GEOMETRIC MUTUAL INFORMATION FOR CAUSAL EFFECT ESTIMATION UNDER NETWORK INTERFERENCE

Anonymous authors
Paper under double-blind review

ABSTRACT

Estimating causal effect on network data considers the situation with the presence of dependencies between individuals. To reduce complex confounding bias caused by network interference, existing methods usually adopt the same information aggregation function for covariates and treatments. However, these methods fail to exploit different geometric structures in the spaces of covariates and treatments, which are vital for information aggregation on network data. In this paper, starting from the causal effect estimation error, we propose a geometric mutual information model to learn balanced representations, so that different geometric structures involved in covariates and treatments are leveraged in representation learning and information aggregation. Specifically, we theoretically reveal that balanced representation learning on network data can be achieved by maximizing the dissimilarity between the geometric structures of covariates and treatments measured by the Gromov-Wasserstein discrepancy. To better characterize the geometry of the network data, we apply the Ricci flow to refine the graph structures for representation learning as well as information aggregation. We conduct experiments on benchmark datasets to demonstrate the effectiveness of our method.

1 INTRODUCTION

Causal effect estimation from observational data aims to evaluate the effect of a treatment based on non-randomized controlled trial data (Yao et al., 2021; Winship & Morgan, 1999). Classic methods rely on the Stable Unit Treatment Value Assumption (SUTVA), *i.e.*, individuals are independent with each other (Rubin, 1980). Nevertheless, there could exist dependencies between individuals in practice, which means that the potential outcome of one unit is affected by the neighbors (Jiang & Sun, 2022). For example, in the domain of epidemiology (Hudgens & Halloran, 2008) or advertisement (Aral & Walker, 2011), the outcome of a person is influenced by the neighbors or friends. Such network interference breaks the SUTVA and introduces extra confounding bias to causal effect estimation.

In order to reduce the complex confounding bias caused by network interference, existing methods usually combine the information of neighbors to learn balanced weights or representations (Ma & Tresp, 2021; Cai et al., 2023). Based on the adjacency matrix that indicates the network structure, Jiang & Sun (2022) aggregated covariates and treatments to learn balanced representations. To further consider heterogeneous interference from neighbors, Zhao et al. (2024) learned attention weights to aggregate covariates and treatments of neighbors. However, these methods adopt the identical aggregation function for covariates and treatments without considering their different geometric structures. Since covariates and treatments are defined in different spaces, they are expected to be associated with different underlying geometric structures, leading to distinct information aggregation mechanisms.

In this paper, we seek to exploit specific geometric structures involved in covariates and treatments to reduce complex confounding bias under the network interference. To this end, we first analyze the effect estimation error with respect to the mutual information between representations and treatments, and then theoretically reveal that balanced representation learning can be achieved by maximizing the dissimilarity between the distinct geometric structures of covariates and treatments. Specifically, the geometric dissimilarity is measured by the Gromov-Wasserstein discrepancy between two measure-metric spaces, which has shown a strong ability to characterize geometric information (Mémoli,

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

2011; Peyré et al., 2016). Motivated by this, we propose a geometric mutual information (GEOMINI) model to exploit geometric structures of covariates and treatments to learn balanced representations. Intuitively, GEOMINI minimizes the similarity between the graph structures of covariates and treatments, in which the dependencies between individuals are captured by the geometry in the graphs. By doing this, the learned representations are invariant to the treatments, reducing the confounding bias on network data.

To better extract geometric structures on network data, we refine the geometric structures of covariates and treatments based on Ricci flow (Hopper & Andrews, 2010). Equipped with Ricci curvature (Ollivier, 2009; Farzam et al., 2024), Ricci flow provides a powerful geometric tool and has been widely applied to graph rewiring (Jin et al., 2008; Nguyen et al., 2023; Ni et al., 2019). Inspired by this, we employ the Ricci flow to refine the graph structures of covariates and treatments, respectively, which are incorporated into our model for balanced representation learning, and are also leveraged for information aggregation of covariates and treatments. Finally, we achieve an optimization objective to learn an outcome prediction model and balanced representations jointly. We summarize our major contributions as follows:

- Motivated by our analysis of effect estimation error regarding the geometric dissimilarity between covariates and treatments, we propose a geometric mutual information model to learn balanced representations for network data.
- Inspired by Ricci flow, we characterize the geometric structures of covariates and treatments for balanced representation learning as well as information aggregation.
- We conduct experiments on two benchmark datasets in terms of different evaluation metrics to demonstrate the effectiveness of our proposed method.

2 Related Works

Causal Inference on Independent data. Traditional causal inference typically relies on the Stable Unit Treatment Value Assumption (SUTVA), which assumes that the potential outcomes of any individual unit are independent of the treatment assignment of others (Rubin, 1980; Imbens & Rubin, 2015). There are two classes of methods to eliminate the confounding bias on independent data. The first class learns sample weights to establish pseudo-balanced groups, *e.g.*, the inverse of propensity weights (Rosenbaum & Rubin, 1983; Robins et al., 1994), or weights to minimize distribution discrepancy, such as maximum mean discrepancy (Kuang et al., 2017) or optimal transport cost (Yan et al., 2024). The other class learns balanced representations based on neural networks to eliminate the confounding bias. This can be achieved by applying different distribution balancing/aligning metrics, *e.g.*, Integral probability metric (Johansson et al., 2016; Shalit et al., 2017), mutual information (Yang et al., 2024), or Gromov-Wasserstein discrepancy (Wang et al., 2025).

Different from them, we focus on the more complex settings where network interference exists between units. In this setting, balancing representation is much more difficult due to complex confounding bias from interference. To address this, we propose a geometric mutual information model between the graphs of treatment and covariates to effectively achieve balance.

Causal Inference on Network Data. There has been considerable recent interest in adapting methods from independent data to network data. Many works extend classic methods to this setting by considering the neighbor covariates as extra confounders. One line of work is the weighting methods (Liu et al., 2016; Forastiere et al., 2021). Another line considers balancing representation learning. Since interference exists, both individual treatment and neighbor treatments are considered as interventions, leading to complex balancing strategies, including adversarial learning (Jiang & Sun, 2022), Hilbert-Schmidt Independence Criterion optimization (Ma & Tresp, 2021), and reweighted representation learning (Cai et al., 2023). Moreover, other methods consider constructing doubly robust estimators (Liu et al., 2023; Ogburn et al., 2024; Chen et al., 2024; Khatami et al., 2025), learning effects on hypergraph (Ma et al., 2022), existence of unobserved confounders (Veitch et al., 2019; Chen et al., 2025; Egami & Tchetgen Tchetgen, 2024; Huang et al., 2023), and extending to heterogeneous interference (Zhao et al., 2024) or unknown interference (Wu et al., 2025).

However, existing methods assume the identical aggregation function for covariates and treatments without considering the different geometric structures. To relax that, we introduce a geometric mutual

information model that maximizes the dissimilarity between the graph structures of covariates and treatments, thereby exploiting inter-individual dependencies in the process of learning balanced representations and information aggregation.

3 Problem Statement

3.1 Notations

Let $X \in \mathcal{X}$ denote the covariates, $T \in \{0,1\}$ denote a binary treatment variable where $T = 1$ indicates a unit is treated and $T = 0$ indicates a unit receives no treatment, and $Y \in \mathcal{Y} \subseteq \mathbb{R}$ be the outcome variable. For the i-th unit, the observed variables are denoted by the triple $\{x_i, t_i, y_i\}$, and the network dataset is denoted as $\mathcal{D} = (\{x_i, t_i, y_i\}_{i=1}^N, A)$, where N denotes the total number of units, and $A \in \{0,1\}^{N\times N}$ denotes the adjacency matrix of the network. Due to the presence of network interference, a unit's potential outcome is influenced not only by its own treatment t_i and own covariates x_i but also by the treatments and covariates of its neighbors $t_{\mathcal{N}_i}$ and $x_{\mathcal{N}_i}$, where $\mathcal{N}_i$ is the neighbor set of the unit i, $t_{\mathcal{N}_i} = \{t_j\}_{j\in\mathcal{N}_i}$ and $x_{\mathcal{N}_i} = \{x_j\}_{j\in\mathcal{N}_i}$ are the neighbor sets of treatments and covariates, respectively. Consequently, the potential outcome is formally denoted as $y_i(t_i, t_{\mathcal{N}_i})$. The single observed outcome for unit i, y_i, is referred to as the factual outcome, while the remaining unobserved potential outcomes constitute the counterfactual outcomes.

Following (Forastiere et al., 2021), we assume that the dependence between the potential outcome and the neighbors' treatments is through a specified summary function $agg : \{0,1\}^{|\mathcal{N}_i|} \to [0,1]$, and z_i is the peer exposure given by the summary function, *i.e.*, $z_i = agg(t_{\mathcal{N}_i})$. Therefore, the potential outcome $y_i(t_i, t_{\mathcal{N}_i})$ can be denoted as $y_i(t_i, z_i)$, which means that under network interference, each unit is affected by two kinds of treatments: the binary individual treatment t_i and the continuous neighborhood exposure z_i. For simplicity, we use $\mathbf{x} \in \mathcal{X} \times \mathcal{X}^{|\mathcal{N}|}$ to denote unit i's covariates along with neighbors' covariates, *i.e.*, $\mathbf{x}_i = (x_i, x_{\mathcal{N}_i})$. Similarly, $\mathbf{t}_i = (t_i, z_i)$ is the treatments of the unit i and the peer exposure.

3.2 Assumptions and Target Estimands

We adopt standard identification assumptions for causal inference under network interference, including network consistency, overlap, neighborhood interference, and unconfoundedness (Jiang & Sun, 2022; Cai et al., 2023; Chen et al., 2024). We define the conditional causal effect between (t, z) and (t', z') as

$$\tau(\mathbf{x}_i)^{(t,z),(t',z')} = f(\mathbf{x}_i, t, z) - f(\mathbf{x}_i, t', z'), \tag{1}$$

where $f(\mathbf{x}_i, t, z) = \mathbb{E}[y_i \mid t_i = t, z_i = z, \mathbf{x}_i]$.

Following prior work, we consider:

Direct effect. The direct effect measures the causal effect of changing unit i's own treatment from untreated to treated while holding its neighborhood exposure fixed at z. It therefore isolates the effect attributable to the unit's own treatment under a given neighborhood environment:$\tau_{DE}(\mathbf{x}_i) = f(\mathbf{x}_i, 1, z) - f(\mathbf{x}_i, 0, z)$,

Spillover effect. The spillover effect captures the causal effect of changing the neighborhood exposure from 0 to z while keeping unit i untreated. This contrast measures how the treatments received by neighboring units affect an untreated focal unit: $\tau_{SE}(\mathbf{x}_i) = f(\mathbf{x}_i, 0, z) - f(\mathbf{x}_i, 0, 0)$,

Total effect. The total effect compares the joint treatment condition $(T_i, Z_i) = (1, z)$ with the baseline condition $(T_i, Z_i) = (0, 0)$. It represents the overall causal effect of simultaneously treating the focal unit and changing its neighborhood exposure from 0 to z: $\tau_{TE}(\mathbf{x}_i) = f(\mathbf{x}_i, 1, z) - f(\mathbf{x}_i, 0, 0)$.

Formal assumptions and definitions are provided in Appendix A.

4 METHOD

4.1 GEOMETRIC MUTUAL INFORMATION

In this part, we first analyze the error bound of causal effect estimation from the perspective of mutual information, and then theoretically reveal the connection between balanced representation learning and geometric discrepancy of representations and treatments.

For two variables A and B, the mutual information between them is defined by the following.

Definition 4.1. Let A and B be random variables defined on a common probability space with joint distribution $p_{A,B}$ and marginal distributions p_A and p_B. With probability density functions $p(a,b)$, $p(a)$, and $p(b)$, the mutual information $I(A;B)$ is defined as:

$$I(A;B) = KL(p_{A,B} \| p_A, p_B) = \iint_{A\times B} p(a,b) \log \frac{p(a,b)}{p(a)p(b)} dadb, \tag{2}$$

where $KL(\cdot\|\cdot)$ is the Kullback-Leibler divergence. Mutual information measures the statistical dependency between two variables. Based on (Chuang et al., 2023; Yang et al., 2024), we define a mutual information estimator $\hat{I}(A;B)$ via Kernel Density Estimation (KDE):

$$\hat{I}(A,B) = \frac{1}{N} \sum_i \log \frac{\frac{1}{N}\sum_j k(a_i,a_j)k(b_i,b_j)}{\frac{1}{N}\sum_j k(a_i,a_j)\frac{1}{N}\sum_j k(b_i,b_j)}, \tag{3}$$

where $k(\cdot,\cdot)$ is a positive definite kernel and implemented as the Gaussian kernel as follows

$$k(x,y) = \exp\big(-d^2(x,y)/\sigma^2\big), \tag{4}$$

where $d(\cdot,\cdot)$ denotes a distance metric, and $\sigma > 0$ is the kernel bandwidth controlling the smoothness of the density estimate.

Now we analyze the error bound of causal effect estimation based on mutual information between representations and treatments. Given the covariate space $\mathcal{X}$, let $\mathcal{R}$ be the representation space, and $\phi : \mathcal{X} \to \mathcal{R}^E$ be an invertible mapping between the two spaces. Without ambiguity, ϕ is performed in an individual-wise manner, such that $\phi(x_{\mathcal{N}_i})$ maps each $x_j \in x_{\mathcal{N}_i}$ to $\mathcal{R}$, thereby obtaining a representation set $\{\phi(x_j)\}_{j\in\mathcal{N}_i}$. For each individual, we further define its neighborhood representation obtained by aggregating representations from its neighbors via an aggregation function $\psi : \mathcal{R}^{|\mathcal{N}|} \to \tilde{\mathcal{R}}$. Based on the learned representations, the potential outcome is predicted by the model $h : \mathcal{R} \times \tilde{\mathcal{R}} \times \{0,1\} \times [0,1] \to \mathcal{Y}$. Let $\hat{\tau}_h(\mathbf{x})^{\mathbf{t},\mathbf{t}'}$ be the estimated ITE and $\tau(\mathbf{x})^{\mathbf{t},\mathbf{t}'}$ be the ground-truth ITE in Eq. (21), we adopt the Precision in Estimation of Heterogeneous Effect (PEHE) as the evaluation metric of causal effect estimation (Hill, 2011), which is defined as $\varepsilon_{PEHE} = \mathbb{E}_{p(\mathbf{t})p(\mathbf{t}')}[\mathbb{E}_{p(\mathbf{x})}[(\hat{\tau}_h(\mathbf{x}_i)^{\mathbf{t},\mathbf{t}'} - \tau(\mathbf{x}_i)^{\mathbf{t},\mathbf{t}'})^2]]$.

Based on the above definitions, we provide the following theorem to establish an error bound on ε_{PEHE} that explicitly depends on the mutual information between the learned representations and the treatment assignments:

Theorem 4.2. *Let $\varepsilon_F = \int_{\mathcal{X}}\int_{\mathcal{T}}(h(\phi(x),\psi(\phi(x_{\mathcal{N}})),t,z) - f(\mathbf{x},\mathbf{t}))^2 p(\mathbf{x},\mathbf{t})d\mathbf{x}d\mathbf{t}$ represent the factual loss, ε_{PEHE} is bounded as follows:*

$$\varepsilon_{PEHE} \le 4\varepsilon_F + 4B^2\sqrt{2I((\phi(X),\phi(X_{\mathcal{N}}));(T,Z))}, \tag{5}$$

where $I((\phi(X),\phi(X_{\mathcal{N}}));(T,Z))$ is the Mutual Information between the representation $(\phi(X),\phi(X_{\mathcal{N}}))$ and the treatment assignments, B is a constant related to the upper bound of the outcome functions.

We provide the details of proof in Appendix D.1.

From Theorem 4.2, we observe that causal effect estimation error is upper bounded by the factual loss ε_F and the mutual information term $I((\phi(X),\phi(X_{\mathcal{N}});(T,Z))$. The factual loss reflects the model's ability to fit factual outcomes, which can be easily minimized by the mean squared error loss on observed factual outcomes. However, ε_F is defined on the biased distribution $p(\mathbf{x},\mathbf{t}) = p(\mathbf{x}|\mathbf{t})p(\mathbf{t})$ and excludes the counterfactual loss in ε_{PEHE}, resulting in biased effect estimation. To alleviate this,

the mutual information between the representations $(\phi(X), \phi(X_{\mathcal{N}}))$ and the treatments (T, Z) acts as a regularizer for learning balanced representations that are invariant with respect to the treatments, thereby generalizing the model for both factual and counterfactual prediction. Motivated by this, we can train the model ϕ and h to minimize the prediction error as well as the mutual information term for causal effect estimation. In Appendix B, we further show that such an objective can also be derived from the information theory.

Nevertheless, it is non-trivial to minimize the mutual information term $I((\phi(X), \phi(X_{\mathcal{N}})); (T, Z))$. To tackle this, we provide a geometric upper bound on it in the following theorem, which introduces dependencies between individuals and provides a tractable objective function to optimize. Before that, we define the Gromov-Wasserstein discrepancy that measures the geometric difference between two variables (Mémoli, 2011; Peyré et al., 2016):

$$\widehat{GW}_2^2(A, B) = \min_{\gamma} \sum_{i,i'} \sum_{j,j'} \ell(d^a_{ii'}, d^b_{jj'}) \gamma_{ij} \gamma_{i'j'}, \quad \text{s.t.} \gamma \in \Pi(\mu_A, \mu_B), \tag{6}$$

where $d^a_{ii'} = d(a_i, a_{i'})$ and $d^b_{jj'} = d(b_j, b_{j'})$ are the distances, $\ell(d^a_{ii'}, d^b_{jj'}) = ((d^a_{ii'})^2 - (d^b_{jj'})^2)^2$ is the squared loss, and $\Pi(\mu_A, \mu_B)$ is the set of joint probability distributions of the two marginal distributions. The Gromov-Wasserstein discrepancy incorporates the metric information into an optimal transport model and is capable of comparing two measure-metric spaces (Peyré et al., 2019).

Now, we are ready to provide the geometric upper bound for mutual information as follows:

Theorem 4.3. *Based on the Gaussian kernel density estimation of mutual information in Eq. (3), the mutual information between the learned representations and the treatment assignments admits the following upper bound:*

$$\hat{I}(\phi(X), \phi(X_{\mathcal{N}}); (T, Z)) \leq -\frac{1}{2\sigma^2}\left(2\widehat{GW}_2^2(\phi(X), T) + \widehat{GW}_2^2(T, Z)\right) + \frac{1}{2N^2\sigma^2}\left(2\|D^\phi - D^t\|_F^2 + \|D^t - D^z\|_F^2\right) + o_p(1) + C, \tag{7}$$

where N is the sample size, $\|\cdot\|_F$ is the Frobenius norm, C is a constant independent of ϕ, D^ϕ, D^t, and D^z are distance matrices. The elements of D^ϕ are $\{d^\phi_{ij}\}$, and D^t, D^z are defined similarly.

We provide the details of proof in Appendix D.2.

According to Theorem 4.3, we find that minimizing the mutual information between representations and treatments can be achieved by maximizing the geometric structure discrepancy between their measure-metric spaces. This provides a principled geometric surrogate for enforcing treatment-invariant representations. Intuitively, the Gromov-Wasserstein discrepancy characterizes the dependencies between individuals in the metric spaces, thereby measuring the dissimilarity between the geometric structures of the representations and treatments. By learning representations to maximize the Gromov-Wasserstein term, it is expected to obtain balanced representations invariant to the treatments, reducing the confounding bias on network data. Here, we omit the term regarding D^ϕ in optimization for simplicity, which is also supported by the extremely small relative coefficient $\frac{1}{N^2}$. We also evaluate the performance with the term of D^ϕ in Appendix I, which shows that the performance difference is quite marginal.

The remaining question is how to exploit the geometry information involved in representations and treatments that is required by the Gromov-Wasserstein model. We answer this question in the next part.

4.2 Graph Refinement based on Ricci Flow

The Gromov-Wasserstein discrepancy provides a flexible framework to compare two geometric structures based on the underlying metrics (Peyré et al., 2016). To better capture geometric information on network data, we apply the discrete Ricci flow (Ni et al., 2019; Bai et al., 2024) to refine the graph structures of covariates and treatments. In Riemannian geometry, Ricci flow leverages the Ricci curvature to model the evolution of the underlying geometric structure and has demonstrated a powerful ability in graph rewiring (Topping et al.). Specifically, let $\pi \in \{\phi, t\}$ be the index for

covariates or treatments, for two individuals i and j connected by an edge, the discrete Ricci flow on the graph is given by the following partial differential equation

$$\frac{\partial}{\partial s} d_{ij}^{(\pi)} = -\kappa_{ij}^{(\pi)} d_{ij}^{(\pi)}, \tag{8}$$

where the $d_{ij}^{(\pi)}$ captures the distance and is introduced in the Gromov-Wasserstein discrepancy, and $\kappa_{ij}^{(\pi)}$ is the Ollivier Ricci curvature (Ollivier, 2009) defined as

$$\kappa_{ij}^{(\pi)} = 1 - W(\mu_i^{(\pi)}, \mu_j^{(\pi)}) / d_{ij}^{(\pi)}, \tag{9}$$

and $W(\mu_i^{(\pi)}, \mu_j^{(\pi)})$ is the Wasserstein distance (Peyré et al., 2019; Villani et al., 2008) between the probability measures defined on one-hop neighbors of i and j. Intuitively, $W(\mu_i^{(\pi)}, \mu_j^{(\pi)})$ measures how far one must transport neighborhood mass around i to the neighborhood mass around j, which reflects the local geometric discrepancy between the pair (i, j). Equipped with the Ollivier Ricci curvature, the Ricci flow tends to enlarge the distance of an edge associated with a large $W(\mu_i^{(\pi)}, \mu_j^{(\pi)})$.

In practice, following (Ni et al., 2019) we adopt the Euler discretization of Eq. (8) to update the distance, which reads

$$d_{ij}^{(\pi)}(s) = (1 - \eta \kappa_{ij}^{(\pi)}(s)) d_{ij}^{(\pi)}(s), \tag{10}$$

where s is the iteration index, and η is the step size. For the calculation of curvature in Eq. (9), we construct the probability measures based on a lazy random walk mechanism with the coefficient $\alpha \in (0, 1)$ as follows

$$\mu_{ik}^{(\pi)} = \begin{cases} \alpha, & k = i, \\ \frac{1-\alpha}{M_i} \cdot \exp(-\frac{1}{\sigma^2}(d_{ik}^{(\pi)})^2), & k \in \mathcal{N}_i, \\ 0, & \text{otherwise}, \end{cases} \tag{11}$$

where $\sigma > 0$ is the kernel bandwidth, and $M_i^{(\pi)} = \sum_{k' \in \mathcal{N}_i} \exp(-\frac{1}{\sigma^2}(d_{ik'}^{(\pi)})^2)$ is the normalization factor. Such a construction assigns probability mass α to the central individual i while the remaining mass to the one-hop neighbors based on the Radial Basis Function kernel of the distance $d_{ik'}^{(\pi)}$.

To learn balanced representations $\phi(X)$ that minimize the negative Gromov-Wasserstein discrepancy in Inequality (7), inspired by the differentiable Ricci curvature in (Sun et al., 2023), we calculate $W(\mu_i^{\phi}, \mu_j^{\phi})$ in Eq. (9) based on the distance on $\phi(X)$. For the iterative update in Eq. (10), we associate each edge (i, j) with an initial length $d_{ij}^{(\pi)}(0) = 1$, and update the values until convergence. Finally, we obtain the refined graphs of representations and treatments described by the converged distances d_{ij}^{ϕ} and d_{ij}^{t}, which are leveraged in the Gromov-Wasserstein term $\widehat{GW}_2^2(\phi(X), T)$ in Theorem 4.3.

To leverage the distances for information aggregation, we constructed the weighted adjacency matrix for covariates by performing the softmax operation on the negative distances d_{ij}^{ϕ} as follows

$$\tilde{A}_{ij}^{\phi} = \frac{\exp(-(d_{ij}^{\phi})^2/\sigma^2)}{\sum_{j \in \mathcal{N}_i} \exp(-(d_{ij}^{\phi})^2/\sigma^2)}. \tag{12}$$

Similarly, the weighted adjacency matrix for treatments is obtained by the following

$$\tilde{A}_{ij}^{t} = \frac{\exp(-(d_{ij}^{t})^2/\sigma^2)}{\sum_{j \in \mathcal{N}_i} \exp(-(d_{ij}^{t})^2/\sigma^2)}. \tag{13}$$

By doing this, different geometric structures of representations and treatments are leveraged to refine the graphs, and the neighbors with small distances will contribute more to information aggregation.

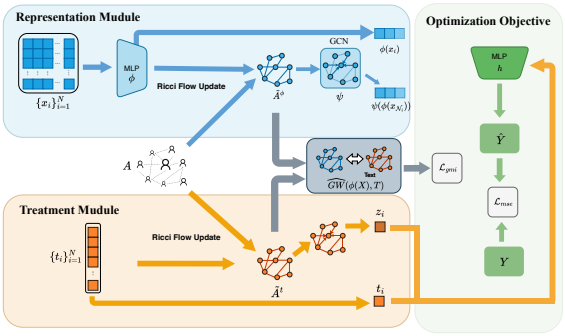


Figure 1: Overview of our proposed method GEOMINI.

4.3 Algorithm

Motivated by Theorems 4.2 and 4.3, we propose to minimize the factual outcome prediction error ε_F and the negative Gromov-Wasserstein discrepancy between representations and treatments, *i.e.*, $-\widehat{GW}_w^2(\phi(X), T)$, in which the geometric structures on $\phi(X)$ and T are constructed based on the graph Ricci flow presented in Eq. (10). We present the overview of our method GEOMINI in Figure 1, including information aggregation, outcome prediction error minimization, and Gromov-Wasserstein discrepancy maximization. We discuss the implementation of our algorithm in the following.

Information aggregation. Based on the refined weighted adjacency matrices in Eqs. (12) and (13), we can aggregate information of representations and treatments to learn balanced representations and predict potential outcomes. Specifically, for the individuals indexed by i and j connected by an edge in A, the representations $\phi(x_i)$ and $\phi(x_j)$ are obtained by the multilayer perceptron model. In order to capture the confounders consisting of representations of neighbors, the representation of the neighbors is aggregated by the following weighting operation

$$\psi(\phi(x_{\mathcal{N}_i})) = \text{GNN}\big(\{\phi(x_j)\}_{j\in\mathcal{N}_i}, \{\tilde{A}^{\phi}_{ij}\}_{j\in\mathcal{N}_i}\big), \tag{14}$$

where $\tilde{A}^{\phi}_{ij}$ is obtained in Eq. (12), and $\psi(\cdot)$ is function implemented by Graph Neural Networks (Defferrard et al., 2016; Kipf, 2016). By doing this, a neighbor j closes to the individual i contribute more compared with others with larger distances.

Similar to the above procedure on representations, the peer exposure from the treatments of neighbors is aggregated based on the weights in Eq. (13) as follows

$$z_i = \sum_{j\in\mathcal{N}_i} \tilde{A}^{t}_{ij} t_j. \tag{15}$$

As a result, different geometric structures of representations and treatments are leveraged for information aggregation.

Outcome prediction error minimization. Based on the above aggregation approach, we can combine the information of individuals and their neighbors to predict potential outcomes. Specifically, for the individual i with the covariates x_i, treatment t_i, and the neighbor set $\mathcal{N}_i$, the predicted potential outcome is obtained by considering the representations $\phi(x_i)$ and $\psi(\phi(x_{\mathcal{N}_i}))$, the treatment t_i as well as the peer exposure z_i, which is given as follows

$$\hat{Y}(\mathbf{x}_i, t_i, z_i) = h\big(\phi(x_i), \psi(\phi(x_{\mathcal{N}_i})), t_i, z_i\big), \tag{16}$$

where the prediction function h is implemented by a multilayer perceptron model. Inspired by the first term in Inequality (5), we minimize the factual outcome prediction error by minimizing the following mean squared error (MSE) loss with respect to ϕ and h

$$\mathcal{L}_{mse} = \frac{1}{N}\sum_{i=1}^{N}(y_i - h(\phi(x_i), \psi(\phi(x_{\mathcal{N}_i})), t_i, z_i))^2. \tag{17}$$

Gromov-Wasserstein discrepancy maximization. Theorem 4.3 shows that balanced representation learning can be achieved by minimizing the negative Gromov-Wasserstein discrepancy between the geometric structures of representations $\phi(X)$ and the treatments T, which motivates us to design the following geometric mutual information loss

$$\mathcal{L}_{gmi} = -\sum_{i=1}^{N}\sum_{i'=1}^{N}\sum_{j=1}^{N}\sum_{j'=1}^{N} \ell(d^{\phi}_{ii'}, d^{t}_{jj'})\gamma^{\star}_{ij}\gamma^{\star}_{i'j'}, \tag{18}$$

where the metric d^{ϕ}_{ij} and d^{t}_{ij} are obtained by the Ricci flow in Eq.(10), and $\gamma^{\star}$ is the solution to the Gromov-Wasserstein problem $\widehat{GW}_2^2(\phi(X), T)$, which is obtained by solving the following problem

$$\gamma^{\star} = \arg\min_{\gamma\in\Pi} \sum_{i=1}^{N}\sum_{i'=1}^{N}\sum_{j=1}^{N}\sum_{j'=1}^{N} \ell(d^{\phi}_{ii'}, d^{t}_{jj'})\gamma_{ij}\gamma_{i'j'} + \rho H(\gamma), \tag{19}$$

where $H(\gamma) = \sum_{ij}\gamma_{ij}(\log\gamma_{ij} - 1)$ is the negative entropy regularization, and the domain of the definition Π is defined as $\Pi = \{\gamma \in \mathbb{R}_{+}^{N\times N} \mid \sum_{j=1}^{N}\gamma_{ij} = \frac{1}{N}, \forall i; \sum_{i=1}^{N}\gamma_{ij} = \frac{1}{N}, \forall j\}$. This optimization problem can be solved by the projected gradient descent algorithm in the KL metric space (Peyré et al., 2016).

Optimization objective. Finally, we train our model by solving the following optimization problem

$$\min_{\phi,h} \ \mathcal{L}_{mse} + \lambda\mathcal{L}_{gmi}, \tag{20}$$

where λ is the trade-off hyperparameter, and the loss functions $\mathcal{L}_{mse}$ and $\mathcal{L}_{gmi}$ are given in Eqs. (17) and (18), respectively.

Appendix C provides the pseudo-code of our method.

5 Experiments

5.1 Datasets and Baselines

In line with prior work (Jiang & Sun, 2022; Chen et al., 2024), we generate semi-synthetic datasets in which the underlying network structure (including features and topology) is derived from real data, while the treatments and potential outcomes are synthetically simulated. These datasets are constructed based on the following two real-world data.

BlogCatalog (BC) is a social network focused on blogging activities, where each node corresponds to a blogger and edges indicate friendship relations. Node attributes are derived from the keyword distributions of the bloggers' articles.

Flickr is an image and video sharing network in which nodes represent users, and edges capture their social connections. User features consist of tag-based interest profiles. Following standard practice, we employ Linear Discriminant Analysis (LDA) to project the original features into a 10-dimensional space.

The details of treatment and potential outcome simulation are provided in Appendix E.

We compare our methods with several baselines, including traditional methods and methods designed for network effect estimation. Traditional methods include **TARNet**, **CFR** (Shalit et al., 2017), and Network Deconfounder (**ND**) (Guo et al., 2020). We also modified **ND** with additionally inputting z, denoted as **ND+z**. We further consider network effect estimators, including **GEst** (Ma & Tresp, 2021), **NetEst** (Jiang & Sun, 2022), **RRNet** (Cai et al., 2023), **TNet** (Chen et al., 2024), **DWR** (Zhao et al., 2024) and **GDML** (Khatami et al., 2024).

5.2 Results and Discussions

Following (Jiang & Sun, 2022; Chen et al., 2024), we adopt the error of ATE ε_{ATE}, and Precision in Estimation of Heterogeneous Effect ε_{PEHE} of Direct Effect, Spillover Effect, and Total Effect, as

Table 1: Results of Direct causal effect estimation on BC Dataset. The PEHE error $\sqrt{\varepsilon_{\text{PEHE}}}$ is reported. The best performing method is bolded, and the second best is underlined.

Interference Degree	Setting	Effect	Methods										
			ND	TARNet+z	CFR+z	ND+z	GEst	NetEst	RRNet	TNet	DWR	GDML	GEOMINI
0.5	Within Sample	Direct	0.0687 ± 0.0311	0.0706 ± 0.0326	0.0644 ± 0.0288	0.0697 ± 0.0299	0.2540 ± 0.0436	0.0681 ± 0.0292	0.1213 ± 0.0141	0.1232 ± 0.0141	0.2460 ± 0.0493	0.2220 ± 0.0921	**0.0410 ± 0.0107**
		Spillover	–	0.1152 ± 0.0510	0.2670 ± 0.0343	0.1295 ± 0.1370	0.4035 ± 0.0537	0.0798 ± 0.0207	0.1526 ± 0.0048	0.0689 ± 0.0070	0.3211 ± 0.0309	0.2765 ± 0.0439	**0.0673 ± 0.0484**
		Total	–	0.1915 ± 0.1186	0.4261 ± 0.0629	0.1908 ± 0.1575	0.2032 ± 0.0483	0.1244 ± 0.0865	0.2378 ± 0.0320	0.1979 ± 0.0627	**0.1231 ± 0.0298**	0.6029 ± 0.1671	0.1286 ± 0.0484
	Out-of Sample	Direct	0.0723 ± 0.0312	0.0710 ± 0.0301	0.0674 ± 0.0304	0.0722 ± 0.0308	0.2533 ± 0.0372	0.0679 ± 0.0276	0.1207 ± 0.0127	0.1279 ± 0.0135	0.2552 ± 0.0505	0.2670 ± 0.0955	**0.0423 ± 0.0104**
		Spillover	–	0.1184 ± 0.0502	0.3649 ± 0.0308	0.1292 ± 0.1313	0.4144 ± 0.0580	0.0785 ± 0.0198	0.1569 ± 0.0058	0.0754 ± 0.0087	0.3170 ± 0.0283	0.3024 ± 0.0420	**0.0696 ± 0.0500**
		Total	–	0.1932 ± 0.1169	0.4260 ± 0.0650	0.1903 ± 0.1582	0.2143 ± 0.0550	0.1238 ± 0.0838	0.2383 ± 0.0311	0.2073 ± 0.0596	**0.1224 ± 0.0295**	0.5849 ± 0.1953	0.1312 ± 0.0477
1	Within Sample	Direct	0.1673 ± 0.0320	0.0839 ± 0.0269	0.0714 ± 0.0247	0.0659 ± 0.0282	0.5344 ± 0.0523	0.0754 ± 0.0200	0.1377 ± 0.0193	0.2239 ± 0.0550	0.3272 ± 0.0414	0.5083 ± 0.1683	**0.0542 ± 0.0087**
		Spillover	–	0.2088 ± 0.1253	0.2889 ± 0.2228	0.1176 ± 0.1122	1.0421 ± 0.1148	0.1213 ± 0.0720	0.2847 ± 0.0091	0.2149 ± 0.0802	0.6169 ± 0.0736	0.4976 ± 0.0528	**0.0845 ± 0.0221**
		Total	–	0.3697 ± 0.1454	0.5246 ± 0.4250	0.1572 ± 0.1384	0.5105 ± 0.0762	0.2306 ± 0.1212	0.3378 ± 0.0623	0.3408 ± 0.1306	**0.1197 ± 0.0542**	1.0167 ± 0.2371	0.1740 ± 0.0435
	Out-of Sample	Direct	0.1639 ± 0.0298	0.0818 ± 0.0264	0.0731 ± 0.0245	0.0661 ± 0.0286	0.5267 ± 0.0552	0.0746 ± 0.0201	0.1408 ± 0.0166	0.2084 ± 0.0467	0.3265 ± 0.0394	0.5323 ± 0.2751	**0.0562 ± 0.0091**
		Spillover	–	0.2109 ± 0.1293	0.2900 ± 0.2215	0.1191 ± 0.1137	1.0279 ± 0.1060	0.1209 ± 0.0718	0.2978 ± 0.0123	0.1808 ± 0.0509	0.6086 ± 0.0787	0.4661 ± 0.0476	**0.0868 ± 0.0228**
		Total	–	0.3658 ± 0.1407	0.5274 ± 0.4249	0.1557 ± 0.1366	0.5043 ± 0.0590	0.2257 ± 0.1161	0.3381 ± 0.0621	0.3158 ± 0.1520	**0.1196 ± 0.0544**	0.9354 ± 0.3185	0.1801 ± 0.0431
2	Within Sample	Direct	0.3520 ± 0.0639	0.1520 ± 0.0374	0.1021 ± 0.0255	0.1322 ± 0.0310	1.0921 ± 0.1065	0.1417 ± 0.0173	0.2102 ± 0.0178	0.3102 ± 0.1781	0.6742 ± 0.0652	0.7814 ± 0.1556	**0.0865 ± 0.0207**
		Spillover	–	0.2326 ± 0.0378	0.2009 ± 0.0719	0.1252 ± 0.0599	2.1404 ± 0.1938	0.1609 ± 0.0493	0.6071 ± 0.0492	0.2496 ± 0.1627	1.1877 ± 0.0725	0.9778 ± 0.1127	**0.1117 ± 0.0403**
		Total	–	0.5930 ± 0.1646	0.4406 ± 0.1998	0.3591 ± 0.1742	1.0029 ± 0.1553	0.4234 ± 0.1439	0.8770 ± 0.2526	0.4395 ± 0.3162	**0.2301 ± 0.0506**	2.1325 ± 0.0581	0.2504 ± 0.0672
	Out-of Sample	Direct	0.3485 ± 0.0640	0.1515 ± 0.0361	0.1019 ± 0.0255	0.1348 ± 0.0322	1.1237 ± 0.0948	0.1449 ± 0.0181	0.2117 ± 0.0149	0.3058 ± 0.1751	0.7209 ± 0.0621	0.7925 ± 0.1697	**0.0891 ± 0.0223**
		Spillover	–	0.2422 ± 0.0390	0.2059 ± 0.0738	0.1312 ± 0.0605	2.2143 ± 0.1575	0.1560 ± 0.0511	0.6139 ± 0.0473	0.2546 ± 0.1648	1.1517 ± 0.0616	0.9795 ± 0.1046	**0.1186 ± 0.0450**
		Total	–	0.5840 ± 0.1623	0.4408 ± 0.1989	0.3597 ± 0.1742	1.0464 ± 0.1368	0.4032 ± 0.1506	0.8754 ± 0.2519	0.4096 ± 0.2751	**0.2378 ± 0.0558**	2.1241 ± 0.0879	0.2534 ± 0.0667

Table 2: Results of PEHE estimation on Flickr Dataset. The PEHE error $\sqrt{\varepsilon_{\text{PEHE}}}$ is reported. The best performing method is bolded, and the second best is underlined.

Interference Degree	Setting	Effect	Methods										
			ND	TARNet+z	CFR+z	ND+z	GEst	NetEst	RRNet	TNet	NetInt	GDML	GEOMINI
0.5	Within Sample	Direct	0.1178 ± 0.0053	0.6955 ± 0.0037	0.1159 ± 0.0082	0.1838 ± 0.0412	0.2568 ± 0.0234	0.0809 ± 0.0255	0.1202 ± 0.0168	0.2115 ± 0.1261	0.1793 ± 0.0226	0.2175 ± 0.0677	**0.0557 ± 0.0012**
		Spillover	–	0.2793 ± 0.0216	0.2805 ± 0.0317	0.3163 ± 0.0356	0.4883 ± 0.0192	**0.1355 ± 0.0777**	0.1538 ± 0.0033	0.1871 ± 0.0481	0.2712 ± 0.0229	0.2360 ± 0.0387	0.1644 ± 0.0239
		Total	–	1.1763 ± 0.0126	0.4797 ± 0.0687	0.5726 ± 0.0753	0.2417 ± 0.0075	0.2103 ± 0.1125	0.2267 ± 0.0262	0.4807 ± 0.1404	**0.0703 ± 0.0472**	0.4976 ± 0.0657	0.2756 ± 0.0235
	Out-of Sample	Direct	0.1169 ± 0.0048	0.7535 ± 0.0052	0.1175 ± 0.0083	0.1614 ± 0.0311	0.2508 ± 0.0276	0.0848 ± 0.0213	0.1213 ± 0.0162	0.2100 ± 0.1182	0.2036 ± 0.0266	0.2317 ± 0.0860	**0.0583 ± 0.0021**
		Spillover	–	0.2770 ± 0.0274	0.2790 ± 0.0375	0.3142 ± 0.0424	0.4826 ± 0.0294	**0.1357 ± 0.0720**	0.1753 ± 0.0068	0.1835 ± 0.0421	0.2570 ± 0.0212	0.2253 ± 0.0262	0.1636 ± 0.0267
		Total	–	1.2448 ± 0.0098	0.4803 ± 0.0705	0.5154 ± 0.0671	0.2422 ± 0.0041	0.2091 ± 0.1108	0.2267 ± 0.0263	0.4704 ± 0.1338	**0.0716 ± 0.0484**	0.4614 ± 0.0797	0.2736 ± 0.0228
1	Within Sample	Direct	0.2784 ± 0.0418	0.6789 ± 0.0353	0.2030 ± 0.0359	0.2289 ± 0.0343	0.5116 ± 0.0690	0.1846 ± 0.0643	0.1654 ± 0.0079	0.6955 ± 0.0037	0.2781 ± 0.0445	0.4577 ± 0.1183	**0.0711 ± 0.0111**
		Spillover	–	0.4983 ± 0.0437	0.4663 ± 0.0102	0.4255 ± 0.0609	0.9488 ± 0.0769	0.1548 ± 0.1248	0.2972 ± 0.0083	0.2793 ± 0.0216	0.5467 ± 0.0173	0.4137 ± 0.0742	**0.0962 ± 0.0393**
		Total	–	1.6505 ± 0.0312	1.0800 ± 0.0237	0.9524 ± 0.0724	0.4273 ± 0.0197	0.4768 ± 0.2667	0.3728 ± 0.0135	1.1763 ± 0.0126	0.2342 ± 0.0708	0.8541 ± 0.0648	**0.2145 ± 0.0527**
	Out-of Sample	Direct	0.2455 ± 0.0340	0.7294 ± 0.0361	0.3043 ± 0.0354	0.2064 ± 0.0340	0.4924 ± 0.0728	0.1787 ± 0.0562	0.1681 ± 0.0088	0.7535 ± 0.0052	0.3070 ± 0.0440	0.4481 ± 0.0913	**0.0740 ± 0.0104**
		Spillover	–	0.4872 ± 0.0534	0.4560 ± 0.0290	0.4140 ± 0.0647	0.9213 ± 0.0960	0.1513 ± 0.1219	0.3360 ± 0.0117	0.2770 ± 0.0274	0.5184 ± 0.0233	0.3942 ± 0.0726	**0.0960 ± 0.0399**
		Total	–	1.7083 ± 0.0274	1.0794 ± 0.0241	0.9065 ± 0.0754	0.4216 ± 0.0376	0.4572 ± 0.2473	0.3729 ± 0.0130	1.2448 ± 0.0098	0.2337 ± 0.0701	0.8241 ± 0.0829	**0.2131 ± 0.0536**
2	Within Sample	Direct	0.4144 ± 0.0624	0.7510 ± 0.0735	0.1529 ± 0.1155	0.2289 ± 0.0343	1.0602 ± 0.0391	0.1476 ± 0.0542	0.1514 ± 0.0362	0.6955 ± 0.0037	0.6509 ± 0.0697	0.9873 ± 0.1283	**0.0653 ± 0.0040**
		Spillover	–	1.0950 ± 0.0371	0.4673 ± 0.4254	0.4255 ± 0.0609	2.1484 ± 0.1211	0.2386 ± 0.0203	0.5933 ± 0.0146	0.2793 ± 0.0216	1.0942 ± 0.0303	0.8317 ± 0.0642	**0.0919 ± 0.0483**
		Total	–	2.7378 ± 0.0728	0.9788 ± 0.9343	0.9524 ± 0.0724	1.0836 ± 0.1859	0.5468 ± 0.2288	0.4839 ± 0.0808	1.1763 ± 0.0126	0.5132 ± 0.0790	1.6281 ± 0.1873	**0.1755 ± 0.1207**
	Out-of Sample	Direct	0.3757 ± 0.0506	0.7945 ± 0.0716	0.1558 ± 0.1085	0.2064 ± 0.0340	1.0971 ± 0.0793	0.1541 ± 0.0641	0.1584 ± 0.0357	0.7535 ± 0.0052	0.7411 ± 0.0931	0.9583 ± 0.1907	**0.0698 ± 0.0036**
		Spillover	–	1.1575 ± 0.0692	0.5090 ± 0.4717	0.4140 ± 0.0647	2.2432 ± 0.1404	0.2125 ± 0.0226	0.7001 ± 0.0112	0.2770 ± 0.0274	1.0315 ± 0.0318	0.8394 ± 0.0628	**0.0999 ± 0.0503**
		Total	–	2.7847 ± 0.0739	0.9780 ± 0.9317	0.9065 ± 0.0754	1.1470 ± 0.1616	0.4910 ± 0.2354	0.4833 ± 0.0800	1.2448 ± 0.0098	0.5314 ± 0.0828	1.6805 ± 0.2598	**0.1747 ± 0.1213**

the evaluation metrics. More details of metrics can be found in Appendix F. We discuss the major results in the following, and provide more results of ε_{ATE} in Appendix G.

Comparison with Baselines. Tables 1 and 2 present the results of PEHE on the BC and Flickr datasets, respectively. Overall, GEOMINI shows superior performance than existing methods, indicating the effectiveness of the proposed method. Specifically, compared with traditional estimators, GEOMINI explicitly models network interference, effectively capturing network effects. Compared with methods that account for network interference, such as NetEst and TNet, GEOMINI achieves better performance on both datasets. This is because our method considers the geometric information of distinct graphs between covariates and between treatments, therefore achieving better representation balancing and lower estimation errors. This necessitates the consideration of accounting for geometric information in network effect estimation.

Robust to Interference Degree. Tables 1 and 2 report the results under varying interference degrees in $\{0.5, 1, 2\}$. As the interference degree increases, all methods perform worse, while GEOMINI stably performs better than the baselines, further supporting its effectiveness. Further, GEOMINI also shows the smallest performance drop when the interference degree reaches 2. This could be contributed by our balancing strategy that considers geometric information of covariates and treatments. Overall, the results showcase the robustness of GEOMINI to different degrees of interference.

6 Conclusion

In this paper, we propose a geometric mutual information model to learn balanced representations for causal effect estimation under network interference. Based on the error bound of causal effect estimation in terms of mutual information, we find that balanced representation learning can be achieved by maximizing the Gromov-Wasserstein discrepancy between the geometric structures of covariates and treatments. We further apply the powerful geometric tool Ricci flow to capture the geometry of network graph data, which is leveraged for representation learning and information aggregation. The experimental results on benchmark datasets verify the effectiveness of our proposed method.

AI USE STATEMENT

In this work, we used generative AI tools for improving the clarity and readability of author-written text and assisting with literature retrieval and the identification of relevant references. We have reviewed all AI-assisted work. We take responsibility for the final content of this work, including text, claims, or artifacts produced with the aid of generative AI.

REFERENCES

Sinan Aral and Dylan Walker. Creating social contagion through viral product design: A randomized trial of peer influence in networks. *Management science*, 57(9):1623–1639, 2011.

Shuliang Bai, Yong Lin, Linyuan Lu, Zhiyu Wang, and Shing-Tung Yau. Ollivier ricci-flow on weighted graphs. *American Journal of Mathematics*, 146(6):1723–1747, 2024.

Robert A Becker. The variance drain and jensen's inequality. 2012.

Ruichu Cai, Zeqin Yang, Weilin Chen, Yuguang Yan, and Zhifeng Hao. Generalization bound for estimating causal effects from observational network data. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management*, pp. 163–172, 2023.

Weilin Chen, Ruichu Cai, Zeqin Yang, Jie Qiao, Yuguang Yan, Zijian Li, and Zhifeng Hao. Doubly robust causal effect estimation under networked interference via targeted learning. In *International Conference on Machine Learning*, pp. 6457–6485. PMLR, 2024.

Weilin Chen, Ruichu Cai, Jie Qiao, Yuguang Yan, and José Miguel Hernández-Lobato. Causal effect estimation under networked interference without networked unconfoundedness assumption. *arXiv preprint arXiv:2502.19741*, 2025.

Ching-Yao Chuang, Stefanie Jegelka, and David Alvarez-Melis. Infoot: Information maximizing optimal transport. In *International Conference on Machine Learning*, pp. 6228–6242. PMLR, 2023.

Michaël Defferrard, Xavier Bresson, and Pierre Vandergheynst. Convolutional neural networks on graphs with fast localized spectral filtering. *Advances in neural information processing systems*, 29, 2016.

Naoki Egami and Eric J Tchetgen Tchetgen. Identification and estimation of causal peer effects using double negative controls for unmeasured network confounding. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 86(2):487–511, 2024.

Amirhossein Farzam, Allen Tannenbaum, and Guillermo Sapiro. From geometry to causality-ricci curvature and the reliability of causal inference on networks. In *Forty-first International Conference on Machine Learning*, 2024.

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

Ruocheng Guo, Jundong Li, and Huan Liu. Learning individual causal effects from networked observational data. In *Proceedings of the 13th international conference on web search and data mining*, pp. 232–240, 2020.

Jennifer L Hill. Bayesian nonparametric modeling for causal inference. *Journal of Computational and Graphical Statistics*, 20(1):217–240, 2011.

Christopher Hopper and Ben Andrews. The ricci flow in riemannian geometry. *Lecture Notes in Mathematics*, 2011, 2010.

Qiang Huang, Jing Ma, Jundong Li, Ruocheng Guo, Huiyan Sun, and Yi Chang. Modeling interference for individual treatment effect estimation from networked observational data. *ACM Transactions on Knowledge Discovery from Data*, 18(3):1–21, 2023.

Michael G Hudgens and M Elizabeth Halloran. Toward causal inference with interference. *Journal of the american statistical association*, 103(482):832–842, 2008.

Guido W Imbens and Donald B Rubin. *Causal inference in statistics, social, and biomedical sciences*. Cambridge university press, 2015.

Song Jiang and Yizhou Sun. Estimating causal effects on networked observational data via representation learning. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*, pp. 852–861, 2022.

Miao Jin, Junho Kim, Feng Luo, and Xianfeng Gu. Discrete surface ricci flow. *IEEE Transactions on Visualization and Computer Graphics*, 14(5):1030–1043, 2008.

Fredrik Johansson, Uri Shalit, and David Sontag. Learning representations for counterfactual inference. In *International conference on machine learning*, pp. 3020–3029. PMLR, 2016.

Seyedeh Baharan Khatami, Harsh Parikh, Haowei Chen, Sudeepa Roy, and Babak Salimi. Graph machine learning based doubly robust estimator for network causal effects. *arXiv preprint arXiv:2403.11332*, 2024.

Seyedeh Baharan Khatami, Harsh Parikh, Haowei Chen, Sudeepa Roy, and Babak Salimi. Graph machine learning based doubly robust estimator for network causal effects. In *International Conference on Artificial Intelligence and Statistics*, pp. 4366–4374. PMLR, 2025.

TN Kipf. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*, 2016.

Kun Kuang, Peng Cui, Bo Li, Meng Jiang, and Shiqiang Yang. Estimating treatment effect in the wild via differentiated confounder balancing. In *Proceedings of the 23rd ACM SIGKDD international conference on knowledge discovery and data mining*, pp. 265–274, 2017.

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

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

Jing Ma, Mengting Wan, Longqi Yang, Jundong Li, Brent Hecht, and Jaime Teevan. Learning causal effects on hypergraphs. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 1202–1212, 2022.

Yunpu Ma and Volker Tresp. Causal inference under networked interference and intervention policy enhancement. In *International Conference on Artificial Intelligence and Statistics*, pp. 3700–3708. PMLR, 2021.

Facundo Mémoli. Gromov–wasserstein distances and the metric approach to object matching. *Foundations of computational mathematics*, 11(4):417–487, 2011.

Khang Nguyen, Nong Minh Hieu, Vinh Duc Nguyen, Nhat Ho, Stanley Osher, and Tan Minh Nguyen. Revisiting over-smoothing and over-squashing using ollivier-ricci curvature. In *International Conference on Machine Learning*, pp. 25956–25979. PMLR, 2023.

Chien-Chun Ni, Yu-Yao Lin, Feng Luo, and Jie Gao. Community detection on networks with ricci flow. *Scientific reports*, 9(1):9984, 2019.

Elizabeth L Ogburn, Oleg Sofrygin, Ivan Diaz, and Mark J Van der Laan. Causal inference for social network data. *Journal of the American Statistical Association*, 119(545):597–611, 2024.

Yann Ollivier. Ricci curvature of markov chains on metric spaces. *Journal of Functional Analysis*, 256(3):810–864, 2009.

Georgia Papadogeorgou, Christine Choirat, and Corwin M Zigler. Adjusting for unmeasured spatial confounding with distance adjusted propensity score matching. *Biostatistics*, 20(2):256–272, 2019.

Gabriel Peyré, Marco Cuturi, and Justin Solomon. Gromov-wasserstein averaging of kernel and distance matrices. In *International conference on machine learning*, pp. 2664–2672. PMLR, 2016.

Gabriel Peyré, Marco Cuturi, et al. Computational optimal transport: With applications to data science. *Foundations and Trends® in Machine Learning*, 11(5-6):355–607, 2019.

James M Robins, Andrea Rotnitzky, and Lue Ping Zhao. Estimation of regression coefficients when some regressors are not always observed. *Journal of the American statistical Association*, 89(427): 846–866, 1994.

Paul R Rosenbaum and Donald B Rubin. The central role of the propensity score in observational studies for causal effects. *Biometrika*, 70(1):41–55, 1983.

Donald B Rubin. Randomization analysis of experimental data: The fisher randomization test comment. *Journal of the American statistical association*, 75(371):591–593, 1980.

Uri Shalit, Fredrik D Johansson, and David Sontag. Estimating individual treatment effect: generalization bounds and algorithms. In *International conference on machine learning*, pp. 3076–3085. PMLR, 2017.

Li Sun, Zhenhao Huang, Hua Wu, Junda Ye, Hao Peng, Zhengtao Yu, and Philip S Yu. Deepricci: Self-supervised graph structure-feature co-refinement for alleviating over-squashing. In *2023 IEEE International Conference on Data Mining (ICDM)*, pp. 558–567. IEEE, 2023.

Jake Topping, Francesco Di Giovanni, Benjamin Paul Chamberlain, Xiaowen Dong, and Michael M Bronstein. Understanding over-squashing and bottlenecks on graphs via curvature. In *International Conference on Learning Representations*.

Victor Veitch, Yixin Wang, and David Blei. Using embeddings to correct for unobserved confounding in networks. *Advances in Neural Information Processing Systems*, 32, 2019.

Cédric Villani et al. *Optimal transport: old and new*, volume 338. Springer, 2008.

Hao Wang, Zhichao Chen, Zhaoran Liu, Xu Chen, Haoxuan Li, and Zhouchen Lin. Proximity matters: Local proximity enhanced balancing for treatment effect estimation. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V. 2*, pp. 2927–2937, 2025.

Christopher Winship and Stephen L Morgan. The estimation of causal effects from observational data. *Annual review of sociology*, 25(1):659–706, 1999.

Anpeng Wu, Haiyi Qiu, Zhengming Chen, Zijian Li, Ruoxuan Xiong, Fei Wu, and Kun Zhang. Causal graph transformer for treatment effect estimation under unknown interference. In *International Conference on Learning Representations*, volume 2025, pp. 99579–99608, 2025.

Yuguang Yan, Zeqin Yang, Weilin Chen, Ruichu Cai, Zhifeng Hao, and Michael Kwok-Po Ng. Exploiting geometry for treatment effect estimation via optimal transport. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 16290–16298, 2024.

Hao Yang, Zexu Sun, Hongteng Xu, and Xu Chen. Revisiting counterfactual regression through the lens of gromov-wasserstein information bottleneck. *arXiv preprint arXiv:2405.15505*, 2024.

Liuyi Yao, Zhixuan Chu, Sheng Li, Yaliang Li, Jing Gao, and Aidong Zhang. A survey on causal inference. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 15(5):1–46, 2021.

Ziyu Zhao, Yuqi Bai, Ruoxuan Xiong, Qingyu Cao, Chao Ma, Ning Jiang, Fei Wu, and Kun Kuang. Learning individual treatment effects under heterogeneous interference in networks. *ACM Transactions on Knowledge Discovery from Data*, 18(8):1–21, 2024.

A Assumptions and Target Estimands

We assume the following assumptions hold.

Assumption A.1 (Network Consistency)**.** The potential outcome is the same as the observed outcome under the same individual treatment and neighborhood exposure, *i.e.*, $y_i = y_i(t_i, z_i)$ if unit i actually receives t_i and z_i .

Assumption A.2 (Network Overlap)**.** Given any individual and neighbors' features, any treatment pair (t, z) has a non-zero probability of being observed in the data, *i.e.*, $\forall x_i, x_{\mathcal{N}_i}, t_i, z_i, \quad 0 < p(t_i, z_i|\mathbf{x}_i) < 1$.

Assumption A.3 (Neighborhood Interference)**.** The potential outcome of a unit is only affected by its own and the first-order neighbors' treatments, and the effect of the neighbors' treatments is through a summary function agg, *i.e.*, $\forall t_{\mathcal{N}_i}, t'_{\mathcal{N}_i}$ which satisfy $agg(t_{\mathcal{N}_i}) = agg(t'_{\mathcal{N}_i})$, the following equation holds: $y(t_i, t_{\mathcal{N}_i}) = y(t_i, t'_{\mathcal{N}_i})$.

Assumption A.4 (Network Unconfoundness)**.** The individual treatment and neighborhood exposure are independent of the potential outcome given the individual and neighbors' features, *i.e.*, $\forall t, z, \quad y_i(t, z) \perp\!\!\!\perp t_i, z_i \mid \mathbf{x}_i$,.

These assumptions are widely used in existing causal inference methods with network interference (Jiang & Sun, 2022; Cai et al., 2023; Chen et al., 2024). Specifically, Assumption A.1 rules out the existence of multiple versions of the treatment. Assumption A.2 ensures that the potential outcome $y(t, z)$ is well defined for all admissible values of the individual treatment t and the neighborhood exposure z. Under Assumption A.3, we adopt the notation $y(t, z)$ to represent the potential outcome associated with a unit receiving treatment t while being exposed to neighborhood treatment configuration z. Assumption A.4 generalizes the standard unconfoundedness condition by explicitly accounting for neighbors' treatments and covariates, thereby enabling valid causal inference in the presence of interference. In this paper, our goal is to identify and estimate the individual treatment effect $\tau(\mathbf{x}_i)^{\mathbf{t},\mathbf{t}'}$ corresponding to a contrast between treatments t and t' while accounting for peer exposure configurations z and z', given by:

$$\begin{aligned}\tau(\mathbf{x}_i)^{(t,z),(t',z')} =& \mathbb{E}[y(t, z) - y(t', z') \mid \mathbf{x}_i] = \mathbb{E}[y(t, z) \mid t, z, \mathbf{x}_i] - \mathbb{E}[y(t', z') \mid t', z', \mathbf{x}_i] \\ =& \mathbb{E}[y \mid t, z, \mathbf{x}_i] - \mathbb{E}[y \mid t', z', \mathbf{x}_i] = f(\mathbf{x}_i, t, z) - f(\mathbf{x}_i, t', z'), \end{aligned} \tag{21}$$

where the second equation holds because of Assumption A.4, and the third equation holds due to Assumption A.1.

Following (Jiang & Sun, 2022; Chen et al., 2024), we focus on estimating three kinds of causal effects:

Definition A.5 (Direct Effect (DE))**.** DE measures the difference in individual outcomes between unit i assigned to $T = 1, Z = z$ and $T = 0, Z = z$: $\tau_{DE}(\mathbf{x}_i) = \tau(\mathbf{x}_i)^{(1,z),(0,z)} = f(\mathbf{x}_i, 1, z) - f(\mathbf{x}_i, 0, z)$.

Definition A.6 (Spillover Effect (SE))**.** SE measures the difference in individual outcomes between unit i assigned to $T = 0, Z = z$ and $T = 0, Z = 0$: $\tau_{SE}(\mathbf{x}_i) = \tau(\mathbf{x}_i)^{(0,z),(0,0)} = f(\mathbf{x}_i, 0, z) - f(\mathbf{x}_i, 0, 0)$.

Definition A.7 (Total Effect (TE))**.** TE measures the difference in individual outcomes between units assigned to $T = 1, Z = z$ and $T = 0, Z = 0$: $\tau_{TE}(\mathbf{x}_i) = \tau(\mathbf{x}_i)^{(1,z),(0,0)} = f(\mathbf{x}_i, 1, z) - f(\mathbf{x}_i, 0, 0)$.

DE captures the change in an individual's outcome attributable to its own treatment assignment, holding the level of external exposure fixed. SE quantifies the impact of external or neighborhood treatment exposure on an individual's outcome when the individual itself is untreated. TE measures the overall change in an individual's outcome when moving from a fully untreated state to a setting in which both the individual and its environment are exposed to treatment.

B Mutual Information Perspective

We propose to minimize the upper bound in Theorem 4.2 to train the model. This optimization problem can also be derived from the mutual information perspective as follows

$$\max_{\phi} \; I((\phi(X), \phi(X_N)); Y|T, Z) - \lambda \, I((\phi(X), \phi(X_N)); (T, Z)). \tag{22}$$

The second term is discussed in Theorem 4.3. We now focus on the first term $I((\phi(X), \phi(X_{\mathcal{N}})); Y|T, Z)$, which is maximized to learn representations highly related to the potential outcome. In practice, the first term can be empirically estimated by

$$\begin{aligned}\hat{I}((\phi(X), \phi(X_{\mathcal{N}})); Y|T, Z) &= \frac{1}{N}\sum_{i}^{N} \log \frac{p(\phi(x_i), \phi(x_{\mathcal{N}_i}), y_i|t_i, z_i)}{p(\phi(x_i), \phi(x_{\mathcal{N}_i})|t_i, z_i)p(y_i|t_i, z_i)} \\ &= \frac{1}{N}\sum_{i}^{N} \log \frac{p(y_i|\phi(x_i), \phi(x_{\mathcal{N}_i}), t_i, z_i)}{p(y_i|t_i, z_i)} \\ &= \frac{1}{N}\sum_{i}^{N} \log p(y_i|\phi(x_i), \phi(x_{\mathcal{N}_i}), t_i, z_i) - \log p(y_i|t_i, z_i). \end{aligned} \tag{23}$$

Based on a parametric outcome prediction model $h : \mathcal{R} \times \tilde{\mathcal{R}} \times \{0, 1\} \times [0, 1] \to \mathcal{Y}$, and the assumption that y_i given $\phi(x_i), \phi(x_{\mathcal{N}_i}), t_i, z_i$ follows a Gaussian distribution with mean $h(\phi(x_i), \psi(\phi(x_{\mathcal{N}_i})), t_i, z_i)$ and standard derivation σ, we have $\log p(y_i|\phi(x), \phi(x_{\mathcal{N}_i}), t_i, z_i) = -\frac{(y_i - h(\phi(x_i), \psi(\phi(x_{\mathcal{N}_i})), t_i, z_i))^2}{\sigma^2} - \frac{1}{2}\log(2\pi\sigma^2)$, so that

$$\begin{aligned}\hat{I}((\phi(X), \phi(X_{\mathcal{N}})); Y|T, Z) &= -\frac{1}{N}\sum_{i}^{N} \frac{(y_i - h(\phi(x_i), \psi(\phi(x_{\mathcal{N}_i})), t_i, z_i))^2}{2\sigma^2} - \frac{1}{2}\log(2\pi\sigma^2) - \log p(y_i|t_i, z_i) \\ &= -\frac{1}{N}\sum_{i}^{N} \frac{(y_i - h(\phi(x_i), \psi(\phi(x_{\mathcal{N}_i})), t_i, z_i))^2}{2\sigma^2} + C_{c,y}. \end{aligned}$$

Minimizing the above mean square loss (*i.e.*, factual loss) corresponds to maximizing the mutual information between the representation ϕ and the outcome Y, up to a constant.

C Pseudo-Code of GEOMINI

Algorithm 1 presents the pseudo-code of our method GEOMINI.

Algorithm 1 Geometric Mutual Information for Networked Interference (GEOMINI)

Input: Covariates $\{x_i\}_{i=1}^N$, treatments $\{t_i\}_{i=1}^N$, factual outcomes $\{y_i\}_{i=1}^N$, and the adjacency matrix $A \in \{0, 1\}^{N\times N}$.

Calculate d_{ij}^t according to Eq. (10).
Construct $\tilde{A}_{ij}^t$ according to Eq. (13).
repeat
  Obtain the representations $\phi(x_i)$.
  Calculate d_{ij}^ϕ according to Eq. (10).
  Construct $\tilde{A}_{ij}^\phi$ according to Eq. (12).
  Aggregate the neighbors' representations $\psi(\phi(x_{\mathcal{N}_i}))$ based on Eq. (14).
  Calculate the peer exposure z_i based on Eq. (15).
  Predict $\hat{y}_i = h(\phi(x_i), \psi(\phi(x_{\mathcal{N}_i})), t_i, z_i)$.
  Calculate the loss function $\mathcal{L}_{mse} + \lambda\mathcal{L}_{gmi}$.
  Update h and ϕ by backpropagating.
until convergence.

D Proofs

D.1 Proof of Theorem 4.2

Theorem 4.2 *Let* $\varepsilon_F = \int_{\mathcal{X}}\int_{\mathcal{T}}(h(\phi(x), \psi(\phi(x_{\mathcal{N}})), t, z) - f(\mathbf{x}, \mathbf{t}))^2 p(\mathbf{x}, \mathbf{t}) d\mathbf{x} d\mathbf{t}$ *represent the factual loss,* ε_{PEHE} *is bounded as follows:*

$$\varepsilon_{PEHE} \leq 4\varepsilon_F + 4B^2\sqrt{2I((\phi(X), \phi(X_{\mathcal{N}})); (T, Z))}, \tag{24}$$

where $I((\phi(X), \phi(X_{\mathcal{N}})); (T, Z))$ is the mutual information between the representation $(\phi(X), \phi(X_{\mathcal{N}}))$ and the treatment assignments, B is a constant related to the upper bound of the outcome functions.

Proof. For simplicity, we use $\mathbf{x}_i$ denote unit i's features along with neighbors' features, *i.e.*, $\mathbf{x}_i = (x_i, x_{\mathcal{N}_i})$ and use $\mathbf{t}_i$ denote treatment and peer exposure for unit i's, *i.e.*, $\mathbf{t}_i = (t_i, z_i)$.

$$\begin{aligned}
\varepsilon_{PEHE}(h, \phi) &= \mathbb{E}_{p(\mathbf{t})p(\mathbf{t}')}\left[\int_{\mathcal{X}} (\tau^{\mathbf{t},\mathbf{t}'}_{h,\phi}(\mathbf{x}) - \tau^{\mathbf{t},\mathbf{t}'}(\mathbf{x}))^2 p(\mathbf{x})\, d\mathbf{x}\right] \\
&= \mathbb{E}_{p(\mathbf{t})p(\mathbf{t}')}\left[\int_{\mathcal{X}} \big(h(\phi(x), \psi(\phi(x_{\mathcal{N}})), \mathbf{t}) - h(\phi(x), \psi(\phi(x_{\mathcal{N}})), \mathbf{t}') - (f(\mathbf{x}, \mathbf{t}) - f(\mathbf{x}, \mathbf{t}'))\big)^2 p(\mathbf{x})\, d\mathbf{x}\right] \\
&\leq 2\int_{\mathcal{T}}\int_{\mathcal{X}} \big(h(\phi(x), \psi(\phi(x_{\mathcal{N}})), \mathbf{t}) - f(\mathbf{x}, \mathbf{t})\big)^2 p(\mathbf{x})p(\mathbf{t})\, d\mathbf{x}d\mathbf{t} \\
&\qquad + 2\int_{\mathcal{T}'}\int_{\mathcal{X}} \big(h(\phi(x), \psi(\phi(x_{\mathcal{N}})), \mathbf{t}') - f(\mathbf{x}, \mathbf{t}')\big)^2 p(\mathbf{x})p(\mathbf{t}')\, d\mathbf{x}d\mathbf{t}' \\
&= 4\int_{\mathcal{T}}\int_{\mathcal{X}} \big(h(\phi(x), \psi(\phi(x_{\mathcal{N}}), \mathbf{t}) - f(\mathbf{x}, \mathbf{t})\big)^2 p(\mathbf{x})p(\mathbf{t})\, d\mathbf{x}d\mathbf{t}.
\end{aligned}$$

Then we define ε_F and Δ as:

$$\varepsilon_F := \int_{\mathcal{T}}\int_{\mathcal{X}} \big(h(\phi(x), \psi(\phi(x_{\mathcal{N}})), \mathbf{t}) - f(\mathbf{x}, \mathbf{t})\big)^2 p(\mathbf{x}, \mathbf{t}) d\mathbf{x}d\mathbf{t}, \tag{25}$$

$$\Delta := \int_{\mathcal{T}}\int_{\mathcal{X}} ((h(\phi(x), \psi(\phi(x_{\mathcal{N}})), \mathbf{t}) - f(\mathbf{x}, \mathbf{t}))^2 \big(p(\mathbf{x})p(\mathbf{t}) - p(\mathbf{x}, \mathbf{t})\big) d\mathbf{x}d\mathbf{t}. \tag{26}$$

We assume $|h(\phi(x), \psi(\phi(x_{\mathcal{N}})), \mathbf{t}) - f(\mathbf{x}, \mathbf{t})| \leq B$,

$$|\Delta| \leq \int_{\mathcal{T}}\int_{\mathcal{X}} B^2 \big|p(\mathbf{x})p(\mathbf{t}) - p(\mathbf{x}, \mathbf{t})\big| d\mathbf{x}d\mathbf{t} = B^2 \|p(\mathbf{x})p(\mathbf{t}) - p(\mathbf{x}, \mathbf{t})\|_1, \tag{27}$$

because of the Pinsker inequation $\|P - Q\|_1 \leq \sqrt{2\mathrm{KL}(P|Q)}$

$$|\Delta| \leq B^2\sqrt{2\mathrm{KL}(p(\mathbf{x}, \mathbf{t})|p(\mathbf{x})p(\mathbf{t}))} = B^2\sqrt{2I(X, X_{\mathcal{N}}; (T, Z))}. \tag{28}$$

Because ϕ is an invertible representation of X and $X_{\mathcal{N}}$, mutual information is preserved, i.e.,

$$I(X, X_{\mathcal{N}}; (T, Z)) = I(\phi(X), \phi(X_{\mathcal{N}}); (T, Z)). \tag{29}$$

Finally,

$$\begin{aligned}
\varepsilon_{PEHE} &\leq 4\int_{\mathcal{T}}\int_{\mathcal{X}} \big(h(\phi(x), \psi(\phi(x_{\mathcal{N}})), \mathbf{t}) - f(x, \mathbf{t})\big)^2 p(\mathbf{x})p(\mathbf{t})\, d\mathbf{x}d\mathbf{t} \\
&= 4\varepsilon_F + 4\Delta \\
&\leq 4\varepsilon_F + 4|\Delta| \\
&\leq 4\varepsilon_F + 4B^2\sqrt{2I(\phi(X), \phi(X_{\mathcal{N}}); (T, Z))}.
\end{aligned} \tag{30}$$

□

D.2 Proof of Theorem 4.3

To prove Theorem 4.3, we present the following lemmas and prove them in the subsequent sections.

Lemma D.1. *Following the approach in (Shalit et al., 2017; Cai et al., 2023), we assume that $\phi : \mathcal{X} \to \mathcal{R} \subset \mathbb{R}^E$ is L-Lipschitz. Mutual information between representations and treatments are upper bounded by the following*

$$I((\phi(X), \phi(X_{\mathcal{N}})); (T, Z)) \leq I(\phi(X); T) + I(\phi(X); Z \mid T) + H(X_{\mathcal{N}}) + E\log L + o_p(1). \tag{31}$$

Lemma D.2. *Based on the Gaussian kernel density estimation of mutual information in Eq. (3), the mutual information between the learned representation $\phi(X)$ and the treatment assignment T admits the following geometric upper bound:*

$$\hat{I}(\phi(X);T) \leq -\frac{1}{2\sigma^2}\widehat{GW}_2^2(\phi(X),T) + \frac{1}{2N^2\sigma^2}\|D^\phi - D^t\|_F^2 + C_1, \tag{32}$$

where C_1 is a constant independent of ϕ.

Lemma D.3. *Based on the Gaussian kernel density estimation of mutual information in Eq. (3), the conditional mutual information $I(\phi(X); Z \mid T)$ admits the following upper bound:*

$$\begin{aligned}\hat{I}(\phi(X); Z \mid T) \leq -\frac{1}{2\sigma^2}\widehat{GW}_2^2(\phi(X),T) - \frac{1}{2\sigma^2}\widehat{GW}_2^2(T,Z) \\ + \frac{1}{2N^2\sigma^2}\left(\|D^\phi - D^t\|_F^2 + \|D^t - D^z\|_F^2\right) + C_2,\end{aligned} \tag{33}$$

where C_2 is a constant independent of ϕ.

Now, we remind Theorem 4.3 and prove it.

Theorem 4.3 *Based on the Gaussian kernel density estimation of mutual information in Eq. (3), the mutual information between the learned representations and the treatment assignments admits the following upper bound:*

$$\begin{aligned}\hat{I}(\phi(X),\phi(X_{\mathcal{N}});(T,Z)) \leq -\frac{1}{2\sigma^2}\left(2\widehat{GW}_2^2(\phi(X),T) + \widehat{GW}_2^2(T,Z)\right) \\ + \frac{1}{2N^2\sigma^2}\left(2\|D^\phi - D^t\|_F^2 + \|D^t - D^z\|_F^2\right) + o_p(1) + C,\end{aligned} \tag{34}$$

where N is the sample size, $\|\cdot\|_F$ is the Frobenius norm, C is a constant independent of ϕ, D^ϕ, D^t, and D^z are distance matrices. The elements of D^ϕ are $\{d_{ij}^\phi\}$, and D^t, D^z are defined similarly.

Proof. Under Lemma D.2, Lemma D.3 and Lemma D.1, it is easy to obtain:

$$\begin{aligned}I((\phi(X),\phi(X_{\mathcal{N}}));(T,Z)) \leq -\frac{1}{\sigma^2}\widehat{GW}_2^2(\phi(X),T) - \frac{1}{2\sigma^2}\widehat{GW}_2^2(T,Z) + \frac{1}{N^2\sigma^2}\|D^\phi - D^t\|_F^2 \\ + \frac{1}{2N^2\sigma^2}\|D^t - D^z\|_F^2 + C,\end{aligned} \tag{35}$$

where $C = C_1 + C_2 + H(X_{\mathcal{N}}) + E\log L$ which is a constant independent of ϕ.

□

D.3 Proof of Lemma D.1

Lemma D.1 *Following the approach in (Shalit et al., 2017; Cai et al., 2023), we assume that $\phi : \mathcal{X} \to \mathcal{R} \subset \mathbb{R}^E$ is L-Lipschitz. Mutual information between representations and treatments are upper bounded by the following*

$$I((\phi(X),\phi(X_{\mathcal{N}}));(T,Z)) \leq I(\phi(X);T) + I(\phi(X);Z \mid T) + H(X_{\mathcal{N}}) + E\log L + o_p(1). \tag{36}$$

Proof. By the chain rule of mutual information, the regularization term $I(\phi;(T,Z))$ can be decomposed as

$$\begin{aligned}
&\hat{I}\big((\phi(X),\phi(X_{\mathcal{N}}));(T,Z)\big)\\
&\le I\big((\phi(X),\phi(X_{\mathcal{N}}));(T,Z)\big)+\Big|\hat{I}\big((\phi(X),\phi(X_{\mathcal{N}}));(T,Z)\big)-I\big((\phi(X),\phi(X_{\mathcal{N}}));(T,Z)\big)\Big| && (37)\\
&\le I(\phi(X);T)+I(\phi(X);Z\mid T)+H(\phi(X_{\mathcal{N}}))\\
&\quad+|\hat{I}\big((\phi(X),\phi(X_{\mathcal{N}}));(T,Z)\big)-I\big((\phi(X),\phi(X_{\mathcal{N}}));(T,Z)\big)| && (38)\\
&\le \hat{I}(\phi(X);T)+\hat{I}(\phi(X);Z\mid T)+H(\phi(X_{\mathcal{N}}))\\
&\quad+\Big|\hat{I}\big((\phi(X),\phi(X_{\mathcal{N}}));(T,Z)\big)-I\big((\phi(X),\phi(X_{\mathcal{N}}));(T,Z)\big)\Big|\\
&\quad+\Big|\hat{I}(\phi(X);T)-I(\phi(X);T)\Big|+\Big|\hat{I}(\phi(X);Z\mid T)-I(\phi(X);Z\mid T)\Big| && (39)\\
&=\hat{I}(\phi(X);T)+\hat{I}(\phi(X);Z\mid T)+H(\phi(X_{\mathcal{N}}))+O_p(a_N+b_N+c_N) && (40)\\
&=\hat{I}(\phi(X);T)+\hat{I}(\phi(X);Z\mid T)+H(\phi(X_{\mathcal{N}}))+o_p(1), && (41)
\end{aligned}$$

where $H(\cdot)$ and $H(\cdot\mid\cdot)$ represent entropy term and conditional entropy term, and we let $|\hat{I}((\phi(X),\phi(X_{\mathcal{N}}));(T,Z))-I((\phi(X),\phi(X_{\mathcal{N}}));(T,Z))|=O_p(a_N)$, $|\hat{I}(\phi(X);T)-I(\phi(X);T)|=O_p(b_N)$ and $|\hat{I}(\phi(X);Z\mid T)-I(\phi(X);Z\mid T)|=O_p(c_N)$.

Eq. (38) holds because of $I(\phi(X_{\mathcal{N}});(T,Z)\mid\phi(X))=H(\phi(X_{\mathcal{N}}\mid\phi(X))-H(\phi(X_{\mathcal{N}}\mid\phi(X),T,Z)$ and $H(\phi(X_{\mathcal{N}}\mid\phi(X),T,Z)\ge 0$ and $H(\phi(X_{\mathcal{N}})\mid\phi(X))\le H(\phi(X_{\mathcal{N}}))$. Since ϕ provides an invertible representation of $X_{\mathcal{N}}$, and $\phi:\mathbb{R}\to\mathbb{R}^E$ is L-Lipschitz. Consequently, the spectral norm of its Jacobian satisfies $\|J_\phi(x_{\mathcal{N}})\|_2\le L$ for all $x_{\mathcal{N}}$.

We define the transformed variable as $r_{\mathcal{N}}=\phi(x_{\mathcal{N}})$, with the inverse relation $x_{\mathcal{N}}=\phi^{-1}(r_{\mathcal{N}})$, and

$$\begin{aligned}
H(\phi(X_{\mathcal{N}}))&=-\int p_{\phi(X_{\mathcal{N}})}(r_{\mathcal{N}})\log p_{\phi(X_{\mathcal{N}})}(r_{\mathcal{N}})dr_{\mathcal{N}}\\
&=-\int p_{X_{\mathcal{N}}}(\phi^{-1}(r_{\mathcal{N}}))|\det J_{\phi^{-1}}(r_{\mathcal{N}})|\log p_{X_{\mathcal{N}}}(r_{\mathcal{N}})|\det J_{\phi^{-1}}(r_{\mathcal{N}})|dr_{\mathcal{N}}\\
&=-\int p_{X_{\mathcal{N}}}(x_{\mathcal{N}}))\log p_{X_{\mathcal{N}}}(x_{\mathcal{N}})|\det J_{\phi^{-1}}(r_{\mathcal{N}})|dx_{\mathcal{N}} && (42)\\
&=-\int p_{X_{\mathcal{N}}}(x_{\mathcal{N}}))\log p_{X_{\mathcal{N}}}(r_{\mathcal{N}})dx_{\mathcal{N}}-\int p_{X_{\mathcal{N}}}(x_{\mathcal{N}}))\log|\det J_{\phi^{-1}}(r_{\mathcal{N}})|dx_{\mathcal{N}}\\
&=H(X_{\mathcal{N}})-\mathbb{E}[\log|\det J_{\phi^{-1}}(r_{\mathcal{N}})|]\\
&=H(X_{\mathcal{N}})+\mathbb{E}[\log|\det J_{\phi}(x_{\mathcal{N}})|] && (43)\\
&\le H(X_{\mathcal{N}})+E\log L. && (44)
\end{aligned}$$

Eq. (42) holds by $x_{\mathcal{N}}=\phi^{-1}(r_{\mathcal{N}})$, Eq. (43) holds by $|\det J_{\phi^{-1}}(r_{\mathcal{N}})|=1/|\det J_\phi(\phi^{-1}(r_{\mathcal{N}}))|=1/|\det J_\phi(x_{\mathcal{N}})|$, and Eq. (44) holds by l norm of its Jacobian satisfies $\|J_\phi(x_{\mathcal{N}})\|_2\le L$. Combine Eq. (41) and Eq. (44):

$$I(\phi(X),\phi(X_{\mathcal{N}});(T,Z))\le I(\phi(X);T)+I(\phi(X);Z\mid T)+H(X_{\mathcal{N}})+E\log L. \quad (45)$$

□

D.4 Proof of Lemma D.2

Lemma D.2 *Based on the Gaussian kernel density estimation of mutual information in Eq. (3), the mutual information between the learned representation $\phi(X)$ and the treatment assignment T admits the following geometric upper bound:*

$$\hat{I}(\phi(X);T)\le-\frac{1}{2\sigma^2}\widehat{GW}_2^2(\phi(X),T)+\frac{1}{2N^2\sigma^2}\|D^\phi-D^t\|_F^2+C_1, \quad (46)$$

where C_1 is a constant independent of ϕ.

Proof. For simplicity, we denote $\phi(X)$ (*resp.*, $\phi(x_i)$) as ϕ (*resp.*, ϕ_i). The mutual information between random variables ϕ and T is

$$I(\phi(X);T) = \iint p(\phi,t)\log\frac{p(\phi,t)}{p(\phi)p(t)}\,d\phi\,dt. \tag{47}$$

Given samples $\{(\phi_n,t_n,z_n)\}_{n=1}^N$, we estimate densities using kernel density estimation:

$$\hat{p}(\phi,t)=\frac{1}{N}\sum_{m=1}^N k(\phi,\phi_m)k(t,t_m),\quad \hat{p}(\phi)=\frac{1}{N}\sum_{m=1}^N k(\phi,\phi_m),\quad \hat{p}(t)=\frac{1}{N}\sum_{m=1}^N k(t,t_m), \tag{48}$$

where $k_\phi(\phi,\phi')=\frac{1}{\sqrt{2\pi}\sigma}\exp\Big(-\frac{\|\phi-\phi'\|_2^2}{2\sigma^2}\Big), k_t(t,t')=\frac{1}{\sqrt{2\pi}\sigma}\exp\Big(-\frac{(t-t')^2}{2\sigma^2}\Big)$. Denote $K^\phi_{mn}=k(\phi_m,\phi_n)$ and $K^t_{mn}=k(t_m,t_n)$.

$$\hat{I}(\phi(X),T)=\frac{1}{N}\sum_n\left[\log\Big(\frac{1}{N}\sum_m K^\phi_{mn}K^t_{mn}\Big)-\log\Big(\frac{1}{N^2}\sum_m K^\phi_{mn}\sum_m K^t_{mn}\Big)\right]. \tag{49}$$

By Jensen's inequality, for a concave function $\log(\cdot)$, we have

$$\mathbb{E}[\log X]\le\log(\mathbb{E}[X]), \tag{50}$$

which leads to

$$\begin{aligned}\hat{I}(\phi(X),T)&=\frac{1}{N}\sum_n\left(\log(\frac{1}{N}\sum_m K^\phi_{mn}K^t_{mn})-\log(\frac{1}{N^2}\sum_m K^\phi_{mn}\sum_m K^t_{mn})\right)\\&=\frac{1}{N}\sum_n\log(\frac{1}{N}\sum_m K^\phi_{mn}K^t_{mn})-\frac{1}{N}\sum_n\log(\frac{1}{N}\sum_m K^\phi_{mn})-\frac{1}{N}\sum_n\log(\frac{1}{N}\sum_m K^t_{mn})\\&\le\log(\frac{1}{N^2}\sum_{m,n}K^\phi_{mn}K^t_{mn})-\frac{1}{N^2}\sum_{m,n}\log(K^\phi_{mn})-\frac{1}{N^2}\sum_{m,n}\log(K^t_{mn}).\end{aligned} \tag{51}$$

For the Gaussian kernel, $\log K^\phi_{mn}=-\frac{(d^\phi(\phi_m,\phi_n))^2}{2\sigma^2}+c$ and $\log K^t_{mn}=-\frac{(d^t(t_m,t_n))^2}{2\sigma^2}+c$, where $c=\log(1/(2\pi\sigma^2))$. Expanding the logarithmic terms and collecting constants yields

$$\hat{I}(\phi(X),T)=\log(\frac{1}{N^2}\sum_{m,n}K^\phi_{mn}K^t_{mn})+\frac{1}{2N^2\sigma^2}\big(d^\phi(\phi_m,\phi_n)^2+d^t(t_m,t_n)^2\big)+2c \tag{52}$$

$$\le\log(\frac{1}{N^2}\sum_{m,n}K^\phi_{mn}K^t_{mn})+\frac{1}{2N^2\sigma^2}\|D^\phi-D^t\|_F^2+\frac{1}{N^2\sigma^2}\sum_{m,n}d^\phi(\phi_m,\phi_n)d^t(t_m,t_n)+2c, \tag{53}$$

where $\|D^\phi-D^t\|_2^2$ represent $\sum_{m,n}(d^\phi(\phi_m,\phi_n)-d^t(t_m,t_n))^2$. Then by Theorem 1 in (Becker, 2012) (taking $f=\log X$ supported on $[a,+\infty)$ with $a>0$), we have

$$\log\mathbb{E}[X]=\mathbb{E}[\log X]-\frac{1}{2}\mu\,\mathrm{Var}(X)\le\mathbb{E}[\log X]+\frac{\mathrm{Var}(X)}{2a^2}, \tag{54}$$

where $\mu\in[-1/a^2,\,0]$. Using the Gaussian kernel and bounding the variance term, we finally obtain

$$\hat{I}(\phi(X),T)\le\frac{1}{N^2}\sum_{m,n}\log K^\phi_{mn}K^t_{mn}+\frac{1}{2N^2\sigma^2}\|D^\phi-D^t\|_F^2+\frac{1}{N^2\sigma^2}\sum_{m,n}d^\phi(\phi_m,\phi_n)d^t(t_m,t_n)+C_{a,c}, \tag{55}$$

where $C_{a,c}\le\frac{\mathrm{Var}(K^\phi K^t)}{2a^2}+2c$. The mixed correlation term can be upper-bounded through a coupling relaxation. Let $\Pi(\frac{1}{N}\mathbf{1}_N,\frac{1}{N}\mathbf{1}_N)$ denote the set of doubly stochastic matrices with uniform marginals. Then

$$\begin{aligned}\hat{I}(\phi(X),T)\le\max_{\gamma\in\Pi(\frac{1}{N}\mathbf{1}_N,\frac{1}{N}\mathbf{1}_N)}\sum_{m,n,k,l}\gamma_{mn}\gamma_{kl}\Big[&\log\Big(K^\phi_{mk}K^t_{nl}\Big)\\&+\frac{1}{\sigma^2}d^\phi(\phi_m,\phi_k)d^t(t_n,t_l)\Big]+\frac{1}{2N^2\sigma^2}\|D^\phi-D^t\|_F^2+C_{a,c}.\end{aligned} \tag{56}$$

Substituting the Gaussian kernel expression $\log K = -\frac{d^2}{\sigma^2} + c$ and simplifying the quadratic terms yields

$$\hat{I}(\phi(X), T) \leq -\frac{1}{2\sigma^2} \min_{\gamma \in \Pi(\frac{1}{N}\mathbf{1}_N, \frac{1}{N}\mathbf{1}_N)} \sum_{m,n,k,l} \gamma_{mn}\gamma_{kl}\big(d^{\phi}(\phi_m, \phi_k) - d^{t}(t_n, t_l)\big)^2 + \frac{1}{2N^2\sigma^2}\|D^{\phi} - D^{t}\|_F^2 + 2c + C_{a,c}. \tag{57}$$

The first term equals to Gromov-Wasserstein (Mémoli, 2011; Peyré et al., 2016) $-\frac{1}{2\sigma^2}\widehat{GW}_2^2(\phi, T)$, so that we have:

$$\hat{I}(\phi(X), T) \leq -\frac{1}{2\sigma^2}\widehat{GW}_2^2(\phi(X), T) + \frac{1}{2N^2\sigma^2}\|D^{\phi} - D^{t}\|_F^2 + C_1. \tag{58}$$

where $C_1 = 2c + C_{a,c}$. □

D.5 Proof of Lemma D.3

Lemma D.3 *Based on the Gaussian kernel density estimation of mutual information in Eq. (3), the conditional mutual information $I(\phi(X); Z \mid T)$ admits the following upper bound:*

$$\hat{I}(\phi(X); Z \mid T) \leq -\frac{1}{2\sigma^2}\widehat{GW}_2^2(\phi, T) - \frac{1}{2\sigma^2}\widehat{GW}_2^2(T, Z) + \frac{1}{2N^2\sigma^2}\Big(\|D^{\phi} - D^{t}\|_F^2 + \|D^{t} - D^{z}\|_F^2\Big) + C_2, \tag{59}$$

where C_2 is a constant independent of ϕ.

Proof. The definition of conditional mutual information:

$$I(\phi(X); Z \mid T) = I((\phi(X), T); Z) - I(T; Z). \tag{60}$$

Given samples $(\phi_n, t_n, z_n)_{n=1}^N$, similar of the proof of $I(\phi(X); T)$, we define the kernel matrices

$$K^{\phi}_{mn} = k(\phi_m, \phi_n), \quad K^{t}_{mn} = k(t_m, t_n), \quad K^{z}_{mn} = k(z_m, z_n). \tag{61}$$

So we can define $I(\phi(X); Z \mid T)$ as:

$$I((\phi(X), T); Z) = \frac{1}{N}\sum_n \log \frac{\frac{1}{N}\sum_m K^{\phi}_{mn}K^{t}_{mn}K^{z}_{mn}}{\frac{1}{N^2}\sum_m K^{\phi}_{mn}K^{t}_{mn}\sum_m K^{z}_{mn}}, \tag{62}$$

$$\hat{I}(T; Z) = \frac{1}{N}\sum_n \log \frac{\frac{1}{N}\sum_m K^{t}_{mn}K^{z}_{mn}}{\frac{1}{N^2}\sum_m K^{t}_{mn}\sum_m K^{z}_{mn}}, \tag{63}$$

$$I(\phi(X); Z \mid T) = \frac{1}{N}\sum_n \log \frac{\Big(\sum_m K^{\phi}_{mn}K^{t}_{mn}K^{z}_{mn}\Big)\Big(\sum_m K^{t}_{mn}\Big)}{\Big(\sum_m K^{\phi}_{mn}K^{t}_{mn}\Big)\Big(\sum_m K^{t}_{mn}K^{z}_{mn}\Big)}. \tag{64}$$

Then, Applying Jensen's inequality to the concave logarithm yields:

$$\hat{I}(\phi; Z \mid T) \leq \log \frac{1}{N^2}\sum_{m,n} K^{\phi}_{mn}K^{t}_{mn}K^{z}_{mn} - \frac{1}{N^2}\sum_{m,n}\log\Big(K^{\phi}_{mn}K^{t}_{mn}\Big) - \frac{1}{N^2}\sum_{m,n}\log\Big(K^{t}_{mn}K^{z}_{mn}\Big) + \log\Big(\frac{1}{N^2}\sum_{m,n}K^{t}_{mn}\Big). \tag{65}$$

Substituting the Gaussian kernel expression $\log K = -\frac{d^2}{2\sigma^2} + c$ and simplifying the quadratic terms yields:

$$\begin{aligned}
\hat{I}(\phi(X); Z \mid T) \leq & \log(\frac{1}{N^2}\sum_{m,n} K^{\phi}_{mn} K^{t}_{mn} K^{z}_{mn}) + \log(\frac{1}{N^2}\sum_{m,n} K^{t}_{mn}) \\
& + \frac{1}{2N^2\sigma^2}\sum_{m,n}[d^{\phi}(\phi_m, \phi_n)^2 + 2d^{t}(t_m, t_n)^2 + d^{z}(z_m, z_n)^2] + 2c \\
= & \log\big(\frac{1}{N^2}\sum_{m,n} K^{\phi}_{mn} K^{t}_{mn} K^{z}_{mn}\big) + \log(\frac{1}{N^2}\sum_{m,n} K^{t}_{mn}) \\
& + \frac{1}{N^2\sigma^2}\sum_{m,n}\big[d^{\phi}(\phi_m, \phi_n) d^{t}(t_m, t_n) + d^{t}(t_m, t_n) d^{z}(z_m, z_n)\big] \\
& + \frac{1}{2N^2\sigma^2}\sum_{m,n}\big(\|D^{\phi} - D^{t}\|_F^2 + \|D^{t} - D^{z}\|_F^2\big) + 4c. \qquad (66)
\end{aligned}$$

Similar of Eq.(55), we have:

$$\begin{aligned}
\hat{I}(\phi(X); Z \mid T) \leq & \frac{1}{N^2}\sum_{m,n}\Big(\log(K^{\phi}_{mn} K^{t}_{mn} K^{z}_{mn}) + \log(K^{t}_{mn})\Big) \\
& + \frac{1}{N^2\sigma^2}\sum_{m,n}\big[d^{\phi}(\phi_m, \phi_n) d^{t}(t_m, t_n) + d^{t}(t_m, t_n) d^{z}(z_m, z_n)\big] \\
& + \frac{1}{2N^2\sigma^2}\sum_{m,n}\big(\|D^{\phi} - D^{t}\|_F^2 + \|D^{t} - D^{z}\|_F^2\big) + C_{a,c} \\
= & -\frac{1}{2\sigma^2 N^2}\sum_{m,n}(\log K^{\phi}_{mn} + \log K^{t}_{mn} + \log K^{z}_{mn} + \log K^{t}_{mn}) \\
& + \frac{1}{N^2\sigma^2}\sum_{m,n}\big[d^{\phi}(\phi_m, \phi_n) d^{t}(t_m, t_n) + d^{t}(t_m, t_n) d^{z}(z_m, z_n)\big] \\
& + \frac{1}{2N^2\sigma^2}\sum_{m,n}\big(\|D^{\phi} - D^{t}\|_F^2 + \|D^{t} - D^{z}\|_F^2\big) + C_{a,c}. \\
= & -\frac{1}{2\sigma^2 N^2}\sum_{m,n}\big[d^{\phi}(\phi_m, \phi_n) - d^{t}(t_m, t_n)\big]^2 \\
& - \frac{1}{2\sigma^2 N^2}\sum_{m,n}\big[d^{t}(t_m, t_n) - d^{z}(z_m, z_n)\big]^2 \\
& + \frac{1}{2N^2\sigma^2}\sum_{m,n}\big(\|D^{\phi} - D^{t}\|_F^2 + \|D^{t} - D^{z}\|_F^2\big) + C_{a,c} + 4c. \qquad (67)
\end{aligned}$$

Similar to Eq.(56):

$$\begin{aligned}
\hat{I}(\phi(X); Z \mid T) \leq & -\frac{1}{2\sigma^2}\sum_{m,n,k,l}\gamma^{\phi t}_{mn}\gamma^{\phi t}_{kl}\big[d^{\phi}(\phi_m, \phi_k) - d^{t}(t_m, t_k)\big]^2 \\
& - \frac{1}{2\sigma^2}\sum_{m,n,k,l}\gamma^{tz}_{mn}\gamma^{tz}_{kl}\big[d^{t}(t_n, t_l) - d^{z}(z_n, z_l)\big]^2 \\
& + \frac{1}{2N^2\sigma^2}\sum_{m,n}\big(\|D^{\phi} - D^{t}\|_F^2 + \|D^{t} - D^{z}\|_F^2\big) + C_{a,c} + 4c \\
= & -\frac{1}{2\sigma^2}\Big(\widehat{GW}_2^2(\phi(X), T) + \widehat{GW}_2^2(T, Z)\Big) + \frac{1}{2N^2\sigma^2}\sum_{m,n}\big(\|D^{\phi} - D^{t}\|_F^2 + \|D^{t} - D^{z}\|_F^2\big) + C_2, \qquad (68)
\end{aligned}$$

where $C_2 = C_{a,c} + 4c$. □

E TREATMENT AND POTENTIAL OUTCOME SIMULATION

Following (Jiang & Sun, 2022; Cai et al., 2023), Treatments and potential outcomes are subsequently simulated based on the following scheme.

Treatments simulation. For network data, the individual treatment t_i is not only affected by i's features x_i, but also affected by its neighbor's features $\{x_j\}_{j\in\mathcal{N}_i}$. To satisfy this setting, we first generate unit i's propensity score $p(t_i|x_i, \{x_j\}_{j\in\mathcal{N}_i})$, and then generate i's treatment according to the mean propensity score as follows:

$$ps_i = g\left(\beta_1(w_1 x_i) + \beta_2\Big(\frac{1}{|\mathcal{N}_i|}\sum_{j\in\mathcal{N}_i} w_2 x_j\Big)\right), \quad \text{s.t.} \quad t_i = \begin{cases} 1, & \text{if } ps_i > \overline{ps}, \\ 0, & \text{otherwise.} \end{cases} \tag{69}$$

where $g(\cdot)$ is sigmoid function, β_1 and β_2 control the relative contributions of individual features and neighbors' features, respectively, w_1 and w_2 are randomly sampled from $\mathcal{N}(-2, 3)$, and $\overline{ps}$ is the average of all ps_i. We set $\beta_1 = 0.5$ and set $\beta_2 = k\beta_1$, where the control parameter $k \in \{0.5, 1.0, 2.0\}$ represents different degrees of neighbor interference. After generate individual treatment t_i, the peer exposure z_i can be easily calculated by Eq. (15).

Potential outcome simulation. We follow the data generation procedure of (Jiang & Sun, 2022). In these datasets, the potential outcomes are generated based on the following rules:

$$y_i(t_i, z_i) = \beta_3 t_i + \beta_4 z_i + \beta_5(w_3 x_i) + \beta_6(w_4 \sum_{j\in\mathcal{N}_i} a_{ij} x_j) + \epsilon_i \tag{70}$$

where ϵ_i is the noise sampled from $\mathcal{N}(0, 1)$, w_3, w_4 are weights generated by $\mathcal{N}(1, 2)$. The parameters β_3, β_4, β_5, β_6 are strengths of individual treatment, neighborhood exposure, individual features, and neighbors' features, respectively. We set $\beta_3 = 1$, $\beta_5 = 0.5$, and like the treatment simulation, we set $\beta_4 = k\beta_3$, $\beta_6 = k\beta_5$, where control parameter $k \in \{0.5, 1.0, 2.0\}$ represents different degrees of neighbor interference.

F EVALUATION METRICS

We evaluate the performance of our method under these metrics:

(1) Direct Effect (DE) $\mathbb{E}[y(t = 1, z) - y(t = 0, z)|X, X_{\mathcal{N}}]$:

$$\varepsilon_{PEHE}^{DE} = \frac{1}{N}\sum_{i}^{N}(\hat{\tau}(\mathbf{x}_i)^{(1,z_i),(0,z_i)} - \tau(\mathbf{x}_i)^{(1,z_i),(0,z_i)})^2 \tag{71}$$

$$\varepsilon_{MAE}^{DE} = |\frac{1}{N}\sum_{i}^{N}\hat{\tau}(\mathbf{x}_i)^{(1,z_i),(0,z_i)} - \tau(\mathbf{x}_i)^{(1,z_i),(0,z_i)}| \tag{72}$$

(73)

(2) Spillover Effect (SE) $\mathbb{E}[y(t = 0, z) - y(t = 0, z = 0)|X, X_{\mathcal{N}}]$:

$$\varepsilon_{PEHE}^{SE} = \frac{1}{N}\sum_{i}^{N}(\hat{\tau}(\mathbf{x}_i)^{(0,z_i),(0,0)} - \tau(\mathbf{x}_i)^{(0,z_i),(0,0)})^2 \tag{74}$$

$$\varepsilon_{MAE}^{SE} = |\frac{1}{N}\sum_{i}^{N}\hat{\tau}(\mathbf{x}_i)^{(0,z_i),(0,0)} - \frac{1}{N}\sum_{i}^{N}\tau(\mathbf{x}_i)^{(0,z_i),(0,0)}| \tag{75}$$

(76)

(3) Total Effect (TE) $\mathbb{E}[y(t = 1, z = 1) - y(t = 0, z = 0)|X, X_{\mathcal{N}}]$:

$$\varepsilon_{PEHE}^{TE} = \frac{1}{N}\sum_{i}^{N}(\hat{\tau}(\mathbf{x}_i)^{(1,1),(0,0)} - \tau(\mathbf{x}_i)^{(1,z_i),(0,0)})^2 \tag{77}$$

$$\varepsilon_{MAE}^{TE} = |\frac{1}{N}\sum_{i}^{N}\hat{\tau}(\mathbf{x}_i)^{(1,1),(0,0)} - \frac{1}{N}\sum_{i}^{N}\tau(\mathbf{x}_i)^{(1,1),(0,0)}| \tag{78}$$

Table 3: Results of Direct causal effect estimation on BC Dataset. The ATE error ε_{ATE} is reported. The best performing method is bolded, and the second best is underlined.

Interference Degree	Setting	Effect	Methods										
			ND	TARNet+z	CFR+z	ND+z	GEst	NetEst	RRNet	TNet	DWR	GDML	GEOMINI
0.5	Within Sample	Direct	0.0572 ± 0.0367	0.0500 ± 0.0399	0.0490 ± 0.0296	0.0473 ± 0.0398	0.2329 ± 0.0442	0.0501 ± 0.0358	0.1168 ± 0.0138	0.0650 ± 0.0254	0.4054 ± 0.0908	0.1388 ± 0.1414	**0.0187 ± 0.0078**
		Spillover	–	0.1063 ± 0.0487	0.2472 ± 0.0346	0.1212 ± 0.1295	0.3765 ± 0.0476	0.0708 ± 0.0203	0.0664 ± 0.0152	**0.0311 ± 0.0115**	0.0560 ± 0.0224	0.2582 ± 0.0431	0.0600 ± 0.0462
		Total	–	0.1805 ± 0.1255	0.4154 ± 0.0658	0.1901 ± 0.1569	0.1611 ± 0.0694	**0.1101 ± 0.0971**	0.2362 ± 0.0332	0.1583 ± 0.0710	0.1243 ± 0.0430	0.3675 ± 0.1765	0.1222 ± 0.0508
	Out-of-Sample	Direct	0.0581 ± 0.0372	0.0512 ± 0.0371	0.0518 ± 0.0294	0.0515 ± 0.0377	0.2318 ± 0.0387	0.0481 ± 0.0344	0.1163 ± 0.0127	0.0673 ± 0.0269	0.4091 ± 0.0902	0.1192 ± 0.1229	**0.0198 ± 0.0092**
		Spillover	–	0.1092 ± 0.0484	0.2436 ± 0.0306	0.1205 ± 0.1228	0.3868 ± 0.0520	0.0692 ± 0.0210	0.0666 ± 0.0144	**0.0324 ± 0.0131**	0.0603 ± 0.0233	0.2826 ± 0.0410	0.0623 ± 0.0479
		Total	–	0.1822 ± 0.1237	0.4146 ± 0.0688	0.1899 ± 0.1577	0.1716 ± 0.0731	**0.1085 ± 0.0959**	0.2367 ± 0.0325	0.1630 ± 0.0667	0.1149 ± 0.0340	0.3258 ± 0.1963	0.1242 ± 0.0500
1	Within Sample	Direct	0.1328 ± 0.0402	0.0537 ± 0.0301	0.0302 ± 0.0338	0.0292 ± 0.0349	0.4961 ± 0.0511	0.0308 ± 0.0340	0.1334 ± 0.0209	0.1432 ± 0.0612	0.4400 ± 0.0515	0.1647 ± 0.2034	**0.0241 ± 0.0158**
		Spillover	–	0.1924 ± 0.1196	0.2689 ± 0.2087	0.1057 ± 0.1084	0.9744 ± 0.1225	0.1061 ± 0.0744	0.1059 ± 0.0184	0.1105 ± 0.0537	0.0876 ± 0.0613	0.4622 ± 0.0573	**0.0734 ± 0.0221**
		Total	–	0.3614 ± 0.1463	0.5210 ± 0.4261	**0.1534 ± 0.1396**	0.4679 ± 0.0845	0.2106 ± 0.1310	0.3371 ± 0.0626	0.2582 ± 0.1912	0.1986 ± 0.0696	0.5557 ± 0.2084	0.1635 ± 0.0493
	Out-of-Sample	Direct	0.1231 ± 0.0392	0.0493 ± 0.0306	0.0301 ± 0.0360	0.0281 ± 0.0339	0.4858 ± 0.0557	0.0273 ± 0.0357	0.1360 ± 0.0185	0.1456 ± 0.0678	0.4406 ± 0.0526	0.2580 ± 0.1965	**0.0273 ± 0.0171**
		Spillover	–	0.1930 ± 0.1221	0.2698 ± 0.2076	0.1071 ± 0.1085	0.9551 ± 0.1180	0.1052 ± 0.0729	0.1047 ± 0.0210	0.1111 ± 0.0515	0.0941 ± 0.0656	0.4321 ± 0.0511	**0.0747 ± 0.0229**
		Total	–	0.3589 ± 0.1413	0.5238 ± 0.4260	**0.1513 ± 0.1388**	0.4579 ± 0.0678	0.2052 ± 0.1290	0.3373 ± 0.0624	0.2577 ± 0.1886	0.1986 ± 0.0800	0.5017 ± 0.2739	0.1697 ± 0.0488
2	Within Sample	Direct	0.3321 ± 0.0699	0.1374 ± 0.0428	0.0924 ± 0.0251	0.1203 ± 0.0346	1.0071 ± 0.1177	0.1284 ± 0.0175	0.1968 ± 0.0297	0.4134 ± 0.0951	0.6143 ± 0.0704	0.3540 ± 0.0881	**0.0532 ± 0.0230**
		Spillover	–	0.2145 ± 0.0314	0.1893 ± 0.0729	0.1034 ± 0.0641	2.0070 ± 0.2105	0.1419 ± 0.0517	0.3135 ± 0.1165	0.1826 ± 0.1137	0.1618 ± 0.0739	0.9104 ± 0.1207	**0.0887 ± 0.0438**
		Total	–	0.5830 ± 0.1660	0.4363 ± 0.2046	0.3574 ± 0.1732	0.9167 ± 0.1765	0.4047 ± 0.1559	0.8739 ± 0.2517	0.7188 ± 0.3116	0.3267 ± 0.0729	1.2644 ± 0.0598	**0.2362 ± 0.0736**
	Out-of-Sample	Direct	0.3272 ± 0.0701	0.1356 ± 0.0402	0.0923 ± 0.0253	0.1207 ± 0.0372	0.1319 ± 0.0167	0.4227 ± 0.0862	0.1975 ± 0.0292	0.4047 ± 0.1031	0.6101 ± 0.0714	0.3399 ± 0.1061	**0.0540 ± 0.0282**
		Spillover	–	0.2222 ± 0.0326	0.1942 ± 0.0745	0.1079 ± 0.0660	2.0792 ± 0.1654	0.1384 ± 0.0540	0.3214 ± 0.1137	0.1942 ± 0.1148	0.1280 ± 0.0743	0.9118 ± 0.1120	**0.0946 ± 0.0462**
		Total	–	0.5765 ± 0.1633	0.4366 ± 0.2036	0.3575 ± 0.1726	0.9590 ± 0.1527	0.3898 ± 0.1592	0.8722 ± 0.2514	0.7017 ± 0.3246	0.3583 ± 0.0701	1.2517 ± 0.0717	**0.2391 ± 0.0725**

Table 4: Results of ATE estimation on Flickr Dataset. The ATE error ε_{ATE} is reported. The best performing method is bolded, and the second best is underlined.

Interference Degree	Setting	Effect	Methods										
			ND	TARNet+z	CFR+z	ND+z	GEst	NetEst	RRNet	TNet	DWR	GDML	GEOMINI
0.5	Within Sample	Direct	0.0698 ± 0.0300	0.6368 ± 0.0102	0.0695 ± 0.0301	0.1039 ± 0.0322	0.2331 ± 0.0226	0.0493 ± 0.0357	0.1172 ± 0.0179	0.1747 ± 0.1416	0.3038 ± 0.0510	0.0857 ± 0.0515	**0.0156 ± 0.0129**
		Spillover	–	0.2546 ± 0.0233	0.2556 ± 0.0325	0.2885 ± 0.0364	0.4461 ± 0.0229	0.1184 ± 0.0819	0.0545 ± 0.0030	0.1581 ± 0.0472	**0.0277 ± 0.0242**	0.2136 ± 0.0388	0.1496 ± 0.0238
		Total	–	1.1427 ± 0.0175	0.4722 ± 0.0704	0.5538 ± 0.0710	0.2151 ± 0.0090	0.2079 ± 0.1125	0.2263 ± 0.0262	0.4639 ± 0.1382	**0.1816 ± 0.0427**	0.2794 ± 0.0598	0.2705 ± 0.0228
	Out-of-Sample	Direct	0.0687 ± 0.0303	0.7223 ± 0.0090	0.0700 ± 0.0327	0.0520 ± 0.0222	0.2203 ± 0.0273	0.0446 ± 0.0372	0.1182 ± 0.0171	0.1719 ± 0.1338	0.3056 ± 0.0522	0.0912 ± 0.0561	**0.0148 ± 0.0144**
		Spillover	–	0.2421 ± 0.0290	0.2438 ± 0.0379	0.2748 ± 0.0427	0.4234 ± 0.0337	0.1111 ± 0.0791	0.0517 ± 0.0047	0.1469 ± 0.0410	**0.0320 ± 0.0177**	0.2033 ± 0.0263	0.1428 ± 0.0264
		Total	–	1.2263 ± 0.0130	0.4729 ± 0.0721	0.4952 ± 0.0631	0.2045 ± 0.0043	0.2059 ± 0.1113	0.2263 ± 0.0262	0.4536 ± 0.1307	**0.1788 ± 0.0438**	0.2543 ± 0.0828	0.2682 ± 0.0219
1	Within Sample	Direct	0.2503 ± 0.0453	0.6494 ± 0.0405	0.1680 ± 0.0333	0.2082 ± 0.0403	0.4644 ± 0.0718	0.1579 ± 0.0773	0.1629 ± 0.0079	0.6368 ± 0.0102	0.3637 ± 0.0505	0.1081 ± 0.0284	**0.0276 ± 0.0235**
		Spillover	–	0.4489 ± 0.0460	0.4195 ± 0.0142	0.3838 ± 0.0597	0.8606 ± 0.0814	0.1227 ± 0.1302	0.1020 ± 0.0117	0.2546 ± 0.0233	**0.0255 ± 0.0195**	0.3690 ± 0.0746	0.0834 ± 0.0370
		Total	–	1.6389 ± 0.0336	1.0742 ± 0.0254	0.9474 ± 0.0734	0.3682 ± 0.0206	0.4728 ± 0.2636	0.3723 ± 0.0134	1.1427 ± 0.0175	0.3184 ± 0.0528	0.4317 ± 0.0376	**0.2079 ± 0.0532**
	Out-of-Sample	Direct	0.2015 ± 0.0418	0.7146 ± 0.0358	0.1700 ± 0.0361	0.1724 ± 0.0461	0.4249 ± 0.0772	0.1475 ± 0.0671	0.1652 ± 0.0087	0.7223 ± 0.0090	0.3645 ± 0.0511	0.0938 ± 0.0390	**0.0318 ± 0.0220**
		Spillover	–	0.4171 ± 0.0579	0.3897 ± 0.0346	0.3550 ± 0.0648	0.7964 ± 0.1047	0.1125 ± 0.1245	0.0946 ± 0.0127	0.2421 ± 0.0290	**0.0154 ± 0.0092**	0.3495 ± 0.0728	0.0789 ± 0.0369
		Total	–	0.5004 ± 0.0396	1.0738 ± 0.0254	0.8994 ± 0.0761	0.3379 ± 0.0443	0.4531 ± 0.2447	0.3724 ± 0.0128	1.2263 ± 0.0130	0.2971 ± 0.0378	0.4218 ± 0.0422	**0.2062 ± 0.0547**
2	Within Sample	Direct	0.3888 ± 0.0720	0.7346 ± 0.0721	0.1241 ± 0.1316	0.2082 ± 0.0403	0.9692 ± 0.0445	0.1069 ± 0.0711	0.1469 ± 0.0370	0.6368 ± 0.0102	0.5243 ± 0.0520	0.1575 ± 0.1421	**0.0259 ± 0.0105**
		Spillover	–	1.0063 ± 0.0378	0.4283 ± 0.3880	0.3838 ± 0.0597	1.9841 ± 0.1210	0.2095 ± 0.0239	0.1723 ± 0.0157	0.2546 ± 0.0233	**0.0243 ± 0.0122**	0.7460 ± 0.0678	0.0687 ± 0.0551
		Total	–	2.7334 ± 0.0722	0.9741 ± 0.9370	0.9474 ± 0.0734	0.9827 ± 0.2050	0.5173 ± 0.2412	0.4833 ± 0.0809	1.1427 ± 0.0175	0.4908 ± 0.0552	0.8445 ± 0.2110	**0.1518 ± 0.1369**
	Out-of-Sample	Direct	0.3367 ± 0.0602	0.7836 ± 0.0736	0.1257 ± 0.1247	0.1724 ± 0.0461	0.9729 ± 0.1038	0.1043 ± 0.0858	0.1521 ± 0.0361	0.7223 ± 0.0090	0.5164 ± 0.0478	0.2408 ± 0.1874	**0.0315 ± 0.0077**
		Spillover	–	1.0363 ± 0.0762	0.4560 ± 0.4226	0.3550 ± 0.0648	2.0244 ± 0.1492	0.1776 ± 0.0353	0.1730 ± 0.0246	0.2421 ± 0.0290	**0.0541 ± 0.0125**	0.7537 ± 0.0662	0.0737 ± 0.0563
		Total	–	2.7817 ± 0.0742	0.9730 ± 0.9346	0.8994 ± 0.0761	1.0155 ± 0.1817	0.4591 ± 0.2533	0.4825 ± 0.0801	1.2263 ± 0.0130	0.5401 ± 0.0533	0.8883 ± 0.2980	**0.1513 ± 0.1371**

G More Results of ε_{ATE}

In addition to ε_{PEHE}, we also report the results of ε_{ATE} on the BC and Flickr datasets in Tables 3 and 4. Our method GEOMINI consistently achieves promising performance in terms of ε_{ATE}.

H External Experiments

Stable Counterfactual Prediction. As shown in Figure 2, we further evaluate GEOMINI and baselines under different treatment flip rates. As the flip rate increases, the performance of all methods degrades. In contrast, GEOMINI exhibits the smallest performance degradation and consistently achieves the lowest ε_{MSE} with small standard deviation. This means that our methods achieve the most accurate counterfactual prediction compared with baselines. Therefore, we conclude that GEOMINI is insensitive to the varying flip rate, effective, and stable.

Insensitive to Hyperparameter λ. In Figure 3, we perform experiments of GEOMINI using different values of hyperparameters λ. When $\lambda = 0$, *i.e.*, removing the Gromov-Wasserstein discrepancy term for balancing, the performance degrades notably, indicating that this term plays a critical role in improving the overall performance. When $\lambda > 0$, as λ varies from 0.05 to 10, GEOMINI maintains consistently stable results, suggesting that our method is robust and not sensitive to the choice of λ.

I External Term on Objective

Motivated by Theorem 4.3, we further incorporate an external term $\frac{1}{N^2}\|D^\phi - D^t\|_F^2$ to learn the model, and evaluate the performance of GEOMINI on the BC dataset (ind = 0.5). The results are shown in Table 5. We observe that the impact of this term is negligible in practice, as the scaling factor $\frac{1}{N^2}$ vanishes quickly with increasing N.

J Experiments on real-world data.

We evaluate our method on a real-world dataset of 473 power plants in the United States, examining the effect of selective catalytic/non-catalytic reduction (SCR/SNCR) installation on NO_x emissions.

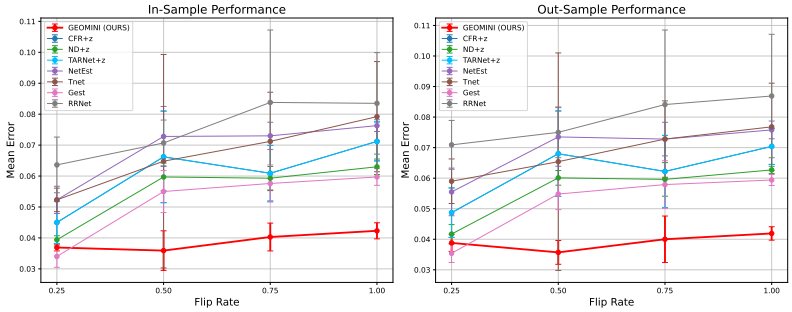
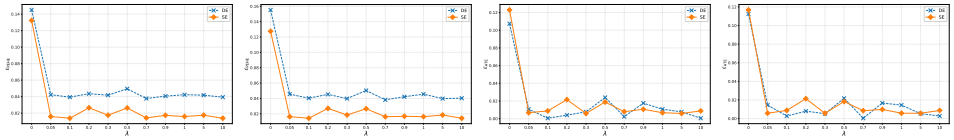


Figure 2: Counterfactual estimation error (MSE) on the BC dataset under different treatment flip rates, where $\varepsilon_{\text{MSE}} = \mathbb{E}\left[(Y - \hat{Y})^2\right]$.



(a) within sample PEHE (b) out of sample PEHE (c) within sample ATE (d) out of sample ATE

Figure 3: Results of varying values of hyperparameters on the BC dataset.

This dataset is consistent with those used in prior studies (Papadogeorgou et al., 2019; Liu et al., 2023; Chen et al., 2024). The installation status of SCR/SNCR is treated as the intervention, NO_x emissions serve as the outcome, and 18 environmental variables (e.g., the 4th highest temperature during the study period) are included as covariates. Following (Liu et al., 2023), each plant is associated with its five nearest neighbors.

We focus on NO_x emissions because prior work (Papadogeorgou et al., 2019) suggests that they are less susceptible to unobserved spatial confounding. The estimated main effect $f(1,0) - f(0,0)$ is -221.6, indicating that SCR/SNCR installation significantly reduces NO_x emissions when no neighboring plants adopt the technology. This result is consistent with (Papadogeorgou et al., 2019), which shows that the installation of SCR/SNCR would reduce on average 205.1 tons of NO_x emissions.

The spillover effects $f(1,z) - f(1,0)$ for $z \in \{0.1, 0.2, 0.3, 0.4\}$ are -7.8, -16.4, -24.0, and -32.2, respectively. These effects are consistently negative and increase in magnitude with z, suggesting positive spillover effects from neighboring adoption.

The total effects $f(1,z) - f(0,z)$ are -239.9, -260.4, -280.1, and -298.8, showing a monotonic increase in magnitude. This indicates that the overall benefit of SCR/SNCR installation is amplified as more neighboring plants adopt the technology.

Overall, the results demonstrate strong direct effects, significant positive spillovers, and increasing total effects, highlighting the importance of coordinated adoption.

K Discussion of Time Complexity

In a single iteration of Ricci flow, let d denotes the average degree of the nodes in the graph, the Wasserstein distance between two nodes is calculated by the Sinkhorn algorithm, which takes $O(d^2)$ operations. For m edges in the graph, the complexity of total computation in one iteration is in $O(md^2)$. The time complexity of calculating the Gromov-Wasserstein discrepancy between two graph with n nodes is $O(n^3)$ (Peyré et al., 2016). The running time results on BC (ind=0.5) are as follows. Although GEOMINI has a longer running time, it achieves promising performance.

Table 5: Results of GEOMINI with the term $\frac{1}{N^2}\|D^\phi - D^t\|_2^2$ on the BC dataset.

	Within sample						Out of Sample					
metric	$\sqrt{\varepsilon_{PEHE}^{DE}}$	ε_{ATE}^{DE}	$\sqrt{\varepsilon_{PEHE}^{SE}}$	ε_{ATE}^{SE}	$\sqrt{\varepsilon_{PEHE}^{TE}}$	ε_{ATE}^{TE}	$\sqrt{\varepsilon_{PEHE}^{DE}}$	ε_{ATE}^{DE}	$\sqrt{\varepsilon_{PEHE}^{SE}}$	ε_{ATE}^{SE}	$\sqrt{\varepsilon_{PEHE}^{TE}}$	ε_{ATE}^{TE}
w/	0.0531 ± 0.0193	**0.0133 ± 0.0050**	0.0689 ± 0.0249	0.0608 ± 0.0254	**0.1209 ± 0.0667**	**0.1061 ± 0.0690**	0.0548 ± 0.0189	**0.0140 ± 0.0037**	0.0707 ± 0.0229	**0.0621 ± 0.0231**	0.1343 ± 0.0659	0.1289 ± 0.0678
w/o	**0.0410 ± 0.0107**	0.0187 ± 0.0078	**0.0673 ± 0.0484**	**0.0600 ± 0.0462**	0.1286 ± 0.0484	0.1222 ± 0.0508	**0.0423 ± 0.0104**	0.0198 ± 0.0092	**0.0696 ± 0.0500**	0.0623 ± 0.0479	**0.1312 ± 0.0477**	**0.1242 ± 0.0500**

Method	Time (min)
NetEsimator	1.9289
ND	1.2505
TNet	5.0401
GEOMINI	15.8117

Table 6: Runtime comparison of different methods.