Almost-Matching-Exactly for Treatment Effect Estimation under Network Interference

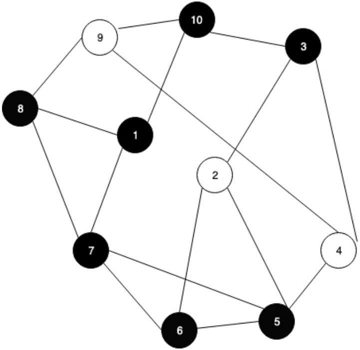
Usaid Awan, Marco Morucci, Vittorio Orlandi

Setting - Causal Inference

- We have $i = 1, \dots, n$ experimental units
- Treatment $t_i \in \{0, 1\}$ with $\mathbf{t} \in \{0, 1\}^n$ is a binary vector with the treatment level of every unit.
- Potential outcomes $Y_i(t_i, \mathbf{t})$ are random variables and depend on both treatment of unit i (1st argument), and treatment of **all other units** (2nd argument).
- Observed treatment $\mathbf{T} \in \{0, 1\}^n$ is assigned **uniformly at random**.
- Observed outcome: $Y_i = T_i Y(1, \mathbf{T}) + (1 - T_i) Y_i(0, \mathbf{T})$
  - Since treatment is randomized: $\mathbb{E}[Y_i | \mathbf{T} = \mathbf{t}, T_i = t] = \mathbb{E}[Y_i(t, \mathbf{T})]$ (Ignorability).
- Units are connected in a network, G, in which unit i's **neighborhood subgraph** is $G_{\mathcal{N}_i^{\mathbf{t}}}$.

The Problem: No SUTVA!

Usually we assume that *units' treatments don't influence other units' outcomes*, but we can't do that here because our units are connected in a network:

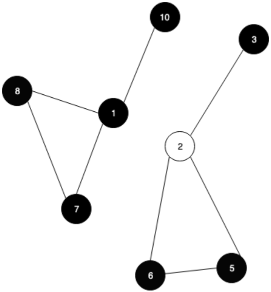


It could be that the treatment assigned to j influences the outcome of i through their connection in the network.

Similar Graphs Carry Similar Interference

Idea

What if the amount of interference experienced by a unit depended on the *shape* of its treated neighborhood subgraph?



Then, in expectation, two units with the same treated neighborhood graph will respond similarly to the treatment.

We can use this idea to do matching to reduce interference.

Assumptions

1. Outcome model: $Y_i = \alpha + t_i \tau_i + f(G_{\mathcal{N}_i^t}) + \epsilon_i$
    - f is some interference function dependent on $G_{\mathcal{N}_i^t}$, the **treated neighborhood subgraph** of unit i.
2. $\mathbb{E}[\epsilon_i] = 0$
    - Ignorability
3. $|\mathbb{E}[f(g)] - \mathbb{E}[f(h)]| \leq K_1 ||g - h||$
    - The more similar the neighborhood graphs of i and j, the more similar the amount of interference they receive.
    - Together with (1), this assumption encodes a version of SANASIA (Airoldi and Sussman, 2018) conditional on unit's neighborhood subgraphs.

Problem

How do we represent similarity between neighborhood graphs?

The Exponential Random Graph Model for Networks

A model in which networks are fully described by a set of statistics, like subgraph counts and nodal attributes.

- $\Pr(G_{\mathcal{N}_i^{\mathbf{t}}} = g) = \frac{exp(\boldsymbol{\phi}^T S(g))}{\sum_{h \in \mathcal{G}} exp \boldsymbol{\phi}^T S(h)}$
- Here $S(g)$ is a function that takes a network and outputs a vector of network statistics
- Two graphs with the same values for the statistics of interest have the same probability

Here we focus on **subgraph counts** as statistics, i.e., how many of each subgraph type does each unit have in its treated neighborhood graph.

Putting It All Together

If we make an additional assumption*, that the network statistics represent correctly each unit's neighborhood graphs, i.e.:

$$|\mathbb{E}[Y_i(t,\mathbf{t})|S(G_{\mathcal{N}_i^{\mathbf{t}}}) = \mathbf{a}] - \mathbb{E}[Y_j(t,\mathbf{t})|S(G_{\mathcal{N}_j^{\mathbf{t}}}) = \mathbf{b}]| \leq K_2||\mathbf{a}-\mathbf{b}||$$

Then if we match units with similar neighborhood graphs, we will reduce bias and eventually recover the correct treatment effect:

$$\mathbb{E}[Y_i(t,\mathbf{t})|S(G_{\mathcal{N}_i^{\mathbf{t}}}) = \mathbf{a}] = \mathbb{E}[Y_j(t,\mathbf{t})|S(G_{\mathcal{N}_j^{\mathbf{t}}}) = \mathbf{a}] \quad \forall i, j$$

* Unclear whether we have to make this assumption or whether it follows from ERGM.

Problem #2

How do we choose which subgraph we should use to represent the treated neighborhood graphs of our units?

Use FLAME to choose which subgraphs to count!

1. Count all possible subgraphs that are in each unit's treated neighborhood graph
2. These are likely a lot and it's unlikely that many units will have the exact same counts
3. Use FLAME to make almost-exact matches on subgraph counts

A Small Change to FLAME

We know from our theoretical setup that network statistics should do two things well:

1. Predict the **outcomes**
2. Predict the **network**

Therefore, it makes sense that the FLAME objective should trade-off between these things:

$$\mathsf{PE} = \sum_{t=0}^{1} \arg\min_{f \in \mathcal{F}} \frac{1}{n} \sum_{i=1}^{n} (Y_i - f(S(G_{\mathcal{N}_i^{\mathbf{t}}})))^2$$
$$- C \arg\max_{\phi \in \mathbb{R}^{|S|_0}} \underbrace{\frac{1}{n} \sum_{i=1}^{n} \phi^T S(G_{\mathcal{N}_i^{\mathbf{t}}}) - \log\left(\sum_{g \in \mathcal{G}} \phi^T S(g)\right)}_{\text{ERGM log-likelihood}}$$

(OLD) Simulation Results

Test 1: Simple Interference

Interference:

$$f_i(\mathbf{z}) = \gamma d_i(\mathbf{z})$$

Outcomes:

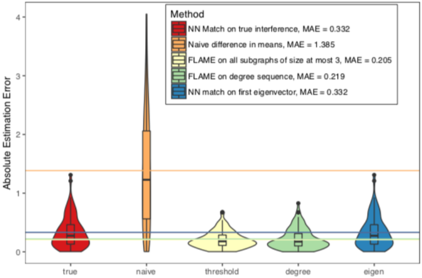
$$Y_i = \alpha_i + \beta_i z_i + \gamma d_i(\mathbf{z})$$

Parameters:

$$N = 100,\ Nsim = 100,\ \alpha_i \sim \mathcal{N}(0,1),\ \beta_i \sim \mathcal{N}(5,1),\ \gamma = 4,\ G \sim ER(0.05),\ Z_i \sim Ber(0.5)$$

3

Results:



(OLD) Simulation Results

Test 3a: Even More complex Interference (denser graph)

Interference:

$$f_i(\mathbf{z}) = d_i(\mathbf{z})(\delta\Delta_i(\mathbf{z}) + \lambda V_i(\mathbf{z}) + \eta\bigstar_i^4(\mathbf{z}))$$

Outcomes:

$$Y_i = \alpha_i + \beta_i z_i + f_i(\mathbf{z}).$$

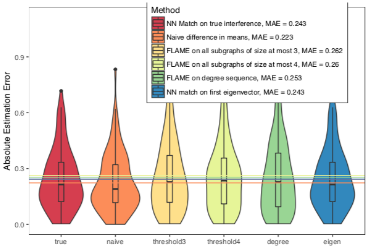
Parameters:

$$N = 100,\ Nsim = 100,\ \alpha_i \sim \mathcal{N}(0,1),\ \beta_i \sim \mathcal{N}(5,1),\ \delta = 2,\ \lambda = 0.5,\ \eta = 4; G \sim ER(0.1),\ Z_i \sim Ber(0.5)$$

.

Results:

6



(OLD) Simulation Results

Test 5b: Interference that is not related to subgraph counts (denser graph)

Interference:

$$f_i(\mathbf{z}) = \delta * Betweenness_i(\mathbf{z}) + \lambda * Closeness_i(\mathbf{z})$$

Outcomes:

$$Y_i = \alpha_i + \beta_i z_i + f_i(\mathbf{z}).$$

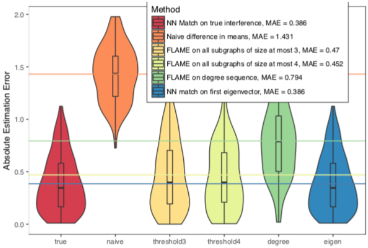
Parameters:

$$N = 100,\ Nsim = 100,\ \alpha_i \sim \mathcal{N}(0,1),\ \beta_i \sim \mathcal{N}(5,1),\ \delta = 40,\ \lambda = 20,\ G \sim ER(0.1),\ Z_i \sim Ber(0.5)$$

.

Results:

10



Bias Bound for oracle AME

As a preliminary result we can say that, under all the assumptions made before, with the true value of ϕ and S known, and if we choose a match for unit i with treated neighborhood graph g such that:

$$j \in \mathtt{MG}(\mathrm{g}) \text{ if } j \in \underset{j=1,\ldots,n,T_j=0}{\arg\min} |\phi^T S(g) - \phi^T S(G_{\mathcal{N}_j^{\mathrm{t}}})|,$$

then the bias for the CATT of i can be upper bounded by:

$$|\mathbb{E}[Y_i - Y_j] - \tau_i| \leq K_1 \sum_{h \in \mathcal{G}} |\phi^T S(g) - \phi^T S(h)| \frac{exp(\phi^T S(h))}{\sum_{\ell \in \mathcal{G}} exp(\phi^T S(\ell))} \times \left[\sum_{d=S(g)-|S(g)-S(h)|}^{S(g)+|S(g)-S(h)|} \frac{D_{\phi,S}(d) exp(d)}{\sum_{\ell \in \mathcal{G}} exp(\phi^T S(\ell))} \right]^{n-1}$$

Plan: We Want to Hit the October 8 AISTATS Deadline

1. We need to implement the revised `PE` function
2. We need to redo the simulations
3. More theory? Statements on how well the subgraph count can "encode" a given graph would be nice
4. Find a good application
5. Write!