Counterfactual learning in networks: an empirical study of model dependence

Usman Shahid and Elena Zheleva
Department of Computer Science
University of Illinois at Chicago
Chicago, IL 60607
{hshahi6,ezheleva}@uic.edu

Abstract

Within the potential outcomes framework for causal inference, the choice of unit features and matching algorithms can impact the estimated causal effects, a problem known as model dependence. Here, we look at this problem in the context of observational network data and recently developed network representations within machine learning. By varying node representations, matching models, and methods for causal effect estimation on synthetic and real-world graph datasets, we show experimentally that estimated causal effects can vary significantly, both in sign and magnitude. With this paper, we aim to highlight some of the challenges of estimating causal effects from observational network data and hope to inspire further studies on model dependence in causal inference.

Introduction

Artificial intelligence and big data technologies are revolutionizing the sciences, engineering, and industry. Predictive systems are used to make sense of rapidly increasing amounts of data and support human decision making, from what to read, to whom to date, whether to invite a job applicant for an interview, and what drug to develop next. Meanwhile, these systems are limited in their ability to answer causal questions from historical data. If a social media user didn't follow fake news media accounts, would they have expressed less radical views? If a recommendation engine showed a more diverse set of job applicants to a hiring manager, would that have led to better hires? The answers to such questions require counterfactual reasoning, what the outcome of interest would have been if the circumstances in which the observed outcome occurred were different.

It is especially challenging to answer counterfactual questions with "big data" which is inherently biased, noisy, and exhibits complex relationships, unlike carefully designed i.i.d. data from surveys (jap 2015). To capture these data properties, it is convenient to represent many real-world data sources as networks (or graphs) and reason about them probabilistically (Getoor and Taskar 2007). In these networks, nodes represent interdependent entities, such as people, companies, websites, and diseases, while edges denote different relationships between these entities, such as friendship, hyperlink, contribution, and spread of disease. In networks, the subjects' outcomes may not independent of each other and the characteristics of a subject can be correlated with the characteristics of the subject's neighbors.

One of the challenges of answering counterfactual questions from observational data is that the process by which treatment (e.g., following fake news media accounts) and control (e.g., not following such accounts) are assigned to units (e.g., social media users) is often unknown, thus breaking desirable causal inference assumptions. Within the potential outcomes framework, researchers have studied matching as a way to deal with selection bias in observational data (Stuart 2010). However, just like the performance of machine learning is heavily dependent on the choice of data representation and model assumptions (Bengio, Courville, and Vincent 2013), potential outcome estimates depend on the choice of features and models (Ho et al. 2007). Different choices can change the sign and statistical significance of discovered causal effects. Thus, two researchers who study the same data may have widely different conclusions due to the difference in their data representations and models. This problem is known as *model dependence* of causal inference (Ho et al. 2007).

Model dependence is especially relevant in the context of big data where there are plethora of machine learning models available to researchers and deciding which specific features to collect and which model and data representation to use is at the discretion of each researcher. This is further exacerbated by the fact that typically there is no ground truth of causal effect in such data which makes it hard to build benchmarks and to improve on state-of-the-art baselines, a common practice in machine learning. Additionally, learning a causal graph (Bareinboim and Pearl 2016; Pearl 2009) on network data is a non-trivial task that requires not only domain knowledge about the constraints for structure learning (Colombo and Maathuis 2014; Heinze-Deml, Maathuis, and Meinshausen 2018; Sridhar, Pujara, and Getoor 2018) but also a way to model the relations between entities in the network (Friedman et al. 1999; Getoor and Taskar 2007).

The goal in this paper is to understand model dependence in relational data and inform the design of further studies on causal inference from network data. We revisit the potential outcomes framework (Rubin 1974) through the lens of recent advances in machine learning for networks and specify the relationship between network representations and causal effect estimation. We propose to capture each network node by leveraging data representation frameworks from graph mining (Khan and Ranu 2017) and statistical relational learning (Getoor and Taskar 2007). We vary the following model components: whether the node features are raw or use embeddings, whether they consider the network structure or not, whether matching utilizes fully blocked or a propensity score model, whether the estimator uses SATT on the matched nodes, SATE with a predicted counterfactual, or a linear model for causal effect estimation. Using both synthetic and real-world network data, we test the hypothesis that model choices lead to different causal effect estimates. To the best of our knowledge, this is the first study that sheds light on the important question of model dependence in causal effect estimation from network data.

Related Work

Our work draws upon three main areas of research: causal inference in networks, network representations, and model dependence. We briefly describe each.

Causal inference in observational network data. Many real world datasets can be naturally represented as networks, and the confounding effect of relational covariates (i.e. covariates derived from the relational structure) needs to be taken into account when estimating causal effects. Arbour et al. (Arbour et al. 2014) developed Relational Propensity Score Matching (RPSM) which accounts for non-trivial relational confounders to allow for the application of propensity score matching. They also developed another method Relational Covariate Adjustment (RCA) to infer networks effects (Arbour, Garant, and Jensen 2016) through an extension of Pearl's backdoor criterion to the relational domain (Pearl 2009).

Network representations. Recent advances in network representation learning have shown that embedded representations of nodes can improve the overall performance of machine learning models. Two prominent embedded representations are $Node2Vec$ (Grover and Leskovec 2016) and $GraphSAGE$ (Hamilton, Ying, and Leskovec 2017b). $Node2Vec$ primarily learns from the structure of the network, whereas $GraphSAGE$ considers the attributes of nodes in addition to the structure. In our work, we compare raw and embedded representations in the context of causal inference.

Model dependence. Model dependence refers to the dependence between the researcher's model choice and the magnitude of the discovered causal effects (Ho et al. 2007; King and Nielsen 2016). Model dependence highlights the problem that a researcher can produce results that are agreeable with their posed hypotheses by just changing the model (King and Langche 2006).

Causal effect estimation in networks

Let $G = (\boldsymbol{V}, \boldsymbol{E})$ denote an undirected, unweighted, attributed graph where $\boldsymbol{V}$ is the set of nodes and $\boldsymbol{E}$ is the set of undirected edges between these nodes. To avoid confusion with the term graph referring to graphical models, we will refer to the data graph G as *network*. If a node v_i has an edge with node v_j then $\{v_i, v_j\} \in \boldsymbol{E}$. Each node $v_i \in \boldsymbol{V}$ has an m-dimensional vector of attributes modeled as random variables $v_i.\boldsymbol{A} \in \mathbb{R}^m$, a treatment assignment $v_i.T$ and a measure of an outcome of interest $v_i.Y$. If $v_i.T = 0$, then node v_i belongs to the control group V_1, while $v_i.T = 1$ signifies that v_i belongs to the treatment group V_0. Note that some of the attribute or treatment values can be missing. Node attributes reflect each node's ego network and can include information about the ego node itself, its edges and its immediate neighbors, from which the node representations can be derived, as discussed in *network representations for matching* section.

Effect Estimation

The main premise of the potential outcomes framework of causal inference is that we can observe the outcome of a target variable for an individual v_i in either the treatment or control group (but not both), and we can estimate the counterfactual, the unobserved outcome if they were in the other group (Rubin 1974). Let $v_i.Y(1)$ and $v_i.Y(0)$ denote the *potential outcomes* of $v_i.Y$ if unit v_i were assigned to treatment ($v_i.T = 1$) or control ($v_i.T = 0$), respectively. The treatment effect (or causal effect) for unit v_i is the difference $g(i) = v_i.Y(1) - v_i.Y(0)$.

For the treatment effect to be estimated, the following assumptions have to hold:

- *Overlap* is the assumption that each unit assigned to the treatment or control group could have been assigned to the other group.
- *Stable unit treatment value assumption* (SUTVA) states that the outcome of unit v_i depends only on the treatment it receives and not on the treatment other units receive.
- *Ignorability* – also known as *conditional independence* (Pearl 2009) and *absence of unmeasured confoundness* (Ho et al. 2007) – is the assumption that all variables $v_i.\boldsymbol{A}$ that can influence the outcome $v_i.Y$ are observed in the data and there are no unmeasured confounding variables (ones that can cause changes in both the treatment and the outcome variables).

To estimate the causal effects, we consider three methods: sample-average treatment effect (SATE), sample-average treatment effect for the treated (SATT), and a linear model (Imbens 2004) . SATE is defined as:

$$SATE = \frac{1}{|V|} \sum_{v_i \in V} (v_i.Y(1) - v_i.Y(0)), \qquad (1)$$

Matching Method	Effect Estimator
PSM	BSATT BSATE Linear Model
FBM	BSATT BSATE Linear Model

Table 1: Different configurations used for effect estimation. PSM indicates Propensity Score Matching and FBM indicates Fully blocked Matching.

while SATT focuses on effect for the treated nodes V_1:

$$SATT = \frac{1}{|V_1|} \sum_{v_i \in V_1} (v_i.Y(1) - v_i.Y(0)) \tag{2}$$

Since for each unit v_i, we can observe only one of the two potential outcomes, e.g. $v_i.Y(1)$, we need to be able to estimate the other, e.g., $v_i.Y(0)$.

Another method for estimating causal effects is through use of a linear model which performs regression adjustment for covariates and has been shown to work best in combination with matching methods (Rubin 1973). The model is given by:

$$v_i.Y = \beta_0 + \tau * v_i.T + W^T v_i.A + \epsilon$$

Here, W and $v_i.A$ are vectors where W represents a vector of weights, τ represents the estimated causal effect and ϵ is a noise term.

Balanced Causal Effect Estimation

In observational network data, subjects are not methodically assigned to treatment and control. It is likely that due to confounding covariates with uneven distributions in both groups, the results will be subjected to selection bias (Arbour, Garant, and Jensen 2016). The basic idea of matching is that subjects that are treated should be compared with similar subjects from the control group for effect estimation; hence, yielding an unbiased estimate of causal effect. We create a subset of nodes, $\mathcal{B} \subseteq V$ in which the covariate distribution is balanced between treatment and control group. This subset is created using a matching method which matches every treatment node in V_t with a single control node based on the similarity of covariates and discarding the remaining nodes, a method known as 1:1 matching (Stuart 2010). Table 1 summarizes the model configurations used in this study.

Balanced SATE. Given this balanced set $\mathcal{B} \subseteq V$, we can estimate causal effect using Balanced SATE (BSATE) as follows:

$$BSATE = \frac{1}{|\mathcal{B}|} \sum_{v_i \in \mathcal{B}} (v_i.\overline{Y}(1) - v_i.\overline{Y}(0)) \tag{3}$$

where

$$v_i.\overline{Y}(t) = \begin{cases} v_i.Y(t) & v_i.T = t \\ E[v_i.Y(t)|v_i.\mathbf{A}] & v_i.T \neq t \end{cases}$$

and $E[v_i.Y(t)|v_i.\mathbf{A}]$ can be any regression estimator. For example Arbour et al. (Arbour, Garant, and Jensen 2016) use Gradient Boosting tees based estimator to predict the missing counterfactual which we also use for our experiments.

Balanced SATT (BSATT). Similarly, given the set of treated nodes in balanced set $\mathcal{B}_1$, we define Balanced SATT as:

$$BSATT = \frac{1}{|\mathcal{B}_1|} \sum_{v_i \in \mathcal{B}_1} (v_i.Y(1) - v_i.\hat{Y}(0)) \tag{4}$$

where, $v_i.\hat{Y}(0) = v_j.Y(0)$ such that, $j = argmin_j(\{Dist(v_i, v_j)|v_j.T = 0\})$.

Balanced linear model. The balanced linear model is a linear model simply built on the matched nodes $\mathcal{B}$.

Effect Estimation Pipeline for Networks

We have broken down the causal inference process into five main steps, in order to compare across different network representations, matching models and effect estimation methods. The process is as follows:

1. Represent relational and/or non-relational covariates associated with each node $v_i \in V$ in the form of an n-dimensional vector $\boldsymbol{X}_i \in \mathbb{R}^n$ using a mapping function:

   $$R(G, i) \rightarrow \boldsymbol{X}_i, \forall v_i \in V$$

   We describe a number of covariate representations in the next section.

2. Define a "distance" metric to estimate pairwise distance between network nodes based on their representations as:

   $$D : \boldsymbol{X} \times \boldsymbol{X} \rightarrow d, d \in \mathbb{R}$$

3. For every node in treatment group, find another node in control group with minimum distance as defined by D. This method is referred to as $1:1$ matching (Stuart 2010). We create a set of triples (v_i, v_j, d) such that v_j is the closest control node to the treatment node v_i based on distance d as:

   $$\{(v_i, v_j, d)|D(\boldsymbol{X}_i, \boldsymbol{X}_j) \leq D(\boldsymbol{X}_i, \boldsymbol{X}_k) \wedge v_i, v_j, v_k \in V \wedge v_i.T = 1 \wedge v_j.T = 0 \wedge v_k.T = 0 \wedge i \neq j \neq k\}$$

   where, $\boldsymbol{X_i} = R(G, i), \boldsymbol{X_j} = R(G, j), \boldsymbol{X_k} = R(G, k)$

4. Sort the triples generated in Step 3 based on distance d in decreasing order and prune the top p percent of triples (i.e., the non-matches) and use the node pairs (v_i, v_j) from the remaining triples to form the set of matched nodes $\mathcal{B}$. To avoid any bias resulting from the choice of p, we estimate effect for a range of p values.

5. Given the remaining pairs $\mathcal{B} \subseteq V$, estimate effect using BSATT, BSATE and balanced linear model as described in the previous subsection.

Distance Metric. We define our distance metrics based on two different matching methods. First, we use Propensity Score Matching (PSM) (Stuart 2010) in which the distance metric is defined as:

Features	**Type**	**Algorithm**
Raw	Non-Relational	NA
	Relational	NA
	Combined	NA
Embeddings	Non-Relational	PCA
	Relational	Node2Vec
	Combined	GraphSAGE

Table 2: Types of representations used in Fully Blocked and Propensity Score Matching.

$$D(X_i, X_j) = (P(T|X = X_i) - P(T|X = X_j))^2$$

where $P(T|X = X_i)$ is probability of a node v_i belonging to the treatment group given its representation X_i. This is normally estimated using a logistic regression model (Stuart 2010). Second, we use Fully Blocked Matching (FB) using Euclidean distance as suggested by (King and Nielsen 2016), it is defined as:

$$D(X_i, X_j) = \|X_i - X_j\|$$

Model dependence

According to (King and Langche 2006) model dependence is defined as "the difference or distance between the predicted outcome values from any two plausible alternative models". Since the causal effects can be estimated using different matching methods and representations, the choice of representation can lead to model dependence. The difference between predicted causal effects based on different model choices can be reported as a measure of model dependence (King and Nielsen 2016). In the presence of ground truth, all models would be compared to the ground truth as a base model.

Limiting spillover

When estimating causal effect in networks, it is very important to account for spillover. If treatment and control nodes have edges between them, then the influence of treatment may flow from a treated to untreated node which is referred to as spillover effect. The presence of edges can break the SUTVA assumption of causal inference and challenges the validity of discovered causal effects. In order to minimize this effect, we take specific measures described in the Experiments section.

Network representations for matching

In attributed networks, a node can have non-relational features, ones that do not depend on the network (e.g., age, gender) and relational features, ones that consider the structure and features of other nodes within the node's neighborhood (e.g., number of neighbors with the same gender). Either of these can be confounding as discussed in (Arbour, Garant, and Jensen 2016). We experiment with non-relational and relational features as well as a combination of both. Each of these feature types is described below.

Raw Features

A node v_i in an attributed network has an m-dimensional vector of attributes $v_i.A$. A researcher may use these attributes as features in their raw form (i.e. without any pre-processing):

- **Raw non-relational features (raw-nrel)** include only node-level attributes and all the relational features are ignored. This representation assumes that both treatment and outcome are independent of the node's ego network.

  $$R_{raw\_nrel}(G, i) = v_i.\boldsymbol{A}$$

- **Raw relational features (raw-rel)** are modeled using a set of aggregate functions applied on the attributes of neighboring nodes, similar to features for collective classification in networks (Sen et al. 2008). Suppose we have a Nbr function defined as:

  $$Nbr(G, i) = \{v_j|\{v_i, v_j\} \in G.E\}$$

  Also, Agg is an aggregate function defined over a set of m-dimensional attribute vectors $\mathcal{A}$ as:

  $$Agg(\mathcal{A}) = (f(\{\boldsymbol{A}[1]|\boldsymbol{A} \in \mathcal{A}\}), f(\{\boldsymbol{A}[2]|\boldsymbol{A} \in \mathcal{A}\}), \\ ....f(\{\boldsymbol{A}[m]|\boldsymbol{A} \in \mathcal{A}\}))$$

  and f can be any aggregate function defined over a set of real values $R \subset \mathbb{R}$ i.e. $f(R) = r, r \in \mathbb{R}$. If

  $$\mathcal{Q}(G, i) = \{v.\boldsymbol{A}|v \in Nbr(G, i)\}$$

  then,

  $$R_{raw\_rel}(G, i) = Agg_m(\mathcal{Q}(G, i)) \oplus Agg_v(\mathcal{Q}(G, i)) \\ \oplus Agg_{mv}(\mathcal{Q}(G, i))$$

  where, $\oplus$ represents the vector concatenation operation and Agg_m, Agg_v, Agg_{mv} use $mean(R), variance(R)$ and $mean(R) * variance(R)$ as aggregate functions respectively (Arbour, Garant, and Jensen 2016).

- **Raw combination features(raw-comb)** are simply the aforementioned vector representations concatenated as follows:

  $$R_{raw\_comb}(G, i) = R_{raw\_nrel}(G, i) \oplus R_{raw\_rel}(G, i)$$

Node Embedding

Network nodes can be efficiently represented using dense embedded representations (Hamilton, Ying, and Leskovec 2017a). A researcher may choose one of these embedding methods as their representation to estimate causal effect.

- **Emdedding non-relational features (emb-nrel)** are constructed using the Principal Component Analysis (PCA) algorithm (Jolliffe 2002). PCA can be seen as a function which maps higher m-dimensional real valued vectors to lower n-dimensional vectors while preserving as much of relevant information as possible. Formally:

  $$PCA : \mathbb{R}^{|V| \times m} \to \mathbb{R}^{|V| \times n}.$$

The **emb-nrel** representation based on PCA is:

$$R_{emb\_nrel}(G, i) = PCA(\boldsymbol{V}.\boldsymbol{A})[i]$$

where $\boldsymbol{V}.\boldsymbol{A}$ is a $|V|$ by m matrix containing attributes for all nodes.

- **Embedding relational features (emb-rel)** are computed using Node2Vec model(Grover and Leskovec 2016). Node2Vec first generates a sequences of nodes using a random walker and then learns a n-dimensional dense vector representation for every node in the network using these sequences in such a way that the nodes which have similar set of neighbors are closer in the distributed vector space and nodes with different neighbors are far apart. Node2Vec can be seen as a function which takes as input our network G and returns a matrix of vector representations for each node.

  $$Node2Vec(G) = \boldsymbol{M}^{|G.V| \times n}$$

  Where $\boldsymbol{M}$ is a matrix. Note that this method does not explicitly use node attributes but still captures them as latent factor in networks with high homophily. Given $Node2Vec$ function, we can describe our node representation as:

  $$R_{emb\_rel}(G, i) = Node2Vec(G)[i]$$

- **Embedding combination features (emb-comb)** are obtained through GraphSAGE algorithm (Hamilton, Ying, and Leskovec 2017b) which propagates and aggregates attribute level information from the node neighborhood using an unsupervised method to generation representation. Similar to Node2Vec, GraphSAGE can be seen as a function on a given network G:

  $$GSAGE(G) = \boldsymbol{M}^{|G.V| \times n}$$

  Where $\boldsymbol{M}$ is a matrix. Given this setup, we can define our representations as:

  $$R_{emb\_comb}(G, i) = GSAGE(G)[i]$$

Experiments

In this section we describe the datasets we have used for our experiments, our experimental setup and results.

Datasets

Synthetic: In real world data actual causal effect are often unknown, and to enable error analysis, we generate synthetic data following closely the process described by Arbour et al. (Arbour, Garant, and Jensen 2016). We create 100 different preferential attachment networks with $1,024$ nodes each. Each new node added to the network forms 3 edges based on a probability distribution determined by node-degree. Higher degree means higher likelihood for attachment. Each node is first initialized with 50 attributes, randomly sampled from a unit normal distribution with 1-hop network effects. The confounding term is generated using a linear combination L_i for node v_i as:

$$L_i = W^T R_{raw\_comb}(G, i)$$

This produces a total of 40 confounders for each node. The weight vector W is also a 40-dimensional vector in which each value is sampled from $U(-1, 1)$. Treatment ($v_i.T$) for node v_i is then sampled from a binomial distribution in such a way that it is dependent on weighted confounding term L_i. Spillover effect is simulated using a label propagation algorithm where the treatment is re-assigned based on $\theta_{nbr,i}$ (proportion of nodes in the neighborhood of v_i which are assigned treatment). Outcome ($v_i.Y$) is generated as a function of treatment ($v_i.T$), confounding term L_i, the proportion of treated nodes in neighborhood,$\theta_{nbr,i}$. Note that the best representation to model our synthetic data is R_{raw_comb} since it contains all the confounding variables. This set of 100 networks will be referred to as *synthetic* dataset.

Additional parameters of the data generation model from (Arbour, Garant, and Jensen 2016) are: true effect $= 2$, $\beta_L = 15$ (confounding factor), $\beta_P = 5$ (Peer effect), $s = 3$ and $\epsilon \sim \mathcal{N}(0, 1)$. To augment our representations, we concatenate $\theta_{nbr,i}$ at the end of our representation vector X_i for node v_i.

Real-world dataset: We use a Twitter dataset of hateful users by (Riberio et al. 2018). This is a large network with $273,344$ user nodes. The edges between nodes signify whether a user retweeted or was retweeted by another user. We classify a user as "hateful" if the use of abusive words in their tweets on average is abnormally high. Given a specific date (October 24, 2017), we classify all users as hateful/not-hateful before and after that date. Users who were not hateful before that date are our subjects, binary outcome is determined by whether they became hateful after specified date or not (1/0). Treatment is whether a person saw and retweeted a hateful user before the specified date or not (also binary). Node attributes are constructed based on the content of tweets using pre-trained word embeddings (Mikolov et al. 2013). This gives us 300 attributes for each node. To account for the spillover effect, we remove all those control nodes which have a treatment node in the ego network. We end up with $4,754$ treatment and $8,222$ control nodes for analysis. However, the entire network is used for computing representations.

Experimental setup

For each of our datasets, we represent the nodes according to the representations described in the previous section. For the relational features, we only consider 1-hop neighborhood, and for PCA dimensions we considered $n = 4$. For **emb-rel**, we create 256-dimensional vectors for each node. We do 20 walks per node with $p = 1$ and $q = 1$, where walks per node, p and q are all hyper-parameters of Node2Vec model, as described in (Grover and Leskovec 2016). For **emb-comb** we create 256-dimensional embeddings using GraphSAGE mean aggregation with 20 walks per node where the aggregation type and number of walks are hyper-parameters of the GraphSAGE model.

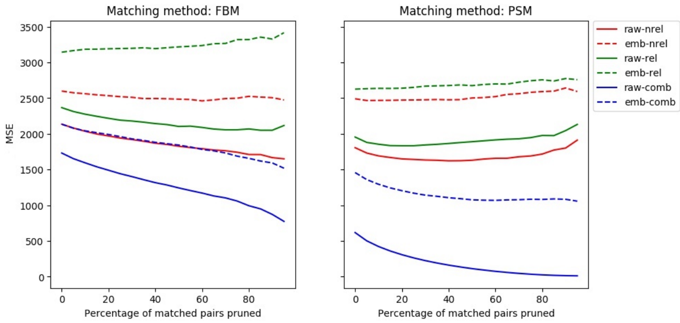

Figure 1: MSE in *synthetic* dataset using BSATT with FBM (left) and PSM (right).

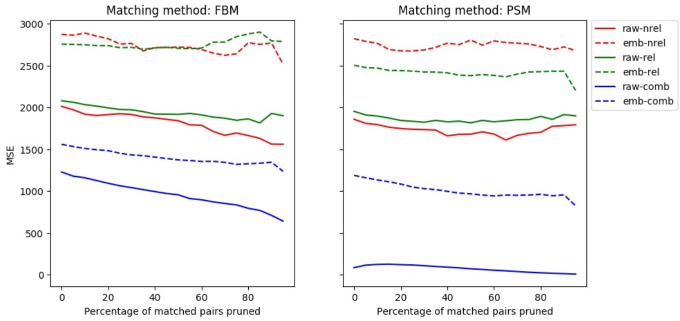

Figure 2: MSE in *synthetic* dataset using BSATE model with FBM(left) and PSM(right)

We perform matching using FBM and PSM, and then prune p percent of worst (most distant) matches where p varies between 0% and 95% with uniform intervals of 5%. For effect estimation, we consider only matched nodes. For the *synthetic* data, we compute the Squared Error (SE) and estimate the causal effect for each synthetic network. SE is given by:

$$SE(E) = (E - E_t)^2$$

where E is the estimated causal effect and E_t is the true effect. Since we have 100 networks, we report the Mean Squared Error (MSE) values over all synthetic networks. Ideally, in synthetic case, MSE should be 0 and Average Effect should be 2. For *hateful-users* where we don't have the ground truth for causal effect, we report the estimated causal effect.

Results

Figures 1, 2, and 3 show Mean Squared Error (MSE) using BSATT, BSATE and linear model effect estimation method respectively.. Since we are unaware of the actual effect in *hateful-users* dataset, we only show the estimated causal effect in Figures 4, 5 and 6 using the same three estimation methods. In each of these networks, embeddings are shown as dotted lines whereas raw representations are shown as solid lines. We have relational, non-relational and combinations for both cases resulting in 6 unique representations for the nodes in our network. Percentage of worse matches pruned (p) is shown on the x-axis.

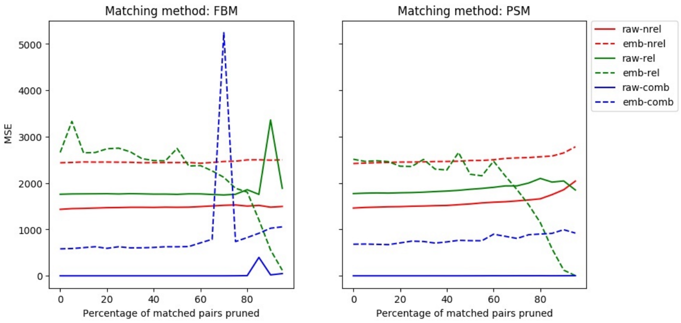

Figure 3: MSE in *synthetic* dataset using linear model with FBM(left) and PSM(right)

First, we observe that the estimated effect significantly varies depending on the representation used in both datasets (Figures 1, 2, 3, 4, 5, and 6) regardless of the choice of effect estimation method or matching method (even with the closest matches i.e. $p = 95$). This shows that the choice of network representation is important and in a real world scenario, a researcher who is not aware of the actual causal effect may end up with widely differing causal effect estimates based on the specific representation they choose. This risk is evident in Figure 4, 6 and 5 where the effect can be negative or positive depending on the choice of network representation.

Non-relational vs. Relational Figure 1, 2 and 3 show that the relational features provide better estimates than non-relational ones in most cases for the *synthetic* dataset. This can be explained by the fact that there are more relational confounders (i.e. 40) as opposed to non-relational (i.e. 10) ones in the data generation model. As expected, **raw-comb** is the best representation as it contains all the original confounding covariates used in *synthetic* dataset generation process.

Raw vs. Embeddings: Figure 1, 2 and 3 show that the embeddings have higher Mean Squared Error (MSE) than their raw feature counterparts of equivalent type. This can be explained by the fact that embeddings are approximating the actual confounding attributes which were raw features. However, **emb-rel** (Node2Vec) works surprisingly well with synthetic data (0 error) in combination with linear regression and high percentage of pruned nodes (Figure 3). This might be happening because after pruning, the treatment/control pairs have the same nodes in their neighborhoods and hence the same confounding covariates, however this needs further investigation. Given that **emb-comb** (GraphSAGE) works on a similar principle, one may expect similar results from it. Although, Figure 3 shows that it works better than other representations but not it is not as good as **emb-rel** when only the best matches are kept.

PSM vs FBM: In Figures 1 and 2, we see that PSM performs relatively better than FBM in terms of MSE. For our best representation (**raw-comb**) of *synthetic*

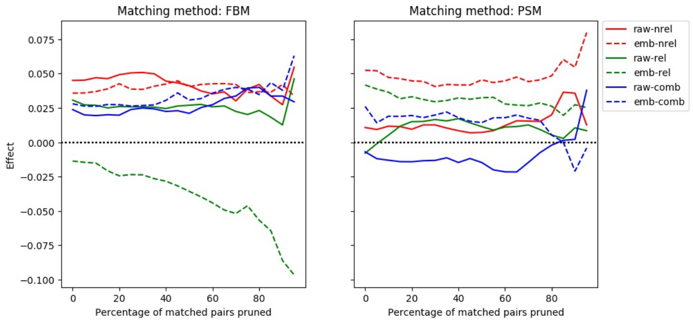

Figure 4: Effect estimate in *hateful-users* dataset using BSATT with FBM(left) and PSM(right).

data we can see that the MSE is close to 0 after pruning. In Figure 3 we can see that PSM is less noisy than the FBM which produces unexpected spikes. In Figure 4, there is less variance in most representations using FBM method, however **emb-rel** is an important exception.

BSATT vs. BSATE vs. linear model: While the BSATT and BSATE estimations benefit more from the process of matching, as shown in Figure 1, the best results are obtained when using a combination of matching with linear model as shown in Figure 3 for *synthetic* data. This is because the underlying data generation process specifies a linear correlation of attributes with outcome and treatment. In *hateful-users* dataset linear model is less noisy and produces less variance in estimated effect. See Figure 4 and Figure 6 for comparison. BSATE in Figure 5 is the noisiest of all methods, most likely because of the complexity of the underlying Gradient Boosted trees model which may cause problems while trying to adjust for noisy covariates.

We also note that the methods which perform well on *synthetic* dataset estimate a positive effect for most representations in *hateful-users* dataset (Figure 6). Another observation is that in *hateful-users* dataset, there is a positive spike in effect when a large number of nodes are pruned, leaving only the best matches. Both of these observations are in-line with our speculation that effect should be positive (i.e., retweeting "hateful" users can make you "hateful").

Discussion and conclusion

We presented an empirical study that highlights the model dependence problem in causal effect estimation in networks. Based on the results, we can draw the general conclusion that without knowledge of the underlying data generation process, causal effect estimates in networks can vary widely in both magnitude and sign. Unfortunately, in real world scenarios, the data generation process is unknown and the causal effect estimates depend on the feature representation, the matching model and the estimation method. This has important implications for any causal inference studies based on real-world network data that was not designed for causal inference. At the same time, matching methods reduce the estimation bias and propensity score matching leads to less noisy estimates than fully-blocked matching when varying the match threshold.

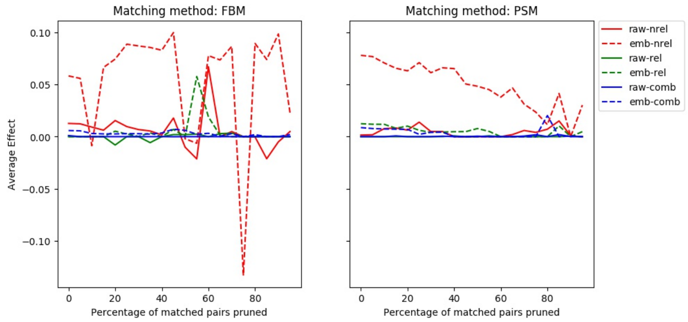

Figure 5: Effect estimate in *hateful-users* dataset using BSATE with FBM(left) and PSM(right).

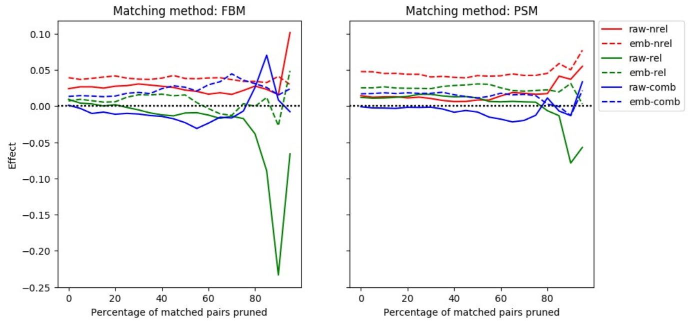

Figure 6: Effect estimate in *hateful-users* dataset using linear model with FBM(left) and PSM(right).

For synthetic data, experimental parameters which reflect the underlying data generation process for networks give the best estimations for synthetic data. Propensity score matching gives more accurate estimates than fully-blocked matching. Embeddings have higher error when the actual confounders are raw features. Linear model performs better than BSATT and BSATE, and a combination of relational and non-relational features gives the best estimates. We also discovered that relational embeddings with a balanced linear model work surprisingly well.

Further investigation is needed to understand why some embedding models lead to better causal effect estimates than others. Another fruitful direction for future work would be to understand the role of different network structures on the estimated effects by identifying which representations, effect estimation models and matching methods are more appropriate for different network generation models and parameters.

References

[Arbour et al. 2014] Arbour, D. T.; Marazopoulou, K.; Garant, D.; and Jensen, D. D. 2014. Propensity score matching for causal inference with relational data. In *UAI Workshop on causal inference*, 25–34.

[Arbour, Garant, and Jensen 2016] Arbour, D.; Garant, D.; and Jensen, D. 2016. Inferring network effects from observational data. In *KDD*.

[Bareinboim and Pearl 2016] Bareinboim, E., and Pearl, J. 2016. Causal inference and the data-fusion problem. *Proceedings of the National Academy of Sciences* 113(27):7345–7352.

[Bengio, Courville, and Vincent 2013] Bengio, Y.; Courville, A.; and Vincent, P. 2013. Representation learning: A review and new perspectives. *IEEE transactions on pattern analysis and machine intelligence* 35(8):1798–1828.

[Colombo and Maathuis 2014] Colombo, D., and Maathuis, M. H. 2014. Order-independent constraint-based causal structure learning. *The Journal of Machine Learning Research* 15(1):3741–3782.

[Friedman et al. 1999] Friedman, N.; Getoor, L.; Koller, D.; and Pfeffer, A. 1999. Learning probabilistic relational models. In *IJCAI*, volume 99, 1300–1309.

[Getoor and Taskar 2007] Getoor, L., and Taskar, B., eds. 2007. *Introduction to statistical relational learning.* MIT Press.

[Grover and Leskovec 2016] Grover, A., and Leskovec, J. 2016. Node2vec: Scalable feature learning for networks. In *KDD*.

[Hamilton, Ying, and Leskovec 2017a] Hamilton, W.; Ying, R.; and Leskovec, J. 2017a. Representation learning on graphs: Methods and applications. *TKDE*.

[Hamilton, Ying, and Leskovec 2017b] Hamilton, W.; Ying, Z.; and Leskovec, J. 2017b. Inductive representation learning on large graphs. In *NIPS*.

[Heinze-Deml, Maathuis, and Meinshausen 2018] Heinze-Deml, C.; Maathuis, M. H.; and Meinshausen, N. 2018. Causal structure learning. *Annual Review of Statistics and Its Application* 5:371–391.

[Ho et al. 2007] Ho, D.; Imai, K.; King, G.; and Stuart, E. 2007. Matching as nonparametric preprocessing for reducing model dependence in parametric causal inference. *Political Analysis* 15(3):199–236.

[Imbens 2004] Imbens, G. W. 2004. Nonparametric estimation of average treatment effects under exogeneity: A review. *Review of Economics and statistics* 86(1):4–29.

[jap 2015] 2015. Big data in survey research: Aapor task force report. *Public Opinion Quarterly* 79(4):839–880.

[Jolliffe 2002] Jolliffe, I. 2002. *Principal component analysis.* Springer.

[Khan and Ranu 2017] Khan, A., and Ranu, S. 2017. Big-graphs: Querying, mining, and beyond. In *Handbook of Big Data Technologies*. Springer. 531–582.

[King and Langche 2006] King, G., and Langche, Z. 2006. The dangers of extreme counterfactuals. *Political Analysis* 14(2):131–159.

[King and Nielsen 2016] King, G., and Nielsen, R. 2016. Why propensity scores should not be used for matching. *Working paper* 378.

[Mikolov et al. 2013] Mikolov, T.; Sutskever, I.; Chen, K.; Corrado, G.; and Dean, J. 2013. Distributed representations of words and phrases and their compositionality. In *NIPS*.

[Pearl 2009] Pearl, J. 2009. *Causality.* Cambridge Univ Press.

[Riberio et al. 2018] Riberio, M.; Calais, P.; Santos, Y.; Almeida, V.; and Meira, W. 2018. Characterizing and detecting hateful users on twitter. In *ICWSM*.

[Rubin 1973] Rubin, D. B. 1973. The use of matched sampling and regression adjustment to remove bias in observational studies. *Biometrics* 29(1):185–203.

[Rubin 1974] Rubin, D. B. 1974. Estimating causal effects of treatments in randomized and nonrandomized studies. *Journal of Educational Psychology* 66(5):688.

[Sen et al. 2008] Sen, P.; Namata, G. M.; Bilgic, M.; Getoor, L.; Gallagher, B.; and Eliassi-Rad, T. 2008. Collective classification in network data. *AI Magazine* 29(3):93–106.

[Sridhar, Pujara, and Getoor 2018] Sridhar, D.; Pujara, J.; and Getoor, L. 2018. Scalable probabilistic causal structure discovery. In *IJCAI*, 5112–5118.

[Stuart 2010] Stuart, E. A. 2010. Matching methods for causal inference: A review and a look forward. *Statist. Sci.* 25(1):1–21.