Causal Inference in The Age of Big Data: Observations and a Linearithmic Algorithm for Blocking/Matching/Clustering

Jasjeet S. Sekhon

ISAT/DARP
What If? Machine Learning for Causal Inference

What's the Big Deal about Big Data?

- One view: We just have to handle the data
  - Build a bigger computer system
  - It is a database problem
- Another view:
  - we need an integration between inferential and algorithmic thinking
- Measuring human activity has generated massive datasets with granular information that can be used for personalization of treatments, creating markets, modeling behavior
- Many inferential issues: e.g., unknown sampling frames, heterogeneity, targeting optimal treatments, compound loss functions

Massive Experiments

- Rising interest in fine-grained inference: e.g., subgroups
- Some traditional experimental design methods have become computationally infeasible—e.g., blocking
- Blocking: create strata and then randomize within strata
- Polynomial time solution not quick enough. Linearithmic is survivable. Sublinear needed in some cases.
- Algorithm can also be used for matching and clustering

A New Blocking Method

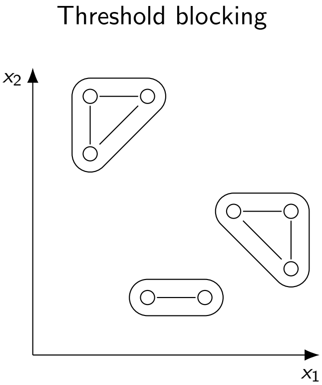
The method minimizes the pair-wise Maximum Within-Block Distance: λ

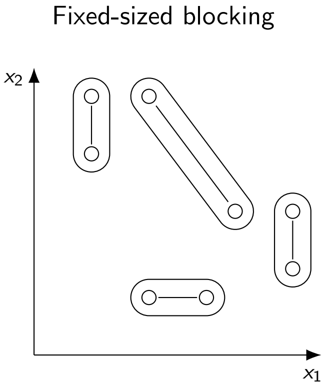
- Any valid distance metric (must satisfy the triangle inequality)
- Ensures good covariate balance by design
- Works for any number of treatments and any minimum number of observations per block
- It is fast: $O(n \log n)$ expected time
- It is memory efficient: $O(n)$ storage
- Approximately optimal: $\leq 4 \times \lambda$
- Special cases
  1. with one covariate: λ
  2. with two covariates: $\leq 2 \times \lambda$

Some Current blocking approaches

- Optimal Multivariate Matching Before Randomization [Greevy, Lu, Silber, and Rosenbaum, 2004]
  - No efficient way to extend approach to more than two treatment categories
  - Even for two treatment categories, doesn't scale well
- Matched-pairs blocking: Pair "most-similar" units together. For each pair, randomly assign one unit to treatment, one to control
  - Natural clustering in the data ignored
  - Cannot estimate conditional variances [Imbens, 2011]
  - Difficulty with treatment effect heterogeneity

Threshold blocking: relaxing the block structure





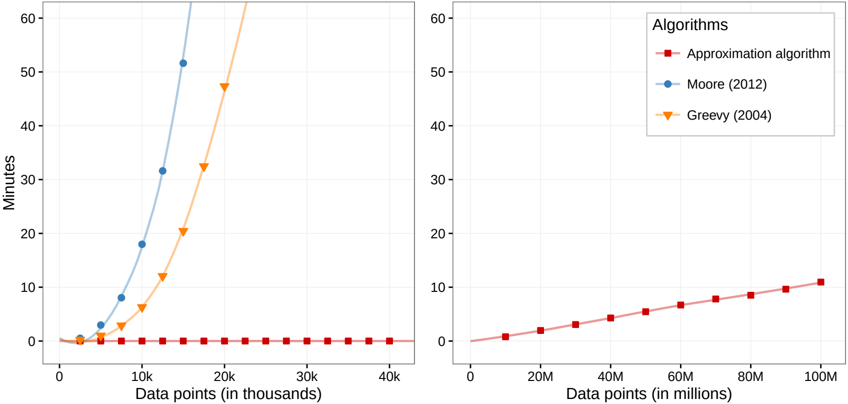
An Advantage

Theorem

For all samples, all objective functions and all desired block sizes, the optimal threshold blocking is always weakly better than the optimal fixed-sized blocking.

- Proof: interpret blocking as an non-linear integer programming problem.
  - The search set of threshold blocking is a superset of fixed-sized blocking

The AppOpt algorithm



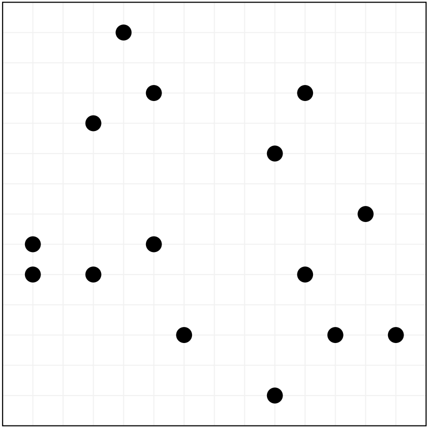
The AppOpt algorithm

Input:

- Units' covariates
- Distance metric
- Minimum block size: $k = 2$

Procedure:

1. A undirected complete graph with distances as edge weights
2. Find $(k-1)$-nearest neighbor graph
3. Construct the second power of NNG
4. Find a maximal independent set (seeds)
5. Form blocks with the seeds and their neighbors in NNG
6. Assign remaining units to a block containing any neighbor



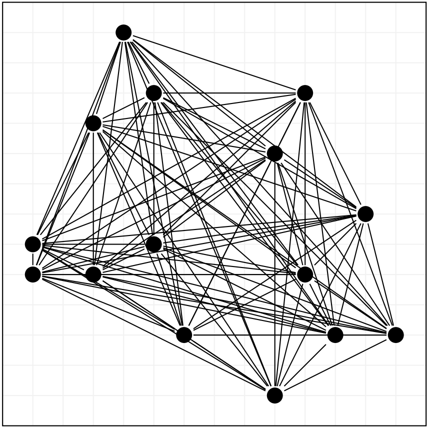
The AppOpt algorithm

Input:

- Units' covariates
- Distance metric
- Minimum block size: $k = 2$

Procedure:

1. A undirected complete graph with distances as edge weights
2. Find $(k-1)$-nearest neighbor graph
3. Construct the second power of NNG
4. Find a maximal independent set (seeds)
5. Form blocks with the seeds and their neighbors in NNG
6. Assign remaining units to a block containing any neighbor



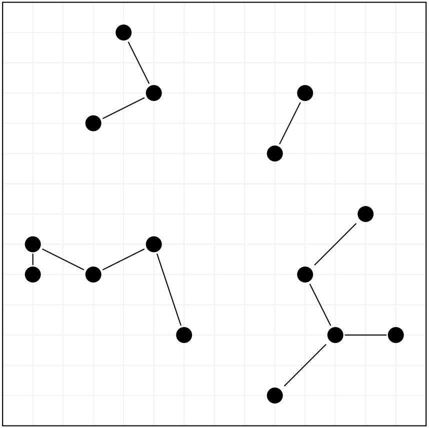
The AppOpt algorithm

Input:

- Units' covariates
- Distance metric
- Minimum block size: $k = 2$

Procedure:

1. A undirected complete graph with distances as edge weights
2. Find $(k-1)$-nearest neighbor graph
3. Construct the second power of NNG
4. Find a maximal independent set (seeds)
5. Form blocks with the seeds and their neighbors in NNG
6. Assign remaining units to a block containing any neighbor



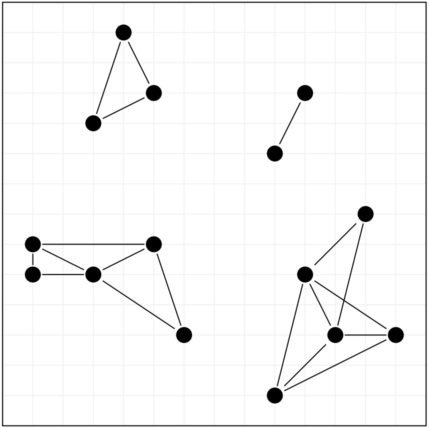
The AppOpt algorithm

Input:

- Units' covariates
- Distance metric
- Minimum block size: $k = 2$

Procedure:

1. A undirected complete graph with distances as edge weights
2. Find $(k-1)$-nearest neighbor graph
3. Construct the second power of NNG
4. Find a maximal independent set (seeds)
5. Form blocks with the seeds and their neighbors in NNG
6. Assign remaining units to a block containing any neighbor



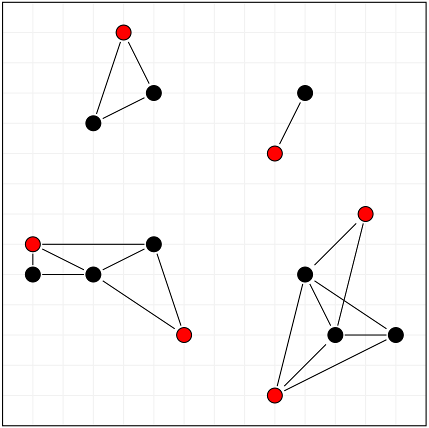
The AppOpt algorithm

Input:

- Units' covariates
- Distance metric
- Minimum block size: $k = 2$

Procedure:

1. A undirected complete graph with distances as edge weights
2. Find $(k-1)$-nearest neighbor graph
3. Construct the second power of NNG
4. Find a maximal independent set (seeds)
5. Form blocks with the seeds and their neighbors in NNG
6. Assign remaining units to a block containing any neighbor



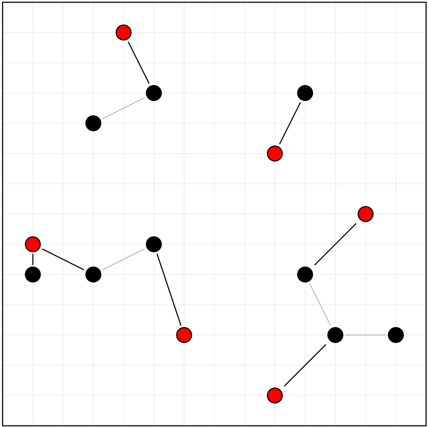
The AppOpt algorithm

Input:

- Units' covariates
- Distance metric
- Minimum block size: $k = 2$

Procedure:

1. A undirected complete graph with distances as edge weights
2. Find $(k-1)$-nearest neighbor graph
3. Construct the second power of NNG
4. Find a maximal independent set (seeds)
5. Form blocks with the seeds and their neighbors in NNG
6. Assign remaining units to a block containing any neighbor



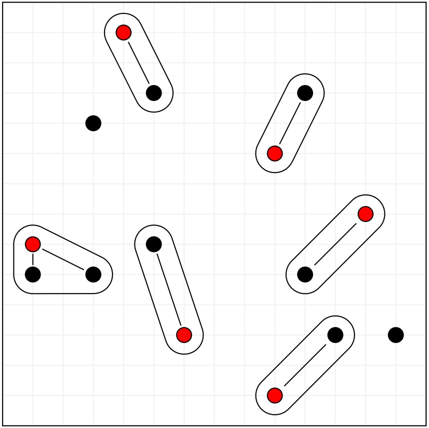
The AppOpt algorithm

Input:

- Units' covariates
- Distance metric
- Minimum block size: $k = 2$

Procedure:

1. A undirected complete graph with distances as edge weights
2. Find $(k-1)$-nearest neighbor graph
3. Construct the second power of NNG
4. Find a maximal independent set (seeds)
5. Form blocks with the seeds and their neighbors in NNG
6. Assign remaining units to a block containing any neighbor



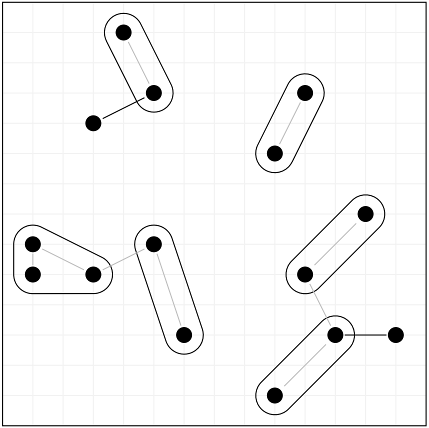
The AppOpt algorithm

Input:

- Units' covariates
- Distance metric
- Minimum block size: $k = 2$

Procedure:

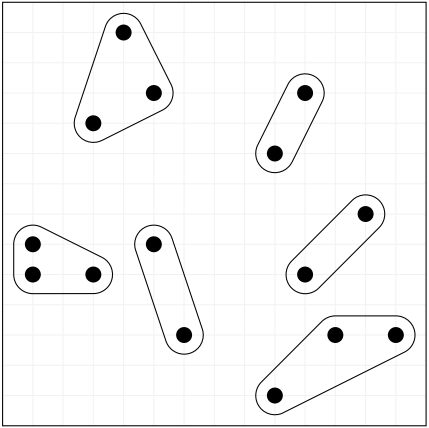
1. A undirected complete graph with distances as edge weights
2. Find $(k-1)$-nearest neighbor graph
3. Construct the second power of NNG
4. Find a maximal independent set (seeds)
5. Form blocks with the seeds and their neighbors in NNG
6. Assign remaining units to a block containing any neighbor



The AppOpt algorithm

Input:

- Units' covariates
- Distance metric
- Minimum block size: $k = 2$



Procedure:

1. A undirected complete graph with distances as edge weights
2. Find $(k-1)$-nearest neighbor graph
3. Construct the second power of NNG
4. Find a maximal independent set (seeds)
5. Form blocks with the seeds and their neighbors in NNG
6. Assign remaining units to a block containing any neighbor

Conclusion

- Closer to clustering than traditional blocking/matching methods
- Fast algorithm:
  - NNG plus $O(d^0 kn)$ time and $O(d^0 kn)$ space
  - K-d trees NN: $O(2^d kn \log n)$ expected time, $O(2^d kn^2)$ worst time, and $O(kn)$ storage
  - Compare with bipartite, network flow methods:
    - e.g., Derigs: $O(n^3 \log n + dn^2)$ worst time and $O(d^0 n^2)$ space

Joint Work with Michael J. Higgins and Fredrick Sävje

But there are problems

- Problem 1: the theorem is for the objective function used to construct the blocks.
  - Might not be the quantity of true interest.
- Problem 2: No help to us if we cannot find the optimum. NP-hard problems

Table: # unique blockings (block size = 2)

# units	Fixed-sized	Threshold
8	105	715
10	945	17,722
12	10,395	580,317
14	135,135	24,011,157
16	2,027,025	1,216,070,380
18	34,459,425	73,600,798,037
20	654,729,075	5.2×10^{12}

Bibliography I

David A. Freedman. On regression adjustments in experiments with several treatments. The annals of applied statistics, 2(1):176–196, 2008.

Robert Greevy, Bo Lu, Jeffrey H. Silber, and Paul Rosenbaum. Optimal multivariate matching before randomization. Biostatistics, 5(4):263—-275, 2004.

Guido W. Imbens. Experimental design for unit and cluster randomized trials. Working Paper, 2011.

Winston Lin. Agnostic notes on regression adjustments to experimental data: Reexamining freedman's critique. Annals of Applied Statistics, 2012.

Luke W. Miratrix, Jasjeet S. Sekhon, and Bin Yu. Adjusting treatment effect estimates by post-stratification in randomized experiments. Journal of the Royal Statistical Society, Series B, 75(2):369–396, 2013.

Ryan T Moore. Multivariate continuous blocking to improve political science experiments. Political Analysis, 20(4):460–479, 2012.

Kari Lock Morgan and Donald B Rubin. Rerandomization to improve covariate balance in experiments. Annals of Statistics, 40(2):1263–1282, 2012.

Bibliography II

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

Jerzy Splawa-Neyman, DM Dabrowska, and TP Speed. On the application of probability theory to agricultural experiments. essay on principles. section 9. Statistical Science, 5(4):465–472, 1990.