\mathcal{N}(i,L)}, \mathbf{A}_{\mathcal{N}(i,L)})$ , which we can directly estimate with an  $L$ -layer GNN.

## 5.2 Primitive Conditions

We next provide primitive conditions for network approximate sparsity. The primary condition is ANI, but this alone is insufficient to show  $p_t(i, \mathbf{X}, \mathbf{A}) \approx p_t(i, \mathbf{X}_{\mathcal{N}(i,L)}, \mathbf{A}_{\mathcal{N}(i,L)})$ . The latter drops from the conditioning event the subnetwork external to the  $L$ -neighborhood, which requires a form of conditional independence. While ANI says that outcomes and treatments primarily depend on units in a relatively small neighborhood, their primitives may be correlated with those of units far from the neighborhood. We consider it reasonable to assume, in the spirit of ANI, that this correlation instead decays with distance, which is the substance of the next condition.

**Assumption 8** (Approximate CI). *There exist a sequence of functions  $\{\lambda_n(\cdot)\}_{n \in \mathbb{N}}$  with  $\lambda_n: \mathbb{R}_+ \rightarrow \mathbb{R}_+$  and a linear function  $r_\lambda: \mathbb{R}_+ \rightarrow \mathbb{R}_+$  such that  $\sup_{n \in \mathbb{N}} \lambda_n(s) \xrightarrow{s \rightarrow \infty} 0$ ,  $r_\lambda(s) \geq s$  for all  $s \in \mathbb{R}_+$ , and*

$$|\mathbf{E}[f(\boldsymbol{\varepsilon}_{\mathcal{N}(i,s)}, \boldsymbol{\nu}_{\mathcal{N}(i,s)}) \mid \mathbf{X}, \mathbf{A}] - \mathbf{E}[f(\boldsymbol{\varepsilon}_{\mathcal{N}(i,r_\lambda(s))}, \boldsymbol{\nu}_{\mathcal{N}(i,r_\lambda(s))}) \mid \mathbf{X}_{\mathcal{N}(i,r_\lambda(s))}, \mathbf{A}_{\mathcal{N}(i,r_\lambda(s))})]| \leq \lambda_n(s)$$

*a.s. for any  $n \in \mathbb{N}$ ,  $i \in \mathcal{N}_n$ ,  $s \geq 0$ , and  $\mathbb{R}$ -valued, bounded, measurable function  $f(\cdot)$ .*

This is perhaps simplest to understand in the case where  $r_\lambda$  is the identity function. Then the assumption requires that the unobservables of an  $s$ -neighborhood are approximately conditionally independent of the network outside of this neighborhood, where the approximation error is shrinking with the radius  $s$ . More generally, we can allow the  $s$ -neighborhood to be approximately conditionally independent of the network outside the greater  $r_\lambda(s)$ -neighborhood for  $r_\lambda(s) \geq s$ .

**Example 11.** Suppose there exist  $K \geq 0$ , a vector-valued function  $H(\cdot)$ , and a random vector  $\mathbf{U}$  independent of  $(\mathbf{X}, \mathbf{A})$  such that  $(\varepsilon_i, \nu_i) = H(\mathbf{U}, \mathbf{X}_{\mathcal{N}(i,K)}, \mathbf{A}_{\mathcal{N}(i,K)})$  for all  $i$ . Then Assumption 8 holds with  $r_\lambda(s) = s + K$  and  $\lambda_n(s) = 0$  for all  $s$ . Here unobservables all depend on the same common shock  $\mathbf{U}$ , which serves to illustrate that they need not be conditionally independent across units to satisfy the assumption. The unobservables depend on the observables only through the ego's  $K$ -neighborhood, so for example,  $\varepsilon_i$  could be larger if  $i$ 's  $K$ -neighborhood contains more units of a certain type, which induces observed confounding.

**Example 12.** Suppose  $\varepsilon_i = v_{n,1}(i, \mathbf{X}, \mathbf{A}, \mathbf{U})$  and  $\nu_i = v_{n,2}(i, \mathbf{X}, \mathbf{A}, \mathbf{U})$  for  $\mathbf{U} = (U_i)_{i=1}^n$ , an array of (possibly dependent) unit-level shocks that are independent of  $(\mathbf{X}, \mathbf{A})$ . Unlike, the previous example, we have high-dimensional confounding. Similar to the models for  $Y_i$  and  $D_i$ , both  $\varepsilon_i$  and  $\nu_i$  can be the outputs of network autoregressive models such as Example 1 or game-theoretic models such as Example 2. We can reparameterize model (1) as  $Y_i = g_n(i, \mathbf{D}, \mathbf{X}, \mathbf{A}, \boldsymbol{\varepsilon}) \equiv \tilde{g}_n(i, \mathbf{D}, \mathbf{X}, \mathbf{A}, \mathbf{U})$  and similarly  $D_i = \tilde{h}_n(i, \mathbf{X}, \mathbf{A}, \mathbf{U})$ . Now suppose ANI holds for the reparameterized model, that is, replacing  $(g_n, h_n)$  with  $(\tilde{g}_n, \tilde{h}_n)$  and  $\boldsymbol{\varepsilon}$  and  $\boldsymbol{\nu}$  with  $\mathbf{U}$  in the statement of Assumption 2. This jointly imposes a version of ANI on the outcome, selection, and random shock models  $v_{n,1}, v_{n,2}$ . Since  $\mathbf{U} \perp\!\!\!\perp (\mathbf{X}, \mathbf{A})$  by supposition, Assumption 8 holds for the reparameterized model with  $r_\lambda(s) = s$  and  $\lambda_n(s) = 0$  for all  $s$ .

**Example 13.** Consider a dyadic network formation model analogous to the one used by [Sánchez-Becerra \(2022\)](#) where  $A_{ij} = \mathbf{1}\{V(X_i, X_j, \zeta_{ij}) > 0\}$  for  $\mathbb{R}$ -valued  $V(\cdot)$  and  $\{\zeta_{ij}\}_{i < j}$  is a set of i.i.d. random variables independent of  $(\mathbf{X}, \boldsymbol{\varepsilon}, \boldsymbol{\nu})$ . Suppose there exists a function  $H(\cdot)$  and unit-level shocks  $\mathbf{U} = (U_i)_{i=1}^n$  independent of all other primitives such that  $(\varepsilon_i, \nu_i) = H(U_i, X_i)$ . Then Assumption 8 holds with  $r_\lambda(s) = s$  and  $\lambda_n(s) = 0$  for all  $s$ . Since Assumption 8 only requires approximate independence, it may be possible to verify when links are weakly dependent as in some models of strategic network formation (e.g. [Leung and Moon, 2023](#)).

We now state our main result, which provides primitive conditions for Definition 1(a). Notably, it sets  $L$  to be order  $\log n$ , which quantifies the sense in which we can allow for shallow GNN architectures.

**Theorem 3.** *Suppose Assumptions 2, 3, and 8 hold,  $\sup_n \max\{\lambda_n(s), \gamma_n(s), \eta_n(s)\} = O(e^{-\alpha s})$  as  $s \rightarrow \infty$  for some  $\alpha > 0$ , and  $n^{-1} \sum_{i=1}^n n(i, 1)^2 = O_p(1)$ . Then (17) holds if*

$$L = r_\lambda(((4 - \epsilon)\alpha)^{-1} \log n + 1)$$

for some  $\epsilon \in (0, 4)$ . Further suppose  $\sup_{n \in \mathbb{N}} n^{-1} \sum_{i=1}^n n(i, s)^2 = O_p(e^{\xi s})$  as  $s \rightarrow \infty$  for some  $\xi < \alpha$ . Under Assumptions 1, 4(b), and 7(a), the analog of (17) holds for  $\mu_t(\cdot)$  if instead  $L = r_\lambda(((2 - \epsilon)(\alpha - \xi))^{-1} \log n + 1)$  for some  $\epsilon \in (0, 2)$ .

PROOF. See §D. ■

The specifications of  $L$  given in the theorem are not feasible, being dependent on unknowns  $r_\lambda(\cdot)$  and  $\alpha$ . This is similar to how finite-sample bounds for the lasso require restrictions on the penalty parameter involving unknown constants. In §6, we illustrate the performance of different choices of  $L$  in simulations.

The first half of the theorem concerns the propensity score, and the assumptions are simple to verify. First, it requires exponential decay of the interference bounds in Assumption 2, which holds in Example 4. Second, real-world networks are typically sparse, usually formalized as  $n^{-1} \sum_{i=1}^n n(i, 1) = O_p(1)$ , which the theorem mildly strengthens to a second-moment condition. The second half of the theorem concerns  $\mu_t(\cdot)$ . The requirement  $n^{-1} \sum_{i=1}^n n(i, s)^2 = O_p(e^{\xi s})$  for  $\xi < \alpha$  says  $s$ -neighborhoods grow at a slower rate  $\xi$  than interference  $\alpha$  decays. The same type of condition is

required for a central limit theorem, as discussed in §4 and §B.

Our final result provides primitive conditions for Definition 1(b).

**Proposition 2.** *Suppose the GNN architecture is given by either Example 8 or 9, and we choose  $L = O(\log n)$  GNN layers. For some  $\kappa < 1/4$ , suppose for each layer  $l$  and  $k \in \{0, 1\}$  that  $\phi_{kl}(\cdot)$  is an MLP with width  $O(n^\kappa \log^2 n)$  and depth  $O(\log n)$ , uniformly in  $l$ . Then Definition 1(b) holds.*

PROOF. The rate conditions on the MLP widths and depths correspond to those of Farrell et al. (2021). These choices result in order  $\rho_n = n^{2\kappa} \log^5 n$  parameters for each MLP uniformly across  $l$  (Farrell et al., 2021, p. 187). Then the number of GNN parameters is  $\sum_{l=1}^L (d_{0l} + d_{1l}) = O(L\rho_n)$ , which is  $o(\sqrt{n})$  given that  $\kappa < 1/4$  and  $L = O(\log n)$ . ■

## 6 Simulation Study

We design a monte carlo study to serve three purposes. The first is to illustrate the finite-sample properties of our proposed estimators for different choices of  $L$ . The second is to compare the performance of GNNs to that of commonly used prespecified controls based on model (7). The third is to demonstrate that shallow GNNs can perform well even on “wide” networks that ordinarily would require many layers in the absence of an approximate sparsity result.

### 6.1 Design

We simulate  $\mathbf{A}$  from two random graph models. The random geometric graph model sets  $A_{ij} = \mathbf{1}\{\|\rho_i - \rho_j\| \leq r_n\}$  for  $\{\rho_i\}_{i=1}^n \stackrel{iid}{\sim} \mathcal{U}([0, 1]^2)$  and  $r_n = (5/(\pi n))^{1/2}$ , where  $\pi$  is the transcendental number. The Erdős-Rényi model sets  $A_{ij} \stackrel{iid}{\sim} \text{Ber}(5/n)$ . Both have limiting average degree equal to five. The former model results in “wide” networks with high average path lengths that grow at a polynomial rate with  $n$ , while the latter results in low average path lengths of  $\log n$  order. For  $n = 2000$ , the average path length is about 39.5 for random geometric graphs and 4.9 for Erdős-Rényi graphs.

Independent of  $\mathbf{A}$ , we draw  $\{\varepsilon_i\}_{i=1}^n \stackrel{iid}{\sim} \mathcal{N}(0, 1)$ ,  $\{\nu_i\}_{i=1}^n \stackrel{iid}{\sim} \mathcal{N}(0, 1)$ , and  $\{X_i\}_{i=1}^n \stackrel{iid}{\sim} \mathcal{U}(\{0, 0.25, 0.5, 0.75, 1\})$ , with all three mutually independent. For some vectors  $\mathbf{W} =$

$(W_i)_{i=1}^n$  and  $\boldsymbol{\nu} = (\nu_i)_{i=1}^n$ , define

$$V_i(\mathbf{W}, \boldsymbol{\nu}; \theta) = \alpha + \beta \frac{\sum_{j=1}^n A_{ij} W_j}{\sum_{j=1}^n A_{ij}} + \delta \frac{\sum_{j=1}^n A_{ij} X_j}{\sum_{j=1}^n A_{ij}} + \gamma X_j + \nu_i + \frac{\sum_{j=1}^n A_{ij} \nu_j}{\sum_{j=1}^n A_{ij}}$$

where  $\theta = (\alpha, \beta, \delta, \gamma)$ . We generate  $\{Y_i\}_{i=1}^n$  from the linear-in-means model, where  $Y_i = V_i(\mathbf{Y}, \boldsymbol{\varepsilon}; \theta_y)$  and  $\theta_y = (0.5, 0.8, 10, -1)$ . We generate  $\{D_i\}_{i=1}^n$  according to Example 2, so that  $D_i = \mathbf{1}\{V_i(\mathbf{D}, \boldsymbol{\nu}; \theta_d) > 0\}$  with  $\theta_d = (-0.5, 1.5, 1, -1)$ . The equilibrium selection mechanism is myopic best-response dynamics starting from the initial condition  $\{D_i^0\}_{i=1}^n$  for  $D_i^0 = \mathbf{1}\{V_i(\mathbf{0}, \boldsymbol{\nu}; \theta_d) > 0\}$ .

The design induces a greater degree of dependence than what our assumptions allow. The error term  $\nu_i + \sum_{j=1}^n A_{ij} \nu_j / \sum_{j=1}^n A_{ij}$  is not conditionally independent across units unlike what Assumption 6(a) requires. Also, back-of-the-envelope calculations indicate that peer effects are sufficiently large in magnitude that Assumption 6(d) is violated.

We use the estimand in Example 5 whose true value is zero. About 57 percent of units select into treatment, so the effective sample size used to estimate the outcome regressions is around  $n/2$  since  $\mathbf{E}[Y_i \mid T_i = t, \mathbf{X}, \mathbf{A}]$  is estimated only with observations for which  $T_i = t$ . We report results for  $n = 1000, 2000, 4000$ .

## 6.2 Nonparametric Estimators

The GNNs use the PNA architecture in Example 9 with aggregator  $\Gamma_2(\cdot)$  defined in the example and  $L = 1, 2, 3$ . Both  $\phi_{0l}$  and  $\phi_{1l}$  are one-layer MLPs with width  $H = 1, 3, 5$ . We optimize the GNNs using the popular Adam variant of stochastic gradient descent with the default PyTorch implementation (Paszke et al., 2019) using random initial parameter values and learning rate 0.01.

For  $\phi_{1l}(\cdot)$ , we use a linear layer (no activation function), which is the default for the PNAConv class in the PyTorch Geometric package (Fey and Lenssen, 2019). That is,  $\phi_{1l}(x) = (\alpha_{l,h,1} + x' \beta_{l,h,1})_{h=1}^H$ , where  $\alpha_{l,h,1}$  is a scalar and  $\beta_{l,h,1}$  a vector. For  $\phi_{0l}(\cdot)$  with  $l < L$ , we use ReLU activation, so  $\phi_{0l}(x) = (\sigma(\alpha_{l,h,0} + x' \beta_{l,h,0}))_{h=1}^H$  for  $\sigma(x) = \max\{x, 0\}$ . Finally,  $\phi_{0L}(\cdot)$  is similar except we use linear activation (replacing  $\sigma$  with the identity function) since it is the output layer.

We compare GNNs to nonparametric estimators using the prespecified controls given in (8), which are analogous to those used in the simulations of Emmenegger et al.

(2022) and Forastiere et al. (2021). For these, we estimate the nuisance functions using GLMs (logistic and linear regression) with polynomial sieves of order 1, 2, or 3. Recall that a GNN with  $L = 1$  corresponds to a receptive field that only encompasses the ego’s 1-neighborhood. This is the same as the implied receptive field of the GLM estimators.

Table 1: Simulation results for random geometric graph

	$L = 1$			$L = 2$			$L = 3$		
	$n$	$n$	$n$	$n$	$n$	$n$	$n$	$n$	$n$
$n$	1000	2000	4000	1000	2000	4000	1000	2000	4000
# treated	567	1137	2277	567	1137	2277	567	1137	2277
$H$	1	3	5	1	3	5	1	3	5
$\hat{\tau}(1, 0)$	0.0783	0.0753	0.0680	0.0937	0.0382	0.0226	0.1288	0.0712	0.0353
CI	0.9316	0.9332	0.9324	0.9318	0.9368	0.9464	0.9360	0.9286	0.9384
SE	0.4279	0.3057	0.2166	0.5134	0.2961	0.2037	0.5745	0.3143	0.2021
Oracle CI	0.9426	0.9434	0.9358	0.9450	0.9498	0.9572	0.9464	0.9420	0.9472
Oracle SE	0.4473	0.3180	0.2190	0.5507	0.3153	0.2116	0.5994	0.3369	0.2094
$W \hat{\tau}(1, 0)$	0.1800	0.1701	0.1555	0.1597	0.1484	0.1356	0.1249	0.1211	0.1116
$W$ CI	0.9160	0.9042	0.8906	0.9200	0.9136	0.9056	0.9174	0.9140	0.9114
$W$ SE	0.4338	0.3082	0.2177	0.4311	0.3072	0.2175	0.4182	0.2998	0.2132
IID CI	0.6968	0.6818	0.6862	0.6688	0.6704	0.6926	0.6658	0.6638	0.6822
IID SE	0.2363	0.1667	0.1174	0.2711	0.1567	0.1078	0.3015	0.1656	0.1063

5k simulations. The estimand is  $\tau(1, 0) = 0$ . “# treated”  $\approx$  effective sample size for GNN regression estimators. GNN depth is  $L$ , and MLP width is  $H$ . Rows beginning with “ $W$ ” use GLMs with prespecified controls and polynomial sieves of order  $L$  in place of GNNs. “CI” rows display the empirical coverage of 95% CIs.

### 6.3 Results

Tables 1 and 2 report the results of 5000 simulations for the random geometric graph and Erdős-Rényi models, respectively. Row “ $\hat{\tau}(1, 0)$ ” reports the average of our estimates, and their absolute values also equal the bias since  $\tau(1, 0) = 0$ . Row “CI” shows the coverage of our CIs using the HAC estimator. The “ $W$ ” rows report estimates using GLMs with polynomial sieves where  $L$  is the order of the polynomial. The “Oracle” rows correspond to true standard errors, computed by taking the standard deviation of  $\hat{\tau}(1, 0)$  across simulation draws. The “IID” rows report i.i.d. standard errors, which illustrate the degree of dependence.

We first compare the GNN estimators with the GLM estimators in the “ $W$ ” rows. The bias of the latter is larger for all polynomial orders, often more than twice the



Table 2: Simulation results for Erdős-Rényi graph

	$L = 1$			$L = 2$			$L = 3$		
	1000	2000	4000	1000	2000	4000	1000	2000	4000
$n$									
# treated	593	1187	2372	593	1187	2372	593	1187	2372
$H$	1	3	5	1	3	5	1	3	5
$\hat{\tau}(1, 0)$	0.0294	0.0366	0.0354	0.0503	0.0244	0.0191	0.0688	0.0443	0.0300
CI	0.9326	0.9276	0.9292	0.9274	0.9298	0.9322	0.9230	0.9170	0.9142
SE	0.1867	0.1336	0.0954	0.2126	0.1318	0.0918	0.2313	0.1388	0.0928
Oracle CI	0.9592	0.9458	0.9402	0.9418	0.9492	0.9472	0.9388	0.9404	0.9374
Oracle SE	0.2072	0.1399	0.0996	0.2291	0.1410	0.0976	0.2472	0.1506	0.0999
$W \hat{\tau}(1, 0)$	0.1310	0.1372	0.1367	0.1111	0.1168	0.1162	0.0810	0.0920	0.0936
$W$ CI	0.8954	0.8376	0.7240	0.9038	0.8584	0.7842	0.9044	0.8774	0.8356
$W$ SE	0.1993	0.1420	0.1012	0.1957	0.1400	0.0999	0.1873	0.1353	0.0978
IID CI	0.8098	0.7972	0.7828	0.8012	0.7996	0.7920	0.7968	0.7768	0.7760
IID SE	0.1324	0.0936	0.0664	0.1504	0.0918	0.0634	0.1640	0.0973	0.0644

5k simulations. The estimand is  $\tau(1, 0) = 0$ . “# treated”  $\approx$  effective sample size for GNN regression estimators. GNN depth is  $L$ , and MLP width is  $H$ . Rows beginning with “ $W$ ” use GLMs with prespecified controls and polynomial sieves of order  $L$  in place of GNNs. “CI” rows display the empirical coverage of 95% CIs.

bias of the GNN estimates. This is the case even for  $L = 1$ , which corresponds to the same receptive field as the GLMs. It suggests that GNNs learn a different function of  $(\mathbf{X}, \mathbf{A})$  than  $W_i$ , one that apparently better adjusts for confounding. The improvement in bias using GNNs does not come at an apparent cost to variance.

Second, we compare the GNN estimators across different choices of  $L$ . The best performance is achieved with  $L = 2$ , which results in low bias. This is the case for both random graph models and is particularly notable for the random geometric graph because its width is substantially larger than the radius of the receptive field when  $L = 2$ . This demonstrates that we can achieve good performance despite only controlling for  $(\mathbf{X}_{\mathcal{N}(i,2)}, \mathbf{A}_{\mathcal{N}(i,2)})$ , which is possible due to approximate sparsity. Our CIs exhibit some undercoverage, which is not unusual for HAC estimators, but coverage tends to the nominal level as  $n$  grows for  $L = 2$ . The oracle CIs achieve coverage close to the nominal level across most sample sizes and architectures, which illustrates the quality of the normal approximation.

Unsurprisingly,  $L = 2$  outperforms  $L = 1$  since the latter only adjusts for 1-neighborhood confounding. In principle,  $L = 3$  accounts for higher-order network confounds, but the bias turns out to be slightly larger and the coverage worse, though the performance still dominates that of GLMs with large enough samples.

A choice of  $L = 2$  is not unusual in the literature. [Zhou et al. \(2021\)](#) compute the prediction error of GNNs on several different datasets with  $L = 2, 4, 8, \dots$  and find that  $L = 2$  has the best performance across several architectures. The fact that GNN performance often fails to improve (and indeed can worsen) with larger  $L$  is well known in the GNN literature, and we survey different explanations in §A.2.<sup>5</sup>

## 7 Empirical Application

We revisit the analysis of [He and Song \(2024\)](#) of the diffusion of microfinance through rural villages in Karnataka, India. They utilize a dataset due to [Banerjee et al. \(2013\)](#) which contains twelve dimensions of social relationships, demographic details, and microfinance adoption decisions from 43 villages involved with Bharatha Swamukti Samsthe’s (BSS) microfinance program in 2007. BSS initiated the program by meeting with a select group of village “leaders” who were asked to spread the word about microfinance.

Following the analysis of [He and Song \(2024\)](#), the unit of observation is the household, and household observables  $X_i$  are (a) the normalized total number of households within each village and indicators for (b) participation in self-help groups, (c) savings activities, and (d) caste composition. They construct three social networks from the multigraph data:  $G_{ee}$  represents connections through material exchanges like borrowing or lending essentials,  $G_{sc}$  captures social activities including advice sharing or joint religious attendance, and  $G_{all}$  is the union of  $G_{ee}$  and  $G_{sc}$ . We report results for **A** set to each one of these options.

We consider three different definitions of the treatment. In the “leader case,”  $D_i$  is an indicator for whether household  $i$  has a leader. In the “leader-adopter case,” it is an indicator for whether the household has a leader who adopts microfinance in the first trimester of the study. These cases are due to [He and Song \(2024\)](#). In the “adopter case,”  $D_i$  is simply an indicator for whether any household member adopts microfinance in the first trimester. The outcome  $Y_i$  is an indicator for whether an individual in household  $i$  adopts microfinance starting in the first trimester of the

---

<sup>5</sup>[Bronstein \(2020\)](#) writes, “Significant efforts have recently been dedicated to coping with the problem of depth in graph neural networks, in hope to [sic] achieve better performance and perhaps avoid embarrassment in using the term ‘deep learning’ when referring to graph neural networks with just two layers.”

study or later.<sup>6</sup>

## 7.1 Comparison with He and Song (2024)

We next discuss causal estimands in the context of the adopter case. [He and Song \(2024\)](#) propose a novel estimand called “average diffusion at the margin” (ADM). This is the average effect of a unit’s initial adoption decision  $D_i$  on neighbors’ subsequent adoption decisions  $Y_j$  for  $j \in \mathcal{N}(i, 1)$ . To identify the ADM, they assume the following. First, initial adoption decisions are unconfounded (their Assumption 2.1), and in the application, they use household observables  $X_i$  as the controls. Second, their selection model, as described in Example 3, is a parametric single-agent discrete choice model. Third, they assume adoption decisions are irreversible in that  $Y_i \geq D_i$ , which is true in the application. Finally, as discussed in Example 3, the econometrician must specify the maximum number of rounds of diffusion that take place between the measurement of  $D_i$  and  $Y_i$ , and they choose  $K = 1$  in the application.

We use a richer set of network controls that includes covariates of higher-order neighbors, without assuming a known function  $W_i$  as in (7). We employ a nonparametric selection model allowing for peer effects in initial adoption. Our outcome model (1) also allows for peer effects in subsequent adoption, as well as higher-order diffusion beyond the  $K = 1$  neighborhood since outcomes depend on the entire initial adoption vector  $\mathbf{D}$ . ANI posits that this dependence decays with distance, which is a feature of most diffusion models, as information from distant units is less likely to diffuse to the ego. Note that ANI does not require knowledge of the number of within-period rounds of diffusion. Finally, we do not require  $Y_i \geq D_i$  since our methodology applies more broadly to non-binary outcomes.

The cost of imposing less structure than [He and Song \(2024\)](#) is that the ADM may not be identified under our assumptions. Instead, we consider two estimands  $\tau(t, t')$  defined by the following exposure mappings:

$$T_i^{(1)} = \begin{cases} 1 & \text{if } \sum_{j=1}^n A_{ij} D_j = 1 \\ 0 & \text{otherwise,} \end{cases} \quad \text{and} \quad T_i^{(2)} = \begin{cases} 1 & \text{if } \sum_{j=1}^n A_{ij} D_j > 1 \\ 0 & \text{otherwise.} \end{cases}$$

---

<sup>6</sup>In the adopter case, the treatment time period intersects with that of the outcome in the first trimester, which generates unwanted feedback between the outcome and selection models. To avoid this, we should define  $Y_i$  as an indicator for adoption *after* the first trimester. Fortunately, this actually coincides with the original definition of  $Y_i$  because adoption decisions are never reversed in the data.

For the first (second) choice,  $\tau(t, t')$  measures the effect of going from 0 to 1 (more than 1) adopting neighbor(s).<sup>7</sup> This sheds light on a different dimension of diffusion relative to the ADM. Whereas the ADM measures how many others are affected by the ego’s adoption, our estimands quantify the effect of having multiple adopting neighbors on the ego’s adoption. We find below that having multiple adopting neighbors has a much larger effect than having only one.

As previously stated, [He and Song \(2024\)](#) define the treatment in two ways. One is a binary indicator for having a leader in the household, the idea being that all leaders were initially informed about microfinance and told to spread the word. However, not all leaders adopted in the first trimester, which perhaps motivates the second definition, a binary indicator for having an adopting leader in the household. In our view, it may be plausible to argue that microfinance adoption in the initial period is as good as random within observable subpopulations, but it is less plausible to make the same case for being a leader, which is likely determined by a complex social process. We therefore consider a third definition, which is simply an indicator for adopting microfinance in the initial period, irrespective of having a leader in the household. Indeed, recall the interpretations of the causal estimands above pertain to the third definition; in our view, the interpretations are less clear when treatment is defined otherwise. Finally, to bolster the plausibility of the unconfounded initial adoptions, we add network controls.

## 7.2 Results

We present estimates of  $\tau(t, t')$  and the ADM for the three network specifications and three treatment definitions introduced above. We use two different estimators for  $\tau(t, t')$ . The first estimates the nuisance functions with GNNs for  $L = 1, 2, 3$  layers. These use the same PNA architecture and learning rate as the simulation study (see §6.2). The MLPs in each layer have  $H = 4$  hidden layers to match the number of household covariates  $X_i$ , and we estimate the GNNs using the PyTorch Geometric package ([Fey and Lenssen, 2019](#)). The second estimator uses GLMs and polynomial sieves of order 1–3 instead of GNNs. As in §6.2, we use a logit to estimate the propensity score and outcome regressions.

---

<sup>7</sup>Assumption 2.1 of [He and Song \(2024\)](#) implies that  $\{D_i\}_{i=1}^n$  is independent conditional on observables. Under the same condition, our estimands identify causal effects by Theorem 1 of [Leung \(2024\)](#).

GNNs FOR NETWORK CONFOUNDING

To compute the estimates, we concatenate the village networks into a single adjacency matrix of size  $n = 4413$ . For the GNN and GLM estimates, we trim observations with propensity scores outside of  $[0.01, 0.99]$ . Standard errors are obtained from the network HAC variance estimator defined in §2.3.

Table 3: Exposure Mapping  $T_i^{(1)}$

	ADM	GNN			GLM		
		1 Layer	2 Layer	3 Layer	Order 1	Order 2	Order 3
Leader case							
$G_{ee}$	-0.052	-0.002 (0.020)	-0.004 (0.021)	0.012 (0.017)	0.000 (0.018)	-0.003 (0.016)	0.037 (0.014)
$G_{sc}$	-0.049	0.018 (0.023)	0.044 (0.019)	0.041 (0.019)	0.014 (0.020)	0.023 (0.020)	-0.025 (0.012)
$G_{all}$	-0.050	0.026 (0.026)	0.022 (0.022)	0.029 (0.022)	0.010 (0.023)	0.018 (0.025)	-0.038 (0.012)
Leader-adopter case							
$G_{ee}$	0.215	0.096 (0.016)	0.085 (0.033)	0.092 (0.025)	0.086 (0.029)	0.086 (0.027)	0.485 (0.017)
$G_{sc}$	0.434	0.032 (0.057)	0.074 (0.022)	0.071 (0.023)	0.073 (0.022)	0.076 (0.020)	0.469 (0.019)
$G_{all}$	0.435	0.076 (0.019)	0.066 (0.021)	0.080 (0.016)	0.069 (0.022)	0.122 (0.021)	0.452 (0.020)
Adopter case							
$G_{ee}$	0.423	0.061 (0.017)	0.056 (0.015)	0.055 (0.016)	0.057 (0.016)	0.054 (0.016)	0.277 (0.018)
$G_{sc}$	0.622	0.034 (0.015)	0.028 (0.014)	0.029 (0.014)	0.025 (0.014)	0.032 (0.014)	0.166 (0.018)
$G_{all}$	0.657	0.024 (0.013)	0.019 (0.014)	0.024 (0.014)	0.019 (0.015)	0.044 (0.014)	0.144 (0.018)

$n = 4413$ . Standard errors are in parentheses. This table presents the effect of a single neighbor adopting microfinance on own adoption.

Table 4: Exposure Mapping  $T_i^{(2)}$

	ADM	GNN			GLM		
		1 Layer	2 Layer	3 Layer	Order 1	Order 2	Order 3
Leader case							
$G_{ee}$	-0.052	-0.022 (0.021)	-0.007 (0.023)	-0.009 (0.022)	-0.013 (0.020)	-0.014 (0.020)	0.077 (0.015)
$G_{sc}$	-0.049	-0.007 (0.023)	0.017 (0.019)	0.017 (0.019)	-0.011 (0.021)	0.001 (0.021)	-0.108 (0.017)
$G_{all}$	-0.050	-0.016 (0.026)	0.004 (0.018)	0.027 (0.018)	-0.010 (0.022)	0.010 (0.024)	-0.137 (0.017)
Leader-adopter case							
$G_{ee}$	0.215	0.302 (0.008)	0.249 (0.010)	-0.172 (0.010)	0.273 (0.065)	0.045 (0.036)	0.522 (0.015)
$G_{sc}$	0.434	0.309 (0.029)	0.335 (0.014)	0.230 (0.025)	0.149 (0.060)	0.118 (0.045)	0.518 (0.017)
$G_{all}$	0.435	0.222 (0.025)	0.418 (0.015)	0.285 (0.018)	0.166 (0.062)	0.153 (0.047)	0.504 (0.017)
Adopter case							
$G_{ee}$	0.423	0.205 (0.033)	0.206 (0.027)	0.185 (0.022)	0.195 (0.031)	0.177 (0.032)	0.381 (0.017)
$G_{sc}$	0.622	0.183 (0.025)	0.176 (0.018)	0.179 (0.020)	0.176 (0.025)	0.187 (0.026)	0.281 (0.018)
$G_{all}$	0.657	0.171 (0.021)	0.166 (0.019)	0.175 (0.021)	0.162 (0.024)	0.184 (0.021)	0.250 (0.018)

$n = 4413$ . Standard errors are in parentheses. This table presents the effect of a multiple neighbors adopting microfinance on own adoption.

Tables 3 and 4 respectively report results for the exposure mapping  $T_i^{(1)}$  and  $T_i^{(2)}$ . We do not report ADM confidence intervals (for the leader and leader-adopter cases these can be found in Table 8 of He and Song (2024)), but the estimates are statistically significant in the leader-adopter and adopter cases and insignificant in

the leader case.

First consider the estimand using  $T_i^{(1)}$ , which contrasts microfinance adoption rates for units with 1 versus 0 initially adopting neighbors. The GNN results are consistent across  $L$ . For the leader case, we obtain precise zeros for almost all estimates, including the ADM. For the leader-adopter case, the GNN estimates are substantially smaller in magnitude than the ADM with an effect size of at most 10 percentage points compared to the smallest ADM estimate of 20 percentage points. This may be attributed to the use of richer network controls. For the adopter case, the contrast is even starker. Our estimates are an order of magnitude smaller than the corresponding ADM estimates. The GLM estimates are typically slightly smaller than the GNN estimates except for the order-3 polynomials, which are outliers in terms of magnitude.

The estimand using  $T_i^{(2)}$  contrasts units with 2+ versus 0 initially adopting neighbors. The estimates for the leader case are similar to those of  $T_i^{(1)}$ . We find sizeable effects for the leader-adopter case, almost of the same order as ADM, but the robustness of the result is partly tempered by the large amount of trimming discussed below. The adopter case sees estimates of around 20 percentage points, whereas the ADM is double or triple that. Once again the GLM estimates are often slightly smaller relative to the GNN estimates, except for the order-3 polynomials.

The number of observations trimmed for  $T_i^{(1)}$  is negligible in the leader and adopter cases. In the leader-adopter case, more units are trimmed since fewer units are leader-adopters, but trimming never drops more than 200 observations. The story is quite different for  $T_i^{(2)}$ , as reported in Table 5. The first three columns of the table report the number of observations for which the number of initially adopting neighbors  $N_i = \sum_{j=1}^n A_{ij}D_j$  satisfies the stated criterion. The last two columns report the smallest sample size after trimming for the GNN and GLM estimates within each category. The problematic category is the leader-adopter case, where the number of observations for which  $N_i$  is two or greater is exceedingly small. As a result, an extremely large amount of observations end up trimmed. This explains the unusual  $-0.172$  GNN point estimate in Table 4, which has a sample size of only 283 after trimming.

Ultimately, the results demonstrate that the addition of network controls attenuates the diffusion effect. However, while the impact of having one adopting neighbor is relatively small, the effect of having multiple adopting neighbors is much larger

in magnitude. In quantifying the impact of having more adopting peers, our estimates complement the ADM, which measures the average effect of own adoption on neighbors’ adoption.

Table 5: Trimming Statistics for  $T_i^{(2)}$ 

	$N_i = 1$	$N_i = 2$	$N_i \geq 3$	GNN $n$	GLM $n$
Leader case					
$G_{ee}$	1481	777	306	4243	4365
$G_{sc}$	1362	956	539	4021	4246
$G_{all}$	1317	964	595	4098	4186
Leader-adopter case					
$G_{ee}$	365	26	0	267	596
$G_{sc}$	539	63	3	2263	1990
$G_{all}$	571	68	3	401	2087
Adopter case					
$G_{ee}$	1087	384	122	4221	4395
$G_{sc}$	1265	506	229	4406	4412
$G_{all}$	1281	557	243	4405	4411

$N_i$  = number of treated neighbors. For the  $N_i$  columns, the rows count the number of observations satisfying the column condition. GNN  $n$  reports the smallest sample size after trimming for any GNN estimate in the category and similarly for GLM  $n$ .

## 8 Conclusion

Existing work on network interference under unconfoundedness assumes that it suffices to control for a known, low-dimensional function of the network and covariates, but the literature lacks selection models justifying common choices of controls. We propose to use GNNs to effectively learn this function and provide a behavioral model under which it is low-dimensional and estimable with shallow GNNs.

Our analysis allows for approximate neighborhood interference in both the outcome and treatment selection stages. [Leung \(2022a\)](#) studies the implications of ANI for asymptotic inference in randomized control trials, and we highlight its utility for handling high-dimensional network confounding, which arises when the decision to select into treatment is subject to peer influence. We provide conditions under which the propensity score and outcome regression, which ordinarily may depend on the entirety of the network, can be approximated by functions of the ego’s  $L$ -neighborhood network for relatively small  $L$ . This is analogous to approximate sparsity conditions

in the lasso literature, which posit that a high-dimensional regression function is well-approximated by a function of a relatively small number of covariates.

## A Additional Results on GNNs

Primitive conditions for Assumption 5 appear to be beyond the scope of the existing GNN literature, but we provide some potentially useful intermediate results. Consider the problem of establishing a rate of convergence for the propensity score:

$$\frac{1}{m_n} \sum_{i \in \mathcal{M}_n} (\hat{p}_t(i, \mathbf{X}, \mathbf{A}) - p_t(i, \mathbf{X}, \mathbf{A}))^2 = o_p(n^{-1/2}).$$

Under network approximate sparsity (Definition 1), the problem simplifies to showing

$$\frac{1}{m_n} \sum_{i \in \mathcal{M}_n} (\hat{p}_t(i, \mathbf{X}, \mathbf{A}) - p_t(i, \mathbf{X}_{\mathcal{N}(i,L)}, \mathbf{A}_{\mathcal{N}(i,L)}))^2 = o_p(n^{-1/2}). \quad (\text{A.1})$$

Since  $\hat{p}_t(i, \mathbf{X}, \mathbf{A})$  is an  $L$ -layer GNN, which only uses information from  $(\mathbf{X}_{\mathcal{N}(i,L)}, \mathbf{A}_{\mathcal{N}(i,L)})$ , this should well approximate  $p_t(i, \mathbf{X}_{\mathcal{N}(i,L)}, \mathbf{A}_{\mathcal{N}(i,L)})$  under appropriate conditions, so (A.1) should be more feasible to verify directly.

Farrell et al. (2021) provide a bound analogous to (A.1) for MLPs, which, were it applicable to our setting, would be of the form

$$\frac{1}{n} \sum_{i=1}^n (\hat{p}_t(i, \mathbf{X}, \mathbf{A}) - p_t(i, \mathbf{X}_{\mathcal{N}(i,L)}, \mathbf{A}_{\mathcal{N}(i,L)}))^2 \leq C \left( \frac{WL \log R}{n} \log n + \frac{\log \log n + \gamma}{n} + \epsilon^2 \right) \quad (\text{A.2})$$

with probability at least  $1 - e^{-\gamma}$ . Here  $W$  is the number of GNN parameters,  $C$  is a constant that does not depend on  $n$ ,  $R$  depends on the architecture through the number of hidden neurons, and  $\epsilon$  is the function approximation error, a measure of the ability of the neural network to approximate any function in a desired class. Establishing a corresponding result for GNNs requires an analog of Lemma 6 of Farrell et al. (2021), which is a bound on the pseudo-dimension of the GNN class, and concentration inequalities for  $\psi$ -dependent data. Jegelka (2022) surveys the few available complexity and generalization bounds for GNNs. These are not sufficiently general for our setup and only apply to settings where the sample consists of many independent networks.



To use a bound of the form (A.2) to verify Assumption 5, we require knowledge of how  $\epsilon$  varies with key aspects of the architecture, such as  $W, R, L, n$ . As a first step toward obtaining such a result, it is necessary to characterize the function class that GNNs can approximate. Our next result, which draws heavily from existing results in the GNN literature, shows that an additional shape restriction on the function class beyond invariance (§3.3) is required.

## A.1 WL Function Class

MLPs can approximate any measurable function (Hornik et al., 1989), so given the discussion in §3.3, a natural question is whether GNNs can approximate any measurable, *invariant* function of graph-structured inputs. In other words, is it enough to require invariance (and regularity conditions), or are stronger restrictions on the function class necessary? For reasons related to the graph isomorphism problem, it turns out stronger restrictions are necessary. We next motivate the need for such restrictions and then state our function approximation result.

Chen et al. (2019) show that, for a function class such as GNNs to approximate any invariant function, some element of the class must be able to separate any pair of non-isomorphic graphs. By “separate,” we mean that for any non-isomorphic “labeled graphs”  $(\mathbf{X}, \mathbf{A}), (\mathbf{X}', \mathbf{A}')$ , the function  $f$  satisfies  $f(\mathbf{X}, \mathbf{A}) \neq f(\mathbf{X}', \mathbf{A}')$ .<sup>8</sup> Hence, a function with separating power of this sort solves the graph isomorphism problem, a problem for which no known polynomial-time solution exists (Kobler et al., 2012; Morris et al., 2021). Since GNNs can be computed in polynomial time, this suggests that approximating any invariant function is too demanding of a requirement.

To define the subclass of invariant functions that GNNs can approximate, we need to take a detour and discuss graph isomorphism tests. The subclass will be defined by a weaker graph separation criterion than solving the graph isomorphism problem, in particular one defined by the *Weisfeiler-Leman (WL) test*. This is a (generally imperfect) test for graph isomorphism on which almost all practical graph isomorphism solvers are based (Morris et al., 2021).

Given a labeled graph  $(\mathbf{X}, \mathbf{A})$ , the WL test outputs a graph coloring (a vector of

---

<sup>8</sup>Their result is for  $\mathbb{R}$ -valued functions  $f(\cdot)$ , so to properly apply GNNs as they define them to isomorphism testing, we would additionally need to take the  $\mathbb{R}^n$ -valued output  $f(\mathbf{X}, \mathbf{A}) = (f(i, \mathbf{X}, \mathbf{A}))_{i=1}^n$  of a GNN as we define it and aggregate it in an invariant manner to obtain  $\mathbb{R}$ -valued output. An example of an invariant aggregator is the sum  $\sum_{i=1}^n f(i, \mathbf{X}, \mathbf{A})$ .

labels for each unit) according to the following recursive procedure, whose definition follows Maron et al. (2019). At each iteration  $t > 0$ , each unit  $i$  is assigned a color  $C_t(i)$  from some set  $\Sigma$  (e.g. the natural numbers) according to

$$C_t(i) = \Phi(C_{t-1}(i), \{C_{t-1}(j) : A_{ij} = 1\}), \quad (\text{A.3})$$

where  $\Phi(\cdot)$  is a bijective function that takes as input a color and a multiset of neighbors' colors.<sup>9</sup> Intuitively, at each iteration, two units are assigned different colors if they differ in the number of identically colored neighbors, so that at iteration  $t$ , colors capture some information about a unit's  $(t - 1)$ -neighborhood. Colors are initialized at  $t = 0$  using a deterministic rule that assigns each  $i$  to the same color  $C_0(i) \in \Sigma$  if and only if they have the same covariates  $X_i$ . At each iteration, the number of assigned colors increases, and the algorithm converges when the coloring is the same in two adjacent iterations. This takes at most  $n - 1$  iterations since there cannot be more than  $n$  distinctly assigned colors.

To test whether two labeled graphs are isomorphic, the procedure is run in parallel on both graphs until some number of iterations, typically until convergence. At this point, if there exists a color such that the number of units assigned that color differs in the two graphs, then the graphs are considered non-isomorphic. This procedure correctly identifies isomorphic graphs, but it is underpowered since there exist non-isomorphic graphs considered isomorphic by the WL test (Morris et al., 2021). Also, because the number of colors increases each iteration, the test is more powerful when run longer.

Morris et al. (2019) and Xu et al. (2018) note the similarity between the GNN architecture (13) and WL test (A.3). The former may be viewed as a continuous approximation of the latter, replacing the hash function  $\Phi(\cdot)$  with a learnable aggregator  $\Phi_{1l}(\cdot)$ . They formally show that any GNN has at most the graph separation power of the WL test and furthermore there exist architectures as powerful.

Returning to the original problem, we now define the class of functions approximated by GNNs in terms of the WL test. Let  $H$  denote the support of  $(\mathbf{X}, \mathbf{A})$ .

**Definition 2.** For any set of functions  $\mathcal{F}$  with domain  $H$ , let  $\rho(\mathcal{F})$  be the subset of

---

<sup>9</sup>Strictly speaking, this is the 1-WL test.

$H^2$  such that

$$(h, h') \in \rho(\mathcal{F}) \quad \text{if and only if} \quad f(h) = f(h') \quad \text{for all} \quad f \in \mathcal{F}.$$

For any two sets of functions  $\mathcal{E}, \mathcal{F}$  with domain  $H$ , we say that  $\mathcal{E}$  is *at most as separating as*  $\mathcal{F}$  if  $\rho(\mathcal{F}) \subseteq \rho(\mathcal{E})$ .

This is essentially Definition 2 of [Azizian and Lelarge \(2021\)](#). Intuitively, if  $\mathcal{E}$  is at most as separating as  $\mathcal{F}$ , the latter is more complex in the sense that some function in  $\mathcal{F}$  can separate weakly more elements of  $H$  than any function in  $\mathcal{E}$ .

Let  $f_{\text{WL},L}$  denote the function of  $(\mathbf{X}, \mathbf{A})$  with range  $\Sigma^n$  that outputs the vector of node colorings from the WL test run for  $L$  iterations. Let  $\mathcal{C}(H)$  be the set of continuous functions with domain  $H$ . For any  $L \in \mathbb{N}$ , define the *WL function class*

$$\mathcal{F}_{\text{WL}}(L) = \{f^* \in \mathcal{C}(H) : \rho(\{f_{\text{WL},L}\}) \subseteq \rho(f^*)\}.$$

This is the set of continuous functions of  $(\mathbf{X}, \mathbf{A})$  that are at most as separating as the WL test with  $L$  iterations.

The next result says that  $p_t(\cdot)$  and  $\mu_t(\cdot)$  can be approximated by  $L$ -layer GNNs under the shape restriction that they are elements of the WL function class. This is a stronger shape restriction than invariance because, by construction, the output of the WL test is invariant, so  $\mathcal{F}_{\text{WL}}(L)$  is a subset of the set of all invariant functions.

Consider the GNN architecture in Example 8 with  $\phi_{\text{ol}}(\cdot), \phi_{\text{ul}}(\cdot)$  being MLPs. For technical reasons, we augment the architecture with an additional MLP layer  $L+1$  at the output stage with  $n$  neurons and the  $i$ th neuron given by  $h_i^{(L+1)} = \phi_{L+1}(h_i^{(L)}, \{h_j^{(L)} : j \in \mathcal{N}_n\})$ . Interpret this as the actual output layer, and let  $L$  only enumerate the number of hidden layers (i.e. not counting the input  $h_i^{(0)}$  and output  $h_i^{(L+1)}$  layers). Let  $\mathcal{F}_{\text{GNN}^*}(L)$  denote the set of such GNNs with  $L$  layers, ranging over the parameter space of the MLPs, including their widths and depths.

**Theorem A.1.** *Fix  $n, L \in \mathbb{N}$ . Suppose that each  $X_i$  has the same common, finite support. For any  $f^* \in \mathcal{F}_{\text{WL}}(L)$ , there exists a sequence of GNNs  $\{f_k\}_{k \in \mathbb{N}} \subseteq \mathcal{F}_{\text{GNN}^*}(L)$  such that*

$$\sup_{(\mathbf{X}, \mathbf{A}) \in H} |f_k(1, \mathbf{X}, \mathbf{A}) - f^*(1, \mathbf{X}, \mathbf{A})| \xrightarrow{k \rightarrow \infty} 0. \quad (\text{A.4})$$

In other words, any function in the class  $\mathcal{F}_{\text{WL}}(L)$  can be approximated by  $L$ -layer GNNs in  $\mathcal{F}_{\text{GNN}^*}(L)$ . The result is a consequence of a Stone-Weierstrauss theorem due to [Azizian and Lelarge \(2021\)](#) and a version of the [Morris et al. \(2019\)](#) and [Xu et al. \(2018\)](#) result on the equivalent separation power of GNNs and the WL test. The proof is given below.

The result is essentially Theorem 4 of [Azizian and Lelarge \(2021\)](#) but with the distinction that they use  $\cup_L \mathcal{F}_{\text{GNN}^*}(L)$  in place of  $\mathcal{F}_{\text{GNN}^*}(L)$  and  $\{f_{\text{WL},\infty}\}$  in place of  $\{f_{\text{WL},L}\}$ . That is, their theorem states that the set of GNNs ranging over all possible numbers of layers can approximate any continuous function at most as separating as the WL test run until convergence.

Theorem A.1 states their result for fixed  $L$ , and the proof is straightforward from prior results. However, our framing clarifies one of the roles of depth, namely that it determines the strength of the shape restriction implicitly imposed on the function being approximated by GNNs. In particular, because the WL test is more powerful when  $L$  is larger, meaning when run for more iterations, Theorem A.1 implies that deeper GNNs can approximate weakly richer function classes, or equivalently, impose weaker shape restrictions. We discuss this point further in the next subsection.

PROOF OF THEOREM A.1. Lemma 35 of [Azizian and Lelarge \(2021\)](#) (in particular the result for  $MGNN_E$ ) shows that there exists a sequence of GNNs  $\{f_k\}_{k \in \mathbb{N}} \subseteq \cup_L \mathcal{F}_{\text{GNN}^*}(L)$  such that (A.4) holds for any  $f^*$  in the class

$$\{f^* \in \mathcal{C}(H) : \rho(\cup_L \mathcal{F}_{\text{GNN}^*}(L)) \subseteq \rho(f^*)\}.$$

In contrast, we would like to establish that, for any fixed  $L$ , there exists a sequence of GNNs  $\{f_k\}_{k \in \mathbb{N}} \subseteq \mathcal{F}_{\text{GNN}^*}(L)$  such that (A.4) holds for any  $f^*$  in the class

$$\{f^* \in \mathcal{C}(H) : \rho(\mathcal{F}_{\text{GNN}^*}(L)) \subseteq \rho(f^*)\}, \tag{A.5}$$

meaning that an  $L$ -layer GNN can arbitrarily approximate any continuous function at most as separating as an  $L$ -layer GNN. The argument in the proof of Lemma 35 actually applies to (A.5) after some minor changes to notation. The first part of the proof (“We now move to the equivariant case. . .”) up to verifying their equation (26) carries over by redefining the  $MGNN_E$  class as having a fixed depth  $L$ . To show (26), [Azizian and Lelarge \(2021\)](#) begin with a GNN  $f$  with  $L$  layers (their notation uses

$T$  in place of  $L$ ) and add an additional MLP layer that implements their equation (26). Here we use the additional MLP layer added to the output of our architecture (see the paragraph prior to the statement of Theorem A.1). In particular, for any  $f \in \mathcal{F}_{\text{GNN}^*}(L)$ , consider the mapping

$$(\mathbf{X}, \mathbf{A}) \mapsto \underbrace{\left( \sum_{i=1}^n f(i, \mathbf{X}, \mathbf{A}), \dots, \sum_{i=1}^n f(i, \mathbf{X}, \mathbf{A}) \right)}_{n \text{ times}} \in \mathbb{R}^n$$

(their (26) in our notation) corresponds to adding a linear output layer  $L + 1$  that is implementable by an MLP of the form  $\phi_{L+1}(h_i^{(L)}, \{h_j^{(L)} : j \in \mathcal{N}_n\})$ . The mapping remains an element of  $\mathcal{F}_{\text{GNN}^*}(L)$ , which completes the argument for (A.5).

By Theorems VIII.1 and VIII.4 of Grohe (2021), which use finiteness of the support of  $X_i$ ,  $\rho(\mathcal{F}_{\text{GNN}^*}(L)) = \rho(\{f_{\text{WL},L}\})$ . That is,  $L$ -layer GNNs have the same separation power as the WL test run for  $L$  iterations. ■

## A.2 Disadvantages of Depth

The receptive field is the main consideration when selecting  $L$ , but Theorem A.1 provides a second consideration, which is imposing a weaker implicit shape restriction. It shows that, for GNNs to approximate a target function well, the target must satisfy a shape restriction stronger than invariance, namely that it is at most as separating as the WL test with  $L$  iterations. The larger the choice of  $L$ , the weaker the shape restriction imposed. However, there are several reasons why shallow architectures remain preferable.

**Low returns to depth.** A natural question is how many iterations are required for the WL test to converge for a given graph, which corresponds to the choice of  $L$  for which the shape restriction is weakest. Unfortunately, the answer is not generally known, being determined by the topology of the input graph in a complex manner. However, there is a range of results bounding the number of iterations required for convergence. For instance, Kiefer and McKay (2020) construct graphs for which the WL test requires  $n - 1$  iterations to converge, so such graphs require  $n - 1$  layers to obtain the weakest shape restriction. This makes the estimation problem extremely high-dimensional, requiring substantially more layers than what is typically required

for the receptive field to encompass the entirety of the network.

Fortunately, theoretical and empirical evidence suggests that such examples are more the exception than the rule and that small choices of  $L$  typically already impose weak shape restrictions. Babai et al. (1980) show that, with probability approaching one as  $n \rightarrow \infty$ , in an  $n$ -unit network drawn uniformly at random from the set of all possible networks, the WL test assigns all units different colors (recall the test must converge at this point) after only *two* iterations (Morris et al., 2021). Thus, roughly speaking, for large networks, the weakest possible shape restriction is generically achieved with only  $L = 2$ . Of course, having the network drawn uniformly at random is a strong assumption, so this result resides in the opposite extreme relative to the worst-case examples requiring  $L = n - 1$ . Nonetheless, it suggests that relatively few layers may often suffice in practice. Indeed, Zopf (2022) provide empirical evidence on this point, showing that the vast majority of graphs in their dataset can be separated using the WL test after a single iteration.

**Cost of depth.** Several explanations have been proposed for why larger  $L$  often results in worse predictive performance. The “oversmoothing” phenomenon (Li et al., 2018; Oono and Suzuki, 2020) posits that node embeddings tend to become indistinguishable across many units as the number of layers grows. In random geometric graphs (see §6),  $L$ -neighborhood sizes grow polynomially with  $L$ , while in Erdős-Rényi graphs, the growth rate is exponential. Accordingly, a small increase in  $L$  can induce a large increase in the number of elements aggregated by  $\Phi_{1l}(\cdot)$ , so by a law of large numbers intuition, the resulting node embeddings tend to concentrate on the same value. Since node embeddings are meant to represent network positions, which tend to be quite heterogeneous across units, this results in poor predictive performance.

The “oversquashing” phenomenon (Alon and Yahav, 2021; Topping et al., 2022) posits that, as  $L$  grows, the GNN aggregates an exceedingly large amount of information due to the growth in neighborhood sizes. This information is stored in node embeddings of relatively small dimension  $H$ , resulting in information loss, so the effective size of the receptive field remains small as  $L$  grows.

Zhou et al. (2021) provide a third explanation, that certain features of common architectures are responsible for variance inflation. In fact, even weaker shape restrictions than that imposed by Theorem A.1 are possible using more complex “ $k$ -GNN” architectures, which would theoretically improve bias, but these have greater computational cost and empirically exhibit worse predictive performance and higher variance

than the standard architecture (13) (Dwivedi et al., 2022). These disadvantages may partly explain the common use in practice of the standard architecture with few layers.

## B Verifying §8 Assumptions

Leung (2022a), §A, verifies analogs of Assumptions 6(d) and 7(c) from an older working paper version of Kojevnikov et al. (2021). This section repeats the exercise for Assumptions 6(d) and 7(c) and (d). We assume throughout that  $\max\{\gamma_n(s/2), \psi_n(s)\} \leq \exp(-c(1 - 4/p)^{-1}s)$  for some  $c > 0$  and  $p$  in Assumption 4(a). As in Leung (2022a), we say a sequence of networks exhibits polynomial neighborhood growth if

$$\sup_n \max_{i \in \mathcal{N}_n} |\mathcal{N}_{\mathbf{A}}(i, s)| = Cs^d$$

for some  $C > 0$ ,  $d \geq 1$ . The sequence exhibits exponential neighborhood growth if

$$\sup_n \max_{i \in \mathcal{N}_n} |\mathcal{N}_{\mathbf{A}}(i, s)| = Ce^{\beta s}$$

for some  $C > 0$  and  $\beta = \log \delta(\mathbf{A})$  (Leung, 2022a, §A discusses this choice of  $\beta$ ).

### B.1 Assumption 6(d)

For polynomial neighborhood growth, choose  $v_n = n^{1/(\alpha d)}$  for  $\alpha > 2$ . The second term in (16) is at most  $n^{3/2} \exp(-cn^{1/(\alpha d)}) = o(1)$ . The third term is at most  $\sum_{s=0}^{\infty} Cs^d \exp(-cs) < \infty$ . The first term is  $\leq n^{-1/2} \sum_{s=0}^{\infty} (Cn^{1/\alpha})(Cs^d) \exp(-cs) = o(1)$  for  $k = 1$ , and for  $k = 2$ , it is at most  $n^{-1} \sum_{s=0}^{\infty} (Cn^{1/\alpha})^2 (Cs^d) \exp(-cs) = o(1)$ .

For exponential neighborhood growth, choose  $v_n = \alpha\beta^{-1} \log n$ ,  $\alpha \in (1.5\beta c^{-1}, 0.5)$ , with  $c$  from the definition of  $\psi_n(s)$  above. Such an  $\alpha$  exists only if  $c > 3\beta$ , which requires  $\psi_n(s)$  to decay sufficiently fast relative to neighborhood growth. The second term in (16) is then at most  $n^{3/2} \exp(-c\alpha\beta^{-1} \log n) = n^{1.5 - c\alpha\beta^{-1}} = o(1)$ . The third term is at most  $\sum_{s=0}^{\infty} C \exp((\beta - c)s) < \infty$ . Finally, for  $k = 1$ , the first term is at most  $n^{-1/2} \sum_{s=0}^{\infty} C^2 \exp(\alpha \log n) \exp((\beta - c)s) = o(1)$ , and for  $k = 2$ , it is at most  $n^{-1} \sum_{s=0}^{\infty} C^2 \exp(2\alpha \log n) \exp((\beta - c)s) = o(1)$ .

## B.2 Bandwidth

We employ a mix of formal and heuristic arguments to show that the bandwidth (12) satisfies Assumption 7(c)–(f). Under polynomial neighborhood growth, as argued in §A.2 of Leung (2022a),  $\mathcal{L}(\mathbf{A}) \approx n^{1/d}$ , in which case  $b_n = \mathcal{L}(\mathbf{A})^{1/4} \approx n^{1/(4d)}$ . Then Assumption 7(d) holds because  $n^{-1} \sum_{i=1}^n n(i, b_n) = Cb_n^d \approx n^{1/4} = o(\sqrt{n})$ , and Assumption 7(e) holds because  $n^{-1} \sum_{i=1}^n n(i, b_n)^2 = Cb_n^{2d} \approx n^{1/2} = O(\sqrt{n})$ . Assumption 7(c) holds because, taking  $\epsilon = 1 - 4/p$ ,

$$\begin{aligned} n^{-1} \sum_{s=0}^{\infty} c_n(s, b_n; 2) \psi_n(s)^{1-4/p} &\leq C^3 n^{-1} \sum_{s=0}^{\infty} b_n^{2d} s^d \exp(-cs) \\ &\approx n^{-1} \sqrt{n} \sum_{s=0}^n s^d \exp(-cs) = O(n^{-1/2}). \end{aligned} \quad (\text{B.1})$$

Finally, Assumption 7(f) holds because

$$\frac{1}{n^2} \sum_{s=0}^n |\mathcal{J}_n(s, b_n)| \psi_n(s) \leq \frac{1}{n^2} \sum_{s=0}^n \sum_{i=1}^n \sum_{j: \ell_{\mathbf{A}}(i,j)=s} n(i, b_n) n(j, b_n) \psi_n(s) \leq (\text{B.1}).$$

Under exponential neighborhood growth, as argued in §A.2 of Leung (2022a),  $\mathcal{L}(\mathbf{A}) \approx \log n / \log \delta(\mathbf{A})$ , in which case  $b_n \approx 0.25 \log n / \log \delta(\mathbf{A})$ . Then Assumption 7(d) holds because  $n^{-1} \sum_{i=1}^n n(i, b_n) = C \exp(\beta b_n) \approx n^{1/4}$ , and Assumption 7(e) holds because  $n^{-1} \sum_{i=1}^n n(i, b_n)^2 = C \exp(2\beta b_n) \approx \sqrt{n}$ . Assumption 7(c) holds because, taking  $\epsilon = 1 - 4/p$ ,

$$\begin{aligned} n^{-1} \sum_{s=0}^{\infty} c_n(s, b_n; 2) \psi_n(s)^{1-4/p} &\leq C^3 n^{-1} \sum_{s=0}^{\infty} \exp(\beta b_n) \exp(\beta s) \exp(-cs) \\ &\approx n^{-1} \exp(0.5 \log n) \sum_{s=0}^n \exp((\beta - c)s) = O(n^{-1/2}), \end{aligned} \quad (\text{B.2})$$

which is  $o(1)$  if  $c > \beta$ , which is weaker than the requirement  $c > 3\beta$  in §B.1. Finally, Assumption 7(f) holds because, if  $c > \beta$ ,  $n^{-2} \sum_{s=0}^n |\mathcal{J}_n(s, b_n)| \psi_n(s) \leq (\text{B.2})$ .



## C Supporting Lemmas

**Lemma C.1.** *Under Assumptions 2, 3, and 8, there exists  $C > 0$  such that for any  $n \in \mathbb{N}$ ,  $i \in \mathcal{N}_n$ , and  $s$  sufficiently large,*

$$\begin{aligned} & |p_t(i, \mathbf{X}, \mathbf{A}) - p_t(i, \mathbf{X}_{\mathcal{N}(i, r_\lambda(s+1))}, \mathbf{A}_{\mathcal{N}(i, r_\lambda(s+1))})| \\ & \leq C(\lambda_n(s+1) + \eta_n(s)(1 + n(i, 1))) \quad a.s. \end{aligned} \quad (\text{C.1})$$

Furthermore, if Assumptions 1, 4(b), 6(b), and 7(a) hold, then there exists  $C > 0$  such that for any  $n \in \mathbb{N}$ ,  $i \in \mathcal{N}_n$ , and  $s$  sufficiently large,

$$\begin{aligned} & |\mu_t(i, \mathbf{X}, \mathbf{A}) - \mu_t(i, \mathbf{X}_{\mathcal{N}(i, r_\lambda(s))}, \mathbf{A}_{\mathcal{N}(i, r_\lambda(s))})| \\ & \leq C(\gamma_n(s/2) + \lambda_n(s) + \eta_n(s/2)(1 + n(i, 1) + \Lambda_n(i, s/2)n(i, s/2))) \quad a.s., \end{aligned} \quad (\text{C.2})$$

where  $\Lambda_n(i, s/2)$  is the Lipschitz constant in Assumption 6(b).

PROOF. Fix  $i \in \mathcal{N}_n$  such that  $n(i, 1) = \gamma \in \Gamma$ .

**Proof of (C.1).** Abbreviate  $V_i = \sum_{j=1}^n A_{ij}D_j$ . Since  $V_i$  is integer-valued and  $\Delta$  defined in Assumption 3 is an interval, by that assumption, there exist  $e > 0$ ,  $a, b, \alpha \in \mathbb{R}$  and  $\beta \in \mathbb{R} \cup \{\infty\}$  with  $a < b$  and  $\alpha < \beta$  such that for any  $\epsilon \in (0, e)$  and  $\mathbf{d} \in \{0, 1\}^n$ ,

$$\begin{aligned} \{f_n(i, \mathbf{d}, \mathbf{A}) = t\} &= \left\{ d_i \in [a, b], \sum_{j=1}^n A_{ij}d_j \in [\alpha, \beta] \right\} \\ &= \left\{ d_i \in [a - \epsilon, b + \epsilon], \sum_{j=1}^n A_{ij}d_j \in [\alpha - \epsilon, \beta + \epsilon] \right\}. \end{aligned} \quad (\text{C.3})$$

For example, if  $T_i = (D_i, \sum_{j=1}^n A_{ij}D_j)$  and  $t = (1, 4)$ , then this holds for  $a = 0.5$ ,  $b = 1.5$ ,  $\alpha = 3.5$ ,  $\beta = 4.5$ , and  $e = 0.1$ .

Fix  $s$ , and abbreviate  $D'_j = h_{n(j,s)}(j, \mathbf{X}_{\mathcal{N}(j,s)}, \mathbf{A}_{\mathcal{N}(j,s)}, \boldsymbol{\nu}_{\mathcal{N}(j,s)})$ ,  $V'_i = \sum_{j=1}^n A_{ij}D'_j$ ,

and  $\mathbf{D}'_B = (D'_j)_{j \in B}$  for any  $B \subseteq \mathcal{N}_n$ . Using the first equality of (C.3),

$$\begin{aligned} p_t(i, \mathbf{X}, \mathbf{A}) &= \mathbf{P}(D'_i + (D_i - D'_i) \in [a, b], V'_i + (V_i - V'_i) \in [\alpha, \beta] \mid \mathbf{X}, \mathbf{A}) \\ &\leq \mathbf{P}(D'_i \in [a - \epsilon, b + \epsilon], V'_i \in [\alpha - \epsilon, \beta + \epsilon] \mid \mathbf{X}, \mathbf{A}) \\ &\quad + \underbrace{\mathbf{P}(|D_i - D'_i| > \epsilon \mid \mathbf{X}, \mathbf{A}) + \mathbf{P}(|V_i - V'_i| > \epsilon \mid \mathbf{X}, \mathbf{A})}_{R_0}. \end{aligned}$$

By (C.3), the right-hand side equals

$$\mathbf{P}(D'_i \in [a, b], V'_i \in [\alpha, \beta] \mid \mathbf{X}, \mathbf{A}) + R_0 = \mathbf{P}(f_n(i, \mathbf{D}', \mathbf{A}) = t \mid \mathbf{X}, \mathbf{A}) + R_0,$$

so that

$$p_t(i, \mathbf{X}, \mathbf{A}) \leq \mathbf{P}(f_n(i, \mathbf{D}', \mathbf{A}) = t \mid \mathbf{X}, \mathbf{A}) + R_0. \quad (\text{C.4})$$

By the same argument,

$$\begin{aligned} \mathbf{P}(f_n(i, \mathbf{D}', \mathbf{A}) = t \mid \mathbf{X}, \mathbf{A}) &= \mathbf{P}(D'_i \in [a, b], V'_i \in [\alpha, \beta] \mid \mathbf{X}, \mathbf{A}) \\ &\leq \mathbf{P}(D'_i + (D_i - D'_i) \in [a - \epsilon, b + \epsilon], V'_i + (V_i - V'_i) \in [\alpha - \epsilon, \beta + \epsilon] \mid \mathbf{X}, \mathbf{A}) \\ &\quad + \mathbf{P}(|D_i - D'_i| > \epsilon \mid \mathbf{X}, \mathbf{A}) + \mathbf{P}(|V_i - V'_i| > \epsilon \mid \mathbf{X}, \mathbf{A}) \\ &= \mathbf{P}(D_i \in [a, b], V_i \in [\alpha, \beta] \mid \mathbf{X}, \mathbf{A}) + R_0 \\ &= p_t(i, \mathbf{X}, \mathbf{A}) + R_0. \end{aligned} \quad (\text{C.5})$$

Combining (C.4) and (C.5),

$$|p_t(i, \mathbf{X}, \mathbf{A}) - \mathbf{P}(f_n(i, \mathbf{D}', \mathbf{A}) = t \mid \mathbf{X}, \mathbf{A})| \leq R_0 \leq \epsilon^{-1}(1 + n(i, 1))\eta_n(s), \quad (\text{C.6})$$

the second inequality due to Markov's inequality and Assumption 2.

Observe that  $f_n(i, \mathbf{D}', \mathbf{A})$  is a deterministic function of  $(\mathbf{X}_B, \mathbf{A}_B, \boldsymbol{\varepsilon}_B, \boldsymbol{\nu}_B)$  for  $B = \mathcal{N}(i, s + 1)$  by definition of  $D'_j$  and Assumption 3. Then by Assumption 8,

$$\begin{aligned} &|\mathbf{P}(f_n(i, \mathbf{D}', \mathbf{A}) = t \mid \mathbf{X}, \mathbf{A}) \\ &\quad - \mathbf{P}(f_n(i, \mathbf{D}', \mathbf{A}) = t \mid \mathbf{X}_{\mathcal{N}(i, r_\lambda(s+1))}, \mathbf{A}_{\mathcal{N}(i, r_\lambda(s+1))})| \leq \lambda_n(s + 1) \end{aligned} \quad (\text{C.7})$$

Combining (C.6) and (C.7) and using the law of iterated expectations,

$$|p_t(i, \mathbf{X}, \mathbf{A}) - p_t(i, \mathbf{X}_{\mathcal{N}(i, r_\lambda(s+1))}, \mathbf{A}_{\mathcal{N}(i, r_\lambda(s+1))})| \leq \lambda_n(s+1) + 2R_0.$$

**Proof of (C.2).** Noting that  $\mu_t(i, \mathbf{X}, \mathbf{A}) = \mathbf{E}[Y_i \mathbf{1}_i(t) \mid \mathbf{X}, \mathbf{A}] / p_t(i, \mathbf{X}, \mathbf{A})$ , we first bound the numerator. For  $B = \mathcal{N}(i, s)$ , define  $Y'_i = g_{n(i, s)}(i, \mathbf{D}'_B, \mathbf{X}_B, \mathbf{A}_B, \boldsymbol{\varepsilon}_B)$ . By Lemma C.2,

$$|\mathbf{E}[Y_i \mathbf{1}_i(t) \mid \mathbf{X}, \mathbf{A}] - \mathbf{E}[Y'_i \mathbf{1}_i(t) \mid \mathbf{X}, \mathbf{A}]| \leq \underbrace{\gamma_n(s) + \Lambda_n(i, s)n(i, s)\eta_n(s)}_{R_1}. \quad (\text{C.8})$$

Recalling Assumption 3, define  $\mathbf{1}_i(t)' = \mathbf{1}\{D'_i = d, \sum_{j=1}^n A_{ij}D'_j \in \boldsymbol{\Delta}\}$ . By Lemma C.3, there exists  $C' > 0$  such that for any  $n \in \mathbb{N}$  and  $i \in \mathcal{N}_n$ ,

$$\mathbf{E}[Y'_i |\mathbf{1}_i(t) - \mathbf{1}_i(t)'| \mid \mathbf{X}, \mathbf{A}] \leq \underbrace{C'(1 + n(i, 1))\eta_n(s)}_{R_2}.$$

This and (C.8) yield

$$|\mathbf{E}[Y_i \mathbf{1}_i(t) \mid \mathbf{X}, \mathbf{A}] - \mathbf{E}[Y'_i \mathbf{1}_i(t)' \mid \mathbf{X}, \mathbf{A}]| \leq R_1 + R_2. \quad (\text{C.9})$$

By Assumption 8, which is applicable because  $Y_i$  is a bounded function by Assumption 7(a),

$$|\mathbf{E}[Y'_i \mathbf{1}_i(t)' \mid \mathbf{X}, \mathbf{A}] - \mathbf{E}[Y'_i \mathbf{1}_i(t)' \mid \mathbf{X}_{\mathcal{N}(i, r_\lambda(2s))}, \mathbf{A}_{\mathcal{N}(i, r_\lambda(2s))}]| \leq \lambda_n(2s)$$

since  $Y'_i \mathbf{1}_i(t)'$  is a deterministic function of  $(\mathbf{X}_{B'}, \mathbf{A}_{B'}, \boldsymbol{\varepsilon}_{B'}, \boldsymbol{\nu}_{B'})$  for  $B' = \mathcal{N}(i, 2s)$ . Using (C.9) and the law of iterated expectations,

$$|\mathbf{E}[Y_i \mathbf{1}_i(t) \mid \mathbf{X}, \mathbf{A}] - \mathbf{E}[Y_i \mathbf{1}_i(t) \mid \mathbf{X}_{\mathcal{N}(i, r_\lambda(2s))}, \mathbf{A}_{\mathcal{N}(i, r_\lambda(2s))}]| \leq \underbrace{\lambda_n(2s) + 2(R_1 + R_2)}_{R_1^*}. \quad (\text{C.10})$$

By (C.1) and (C.10),

$$\begin{aligned} \mu_t(i, \mathbf{X}, \mathbf{A}) &= \frac{\mathbf{E}[Y_i \mathbf{1}_i(t) \mid \mathbf{X}, \mathbf{A}]}{p_t(i, \mathbf{X}, \mathbf{A})} = \frac{\mathbf{E}[Y_i \mathbf{1}_i(t) \mid \mathbf{X}_{\mathcal{N}(i, r_\lambda(2s))}, \mathbf{A}_{\mathcal{N}(i, r_\lambda(2s))}] + R_1^*}{p_t(i, \mathbf{X}_{\mathcal{N}(i, r_\lambda(2s))}, \mathbf{A}_{\mathcal{N}(i, r_\lambda(2s))}) + R_2^*} \\ &= \mu_t(i, \mathbf{X}_{\mathcal{N}(i, r_\lambda(2s))}, \mathbf{A}_{\mathcal{N}(i, r_\lambda(2s))}) + R_3^* \end{aligned}$$

where, using Assumption 4(b),

$$\begin{aligned} |R_1^*| &\leq \lambda_n(2s) + 2(\gamma_n(s) + \Lambda_n(i, s)n(i, s)\eta_n(s) + C'(1 + n(i, 1))\eta_n(s)), \\ |R_2^*| &\leq C(\lambda_n(2s) + (1 + n(i, 1))\eta_n(2s - 1)), \quad \text{and} \\ |R_3^*| &\leq C''(|R_1^*| + |R_2^*|) \end{aligned}$$

for some universal  $C'' > 0$ . Substituting  $s/2$  for  $s$  yields the result.  $\blacksquare$

**Lemma C.2.** *Define  $B_i = \mathcal{N}(i, s)$ ,  $D'_j = h_{n(j, s)}(j, \mathbf{X}_{B_j}, \mathbf{A}_{B_j}, \boldsymbol{\nu}_{B_j})$ ,  $\mathbf{D}'_{B_i} = (D'_j)_{j \in B_i}$ , and  $Y'_i = g_{n(i, s)}(i, \mathbf{D}'_{B_i}, \mathbf{X}_{B_i}, \mathbf{A}_{B_i}, \boldsymbol{\varepsilon}_{B_i})$ . Under Assumptions 2, 1, and 6(b),*

$$|\mathbf{E}[Y_i \mathbf{1}_i(t) \mid \mathbf{X}, \mathbf{A}] - \mathbf{E}[Y'_i \mathbf{1}_i(t) \mid \mathbf{X}, \mathbf{A}]| \leq \gamma_n(s) + \Lambda_n(i, s)n(i, s)\eta_n(s),$$

where  $\Lambda_n(i, s)$  is defined in Assumption 6(b).

PROOF. By Assumption 2,

$$|\mathbf{E}[Y_i \mathbf{1}_i(t) \mid \mathbf{D}, \mathbf{X}, \mathbf{A}] - \mathbf{E}[g_{n(i, s)}(i, \mathbf{D}_{B_i}, \mathbf{X}_{B_i}, \mathbf{A}_{B_i}, \boldsymbol{\varepsilon}_{B_i}) \mathbf{1}_i(t) \mid \mathbf{D}, \mathbf{X}, \mathbf{A}]| \leq \gamma_n(s).$$

By Assumption 1,

$$\begin{aligned} &\mathbf{E}[g_{n(i, s)}(i, \mathbf{D}_{B_i}, \mathbf{X}_{B_i}, \mathbf{A}_{B_i}, \boldsymbol{\varepsilon}_{B_i}) \mathbf{1}_i(t) \mid \mathbf{D} = \mathbf{d}, \mathbf{X} = \mathbf{x}, \mathbf{A} = \mathbf{a}] \\ &= \mathbf{E}[g_{n(i, s)}(i, \mathbf{d}_{B_i}, \mathbf{X}_{B_i}, \mathbf{A}_{B_i}, \boldsymbol{\varepsilon}_{B_i}) \mathbf{1}_i(t) \mid \mathbf{X} = \mathbf{x}, \mathbf{A} = \mathbf{a}], \end{aligned}$$

which, together with the law of iterated expectations and Assumption 6(b), implies

$$\begin{aligned} &|\mathbf{E}[g_{n(i, s)}(i, \mathbf{D}_{B_i}, \mathbf{X}_{B_i}, \mathbf{A}_{B_i}, \boldsymbol{\varepsilon}_{B_i}) \mathbf{1}_i(t) \mid \mathbf{X}, \mathbf{A}] \\ &\quad - \mathbf{E}[g_{n(i, s)}(i, \mathbf{D}'_{B_i}, \mathbf{X}_{B_i}, \mathbf{A}_{B_i}, \boldsymbol{\varepsilon}_{B_i}) \mathbf{1}_i(t) \mid \mathbf{X}, \mathbf{A}]| \\ &\quad \leq \Lambda_n(i, s) \sum_{j \in B_i} \mathbf{E}[|D_j - D'_j| \mid \mathbf{X}, \mathbf{A}] \leq \Lambda_n(i, s)n(i, s)\eta_n(s), \end{aligned}$$

where the last inequality uses Assumption 2. Therefore,

$$\mathbf{E}[Y_i \mathbf{1}_i(t) \mid \mathbf{X}, \mathbf{A}] = \mathbf{E}[Y'_i \mathbf{1}_i(t) \mid \mathbf{X}, \mathbf{A}] + R_1$$

for  $|R_1| \leq \gamma_n(s) + \Lambda_n(i, s)n(i, s)\eta_n(s)$ . ■

**Lemma C.3.** *Define  $Y'_i, D'_i$  as in Lemma C.2 and  $\mathbf{1}_i(t)' = \mathbf{1}\{D'_i = d, \sum_{j=1}^n A_{ij}D'_j \in \Delta\}$ . Under Assumptions 2, 1, 3, and 4(a), there exists  $C > 0$  such that for any  $n \in \mathbb{N}$ ,  $i \in \mathcal{N}_n$ , and  $s \geq 0$ ,*

$$\mathbf{E}[Y_i|\mathbf{1}_i(t) - \mathbf{1}_i(t)' | \mathbf{X}, \mathbf{A}] \leq C(1 + n(i, 1))\eta_n(s).$$

PROOF. Recall the definition of  $a, b, \alpha, \beta, \epsilon$  prior to (C.3). Define  $V_i = \sum_{j=1}^n A_{ij}D_j$ ,  $V'_i = \sum_{j=1}^n A_{ij}D'_j$ , and  $\mathcal{C} = \{|D_i - D'_i| \leq \epsilon, |V_i - V'_i| \leq \epsilon\}$ . Then

$$\begin{aligned} & \mathbf{E}[Y_i|\mathbf{1}_i(t) - \mathbf{1}_i(t)' | \mathbf{X} = \mathbf{x}, \mathbf{A} = \mathbf{a}] \\ & \leq \mathbf{E}[Y_i|\mathbf{1}_i(t) - \mathbf{1}_i(t)' | \mathcal{C}, \mathbf{X} = \mathbf{x}, \mathbf{A} = \mathbf{a}] + C \mathbf{P}(\mathcal{C}^c | \mathbf{X} = \mathbf{x}, \mathbf{A} = \mathbf{a}) \end{aligned} \quad (\text{C.11})$$

for some universal  $C > 0$  by Assumptions 1 and 4(a). By Assumption 3,

$$\mathbf{1}_i(t) = \mathbf{1}\{D_i \in [a, b], V_i \in [\alpha, \beta]\} \quad \text{and} \quad \mathbf{1}_i(t)' = \mathbf{1}\{D'_i \in [a, b], V'_i \in [\alpha, \beta]\}.$$

Under event  $\mathcal{C}$ ,

$$\begin{aligned} \mathbf{1}\{D_i \in [a, b], V_i \in [\alpha, \beta]\} &= \mathbf{1}\{D'_i + (D_i - D'_i) \in [a, b], V'_i + (V_i - V'_i) \in [\alpha, \beta]\} \\ &\leq \mathbf{1}\{D'_i \in [a - \epsilon, b + \epsilon], V'_i \in [\alpha - \epsilon, \beta + \epsilon]\} \\ &= \mathbf{1}\{D'_i \in [a, b], V'_i \in [\alpha, \beta]\}, \end{aligned}$$

where the inequality is due to event  $\mathcal{C}$  and the last equality is due to (C.3). By the same argument,  $\mathbf{1}\{D'_i \in [a, b], V'_i \in [\alpha, \beta]\} \leq \mathbf{1}\{D_i \in [a, b], V_i \in [\alpha, \beta]\}$ , so  $\mathbf{1}_i(t) = \mathbf{1}_i(t)'$  under event  $\mathcal{C}$ . Hence, by Markov's inequality and Assumption 2,

$$(\text{C.11}) \leq C\epsilon^{-1}(1 + n(i, 1))\eta_n(s).$$

■

The following notion of weak network dependence is due to [Kojevnikov et al. \(2021\)](#). For any  $H, H' \subseteq \mathcal{N}_n$ , define  $\ell_{\mathbf{A}}(H, H') = \min\{\ell_{\mathbf{A}}(i, j) : i \in H, j \in H'\}$ . Let  $\{Z_i\}_{i=1}^n \subseteq \mathbb{R}$  be a triangular array,  $\mathbf{Z}_H = (Z_i)_{i \in H}$ ,  $\mathcal{L}_d$  be the set of bounded  $\mathbb{R}$ -valued

Lipschitz functions on  $\mathbb{R}^d$ ,  $\text{Lip}(f)$  be the Lipschitz constant of  $f \in \mathcal{L}_d$ , and

$$\mathcal{P}_n(h, h'; s) = \{(H, H') : H, H' \subseteq \mathcal{N}_n, |H| = h, |H'| = h', \ell_{\mathbf{A}}(H, H') \geq s\}.$$

**Definition C.1.** A triangular array  $\{Z_i\}_{i=1}^n$  is *conditionally  $\psi$ -dependent given  $\mathcal{F}_n$*  if there exist  $C \in (0, \infty)$  and an  $\mathcal{F}_n$ -measurable sequence  $\{\psi_n(s)\}_{s, n \in \mathbb{N}}$  with  $\psi_n(0) = 1$  for all  $n$  such that

$$|\text{Cov}(f(\mathbf{Z}_H), f'(\mathbf{Z}_{H'}))| \leq Chh'(\|f\|_\infty + \text{Lip}(f))(\|f'\|_\infty + \text{Lip}(f'))\psi_n(s) \quad \text{a.s.} \quad (\text{C.12})$$

for all  $n, h, h' \in \mathbb{N}$ ;  $s > 0$ ;  $f \in \mathcal{L}_h$ ;  $f' \in \mathcal{L}_{h'}$ ; and  $(H, H') \in \mathcal{P}_n(h, h'; s)$ . We call  $\psi_n(s)$  the *dependence coefficient* of  $\{Z_i\}_{i=1}^n$ .

**Lemma C.4.** *Under Assumptions 1, 2, 3, 4(a) and (b), and 6(a) and (b), for any  $t, t' \in \mathcal{T}$ ,  $\{\varphi_{t, t'}(i)\}_{i=1}^n$  is conditionally  $\psi$ -dependent given  $(\mathbf{X}, \mathbf{A})$  (Definition C.1) with dependence coefficient  $\psi_n(s)$  defined in (15).*

PROOF. Let  $\mathcal{F}_n$  be the  $\sigma$ -algebra generated by  $(\mathbf{X}, \mathbf{A})$ ,  $(h, h') \in \mathbb{N} \times \mathbb{N}$ ,  $(f, f') \in \mathcal{L}_h \times \mathcal{L}_{h'}$ ,  $s > 0$ , and  $(H, H') \in \mathcal{P}_n(h, h'; s)$ . Define  $Z_i = \varphi_{t, t'}(i)$ ,  $\mathbf{Z}_H = (Z_i)_{i \in H}$ ,  $\xi = f(\mathbf{Z}_H)$ ,  $\zeta = f'(\mathbf{Z}_{H'})$ , and

$$D_i^{(s)} = h_{n(i, s)}(i, \mathbf{X}_{\mathcal{N}(i, s)}, \mathbf{A}_{\mathcal{N}(i, s)}, \boldsymbol{\nu}_{\mathcal{N}(i, s)}).$$

For  $\mathbf{D}_{\mathcal{N}(i, s')}^{(s)} = (D_j^{(s)})_{j \in \mathcal{N}(i, s')}$ , let

$$\begin{aligned} \mathbf{1}_i^{(s)}(t) &= \mathbf{1}\{f_{n(i, s/2)}(i, \mathbf{D}_{\mathcal{N}(i, s/2)}^{(s/2)}, \mathbf{A}_{\mathcal{N}(i, s/2)}) = t\}, \\ Y_i^{(s)} &= g_{n(i, s/2)}(i, \mathbf{D}_{\mathcal{N}(i, s/2)}^{(s/2)}, \mathbf{X}_{\mathcal{N}(i, s/2)}, \mathbf{A}_{\mathcal{N}(i, s/2)}, \boldsymbol{\epsilon}_{\mathcal{N}(i, s/2)}), \\ Z_i^{(s)} &= \frac{\mathbf{1}_i^{(s)}(t)(Y_i^{(s)} - \mu_t(i, \mathbf{X}, \mathbf{A}))}{p_t(i, \mathbf{X}, \mathbf{A})} + \mu_t(i, \mathbf{X}, \mathbf{A}) \\ &\quad - \frac{\mathbf{1}_i^{(s)}(t')(Y_i^{(s)} - \mu_{t'}(i, \mathbf{X}, \mathbf{A}))}{p_{t'}(i, \mathbf{X}, \mathbf{A})} - \mu_{t'}(i, \mathbf{X}, \mathbf{A}) - \tau_i(t, t'). \end{aligned}$$

Finally, let  $\xi^{(s)} = f((Z_i^{(s)})_{i \in H})$  and  $\zeta^{(s)} = f'((Z_i^{(s)})_{i \in H'})$ .

By Assumption 6(a),  $(Z_i^{(s/2, \xi)})_{i \in H} \perp\!\!\!\perp (Z_j^{(s/2, \zeta)})_{j \in H'} \mid \mathcal{F}_n$ , so

$$\begin{aligned} |\text{Cov}(\xi, \zeta \mid \mathcal{F}_n)| &\leq |\text{Cov}(\xi - \xi^{(s/2)}, \zeta \mid \mathcal{F}_n)| + |\text{Cov}(\xi^{(s/2)}, \zeta - \zeta^{(s/2)} \mid \mathcal{F}_n)| \\ &\leq 2\|f'\|_\infty \mathbf{E}[|\xi - \xi^{(s/2)}| \mid \mathcal{F}_n] + 2\|f\|_\infty \mathbf{E}[|\zeta - \zeta^{(s/2)}| \mid \mathcal{F}_n] \\ &\leq 2(h\|f'\|_\infty \text{Lip}(f) + h'\|f\|_\infty \text{Lip}(f')) \max_{i \in \mathcal{N}_n} \mathbf{E}[|Z_i - Z_i^{(s/2)}| \mid \mathcal{F}_n]. \end{aligned}$$

By Assumption 4(a) and (b), there exists  $C > 0$  such that for any  $n \in \mathbb{N}$  and  $i \in \mathcal{N}_n$ ,

$$\mathbf{E}[|Z_i - Z_i^{(s/2)}| \mid \mathcal{F}_n] \leq C(\mathbf{E}[|\mathbf{1}_i(t) - \mathbf{1}_i^{(s/2)}(t)| \mid \mathcal{F}_n] + \mathbf{E}[|Y_i - Y_i^{(s/2)}| \mid \mathcal{F}_n]).$$

By an argument similar to the proof of Lemma C.2,

$$\mathbf{E}[|Y_i - Y_i^{(s)}| \mid \mathcal{F}_n] \leq \gamma_n(s/2) + \Lambda_n(i, s/2)n(i, s/2)\eta_n(s/2).$$

By an argument similar to the proof of Lemma C.3,

$$\mathbf{E}[|\mathbf{1}_i(t) - \mathbf{1}_i^{(s)}(t)| \mid \mathcal{F}_n] \leq C'(1 + n(i, 1))\eta_n(s/2)$$

for some universal constant  $C' > 0$ . Hence, for some universal  $C'' > 0$ ,

$$\begin{aligned} \max_{i \in \mathcal{N}_n} \mathbf{E}[|Z_i - Z_i^{(s/2)}| \mid \mathcal{F}_n] \\ \leq C'' \underbrace{\max_{i \in \mathcal{N}_n} (\gamma_n(s/4) + \eta_n(s/4)(1 + n(i, 1) + \Lambda_n(i, s/4)n(i, s/4)))}_{\psi_n(s)}. \end{aligned}$$

■

**Lemma C.5.** *Under Assumptions 2, 1, 4(a), and 6(a),  $\text{Cov}(Y_i, Y_j \mid \mathbf{D}, \mathbf{X}, \mathbf{A}) \leq C\gamma_n(s/2)^{1-2/p}$  a.s. for  $p$  given in Assumption 4(a) and some universal constant  $C > 0$ .*

PROOF. Let  $\mathcal{F}'_n$  be the  $\sigma$ -algebra generated by  $(\mathbf{D}, \mathbf{X}, \mathbf{A})$ . We show that  $\{Y_i\}_{i=1}^n$  is conditionally  $\psi$ -dependent given  $\mathcal{F}'_n$  (Definition C.1) with dependence coefficient  $\gamma_n(s/2)$  (cf. [Kojevnikov et al., 2021](#), Proposition 2.3). Define  $(h, h') \in \mathbb{N} \times \mathbb{N}$ ,  $(f, f') \in$

$\mathcal{L}_h \times \mathcal{L}_{h'}$ ,  $s > 0$ ,  $(H, H') \in \mathcal{P}_n(h, h'; s)$ ,

$$Y_i^{(s)} = g_{n(i,s)}(i, \mathbf{D}_{\mathcal{N}(i,s)}, \mathbf{X}_{\mathcal{N}(i,s)}, \mathbf{A}_{\mathcal{N}(i,s)}, \boldsymbol{\varepsilon}_{\mathcal{N}(i,s)}),$$

$\xi = f((Y_i)_{i \in H})$ ,  $\zeta = f'((Y_i)_{i \in H'})$ ,  $\xi^{(s)} = f((Y_i^{(s)})_{i \in H})$ , and  $\zeta^{(s)} = f'((Y_i^{(s)})_{i \in H'})$ . By Assumption 6(a),

$$\begin{aligned} |\text{Cov}(\xi, \zeta \mid \mathcal{F}'_n)| &\leq |\text{Cov}(\xi - \xi^{(s/2)}, \zeta \mid \mathcal{F}'_n)| + |\text{Cov}(\xi^{(s/2)}, \zeta - \zeta^{(s/2)} \mid \mathcal{F}'_n)| \\ &\leq 2\|f'\|_\infty \mathbf{E}[|\xi - \xi^{(s/2)}| \mid \mathcal{F}'_n] + 2\|f\|_\infty \mathbf{E}[|\zeta - \zeta^{(s/2)}| \mid \mathcal{F}'_n] \\ &\leq 2(h\|f'\|_\infty \text{Lip}(f) + h'\|f\|_\infty \text{Lip}(f')) \max_{i \in \mathcal{N}_n} \mathbf{E}[|Y_i - Y_i^{(s/2)}| \mid \mathcal{F}'_n] \\ &\leq 2(h\|f'\|_\infty \text{Lip}(f) + h'\|f\|_\infty \text{Lip}(f')) \gamma_n(s/2), \end{aligned}$$

the last line using Assumption 2. Given  $\psi$ -dependence, the claim follows from Corollary A.2 of [Kojevnikov et al. \(2021\)](#), which we may apply in light of the moment conditions implied by Assumptions 1 and 4(a).  $\blacksquare$

## D Proofs of Main Results

PROOF OF PROPOSITION 1. By definition,  $T_{\pi(i)} = f_n(\pi(i), \mathbf{D}, \mathbf{A})$  and

$$Y_{\pi(i)} = g(\pi(i), \mathbf{D}, \mathbf{X}, \mathbf{A}, \boldsymbol{\varepsilon}) = g(\pi(i), (h_n(j, \mathbf{X}, \mathbf{A}, \boldsymbol{\nu}))_{j=1}^n, \mathbf{X}, \mathbf{A}, \boldsymbol{\varepsilon}).$$

By the invariance assumptions on  $f_n, g_n, h_n$ ,

$$\begin{aligned} Y_i &= g(i, \mathbf{D}, \mathbf{X}, \mathbf{A}, \boldsymbol{\varepsilon}) = g_n(\pi(i), \pi(\mathbf{D}), \pi(\mathbf{X}), \pi(\mathbf{A}), \pi(\boldsymbol{\varepsilon})) \\ &= g_n(\pi(i), (h_n(\pi(j), \pi(\mathbf{X}), \pi(\mathbf{A}), \pi(\boldsymbol{\nu})))_{j=1}^n, \pi(\mathbf{X}), \pi(\mathbf{A}), \pi(\boldsymbol{\varepsilon})), \end{aligned}$$

and

$$T_i = f_n(\pi(i), \pi(\mathbf{D}), \pi(\mathbf{A})) = f_n(\pi(i), (h_n(\pi(j), \pi(\mathbf{A}), \pi(\boldsymbol{\nu})))_{j=1}^n, \pi(\mathbf{X}), \pi(\mathbf{A})),$$

so by the distributional exchangeability assumption,

$$(Y_i, T_i, \mathbf{X}, \mathbf{A}) \stackrel{d}{=} (Y_{\pi(i)}, T_{\pi(i)}, \pi(\mathbf{X}), \pi(\mathbf{A})).$$



It follows that

$$\begin{aligned}\mu_t(i, \mathbf{X}, \mathbf{A}) &= \mathbf{E}[Y_i \mid T_i = t, \mathbf{X}, \mathbf{A}] \\ &= \mathbf{E}[Y_{\pi(i)} \mid T_{\pi(i)} = t, \pi(\mathbf{X}), \pi(\mathbf{A})] = \mu_t(\pi(i), \pi(\mathbf{X}), \pi(\mathbf{A}))\end{aligned}$$

and similarly for the generalized propensity score. ■

PROOF OF THEOREM 1. Decompose

$$\sqrt{m_n}(\hat{\tau}(t, t') - \tau(t, t')) = \frac{1}{\sqrt{m_n}} \sum_{i \in \mathcal{M}_n} \varphi_{t,t'}(i) - R_{1t} + R_{1t'} - R_{2t} + R_{2t'},$$

where

$$\begin{aligned}R_{1t} &= \frac{1}{\sqrt{m_n}} \sum_{i \in \mathcal{M}_n} \frac{\mathbf{1}_i(t)(Y_i - \mu_t(i, \mathbf{X}, \mathbf{A}))}{\hat{p}_t(i, \mathbf{X}, \mathbf{A})p_t(i, \mathbf{X}, \mathbf{A})} (\hat{p}_t(i, \mathbf{X}, \mathbf{A}) - p_t(i, \mathbf{X}, \mathbf{A})), \\ R_{2t} &= \frac{1}{\sqrt{m_n}} \sum_{i \in \mathcal{M}_n} (\hat{\mu}_t(i, \mathbf{X}, \mathbf{A}) - \mu_t(i, \mathbf{X}, \mathbf{A})) \left(1 - \frac{\mathbf{1}_i(t)}{\hat{p}_t(i, \mathbf{X}, \mathbf{A})}\right),\end{aligned}$$

and likewise for  $R_{1t'}$  and  $R_{2t'}$ . Let  $\mathcal{F}_n$  denote the  $\sigma$ -algebra generated by  $(\mathbf{X}, \mathbf{A})$ . By Lemma C.4,  $\{\varphi_{t,t'}(i)\}_{i=1}^n$  is conditionally  $\psi$ -dependent given  $\mathcal{F}_n$  in the sense of Definition C.1 with dependence coefficient  $\psi_n(s)$ . By Assumptions 4 and 6(c) and (d), we may apply Theorem 3.2 of [Kojevnikov et al. \(2021\)](#) to  $n^{-1/2} \sum_{i=1}^n \sqrt{n/m_n} \varphi_{t,t'}(i) \mathbf{1}\{i \in \mathcal{M}_n\}$  to obtain

$$\sigma_n^{-1} \frac{1}{\sqrt{m_n}} \sum_{i \in \mathcal{M}_n} \varphi_{t,t'}(i) \xrightarrow{d} \mathcal{N}(0, 1).$$

It therefore remains to show that the remainder terms  $R_{1t}, R_{2t}$  are  $o_p(1)$ .

We first bound  $R_{1t}$ . The argument is more complicated than the i.i.d. case due to covariance terms. Abbreviate  $\mu_i = \mu_t(i, \mathbf{X}, \mathbf{A})$ ,  $p_i = p_t(i, \mathbf{X}, \mathbf{A})$ , and  $\hat{p}_i =$

$\hat{p}_t(i, \mathbf{X}, \mathbf{A})$ . For some universal constants  $C, C' > 0$ ,  $\mathbf{E}[R_{1t}^2]$  equals

$$\begin{aligned} & \frac{1}{m_n} \sum_{i \in \mathcal{M}_n} \sum_{j \in \mathcal{M}_n} \mathbf{E} \left[ \mathbf{E} [(Y_i - \mu_i)(Y_j - \mu_j) \mid \mathbf{D}, \mathbf{X}, \mathbf{A}] \frac{\mathbf{1}_i(t) \mathbf{1}_j(t) (\hat{p}_i - p_i) (\hat{p}_j - p_j)}{\hat{p}_i p_i \hat{p}_j p_j} \right] \\ & \leq C \sum_{s=0}^{\infty} \gamma_n(s/2)^{1-2/p} \frac{n}{m_n} \frac{1}{n} \sum_{i=1}^n \sum_{j=1}^n \mathbf{1}\{\ell_{\mathbf{A}}(i, j) = s\} C' \mathbf{E} [|\hat{p}_i - p_i|] \\ & \leq C C' \sum_{s=0}^{\infty} \gamma_n(s/2)^{1-2/p} \frac{n}{m_n} \left( \frac{1}{n} \sum_{i=1}^n |\mathcal{N}^\partial(i, s)|^2 \right)^{1/2} \left( \frac{1}{n} \sum_{i=1}^n \mathbf{E} [(\hat{p}_i - p_i)^2] \right)^{1/2} \end{aligned}$$

where the second line uses Lemma C.5 and Assumption 4(b). The last line is  $o_p(1)$  by Assumptions 4(b), 5, and 6(d).

Finally, from the proof of Theorem 3.1 of Farrell (2018),  $R_{2t} = o_p(1)$ . This part of the argument only uses Assumptions 4 and 5.  $\blacksquare$

PROOF OF THEOREM 2. Define

$$\tilde{\sigma}^2 = \frac{1}{m_n} \sum_{i \in \mathcal{M}_n} \sum_{j \in \mathcal{M}_n} \varphi_{t,t'}(i) \varphi_{t,t'}(j) \mathbf{1}\{\ell_{\mathbf{A}}(i, j) \leq b_n\}.$$

We first show that  $|\hat{\sigma}^2 - \tilde{\sigma}^2| \xrightarrow{p} 0$ . For  $\hat{\varphi}_{t,t'}(i) = \hat{\tau}_i(t, t') - \hat{\tau}(t, t')$ ,

$$\begin{aligned} |\hat{\sigma}^2 - \tilde{\sigma}^2| &= \left| \frac{1}{m_n} \sum_{i \in \mathcal{M}_n} (\hat{\varphi}_{t,t'}(i) - \varphi_{t,t'}(i)) \sum_{j \in \mathcal{M}_n} (\hat{\varphi}_{t,t'}(j) + \varphi_{t,t'}(j)) \mathbf{1}\{\ell_{\mathbf{A}}(i, j) \leq b_n\} \right| \\ &\leq \frac{n}{m_n} \left( \frac{1}{n} \sum_{i=1}^n (\hat{\varphi}_{t,t'}(i) - \varphi_{t,t'}(i))^2 \frac{1}{n} \sum_{i=1}^n \max_{j \in \mathcal{N}_n} (\hat{\varphi}_{t,t'}(j) + \varphi_{t,t'}(j))^2 n(i, b_n)^2 \right)^{1/2}. \quad (\text{D.1}) \end{aligned}$$

By Assumptions 4(b) and 7(a) and (e), for some universal  $C > 0$

$$\frac{1}{n} \sum_{i=1}^n \max_{j \in \mathcal{N}_n} (\hat{\varphi}_{t,t'}(j) + \varphi_{t,t'}(j))^2 n(i, b_n)^2 \leq C \frac{1}{n} \sum_{i=1}^n n(i, b_n)^2 = O_p(\sqrt{n}). \quad (\text{D.2})$$

We next show that

$$\frac{1}{n} \sum_{i=1}^n (\hat{\varphi}_{t,t'}(i) - \varphi_{t,t'}(i))^2 = o_p(n^{-1/2}), \quad (\text{D.3})$$

Abbreviate  $\mu_t(i) = \mu_t(i, \mathbf{X}, \mathbf{A})$ ,  $p_t(i) = p_t(i, \mathbf{X}, \mathbf{A})$ ,  $\hat{\mu}_t(i) = \hat{\mu}_t(i, \mathbf{X}, \mathbf{A})$ ,  $\hat{p}_t(i) =$

$\hat{p}_t(i, \mathbf{X}, \mathbf{A})$ , and

$$\Delta_i(t) = (\hat{\mu}_t(i) - \mu_t(i)) \frac{p_t(i) - \mathbf{1}_i(t)}{p_t(i)} - \frac{\mathbf{1}_i(t)(Y_i - \hat{\mu}_t(i))(\hat{p}_t(i) - p_t(i))}{\hat{p}_t(i)p_t(i)}.$$

The left-hand side of (D.3) equals

$$\frac{1}{n} \sum_{i=1}^n \left( \Delta_i(t) - \Delta_i(t') - \hat{\tau}(t, t') + \tau(t, t') \right)^2.$$

Using Assumption 7(a) and (b) and Theorem 1, this is  $o_p(n^{-1/2})$ , which establishes (D.3). It follows from (D.2) and Assumption 4(b) that (D.1) =  $o_p(1)$ .

Next, the proof of Theorem 4 of [Leung \(2022a\)](#) can be applied to show that

$$\tilde{\sigma}^2 = \hat{\sigma}_*^2 + R_n + o_p(1).$$

The argument follows from substituting  $\tilde{\varphi}_{t,t'}(i)$  for his  $Z_i - \tau_i(t, t')$  and our Assumptions 7(d)–(f) for his Assumptions 7(b)–(d).

Finally, we apply Proposition 4.1 of [Kojevnikov et al. \(2021\)](#) to show  $|\hat{\sigma}_*^2 - \sigma_n^2| \xrightarrow{p} 0$ . First,  $\mathbf{E}[\tilde{\varphi}_{t,t'}(i) \mid \mathbf{X}, \mathbf{A}] = 0$  under Assumption 1, as required by their setup. Their Assumption 2.1 is a consequence of our Lemma C.4 and Assumption 6(c). Their Assumption 4.1(i) is satisfied due to our Assumption 7(a). Their Assumption 4.1(ii) is a consequence of their Proposition 4.2. Lastly, their Assumption 4.1(iii) corresponds to our Assumption 7(c).  $\blacksquare$

**PROOF OF THEOREM 3.** Under the assumptions of the theorem, Lemma C.1 holds. The average (over  $i$ ) of the square right-hand side of (C.1) is at most of order  $e^{-2\alpha s} n^{-1} \sum_{i=1}^n n(i, 1)^2$ , which is  $o_p(n^{-1/2})$  if  $s = ((4 - \epsilon)\alpha)^{-1} \log n$ . From the left-hand side of (C.1), this choice of  $s$  corresponds to  $L = r_\lambda(((4 - \epsilon)\alpha)^{-1} \log n + 1)$ .

The average (over  $i$ ) of the square of the right-hand side of (C.2) is at most of order  $e^{-\alpha s} (n^{-1} \sum_{i=1}^n n(i, 1)^2 + n^{-1} \sum_{i=1}^n \Lambda_n(i, s/2)^2 n(i, s/2)^2)$ . Under Assumption 7(a), we satisfy Assumption 6(b) by choosing  $\lambda_n(i, s/2) = 2M$ . Then the assumptions of the theorem imply that this is  $o_p(n^{-1/2})$  if  $s = ((2 - \epsilon)(\alpha - \xi))^{-1} \log n$ . From the left-hand side of (C.2), this corresponds to  $L = r_\lambda(((2 - \epsilon)(\alpha - \xi))^{-1} \log n + 1)$ .  $\blacksquare$

## References

- Alon, U. and E. Yahav**, “On the Bottleneck of Graph Neural Networks and its Practical Implications,” in “International Conference on Learning Representations” 2021.
- Aronow, P. and C. Samii**, “Estimating Average Causal Effects Under General Interference, with Application to a Social Network Experiment,” *Annals of Applied Statistics*, 2017, 11 (4), 1912–1947.
- Athey, S., D. Eckles, and G. Imbens**, “Exact  $p$ -Values for Network Interference,” *Journal of the American Statistical Association*, 2018, 113 (521), 230–240.
- , **G. Imbens, J. Metzger, and E. Munro**, “Using Wasserstein Generative Adversarial Networks for the Design of Monte Carlo Simulations,” *Journal of Econometrics*, 2021.
- Auerbach, E.**, “Identification and Estimation of a Partially Linear Regression Model Using Network Data,” *Econometrica*, 2022, 90 (1), 347–365.
- and **M. Tabord-Meehan**, “The Local Approach to Causal Inference Under Network Interference,” *arXiv preprint arXiv:2105.03810*, 2023.
- Azizian, W. and M. Lelarge**, “Expressive Power of Invariant and Equivariant Graph Neural Networks,” in “International Conference on Learning Representations” 2021.
- Babai, L., P. Erdős, and S. Selkow**, “Random Graph Isomorphism,” *SIAM Journal on Computing*, 1980, 9 (3), 628–635.
- Balat, J. and S. Han**, “Multiple Treatments with Strategic Substitutes,” *Journal of Econometrics*, 2023, 234 (2), 732–757.
- Banerjee, A., A. Chandrasekhar, E. Duflo, and M. Jackson**, “The Diffusion of Microfinance,” *Science*, 2013, 341 (6144), 1236498.
- Belloni, A., V. Chernozhukov, and C. Hansen**, “Inference on Treatment Effects After Selection Among High-Dimensional Controls,” *Review of Economic Studies*, 2014, 81 (2), 608–650.

- Bronstein, M.**, “Do We Need Deep Graph Neural Networks?,” <https://towardsdatascience.com/do-we-need-deep-graph-neural-networks-be62d3ec5c59> 2020. Accessed: 2022-07-02.
- , **J. Bruna, T. Cohen, and P. Veličković**, “Geometric Deep Learning: Grids, Groups, Graphs, Geodesics, and Gauges,” *arXiv preprint arXiv:2104.13478*, 2021.
- Chen, Z., S. Villar, L. Chen, and J. Bruna**, “On the Equivalence Between Graph Isomorphism Testing and Function Approximation with GNNs,” in “Advances in Neural Information Processing Systems,” Vol. 32 2019.
- Chernozhukov, V., D. Chetverikov, M. Demirer, E. Duflo, C. Hansen, W. Newey, and J. Robins**, “Double/Debiased Machine Learning for Treatment and Structural Parameters,” *The Econometrics Journal*, 2018, 21, C1–C68.
- Corso, G., L. Cavalleri, D. Beaini, P. Liò, and P. Veličković**, “Principal Neighbourhood Aggregation for Graph Nets,” in “Advances in Neural Information Processing Systems,” Vol. 33 2020, pp. 13260–13271.
- DeGroot, M.**, “Reaching a Consensus,” *Journal of the American Statistical Association*, 1974, 69 (345), 118–121.
- DiTraglia, F., C. Garcia-Jimeno, R. O’Keeffe-O’Donovan, and A. Sanchez-Becerra**, “Identifying Causal Effects in Experiments with Spillovers and Non-compliance,” *Journal of Econometrics*, 2023, 235 (2), 1589–1624.
- Dwivedi, V., C. Joshi, T. Laurent, Y. Bengio, and X. Bresson**, “Benchmarking Graph Neural Networks,” *arXiv preprint arXiv:2003.00982*, 2022.
- Emmenegger, C., M. Spohn, and P. Bühlmann**, “Treatment Effect Estimation from Observational Network Data Using Augmented Inverse Probability Weighting and Machine Learning,” *arXiv preprint arXiv:2206.14591*, 2022.
- Farrell, M.**, “Robust Inference on Average Treatment Effects with Possibly more Covariates than Observations,” *arXiv preprint arXiv:1309.4686v3*, 2018.
- , **T. Liang, and S. Misra**, “Deep Neural Networks for Estimation and Inference,” *Econometrica*, 2021, 89 (1), 181–213.

- Fey, M. and J. Lenssen**, “Fast Graph Representation Learning with PyTorch Geometric,” *arXiv preprint arXiv:1903.02428*, 2019.
- Forastiere, L., E. Airoidi, and F. Mealli**, “Identification and Estimation of Treatment and Interference Effects in Observational Studies on Networks,” *Journal of the American Statistical Association*, 2021, 116 (534), 901–918.
- Grohe, M.**, “The Logic of Graph Neural Networks,” in “2021 36th Annual ACM/IEEE Symposium on Logic in Computer Science” IEEE 2021, pp. 1–17.
- He, X. and K. Song**, “Measuring Diffusion over a Large Network,” *Review of Economic Studies (forthcoming)*, 2024.
- Hornik, K., M. Stinchcombe, and H. White**, “Multilayer Feedforward Networks are Universal Approximators,” *Neural Networks*, 1989, 2 (5), 359–366.
- Hoshino, T. and T. Yanagi**, “Causal Inference with Noncompliance and Unknown Interference,” *Journal of the American Statistical Association*, 2023, pp. 1–12.
- and —, “Treatment Effect Models with Strategic Interaction in Treatment Decisions,” *Journal of Econometrics*, 2023, 236 (2), 105495.
- Imbens, G.**, “The Role of the Propensity Score in Estimating Dose-Response Functions,” *Biometrika*, 2000, 87 (3), 706–710.
- Jackson, M., Z. Lin, and N. Yu**, “Adjusting for Peer-Influence in Propensity Scoring when Estimating Treatment Effects,” *SSRN 3522256*, 2020.
- Jegelka, S.**, “Theory of Graph Neural Networks: Representation and Learning,” *arXiv preprint arXiv:2204.07697*, 2022.
- Kaji, T., E. Manresa, and G. Pouliot**, “An Adversarial Approach to Structural Estimation,” *arXiv preprint arXiv:2007.06169*, 2020.
- Kang, H. and G. Imbens**, “Peer Encouragement Designs in Causal Inference with Partial Interference and Identification of Local Average Network Effects,” *arXiv preprint arXiv:1609.04464*, 2016.

- Kiefer, S. and B. McKay**, “The Iteration Number of Colour Refinement,” in “47th International Colloquium on Automata, Languages, and Programming,” Vol. 168 2020, p. 73.
- Kim, B.**, “Analysis of Randomized Experiments with Network Interference and Non-compliance,” *arXiv preprint arXiv:2012.13710*, 2020.
- Kobler, J., U. Schöning, and J. Torán**, *The Graph Isomorphism Problem: Its Structural Complexity*, Springer Science & Business Media, 2012.
- Kojevnikov, D., V. Marmar, and K. Song**, “Limit Theorems for Network Dependent Random Variables,” *Journal of Econometrics*, 2021, *222* (2), 882–908.
- Kriege, N., F. Johansson, and C. Morris**, “A Survey on Graph Kernels,” *Applied Network Science*, 2020, *5* (1), 1–42.
- Leung, M.**, “Causal Inference Under Approximate Neighborhood Interference,” *arXiv preprint arXiv:1911.07085v1*, 2019.
- , “Causal Inference Under Approximate Neighborhood Interference,” *Econometrica*, 2022, *90* (1), 267–293.
- , “Rate-Optimal Cluster-Randomized Designs for Spatial Interference,” *Annals of Statistics*, 2022, *50* (5), 3064–3087.
- , “Causal Interpretation of Estimands Defined by Exposure Mappings,” *arXiv preprint arXiv:2403.08183*, 2024.
- **and R. Moon**, “Normal Approximation in Large Network Models,” *arXiv preprint arXiv:1904.11060*, 2023.
- Li, Q., Z. Han, and X. Wu**, “Deeper Insights into Graph Convolutional Networks for Semi-Supervised Learning,” in “32nd AAAI Conference on Artificial Intelligence” 2018.
- Li, S. and S. Wager**, “Random Graph Asymptotics for Treatment Effect Estimation Under Network Interference,” *The Annals of Statistics*, 2022, *50* (4), 2334–2358.

- Lin, Z. and F. Vella**, “Selection and Endogenous Treatment Models with Social Interactions: An Application to the Impact of Exercise on Self-Esteem,” *IZA DP No. 14167*, 2021.
- Liu, L., M. Hudgens, B. Saul, J. Clemens, M. Ali, and M. Emch**, “Doubly Robust Estimation in Observational Studies with Partial Interference,” *Stat*, 2019, 8 (1), e214.
- Manski, C.**, “Identification of Endogenous Social Effects: The Reflection Problem,” *Review of Economic Studies*, 1993, 60 (3), 531–542.
- , “Identification of Treatment Response with Social Interactions,” *The Econometrics Journal*, 2013, 16 (1), S1–S23.
- Maron, H., H. Ben-Hamu, H. Serviansky, and Y. Lipman**, “Provably Powerful Graph Networks,” in “Advances in Neural Information Processing Systems,” Vol. 32 2019.
- Morris, C., M. Ritzert, M. Fey, W. Hamilton, J. Lenssen, G. Rattan, and M. Grohe**, “Weisfeiler and Leman Go Neural: Higher-Order Graph Neural Networks,” in “Proceedings of the AAAI Conference on Artificial Intelligence,” Vol. 33 2019, pp. 4602–4609.
- , **Y. Lipman, H. Maron, B. Rieck, N. Kriege, M. Grohe, M. Fey, and K. Borgwardt**, “Weisfeiler and Leman Go Machine Learning: The Story So Far,” *arXiv preprint arXiv:2112.09992*, 2021.
- Ogburn, E., O. Sofrygin, I. Diaz, and M. van der Laan**, “Causal Inference for Social Network Data,” *arXiv preprint arXiv:1705.08527*, 2022.
- Oono, K. and T. Suzuki**, “Graph Neural Networks Exponentially Lose Expressive Power for Node Classification,” in “International Conference on Learning Representations” 2020.
- Paszke, A., S. Gross, F. Massa, A. Lerer, J. Bradbury et al.**, “PyTorch: An Imperative Style, High-Performance Deep Learning Library,” in “Advances in Neural Information Processing Systems 32,” Curran Associates, Inc., 2019, pp. 8024–8035.



- Pollmann, M.**, “Causal Inference for Spatial Treatments,” *arXiv preprint arXiv:2011.00373*, 2023.
- Qu, Z., R. Xiong, J. Liu, and G. Imbens**, “Efficient Treatment Effect Estimation in Observational Studies under Heterogeneous Partial Interference,” *arXiv preprint arXiv:2107.12420*, 2022.
- Sánchez-Becerra, A.**, “Spillovers, Homophily, and Selection into Treatment: The Network Propensity Score,” *arXiv preprint arXiv:2209.14391*, 2022.
- Sävje, F.**, “Causal Inference with Misspecified Exposure Mappings,” *Biometrika*, 2024, *111* (1), 1–15.
- Topping, J., F. Di Giovanni, B. Chamberlain, X. Dong, and M. Bronstein**, “Understanding Over-Squashing and Bottlenecks on Graphs via Curvature,” in “International Conference on Learning Representations” 2022.
- Toulis, P. and E. Kao**, “Estimation of Causal Peer Influence Effects,” in “International Conference on Machine Learning” 2013, pp. 1489–1497.
- Veitch, V., Y. Wang, and D. Blei**, “Using Embeddings to Correct for Unobserved Confounding in Networks,” in “Advances in Neural Information Processing Systems,” Vol. 32 2019.
- Wu, Z., S. Pan, F. Chen, G. Long, C. Zhang, and S. Philip**, “A Comprehensive Survey on Graph Neural Networks,” *IEEE Transactions on Neural Networks and Learning Systems*, 2020, *32* (1), 4–24.
- Xu, H.**, “Social Interactions in Large Networks: A Game Theoretic Approach,” *International Economic Review*, 2018, *59* (1), 257–284.
- Xu, K., W. Hu, J. Leskovec, and S. Jegelka**, “How Powerful are Graph Neural Networks?,” in “International Conference on Learning Representations” 2018.
- Zhou, K., Y. Dong, K. Wang, W. Lee, B. Hooi, H. Xu, and J. Feng**, “Understanding and Resolving Performance Degradation in Deep Graph Convolutional Networks,” in “Proceedings of the 30th ACM International Conference on Information & Knowledge Management” 2021, pp. 2728–2737.

**Zopf, M.**, “1-WL Expressiveness Is (Almost) All You Need,” *arXiv preprint arXiv:2202.10156*, 2022.