

# GRAPH NEURAL NETWORKS FOR CAUSAL INFERENCE UNDER NETWORK CONFOUNDING

Marc ae P Leun<sup>†</sup>      Pante s Loupos<sup>‡</sup>

Marc

**Abstract.**    s p p    s    s | n<sup>e</sup> n<sup>e</sup>    o    s    on | n<sup>e</sup>    s

A    n n    s p    o<sup>i</sup>    s    n    s    e    po    s |    o<sup>i</sup> n<sup>e</sup>    f<sup>e</sup> n<sup>e</sup>    n

o    po<sup>e</sup> n | o    o<sup>e</sup>    s    n    s |    on n o<sup>e</sup>    e<sup>n</sup>    o<sup>e</sup>    p<sup>i</sup>    e<sup>o</sup> p<sup>e</sup>

o<sup>e</sup>    s    n<sup>e</sup>    e<sup>s</sup>    e<sup>e</sup>    e<sup>e</sup>    o<sup>e</sup>    on<sup>s</sup>    e<sup>e</sup>    nonp    e<sup>e</sup>    s    p    n<sup>e</sup>    s

o    s    e<sup>s</sup>    e<sup>e</sup>    e<sup>e</sup>    o<sup>e</sup>    s |    n<sup>e</sup> o<sup>e</sup>    on<sup>s</sup>    o<sup>e</sup> |    s    s |    s    n

e<sup>n</sup>    s    on | n<sup>e</sup>    s    on<sup>o</sup> n n    s    e<sup>e</sup>    e<sup>n</sup>    s    n    o<sup>v</sup>    e<sup>s</sup>    o<sup>i</sup> ||

n    s    on<sup>s</sup>    e<sup>s</sup>    o<sup>e</sup>    s    o<sup>i</sup>    s |    on    s    e<sup>e</sup>    |    e<sup>e</sup>    p<sup>e</sup>    o    n n |    s    e<sup>s</sup>

on<sup>o</sup> n n    n    e<sup>s</sup>    e<sup>e</sup>    non<sup>n</sup> | o<sup>v</sup>    e<sup>n</sup>    s    on | n<sup>e</sup>    on<sup>o</sup>

e<sup>s</sup>    o<sup>e</sup>    s    n    s    n<sup>i</sup>    s    |    on    o<sup>e</sup> |    s    o    on    o<sup>e</sup>    s

o<sup>i</sup> n<sup>e</sup>    on<sup>s</sup>    W    s    o<sup>v</sup>    p    n<sup>e</sup>    |    n<sup>e</sup>    s    e<sup>e</sup>    s    e<sup>e</sup> ||    s    e<sup>e</sup>    o

s    o<sup>e</sup>    W    e<sup>n</sup>    s    on | n<sup>e</sup>    s    on<sup>o</sup> n n    e<sup>e</sup>    s    |    s    n<sup>e</sup>    s    n

o<sup>i</sup> o<sup>i</sup>    pp o    e<sup>s</sup>    p    s    n<sup>e</sup>    p    s    on    on<sup>s</sup>    on    n<sup>e</sup>    e<sup>e</sup>    n<sup>e</sup>    s

e<sup>e</sup>    on<sup>s</sup>    e<sup>s</sup>    e<sup>e</sup>    o<sup>e</sup> |    s | o<sup>v</sup>    e<sup>n</sup>    s    on |    s    e<sup>e</sup>    e<sup>s</sup>    s

# 1 Introduction

U VA

x m y, x

A, X, A

## GNNs for Network Confounding

# Leung and Loupos

! ' A |  
, ' A | "  
! ' | ' ,  
! ' | ' ,

## GNNs for Network Confounding

“The main idea is to use a GNN to learn a representation of the network structure that is invariant to the confounding effect of the hidden variables. This is achieved by using a message-passing neural network (MPNN) that takes as input the adjacency matrix and the node features, and outputs a representation of the network structure. The MPNN is trained to minimize the loss between the representation and the ground truth network structure. The resulting representation is then used to predict the network structure, which is compared against the ground truth network structure to evaluate the performance of the GNN.”

“A GNN is a type of neural network that can process graph-structured data. It consists of a series of layers of nodes, where each node is connected to its neighbors. The nodes in each layer are updated based on the information from their neighbors in the previous layer. The final output of the GNN is a representation of the network structure, which can be used for various tasks such as classification, regression, and recommendation.”

“The GNN is trained on a dataset of network structures, where each network structure is represented by an adjacency matrix and a set of node features. The GNN is trained to minimize the loss between the representation and the ground truth network structure. The resulting representation is then used to predict the network structure, which is compared against the ground truth network structure to evaluate the performance of the GNN.”

“The GNN is trained on a dataset of network structures, where each network structure is represented by an adjacency matrix and a set of node features. The GNN is trained to minimize the loss between the representation and the ground truth network structure. The resulting representation is then used to predict the network structure, which is compared against the ground truth network structure to evaluate the performance of the GNN.”

### 1.2 Related Literature

“The related literature includes several papers that discuss the use of GNNs for network confounding. The first paper is by [1], which introduces a GNN-based approach for network confounding. The second paper is by [2], which discusses the use of GNNs for network confounding in a different context. The third paper is by [3], which discusses the use of GNNs for network confounding in a different context. The fourth paper is by [4], which discusses the use of GNNs for network confounding in a different context. The fifth paper is by [5], which discusses the use of GNNs for network confounding in a different context.”

“The related literature includes several papers that discuss the use of GNNs for network confounding. The first paper is by [1], which introduces a GNN-based approach for network confounding. The second paper is by [2], which discusses the use of GNNs for network confounding in a different context. The third paper is by [3], which discusses the use of GNNs for network confounding in a different context. The fourth paper is by [4], which discusses the use of GNNs for network confounding in a different context. The fifth paper is by [5], which discusses the use of GNNs for network confounding in a different context.”



# GNNs for Network Confounding

$\mathcal{N}_n = \{1, \dots, n\}$  is the set of nodes.  $\mathbf{A} \in \mathbb{R}^{n \times n}$  is the adjacency matrix.  $\mathbf{K} \in \mathbb{R}^{n \times n}$  is the confounding matrix.

## 2 Setup

For each node  $i \in \mathcal{N}_n$ , we observe a vector  $\mathbf{X}_i \in \mathbb{R}^{d_x}$  and a vector  $\mathbf{Z}_i \in \mathbb{R}^{d_z}$ . The observed data is  $(\mathbf{X}_i, \mathbf{Z}_i)$ .

The outcome  $Y_i$  is generated by a function  $g_n(\mathbf{i}, D, \mathbf{X}, \mathbf{A}, \varepsilon)$  and the confounding variable  $D_i$  is generated by a function  $h_n(\mathbf{i}, \mathbf{X}, \mathbf{A}, \nu)$ .

The functions  $g_n$  and  $h_n$  are defined as:

$$g_n(\mathbf{i}, D, \mathbf{X}, \mathbf{A}, \varepsilon) = \mathbb{E}[Y_i | D_i, \mathbf{X}_i, \mathbf{A}] + \varepsilon_i$$

$$h_n(\mathbf{i}, \mathbf{X}, \mathbf{A}, \nu) = \mathbb{E}[D_i | \mathbf{X}_i, \mathbf{A}] + \nu_i$$

The parameters  $\varepsilon$  and  $\nu$  are independent and identically distributed random variables.

Example 1

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

$$Z_i = (D_i, \mathbf{X}_i')$$

# Leung and Loupos

A

$$Y = \frac{1}{1+Z} + \sum_{k=0}^{\infty} \tilde{A}^{k+1} Z + \sum_{k=0}^{\infty} \tilde{A}^k \epsilon.$$

$$Y_i = g_n(i, D, X, A, \epsilon)$$

E a P

$$D_i = 1 + \frac{\sum_{j=1}^n A_{ij} D_j}{\sum_{j=1}^n A_{ij}} + \frac{\sum_{j=1}^n A_{ij} Z'_j}{\sum_{j=1}^n A_{ij}} + Z'_i + \epsilon_i$$

U

$(X, A, \nu$

D

$$D_i = h_n(i, X, A, \nu)$$

$(i, X, A, \nu$

$$D_j = 1$$

$$D_i$$

$$h_n(i, X, A, \nu)$$

E a P

$$D_i = i'$$

$$Y_i$$

$$Y_i = g_n(D_{N(i,K)}, i, D_i = 1, W'_i, i, \dots)$$

$W_i$

$(X, A, K$

$$Y_i(d) = g_n(i, d, X, A, \epsilon)$$

$$Y_i(d)$$

$$D_i$$



# GNNs for Network Confounding

$(X, A)$   
 $\varepsilon$

Assumption 1:  $U$  is independent of  $(X, A)$

$A_i = X_i + \varepsilon_i$

$S \subseteq \mathcal{N}$ ,  $D_S = \{D_i \mid i \in S\}$ ,  $X_S$   
 $A_S$

$\mathcal{S}$   
 $\mathcal{S}$   
 $\mathcal{S}$   
 $Y_i$   
 $D_i$   
 $E$  a  $\mathcal{H}$   
 $C$   $0$   
 $\sup_n \mathcal{N}(\mathcal{S}, C) | \mathcal{S}$   
 $\sup_n \mathcal{N}(\mathcal{S}, \mathcal{S})$   
 $D_{\mathcal{N}(i, K)}$   
 $n(\mathcal{S}, c_1 \mathcal{S}, K)$   
 $W_i, X_i$   
 $n(\mathcal{S}, 0) \mathcal{S}$

## 2.1 Related Literature

$U, V, A$   
 $Y_i = g(D_i, X_i, \epsilon_i)$   
 $D_i = X_i$   
 $T_i = f_n(i, D, A)$   
 $W_i = q_n(i, X, A)$   
 $f_n(\epsilon_i)$   
 $q_n(\epsilon_i)$   
 $n, A$

# GNNs for Network Confounding

$o \quad oo \quad n \quad n$

$$Y_i = g(T_i, W_i, \dots)$$

$$T_i \sim D_{i, \Psi} \prod_{j=1}^n A_{ij} D_j \quad W_i \sim X_{i, \Psi} \prod_{j=1}^n A_{ij} \frac{\sum_{j=1}^n A_{ij} X_j}{\sum_{j=1}^n A_{ij}}$$

$$T_i \quad Y_i \quad D \quad V$$

$D_{N(i,1)}$

$W_i$

$D_{N(i,K)}$

$K \quad A$

$A$

$T_i$

$W_i$

$U$

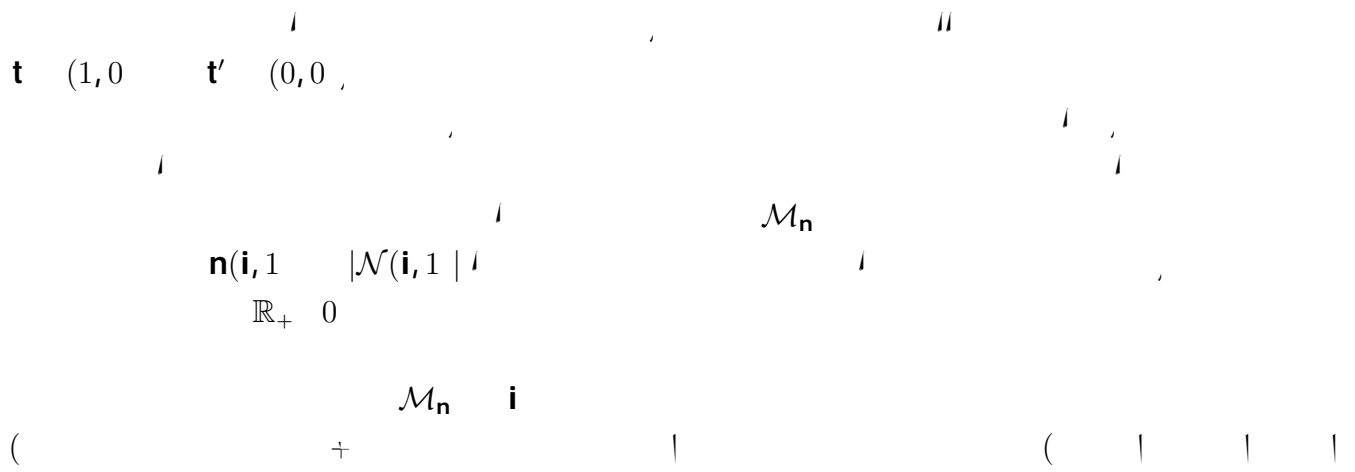
$W_i \quad X_i$

$X \quad A$

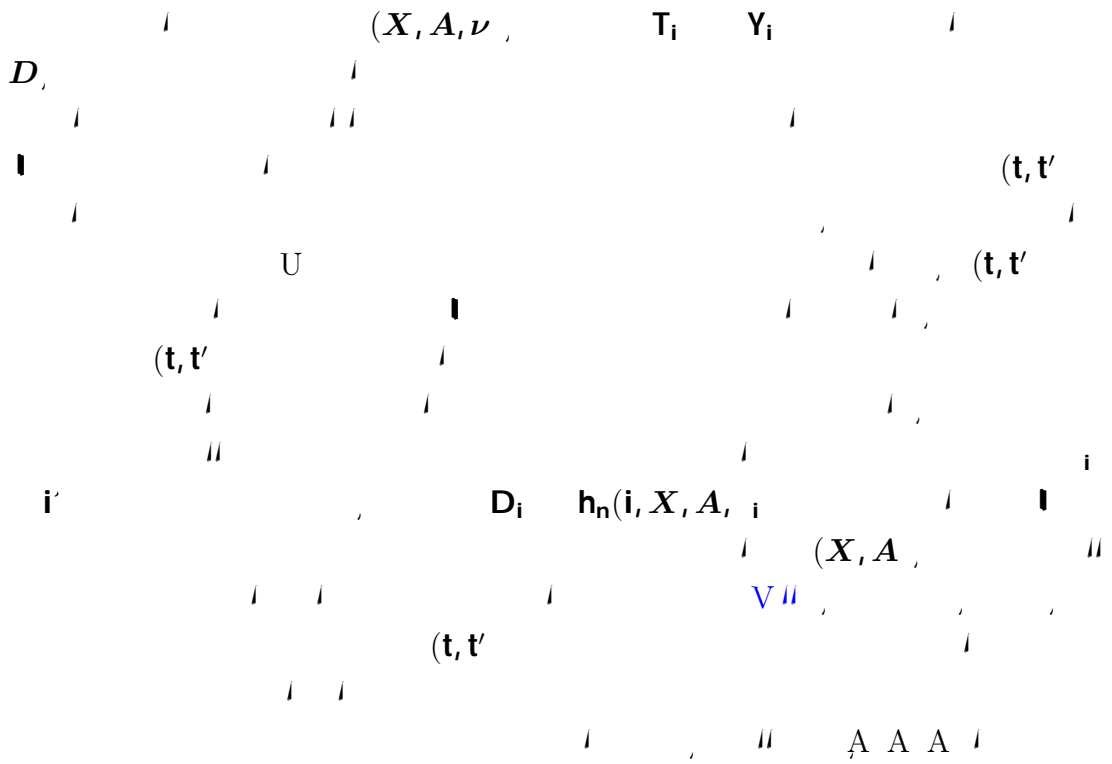
$(X, A)$



## GNNs for Network Confounding



Leung and Loupos



## GNNs for Network Confounding

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

### **3.1 Architecture**



## GNNs for Network Confounding

## Leung and Loupos

$$\begin{aligned}
 & \mathbf{X}_i \\
 & \Gamma(\mu, \Sigma, \min, \max) \\
 & \Gamma_1(\mu, \Sigma, \min, \max) \\
 & \Gamma_1(\mu, \Sigma, \min, \max) \\
 & \mathbf{n}(i, 1) \\
 & \Gamma_1(\mu, \Sigma, \min, \max) \\
 & \mathbf{S}(\mu, \Sigma, \min, \max) \\
 & \frac{1}{n} \sum_{i=1}^n \log \sum_{j=1}^n \mathbf{A}_{ij} + 1 \\
 & 1 \\
 & 0
 \end{aligned}$$

## GNNs for Network Confounding

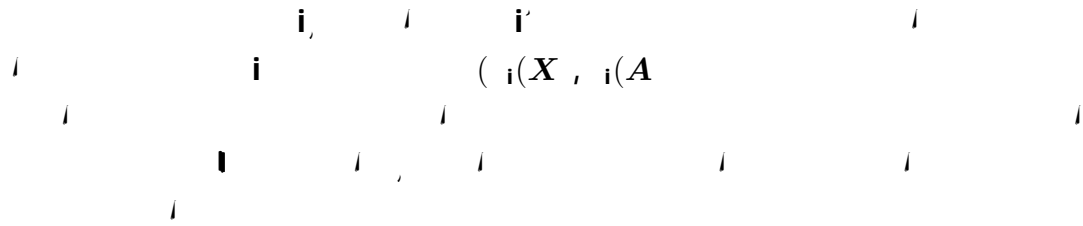
(

$$\hat{\mathbf{f}}_{\text{GNN}} = \underset{\mathbf{f} \in \mathcal{F}_{\text{GNN}}(\mathbf{L})}{\text{argmin}} \|\mathbf{y} - \mathbf{f}\|_2$$

A

$\mathbf{f} \in \mathcal{F}_{\text{GNN}}(\mathbf{L})$

## GNNs for Network Confounding



Proposition 1. For any  $n \in \mathbb{N}$ ,  $n \geq 2$ , and any  $\epsilon > 0$ ,

$$\begin{aligned}
 \mathbf{f}_n(i, D, A) &= \mathbf{f}_n(i, (D, (A, \\
 \mathbf{g}_n(i, D, X, A, \epsilon) &= \mathbf{g}_n(i, (D, (X, (A, (\epsilon, n
 \end{aligned}$$

# Leung and Loupos

$$\begin{aligned}
 & \mu_{t,t'}(\mathbf{i}) = \frac{1}{\mathbf{p}_t(\mathbf{i}, \mathbf{X}, \mathbf{A})} \sum_{\mathbf{d}} \mathbf{T}_i(\mathbf{d}) \frac{\mathbf{Y}_i(\mathbf{d} | \mathbf{X}, \mathbf{A})}{\mathbf{p}_t(\mathbf{i}, \mathbf{X}, \mathbf{A})} + \mu_{t'}(\mathbf{i}, \mathbf{X}, \mathbf{A}) \\
 & \mu_{t,t'}(\mathbf{i}) = \frac{1}{\mathbf{p}_{t'}(\mathbf{i}, \mathbf{X}, \mathbf{A})} \sum_{\mathbf{d}} \mathbf{T}_i(\mathbf{d}) \frac{\mathbf{Y}_i(\mathbf{d} | \mathbf{X}, \mathbf{A})}{\mathbf{p}_{t'}(\mathbf{i}, \mathbf{X}, \mathbf{A})} + \mu_{t'}(\mathbf{i}, \mathbf{X}, \mathbf{A}) \quad (t, t', \mathbf{i}) \\
 & \mathbf{i} \in \mathcal{M}_n \\
 & \mathbf{V} = \frac{1}{m_n} \sum_{\mathbf{i} \in \mathcal{M}_n} \mathbf{Y}_{t,t'}(\mathbf{i}, \mathbf{X}, \mathbf{A})^2
 \end{aligned}$$

**Assumption**  $M$   $n$   $p$   $4$   $u$   $o$   $n$   
 $n \in \mathbb{N}$ ,  $\mathbf{i} \in \mathcal{M}_n$ ,  $n$   $d \in \{0, 1\}^n$ ,  $\|\mathbf{Y}_i(\mathbf{d} | \mathbf{X}, \mathbf{A})\| \leq M$   
 $\mathbf{p}_t(\mathbf{i}, \mathbf{X}, \mathbf{A}) \in (0, 1)^u$ ,  $\hat{\mathbf{p}}_t(\mathbf{i}, \mathbf{X}, \mathbf{A})$ ,  $\mathbf{p}_t(\mathbf{i}, \mathbf{X}, \mathbf{A}) \in \mathbb{R}^u$

## GNNs for Network Confounding

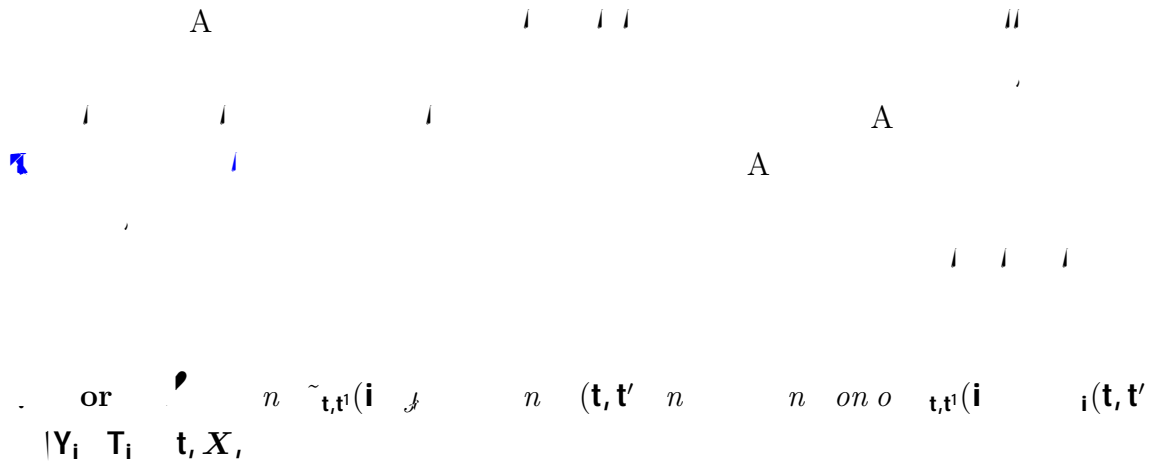
$$\mathbb{E}[y_i | x_i] = \mathbb{E}[y_i | x_i, z_i]$$

# Leung and Loupos

$$\Lambda_n(s) \quad 2M \quad M \quad t, t^{(i)} \quad \prod_{i=1}^n$$



# GNNs for Network Confounding



$$\| \hat{\mathbf{X}} - \mathbf{A} \|_F^2$$

## 5 Approximate Sparsity

$$\mathbf{A} \underset{(\mathbf{X}_{\mathcal{N}(i,L)})}{\text{minimize}} \quad \mathbf{L}$$









## GNNs for Network Confounding

$$(W_i)_{i=1}^n \sim \nu \quad (i)_{i=1}^n$$

$$V_i(W, \nu; \epsilon) = \frac{\sum_{j=1}^n A_{ij} W_j}{\sum_{j=1}^n A_{ij}} + \frac{\sum_{j=1}^n A_{ij} X_j}{\sum_{j=1}^n A_{ij}} + X_j + \epsilon + \frac{\sum_{j=1}^n A_{ij} j}{\sum_{j=1}^n A_{ij}}$$

$$Y_i \sim V_i(Y, \epsilon; y) \quad y \in (0.5, 0.8, 10, 1) \quad D_i \sim D_i \quad (0.5, 1.5, 1, 1)$$

$$D_i^0 \sim D_i^0 \quad (0, \nu; d = 0)$$

$$A_{ij} = \frac{\sum_{j=1}^n A_{ij} j}{\sum_{j=1}^n A_{ij}}$$

$$A_{ij} = \frac{\sum_{j=1}^n A_{ij} j}{\sum_{j=1}^n A_{ij}}$$

$$T_i \sim t \quad (Y_i, T_i, t, X, A) \quad n = 1000, 2000, 4000$$

### 6.2 Nonparametric Estimators

$$L = 1, \quad \Gamma_2(\dots)$$

# Leung and Loupos

$L = 1$

---

	$L = 1$	$L = 2$	$L = 3$
$n$	$\overbrace{111}^{111} \quad \overbrace{111}^{111} \quad \overbrace{111}^{111}$	$\overbrace{111}^{111} \quad \overbrace{111}^{111} \quad \overbrace{111}^{111}$	$\overbrace{111}^{111} \quad \overbrace{111}^{111} \quad \overbrace{111}^{111}$
$e$			
$e$			
$H$			
$\hat{\tau}(1,0)$	$\begin{matrix} \overline{\overline{11}} \\ \overline{11} \end{matrix}$	$\begin{matrix} \overline{\overline{11}} \\ \overline{11} \end{matrix}$	$\begin{matrix} \overline{\overline{11}} \\ \overline{11} \end{matrix}$
$\mathbb{C} \downarrow$	$\begin{matrix} \overline{\overline{11}} \\ \overline{11} \end{matrix}$	$\begin{matrix} \overline{\overline{11}} \\ \overline{11} \end{matrix}$	$\begin{matrix} \overline{\overline{11}} \\ \overline{11} \end{matrix}$
$\begin{matrix} e \\ e \\ \mathbb{C} \downarrow \end{matrix}$	$\begin{matrix} \overline{\overline{11}} \\ \overline{11} \end{matrix}$	$\begin{matrix} \overline{\overline{11}} \\ \overline{11} \end{matrix}$	$\begin{matrix} \overline{\overline{11}} \\ \overline{11} \end{matrix}$
$W \hat{\tau}(1,0)$	$\begin{matrix} \overline{\overline{11}} \\ \overline{11} \end{matrix}$	$\begin{matrix} \overline{\overline{11}} \\ \overline{11} \end{matrix}$	$\begin{matrix} \overline{\overline{11}} \\ \overline{11} \end{matrix}$
$W \mathbb{C} \downarrow$	$\begin{matrix} \overline{\overline{11}} \\ \overline{11} \end{matrix}$	$\begin{matrix} \overline{\overline{11}} \\ \overline{11} \end{matrix}$	$\begin{matrix} \overline{\overline{11}} \\ \overline{11} \end{matrix}$
$W$	$\begin{matrix} \overline{\overline{11}} \\ \overline{11} \end{matrix}$	$\begin{matrix} \overline{\overline{11}} \\ \overline{11} \end{matrix}$	$\begin{matrix} \overline{\overline{11}} \\ \overline{11} \end{matrix}$
$\mathbb{H} \mathbb{C} \downarrow$	$\begin{matrix} \overline{\overline{11}} \\ \overline{11} \end{matrix}$	$\begin{matrix} \overline{\overline{11}} \\ \overline{11} \end{matrix}$	$\begin{matrix} \overline{\overline{11}} \\ \overline{11} \end{matrix}$
$\mathbb{H}$	$\begin{matrix} \overline{\overline{11}} \\ \overline{11} \end{matrix}$	$\begin{matrix} \overline{\overline{11}} \\ \overline{11} \end{matrix}$	$\begin{matrix} \overline{\overline{11}} \\ \overline{11} \end{matrix}$

• In situations, the estimator is  $\hat{\tau}(1,0) = 0$ , treated  $\approx$







,

## **7.1 Comparison with He and Song (2024)**



# GNNs for Network Confounding

$n = 4413$

$\{0.01, 0.99\}$

$\mathbf{A}$

$\mathbf{T}_i^{(1)}$

	ADM	GNN			GLM		
		Layer	Layer	Layer	Order	Order	Order
Leader case							
$G_{ee}$	003	000	000	000	000	000	000
$G_{sc}$	00	00	00	00	00	00	00
$G_{all}$	003	00	00	00	00	003	00
Leader adopter case							
$G_{ee}$	0'3	00	00	003	00	00	0'3
$G_{sc}$	0'	00	003	00	00	00	0'00
$G_{all}$	0'3	00	00	00	00	0'	0'3
Adopter case							
$G_{ee}$	0'	00	00	003	003	003	0'00
$G_{sc}$	0'	00	003	00	00	003	0'00
$G_{all}$	0'3	00	00	00	00	003	0'00

$n = 4413$





## A Additional Results on GNNs

$$\begin{aligned}
 & \frac{1}{m_n} \sum_{i \in \mathcal{M}_n} \|\hat{\mathbf{y}}_t(i, \mathbf{X}, \mathbf{A}) - \mathbf{y}_t(i, \mathbf{X}, \mathbf{A})\|^2 = o_p(n^{-1/2}). \\
 & \frac{1}{m_n} \sum_{i \in \mathcal{M}_n} \|\hat{\mathbf{y}}_t(i, \mathbf{X}, \mathbf{A}) - \mathbf{y}_t(i, \mathbf{X}_{\mathcal{N}(i,L)}, \mathbf{A}_{\mathcal{N}(i,L)})\|^2 = o_p(n^{-1/2}). \\
 & \|\hat{\mathbf{y}}_t(i, \mathbf{X}, \mathbf{A}) - \mathbf{y}_t(i, \mathbf{X}_{\mathcal{N}(i,L)}, \mathbf{A}_{\mathcal{N}(i,L)})\| \leq C \frac{WL \log R}{n} \log n + \frac{\log \log n}{n} + \frac{1}{n}. \\
 & \frac{1}{n} \sum_{i=1}^n \|\hat{\mathbf{y}}_t(i, \mathbf{X}, \mathbf{A}) - \mathbf{y}_t(i, \mathbf{X}_{\mathcal{N}(i,L)}, \mathbf{A}_{\mathcal{N}(i,L)})\|^2 \leq C \frac{WL \log R}{n} \log n + \frac{\log \log n}{n} + \frac{1}{n}.
 \end{aligned}$$



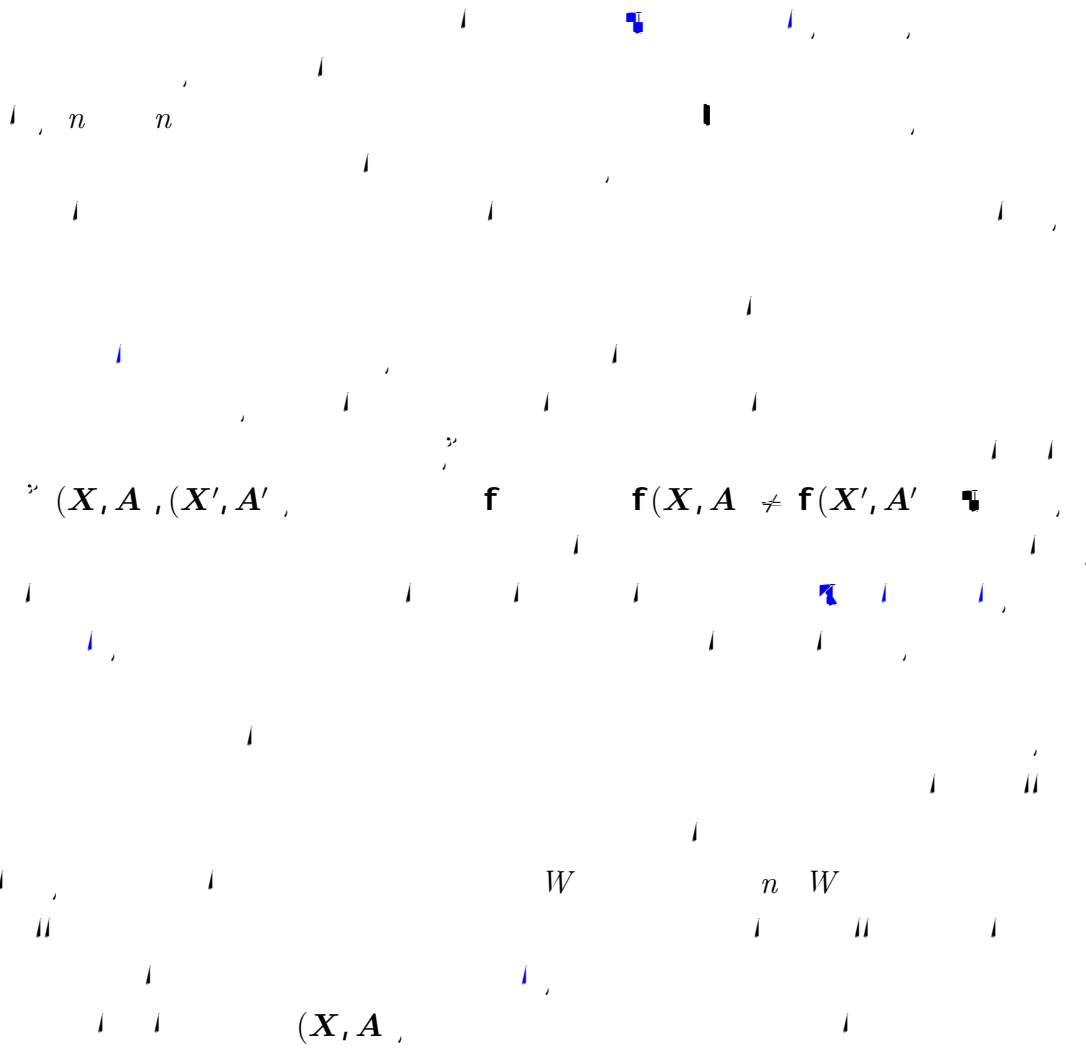
# GNNs for Network Confounding

A

A

$W, R, L, n$

## A.1 WL Function Class



## Leung and Loupos

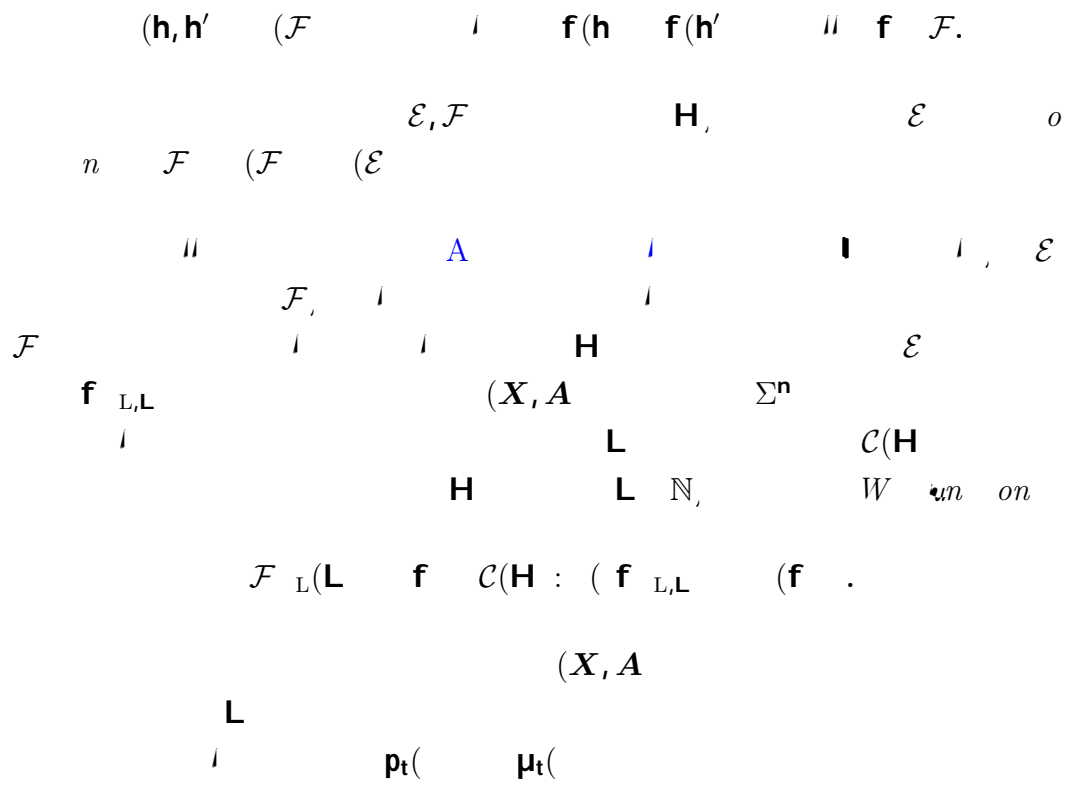
||

||

,

# GNNs for Network Confounding

$H^2$





# GNNs for Network Confounding

$\mathcal{F}_{\text{GNN}}(\mathbf{L}, \mathbf{X}, \mathbf{A})$

$(\mathbf{X}, \mathbf{A}) \rightarrow \mathbf{y} = f(i, \mathbf{X}, \mathbf{A}, \dots, \mathbf{y}) \in \mathbb{R}^n$

$\mathbf{L}_{L+1}(\mathbf{h}_i^{(L)}, \mathbf{h}_j^{(L)}; \mathbf{A})$

$\mathbf{X}_i, (\mathcal{F}_{\text{GNN}}(\mathbf{L}, \mathbf{X}, \mathbf{A}))$

$n$  times

$\blacksquare$

## A.2 Disadvantages of Depth

As the depth of the GNN increases, the number of parameters grows exponentially. This leads to a significant increase in the number of parameters, which can be a major disadvantage in terms of computational cost and model complexity.

Additionally, the model may suffer from overfitting as the depth increases, especially when the number of nodes in the graph is limited. This can result in poor generalization performance on new data.

Furthermore, the model's performance may be sensitive to the choice of hyperparameters, such as the learning rate and the number of layers, which can make the training process more challenging.

In summary, while GNNs are powerful tools for network analysis, their depth can be a significant disadvantage due to the exponential growth of parameters and the risk of overfitting.



# GNNs for Network Confounding

## B Verifying §8 Assumptions

$$\begin{aligned}
 & \mathbb{P} \left( \max_{\mathbf{A}} \sum_{\mathbf{s}} \left| \mathcal{N}_{\mathbf{A}}(\mathbf{i}, \mathbf{s}) - \mathbb{E}[\mathcal{N}_{\mathbf{A}}(\mathbf{i}, \mathbf{s})] \right| > c(1 - 4p)^{-1} \mathbf{s} \right) \\
 & \leq \mathbb{P} \left( \max_{\mathbf{A}} \sum_{\mathbf{s}} \left| \mathcal{N}_{\mathbf{A}}(\mathbf{i}, \mathbf{s}) - \mathbb{E}[\mathcal{N}_{\mathbf{A}}(\mathbf{i}, \mathbf{s})] \right| > c \cdot 0 \cdot \mathbf{p} \cdot \mathbf{A} \right) \\
 & \leq \mathbb{P} \left( \max_{\mathbf{A}} \sum_{\mathbf{s}} \left| \mathcal{N}_{\mathbf{A}}(\mathbf{i}, \mathbf{s}) - \mathbb{E}[\mathcal{N}_{\mathbf{A}}(\mathbf{i}, \mathbf{s})] \right| > c \cdot 0 \cdot \mathbf{p} \cdot \mathbf{A} \right)
 \end{aligned}$$

$$\sup_{\mathbf{n}} \max_{\mathbf{i} \in \mathcal{N}_{\mathbf{n}}} |\mathcal{N}_{\mathbf{A}}(\mathbf{i}, \mathbf{s})| \leq C \mathbf{s}^d$$

**C**  $0, d > 1$







# Leung and Loupos

$$D'_B = (D'_j)_{j \in B} \quad B \in \mathcal{N}_n \cup \{ \emptyset \}$$

$$p_t(i, X, A) = (D'_i + (D_i - D'_i) \mathbb{1}_{\{a, b\}} + (V_i - V'_i) \mathbb{1}_{\{ \cdot \}}) \cdot (D'_i \mathbb{1}_{\{a, b\}} + V'_i \mathbb{1}_{\{ \cdot \}}) + X, A$$

$$+ \mathbb{1}_{\{ \cdot \}} \cdot (D_i - D'_i) \mathbb{1}_{\{a, b\}} + (V_i - V'_i) \mathbb{1}_{\{ \cdot \}} \cdot X, A$$

$$R_0$$

$$(D'_i \mathbb{1}_{\{a, b\}})$$

## GNNs for Network Confounding

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

$$\text{root } \mu_t(\mathbf{i}, \mathbf{X}, \mathbf{A}) = \mathbb{E}[Y_i | \mathbf{i}, \mathbf{X}, \mathbf{A}] = p_t(\mathbf{i}, \mathbf{X}, \mathbf{A}),$$

$$\mathbf{B} = \mathcal{N}(\mathbf{i}, s), \quad Y_i' = g_{\mathbf{n}(\mathbf{i}, s)}(\mathbf{i}, D_{\mathbf{B}}, \mathbf{X}_{\mathbf{B}}, \mathbf{A}_{\mathbf{B}}, \epsilon_{\mathbf{B}})$$

$$| \mathbb{E}[Y_i | \mathbf{i}, \mathbf{X}, \mathbf{A}] - \mathbb{E}[Y_i' | \mathbf{i}, \mathbf{X}, \mathbf{A}] | \leq \frac{1}{n(s)} + \Lambda_{\mathbf{n}(\mathbf{i}, s)} \frac{1}{n(\mathbf{i}, s)} \frac{1}{n(s)}$$

, A ,

$|R_1|$  n

## GNNs for Network Confounding

$$|R_1| = \frac{1}{n} \sum_{i \in \mathcal{N}_n} \Lambda_n(i, s) \frac{1}{n} \sum_{i \in \mathcal{N}_n} \Lambda_n(i, s) \quad \blacksquare$$

Let  $C = \frac{1}{n} \sum_{i \in \mathcal{N}_n} Y_i' D_i'$  and  $d = \frac{1}{n} \sum_{j=1}^n A_{ij} D_j'$ .  
 Let  $\mathcal{C} = \{C, d\}$ .  
 $n \in \mathbb{N}, i \in \mathcal{N}_n, n \geq 1$ .

$$|Y_i|1_i(t) - 1_i(t') | X, A = C(1 + n(i, 1) - n(s)).$$

**Proof.** //  $a, b, \dots, V_i = \frac{1}{n} \sum_{j=1}^n A_{ij} D_j'$ ,  
 $V_i' = \frac{1}{n} \sum_{j=1}^n A_{ij} D_j'$ ,  $C = \frac{1}{n} \sum_{i \in \mathcal{N}_n} |D_i - D_i'|, |V_i - V_i'|$

$$|Y_i|1_i(t) - 1_i(t') | X = x, A = a$$

$$|Y_i|1_i(t) - 1_i(t') | C, X = x, A = a + C = (C^c | X = x, A = a$$

$$|C = 0, A = A$$

$$1_i(t) - 1_i(t') | D_i = |a, b, V_i|, \quad 1_i(t') - 1_i(t) | D_i' = |a, b, V_i'|, \dots$$

U  $C,$

$$1_i(t) - 1_i(t') | D_i = |a, b, V_i|, \quad \left( 1_i(t') - 1_i(t) | D_i' + (D_i - D_i') | a, b, V_i' + (V_i - V_i') | D_i' \right)$$



## GNNs for Network Confounding

$$\begin{aligned}
 & A \quad \left( \mathbf{Z}_i^{(s/2, \cdot)} \right)_{i \in H} \quad \left( \mathbf{Z}_j^{(s/2, \cdot)} \right)_{j \in H^1} \quad \mathcal{F}_n, \\
 & \left| \left( \cdot, \mathcal{F}_n \right) \right| \quad \left| \left( \cdot^{(s/2)}, \mathcal{F}_n \right) \right| + \left| \left( \cdot^{(s/2)}, \mathcal{F}_n \right) \right| \\
 & 2 \|\mathbf{f}'\|_\infty \quad \left| \left( \cdot^{(s/2)}, \mathcal{F}_n \right) \right| + 2 \|\mathbf{f}\|_\infty \quad \left| \left( \cdot^{(s/2)}, \mathcal{F}_n \right) \right| \\
 & 2 \mathbf{h} \|\mathbf{f}'\|_\infty \quad (\mathbf{f} + \mathbf{h}' \|\mathbf{f}\|_\infty) \quad (\mathbf{f}' \max_{i \in \mathcal{N}})
 \end{aligned}$$

## Leung and Loupos

$$\mathcal{L}_h = \mathcal{L}_{h^1}, \mathbf{s} = 0, (\mathbf{H}, \mathbf{H}') = \mathcal{P}_n(\mathbf{h}, \mathbf{h}'; \mathbf{s}),$$

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

$$\mathbf{f}(\mathbf{Y}_{i \in \mathcal{H}}), \mathbf{f}'(\mathbf{Y}_{i \in \mathcal{H}^1}), \mathbf{f}^{(s)}(\mathbf{Y}_{i \in \mathcal{H}}^{(s)}), \mathbf{f}'^{(s)}(\mathbf{Y}_{i \in \mathcal{H}^1}^{(s)})$$

A

$$\begin{aligned} & \left| \left( \cdot, \mathcal{F}'_n \right) \right| \left| \left( \cdot^{(s/2)}, \mathcal{F}'_n \right) \right| + \left| \left( \cdot^{(s/2)}, \mathcal{F}'_n \right) \right| \left| \left( \cdot^{(s/2)}, \mathcal{F}'_n \right) \right| \\ & 2 \|\mathbf{f}'\|_\infty \left| \left( \cdot^{(s/2)}, \mathcal{F}'_n \right) \right| + 2 \|\mathbf{f}\|_\infty \left| \left( \cdot^{(s/2)}, \mathcal{F}'_n \right) \right| \\ & 2 \|\mathbf{h}\| \|\mathbf{f}'\|_\infty (\|\mathbf{f}\| + \|\mathbf{h}'\| \|\mathbf{f}\|_\infty) (\mathbf{f}' \max_{i \in \mathcal{N}_n} \|\mathbf{Y}_i - \mathbf{Y}_i^{(s/2)}\|) \mathcal{F}'_n \\ & 2 \|\mathbf{h}\| \|\mathbf{f}'\|_\infty (\|\mathbf{f}\| + \|\mathbf{h}'\| \|\mathbf{f}\|_\infty) (\mathbf{f}' \mathbf{n}(\mathbf{s}^2), \end{aligned}$$

$$\left| \left( \cdot, \mathcal{F}'_n \right) \right| \left| \left( \cdot^{(s/2)}, \mathcal{F}'_n \right) \right| \left| \left( \cdot^{(s/2)}, \mathcal{F}'_n \right) \right| \left| \left( \cdot^{(s/2)}, \mathcal{F}'_n \right) \right|$$

A





## Leung and Loupos

$$\hat{\rho}_t(i, X, A) \quad | \quad C, C' = 0, \quad | \mathbf{R}_{it}^2 \quad |$$

$$\frac{1}{m_n} \sum_{i \in \mathcal{M}_n} \sum_{j \in \mathcal{M}_n} \left( Y_i - \mu_i \right) \left( Y_j - \mu_j \right) D, X, A \quad 1_i(t) 1_j(t) ($$

## GNNs for Network Confounding

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

$$\Delta_i(t) = (\hat{\rho}_t(\mathbf{i}) - \mu_t(\mathbf{i})) \mathbf{p}_t(\mathbf{i}) \mathbf{1}_i(t)$$

Leung and Loupos

## GNNs for Network Confounding

Bronstein

<https://towardsdatascience.com/do-we-need-deep-graph-neural-networks>  
be 2dec A

J Bruna, Courville, and Leventas

X n X 0

Corn Zou, Wang, and Bruna

A A

V

Cornu, D'Amico, and Durugbo

wang, Jones

ono ou n

Corso, Cavallari, and Baccarelli

A A

V

D Groot

ou n o A n A o

on

Dora, Ferrara, Garofalo, and Donovan

Berra

ou n o ono 5

Dworkin, Joshi, and Yezhov

X n X 00 00

Eisenstein, Pons, and Brann

U A

X n X 0 5

Farrugia

X n X 0

an an sra

ono

# Leung and Loupos

F an` J nss n,  
X n X 0 0 ,

Forast r E A rd` an` F a h` l  
A n A o on, , ou n o

Gro A i` A i`

H X an` on , o o  
no u o o n ,

Horn U A t n o an` H t` l l  
U A u o , ,

## GNNs for Network Confounding

Y  
I  
r  
an  
B  
A





# GNNs for Network Confounding

$\mathcal{G}$   $\text{ann}$   $\mathcal{I}$   $\mathcal{I}$   $\mathcal{I}$   $X$   $n$   
 $X$   $\theta$   $\theta$   $\theta$

# Leung and Loupos

Zop

X  $\theta$   $\theta$  5 ,

! A! A! Y ; X n