Causal inference in social network data

Oleg Sofrygin, Mark J. van der Laan

(UC Berkeley)

June 30, 2016

Outline

- **I.** Background
- **II**. Network-dependent data as high-dimensional data problem
- **III**. Simulations & Estimation in **R** (`simcausal`/`tmlenet`)
- **IV.** Simulation study

Outline

1. Background
2. Network-dependent data as high-dimensional data problem
3. Estimation with `tmlenet` / Simulations with `simcausal`
4. Simulation Study

Background

- This talk is not about statistical modeling of network formation
- Data has been gathered on individual people that are known to be connected by a social network
- The field has been gaining interest:
  - New ways of gathering data (online social networks, mobile fitness censors)
- Want to known estimate an effect of some intervention among these people
- We hypothesize that network plays a role in the way the personal-level data was generated
  - The intervention might propagate amongst people
  - May induce dependence among units

Background

- Christakis and Fowler (2007, 2008, 2009, 2010, 2011, 2012) initiated a wave of interest
- Widely publicized results with significant peer effects for **obesity**, **smoking**, **alcohol consumption**, **sleep habits**, etc.
- Criticized for ignoring the dependent nature of the data and for making unrealistic modeling assumptions
- Two problems for traditional (independent data) inference:
  - CLT may not hold
  - Not taking into account for dependence may result in S.E.s that are too small (anti-conservative)

Main Goals

- Framework for estimation and inference in such data
- Software for simulation of synthetic population data under network dependence (`simcausal`)
- Software for estimation of effects in network-dependent data (`tmlenet`)
- Correct inference (a good estimate of the variance of our estimate)
- Side note: the issues discussed here are applicable to non-network (independent) high dimensional data

Outline

1. Background
2. Network-dependent data as high-dimensional data problem
3. Estimation with tmlenet / Simulations with simcausal
4. Simulation Study

Classical causal framework with IID setting

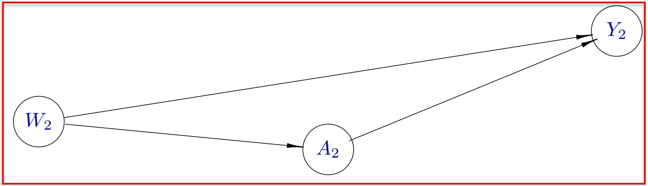
- Suppose we have N individuals (units) enrolled in a study
- $O_i = (W_i, A_i, Y_i)$ denotes the data collected on each unit, for $i = 1, \ldots, N$:
  - W_i - are the baseline covariates
  - A_i - exposure (0/1)
  - Y_i - outcome
- Want to estimate the ATE:

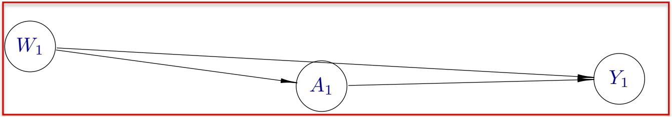
$$E_W \left[E(Y_i | A_i = 1, W_i) - E(Y_i | A_i = 0, W_i) \right]$$

- This parameter is a function of the true distribution of the data, P_0
- It has causal interpretation under additional assumptions
- It is **interpretable** even when we don't believe in these assumptions!
- We can use state-of-the-art machine learning without ever loosing this interpretability

Two independent units with DAGs

- Consider a typical causal DAG for two i.i.d. observations (**1** & **2**) with treatment A, baseline covariates W and outcome Y:

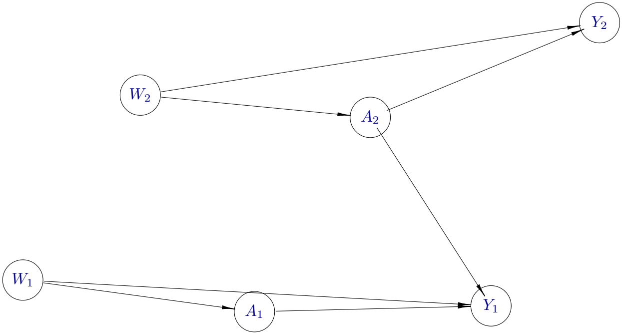




- Now these two units are also "connected" by a network (set of friends F_1 and F_2 that was also measured)

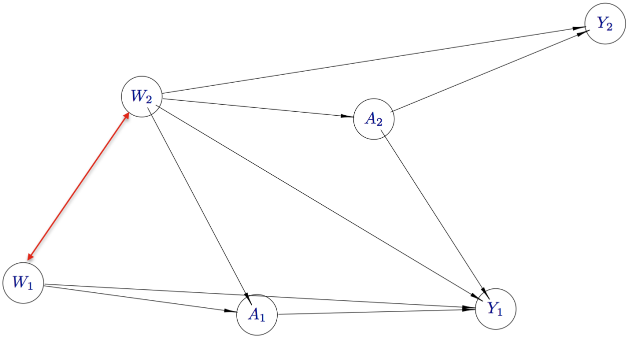
What we mean by a "network"?

- Suppose unit 1 lists unit 2 is her "friend" (but not vice versa),
  - Allow **spillover**: Y_1 depends on the treatment assignment of unit 2, A_2.



What we mean by a "network"?

- Y_1 depends on W_2 (baseline covariates of unit 1); and
- May allow W_1 and W_2 to be dependent (correlated) if units 1 and 2 are friends
- We may also assume A_1 depends on W_2 (in addition to W_1)



Analogue to ATE in a network setting

- The ATE in IID data:

$$E_W\left[E(Y_i|A_i = 1, W_i) - E(Y_i|A_i = 0, W_i)\right]$$

- Network:
  - Want to know the effect of setting A_j for $j \in F_i$ on Y_i
  - All W_j, for $j \in F_i$ are all confounders - need to adjust for them
- The ATE analogue in "networked" data:

$$\begin{aligned} & \frac{1}{N}\sum_i E(Y_i|A_i = 1, \mathbf{A}_{F_i} = \mathbf{1}^{|F_i|}, W_i, W_j : j \in F_i) \\ - \quad & \frac{1}{N}\sum_i E(Y_i|A_i = 0, \mathbf{A}_{F_i} = \mathbf{0}^{|F_i|}, W_i, W_j : j \in F_i) \end{aligned}$$

Network curse of dimensionality

- Suppose that i has 100 friends ($|F_i| = 100$)
- Have to adjust for W_i plus additional 100 ($W_j : j \in F_i$)
- Have to fit a model for the effect of A_i on Y_i plus the effect of 100 additional exposures ($A_j : j \in F_i$) on Y_i
- To have any hope of fitting the outcome model we have to assume some common model for N observations
  - But i and j can have different number of friends! How can we even have a common model?
- Ways around it:
  - Assume same number of friends for everybody
  - Assume very small number of friends (a most 2) - only household members
  - Clearly this is not a good representation of real data

Network curse of dimensionality: network summaries

- Assume that my outcome (Y_i) depends only on some functions (***network summaries***):

$$W_i^s := w_i^s(\mathbf{W}_{F_i}, W_i) \text{ and } A_i^s := a_i^s(\mathbf{A}_{F_i}, A_i)$$

- They have the same and **fixed** dimension for all i and are otherwise arbitrary
- Assume:
  1. Conditional probability $P(A_i \mid \cdot)$ is only a function of summary $w_i^s(\mathbf{W}_{F_i}, W_i)$
  2. Conditional density $P(Y_i \mid \cdot)$ is only a function of $w_i^s(\mathbf{W}_{F_i} W_i)$ and $a_i^s(\mathbf{A}_{F_i}, A_i)$
- Simplifies the notation:
  - Data on N units can be represented:

$$O_i^s = (W_i^s, A_i^s, Y_i), \text{ for } i = 1, \ldots, N$$

  - Our estimand (ATE):

$$\frac{1}{N}\sum_{i=1}^{N}[E(Y_i|A_i^s = a_i^s(\mathbf{o}), W_i^s) - E(Y_i|A_i^s = a_i^s(\mathbf{1}), W_i^s)]$$

Outline

1 Background

2 Network-dependent data as high-dimensional data problem

3 Estimation with `tmlenet` / Simulations with `simcausal`

4 Simulation Study

Syntax for network summaries in **R** (`tmlenet` and `simcausal`)

- Define network baseline summaries / features W^s with function **def_sW**:

```
def_sW(netW1W2 = sum(W1[[1:Kmax]]*W2[[1:Kmax]]))
```

- Define network exposure summaries / features A^s with function **def_sA**:

```
def_sA(A, sum.net.A = (sum(A[[1:Kmax]])))
```

`tmlenet`

- Implements 3 estimators for network data
- **IPW**: Inverse Probability Weighted Estimator
  - Re-weights the outcomes Y_i by the inverse probability of receiving the network exposure summary (the effective exposure)
- **GCOMP**: G-Computation Estimator
  - Directly fit the outcome model: $(E(Y_i|A_i^s, W_i^s)$
- **TMLE**: Targeted Maximum Likelihood Estimator
  - Combines IPW and GCOMP into a single estimator to take advantage of both
  - Involves only a single additional modeling step (at low computational cost)
  - Recovers the CLT for the estimator (allows ML)
  - Provides asymptotically valid confidence intervals
- `tmlenet` will work with independent data just as well (no network)
- For network data, `tmlenet` implements two approaches for estimating variance that adjusts for dependence

`tmlenet`

- Defining "effective" exposure A_i^s created another problem:
  - Even when $A_i \in \{0, 1\}$, the summary A_i^s is likely to be continuous
- The "effective" exposure model is now a **multivariate conditional** density rather than a binary classification problem: $p_{A_i^s|W_i^s}(a^s|w^s)$
- `tmlenet` implements conditional histogram density estimator for $p_{A_i^s|W_i^s}$
  - Discretize range of A_i^s by splitting it into intervals (bins)
  - Fit a separate binary classification/regression for each bin as a function of the baseline summaries W_i^s
  - Automatically detects the type of the exposure summary and then decides how to fit it
- `tmlenet` allows for stochastic interventions, among others:
  - **Stochastic Intervention**: covered a random 40% of the community?
  - **Targeted Intervention**: covered only the top 10% most connected community members?
  - **Network intervention**: remove or add a new friend?

Network simulation with `simcausal` - example

- `simcausal`:
  - Simulates synthetic datasets to test statistical methods applied in causal inference
  - Time-varying (longitudinal data) and network-dependent data
  - Single pipeline for conducting a "typical" simulation study in causal inference
  - Supports arbitrary univariate and multivariate (conditional) distributions

```
node("Y", distr = "rbern", prob = plogis(0.5*W - 0.35*A - 0.5*sum(A[[1:Kmax]])))
```

- Above defined $P(Y_i = 1|\cdot)$ as logit-linear function of:
  - Baseline covariate (**W**), exposure (**A**), and
  - Sum of friends' exposures (**sum(A[[1:Kmax]])**)
- Note: **Kmax** is a special constant - maximum number of friends and is evaluated automatically by simcausal

Estimation with *tmlenet* - example

- Define baseline summaries / features W^s with function **def_sW**:

```
sW <-  def_sW(W1, W2) +
       def_sW(netW1W2 = sum(W1[[1:Kmax]]*W2[[1:Kmax]]),
              nF.PA = sum(PA[[1:Kmax]]),
              nFPAeq0.PAeq1 = (nF.PA < 1) * (PA == 1),
              replaceNAw0 = TRUE)
```

- Define exposure summaries / features A^s with function **def_sA**:

```
sA <- def_sA(A, A.PAeq0 = A * (PA == 0)) +
     def_sA(sum.net.A = (sum(A[[1:Kmax]])*(HUB==0) +
                         sum((W1[[1:Kmax]] > 4)*A[[1:Kmax]])*(HUB==1)),
                replaceNAw0 = TRUE)
```

Estimation with *tmlenet* - example

- Define interventions with function **def_new_sA**:

```
intervene_1 <-  def_new_sA(A = 0)
intervene_2 <-  def_new_sA(A = 1 - A)
intervene_stoch <-  def_new_sA(A = rbinom(n = length(A), size = 1, prob = 0.35))
intervene_dyn <-  def_new_sA(A = rbinom(n = length(A), size = 1,
                                         prob = ifelse(nF >= 20, 0.9, 0.1)))
```

Estimation with *tmlenet* - example

- Function **tmlenet** performs estimation (also requires the network matrix and the input data):

```
# REGRESSION FORMULAS
Qform <- "Y ~ nF.PA + A.PAeq0 + nFPAeq0.PAeq1 + sum.net.A + PA + W1 + W2"
hform.g0 <- "A + sum.net.A ~ HUB + PA + nF.PA + nFPAeq0.PAeq1"
# EFFECT ESTIMATION
res <- tmlenet(data = sim_dat, sW = sW, sA = sA,
              NETIDmat = NetInd_mat,
              Kmax = ncol(NetInd_mat),
              intervene1.sA = intervene_stoch,
              Qform = Qform,
              hform.g0 = hform.g0,
              hform.gstar = hform.g0,
              optPars = list(
                bootstrap.var = FALSE)
              )
```

Outline

1 Background

2 Network-dependent data as high-dimensional data problem

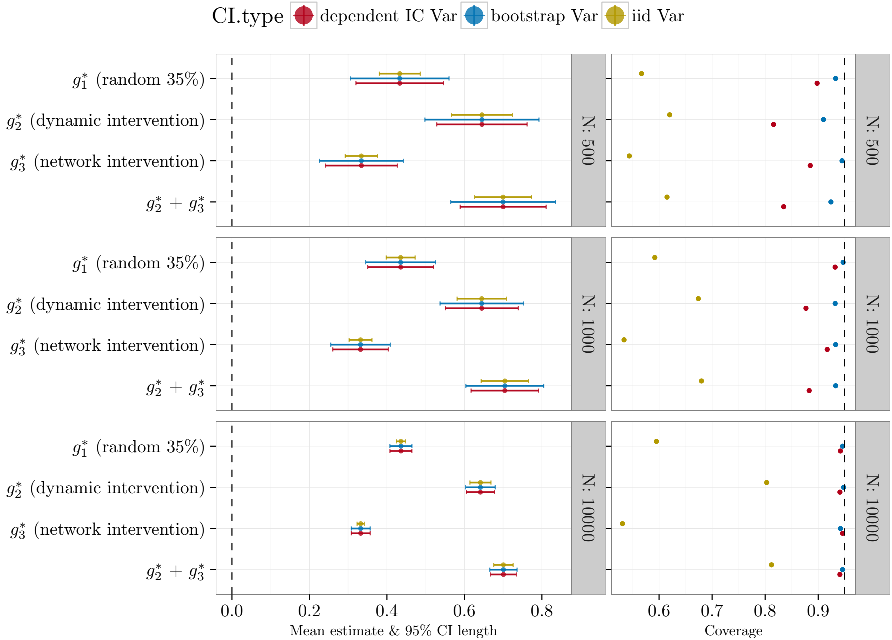
3 Estimation with `tmlenet` / Simulations with `simcausal`

4 Simulation Study

Simulation - Peer Effects of Exercise

- Simulated a small world network
- Study designed to increase the levels of physical activity in a highly-connected community
- Individuals randomly received vouchers to attend a local gym
- Outcome is a binary indicator of maintaining gym membership
- Estimated the effects of:
  - Assigning exposure to random 35%
  - Targeted exposure assignment to top 10% most connected units
  - Effect of combining the exposure with network interventions (additional physically active friend for each units with <10 friends)

Simulation Results - Small World Network



Concluding remarks

- **tmlenet** solves some estimation challenges in network-dependent data
- Allows continuous exposures & arbitrary stochastic interventions
- Flexible interface for defining arbitrary summaries/features of network covariates
- Two ways of doing inference while adjusting for dependence
- Ongoing work, new features are being added (e.g., networks over multiple time-points)
- See **simcausal** vignette on CRAN and JSS paper to appear `https://cran.r-project.org/web/packages/simcausal`
- Github:
  - **simcausal**: `https://github.com/osofr/simcausal`
  - **tmlenet:** `https://github.com/osofr/tmlenet`
  - **stremr** (most recent expansion of **tmlenet** code into longitudinal IID data, estimation with h2o ML libraries): `https://github.com/osofr/stremr`

REFERENCES

1. Sofrygin, O and van der Laan, M J, "Semi-Parametric Estimation and Inference for the Mean Outcome of the Single Time-Point Intervention in a Causally Connected Population" (December 2015). *U.C. Berkeley Division of Biostatistics Working Paper Series.* Working Paper 344.
2. Sofrygin, O. and van der Laan, M. J. (2015). simcausal R Package: Conducting Transparent and Reproducible Simulation Studies of Causal Effect Estimation with Complex Longitudinal Data. *Submitted to J of Stat Soft.*
3. Sofrygin, O. and van der Laan, M. J. (2015). tmlenet: Targeted Maximum Likelihood Estimation for Network Data. R package version 0.1.0.
4. van der Laan, M. J. (2014). Causal Inference for a Population of Causally Connected Units. *Journal of Causal Inference*, 2(1):1–62.

- **FUNDING ACKNOWLEDGEMENT**:
  - This work was partially supported through an NIH grant (R01 AI074345-07) and a Patient-Centered Outcomes Research Institute (PCORI) Award (ME-1403-12506).