Improving Massive Experiments with Threshold Blocking

Jasjeet S. Sekhon

Drawing Causal Inference from Big Data
Sackler Colloquia, March 26, 2015

Massive Experiments

- Rising interest in fine-grained inference: e.g., subgroups, heterogeneous effects
- Some traditional experimental design methods have become computationally infeasible
- Researcher's degrees of freedom has increased
- Big rise in false positive rate

Why We Randomize?

- Unbiased estimator by design
- Make probability statements; "reasoned basis for inference" (Fisher, Peirce)
- Separate design from analysis (Cochran, Rubin)

A New Blocking Method

A new blocking method with nice theoretical properties

- Blocking: create strata and then randomize within strata
- Some analytical benefits for blocking, but the main one is transparency and minimizing fishing

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: $\leq 2 \times \lambda$

Covariate imbalance in randomized experiments

- PROBLEM: In finite samples, there is a probability of bad covariate balance between treatment groups
- Bad imbalance on important covariates:
  - $\rightarrow$ Imprecise estimates of treatment effects
  - $\rightarrow$ Conditional bias
- In large samples problems remain: we want to estimate treatment effects for subgroups

Some theoretical results about blocking

- Blocking cannot hurt the precision of the estimator:
  - if no worse than random matching
  - if sample from an infinite super population
- Blocking may increase the estimated variance. But this is specific to the estimator used (degrees of freedom). e.g., randomization inference solves the problem.

Adjustment and covariate imbalance

- Regression adjustment [Freedman, 2008, Lin, 2012]
- Post-stratification [Miratrix, Sekhon, and Yu, 2013]:
  - Group similar units together after *after* randomization
  - SATE/PATE results good; *ex post* problems arise
  - Data mining concerns
- Re-randomization [Morgan and Rubin, 2012]:
  - Repeat randomly assigning treatments until covariate balance is "acceptable"
- LESSON: design the randomization to build in adjustment

Some Current blocking approaches

- Optimal Multivariate Matching Before Randomization [Greevy, Lu, Silber, and Rosenbaum, 2004]
- Matched-pairs blocking: Pair "most-similar" units together. For each pair, randomly assign one unit to treatment, one to control
- Optimal-greedy blocking [e.g. Moore, 2012]
- Some methods make principled probability statements impossible

Matched-Pairs

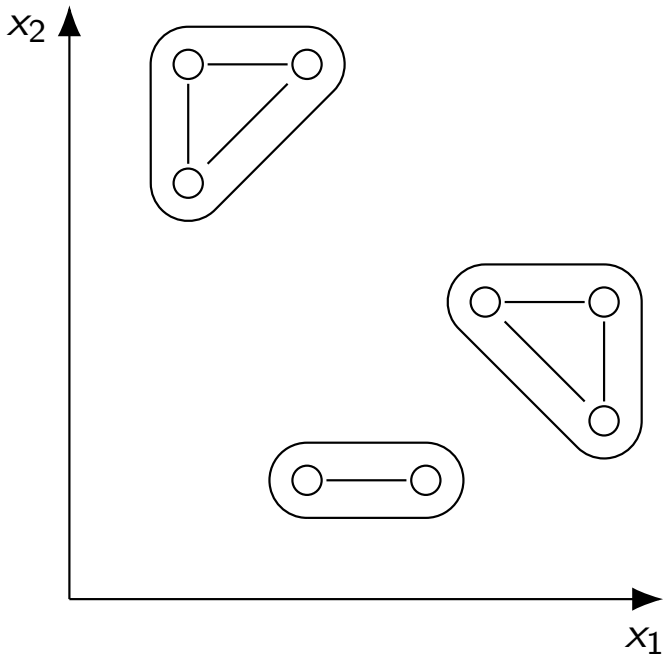
- No efficient way to extend approach to more than two treatment categories
- Fixed block sizes (2 units): design may pair units from different clusters
- Cannot estimate conditional variances [Imbens, 2011]
- Difficulty with treatment effect heterogeneity

Blocking by minimizing the Maximum Within-Block Distance (MWBD)

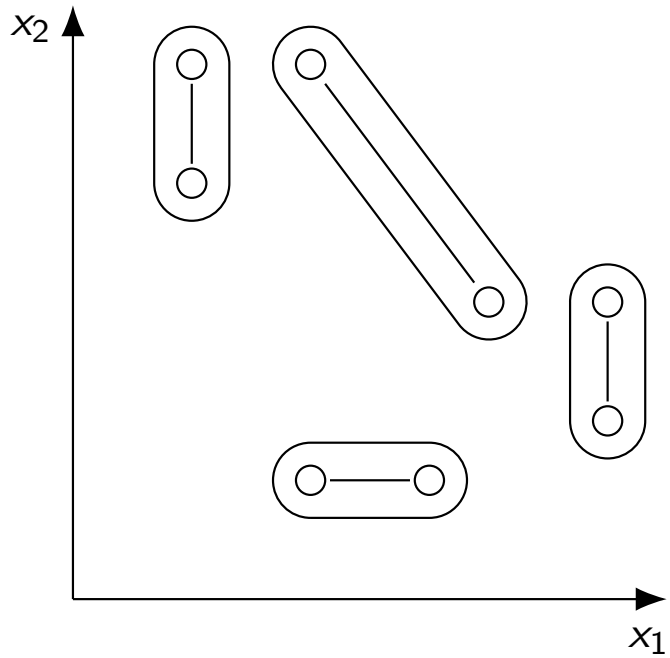
- Experiment with n units and t treatment categories
- Select a threshold $k \geq t$ for a minimum number of units to be contained in a block
- Block units so that each block contains at least k units, and so that the maximum distance between any two units within a block—the MWBD—is minimized
- Threshold k: Allows designs with multiple treatment categories, multiple replications of treatments within a block

Threshold blocking: relaxing the block structure

Threshold blocking



Fixed-sized blocking



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.

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}

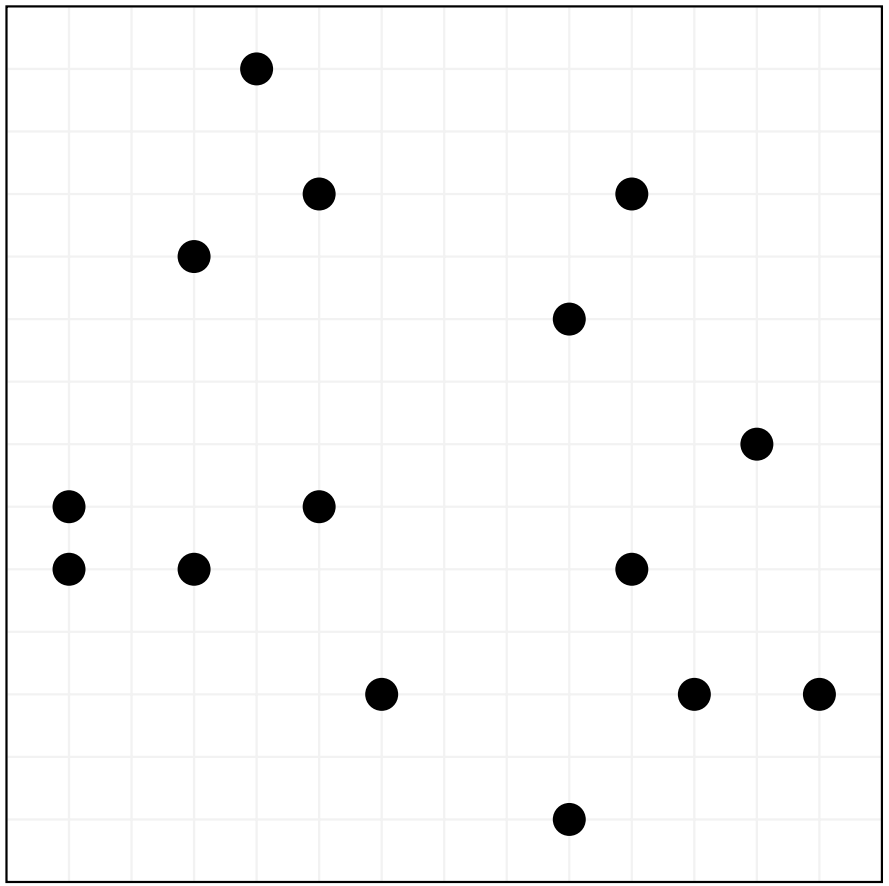
The AppOpt algorithm

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



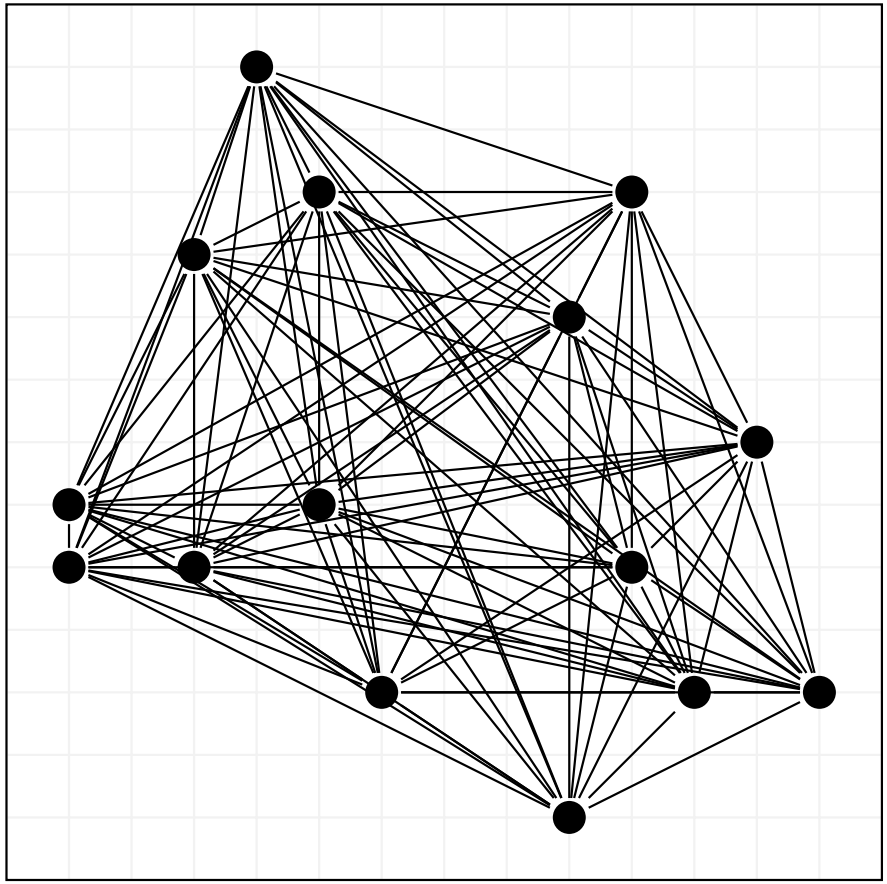
The AppOpt algorithm

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



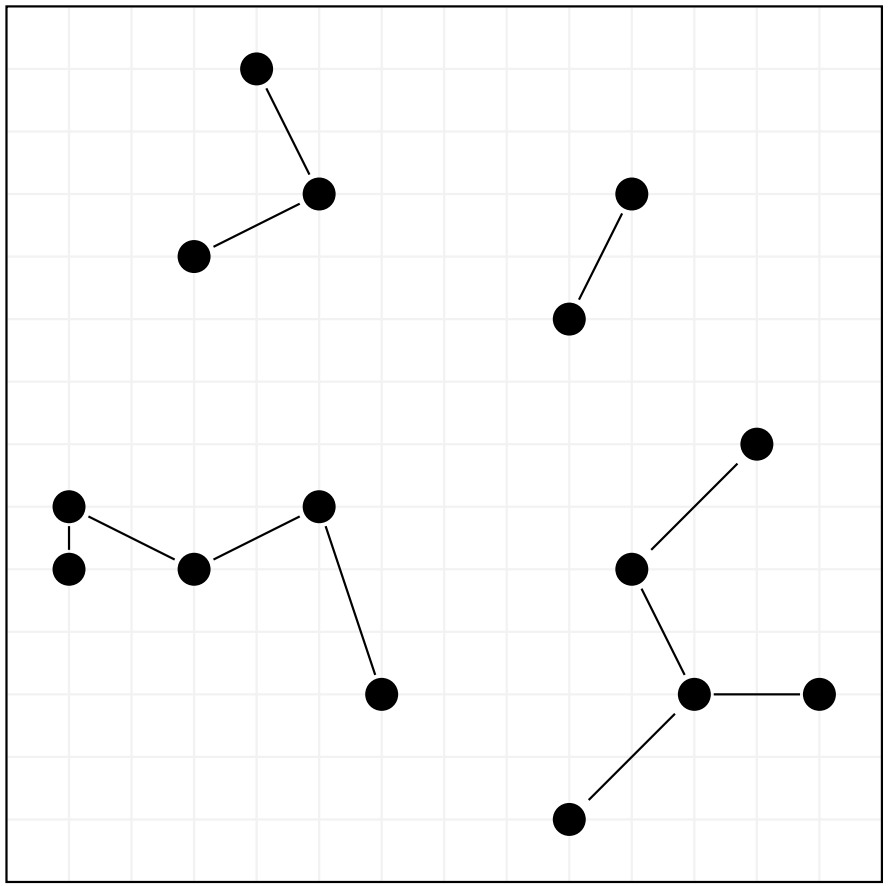
The AppOpt algorithm

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



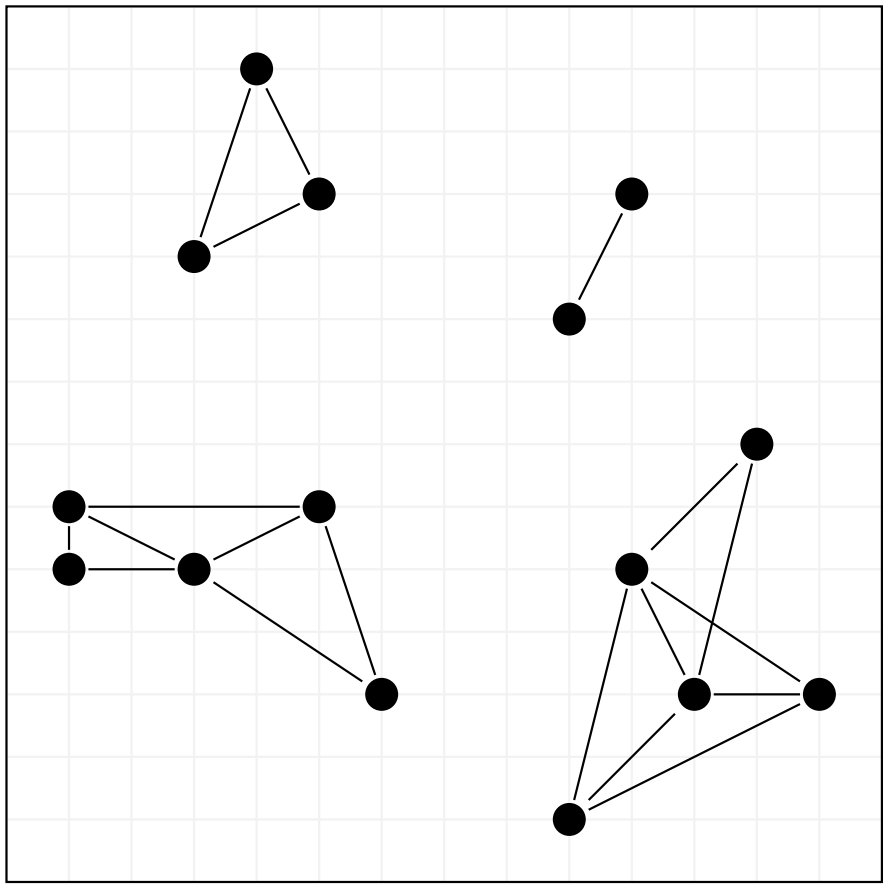
The AppOpt algorithm

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



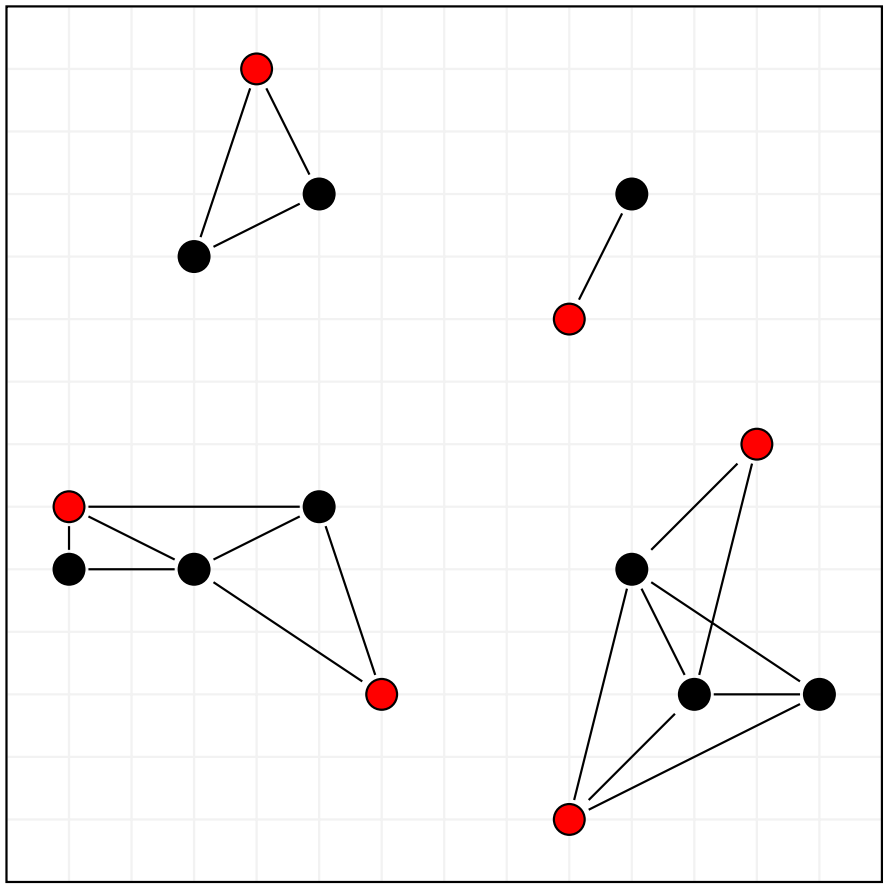
The AppOpt algorithm

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



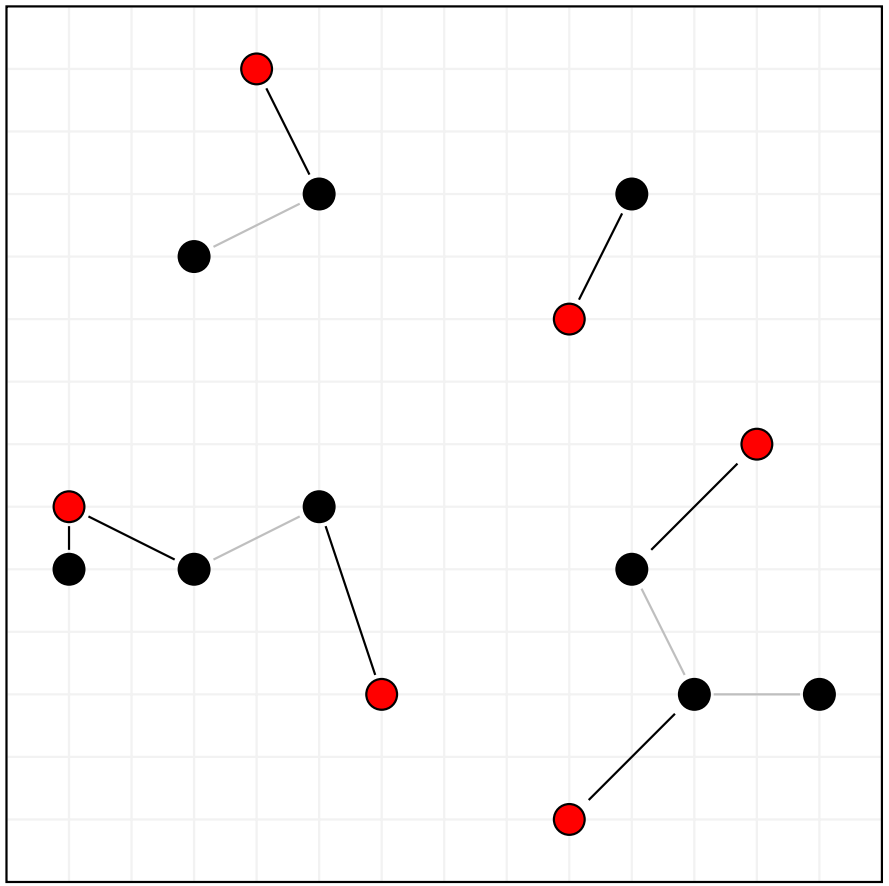
The AppOpt algorithm

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



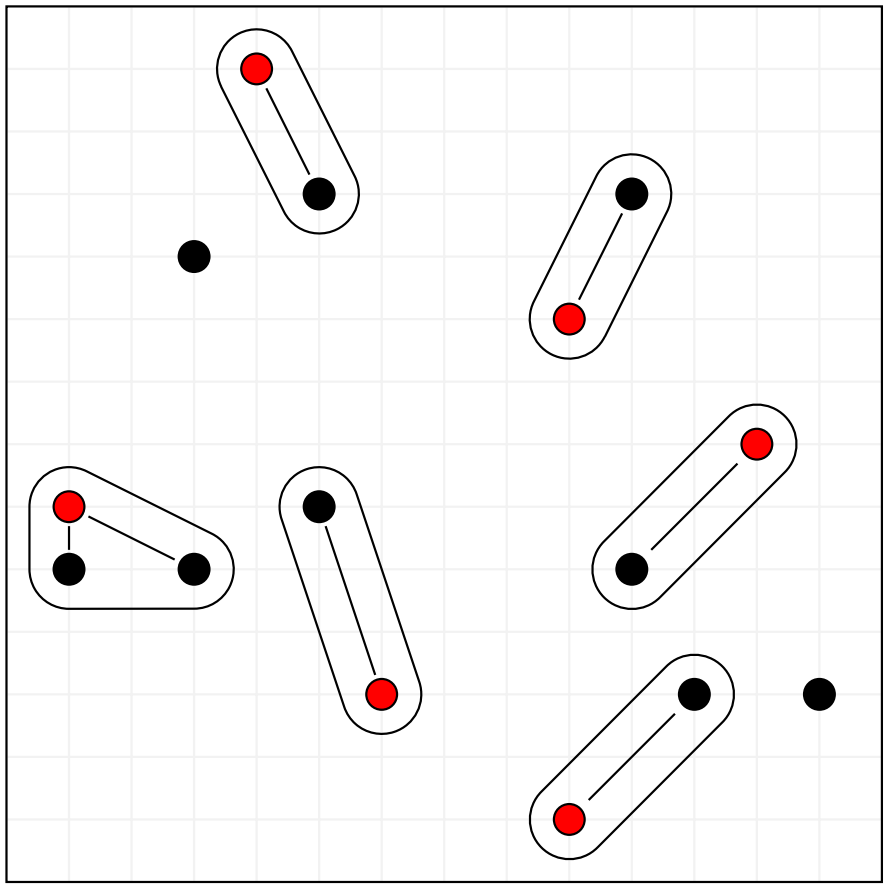
The AppOpt algorithm

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



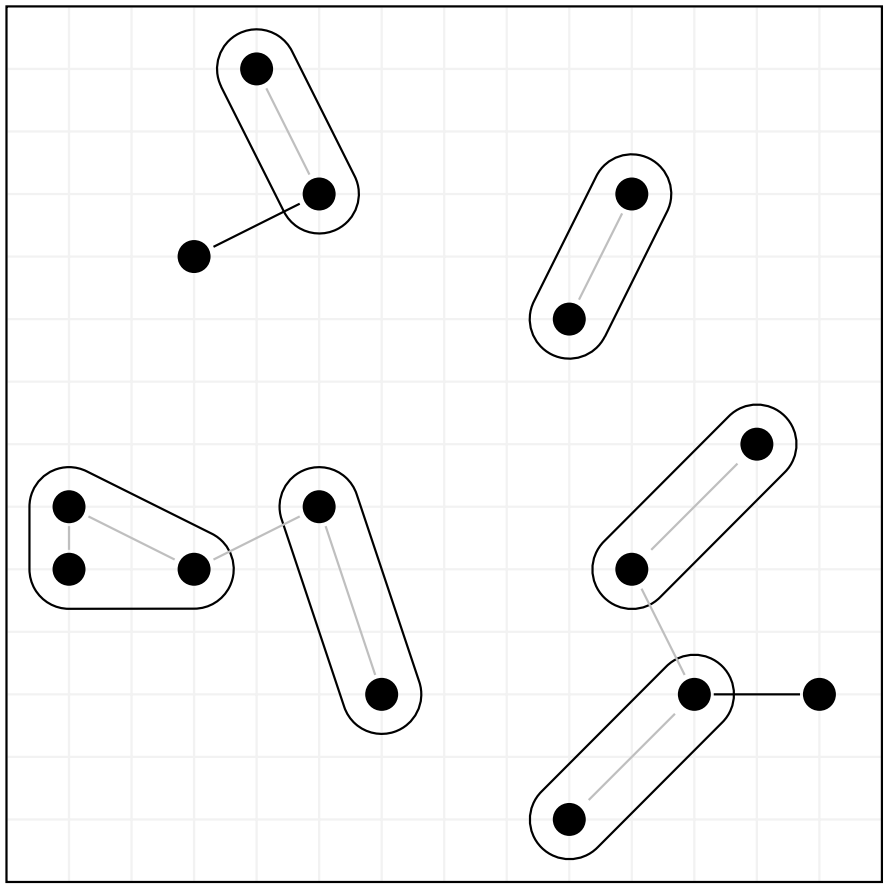
The AppOpt algorithm

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



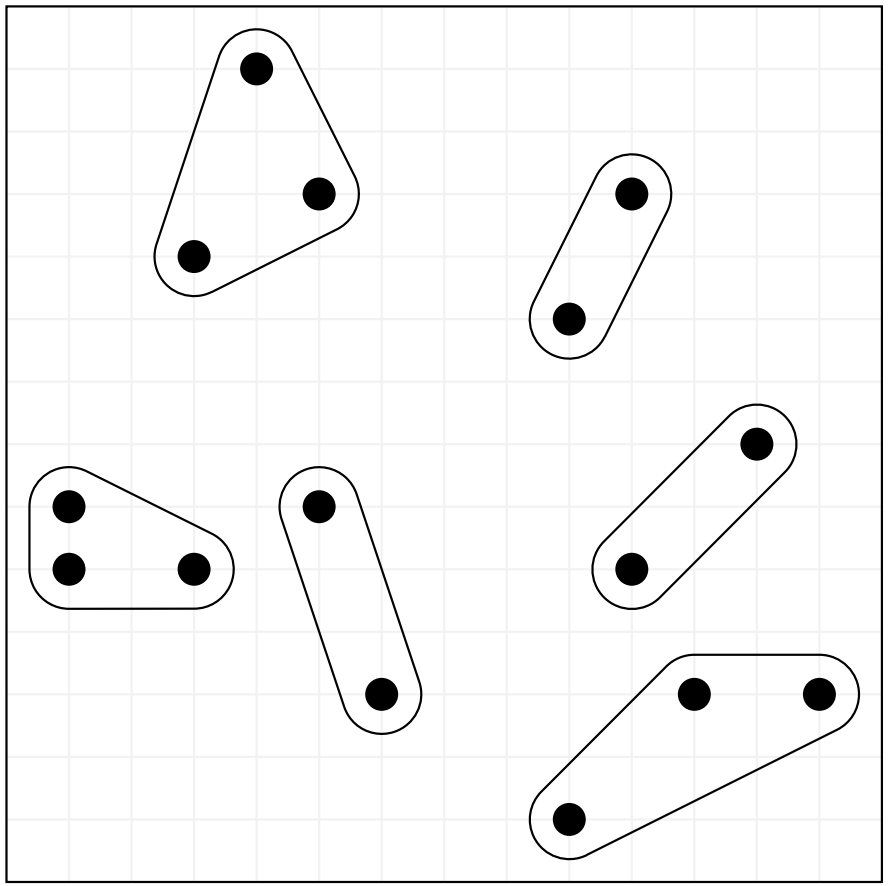
The AppOpt algorithm

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



Preliminary simulation results: Complexity

	Non-bipartite		Opt. Greedy		AppOpt	
n	CPU	Memory	CPU	Memory	CPU	Memory
10^2	0.0	34.1	0.1	34.1	0.0	29.1
10^3	1.6	84.0	4.3	55.2	0.0	29.2
10^4	352.3	4990.7	1050.5	2154.7	0.0	30.0
10^5	?	> 64000	?	> 64000	0.3	36.2
10^6					3.4	98.9
10^7					44.3	729.8
10^8		$> 10^{11}$		$> 10^{11}$	679.5	7038.7

- CPU: Average running time (seconds).
- Memory: Average maximum RAM-use for one run (MB).

Preliminary simulation results: Performance

- Setting: two-dimensional covariate space, uniform distribution.

		Non-bipartite		Opt. Greedy	
n	k	Max	Avg.	Max	Avg.
10^2	2	0.5%	-14.9%	283.5%	6.8%
10^2	4			187.8%	11.0%
10^3	2	-4.2%	-16.4%	881.3%	5.2%
10^3	4			677.2%	11.0%
10^4	2	-7.1%	-17.0%	2565.8%	3.4%
10^4	4			2161.4%	9.9%

- Max: Maximum within-block distance (relative to AppOpt).
- Avg.: Average within-block distance (relative to AppOpt).

Conclusion

- 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
- Closer to clustering than traditional blocking methods
- Important for separating design from analysis
- Lots of questions about best way to handle estimation
  - Design based estimators: Difference of means; Horvitz-Thompson estimator; double Hájek estimator
  - Probably do want to run a model on the blocked data. What if there is heterogeneity by blocks? $\frac{p}{n} \neq 0$

Joint Work with Michael J. Higgins and Fredrick Sävje

Neyman-Rubin potential outcomes model

- The Neyman-Rubin potential outcomes framework assumes the following model for response [Splawa-Neyman, Dabrowska, and Speed, 1990, Rubin, 1974]:

$$Y_{kc} = y_{kc1} T_{kc1} + y_{kc2} T_{kc2} + \ldots + y_{kcr} T_{kcr}.$$

- Y_{kc}: Observed response of kth unit in block c.
- y_{kct}: Potential outcome of the unit under treatment t.
- T_{kct}: Treatment indicators. $T_{kct} = 1$ if the unit receives treatment t, $T_{kct} = 0$ otherwise.

Parameters of interest and estimators

- Parameters of interest: Sample average treatment effect of treatment s relative to treatment t (SATE_{st}):

$$\text{SATE}_{st} = \sum_{c=1}^{b} \sum_{k=1}^{n_c} \frac{y_{kcs} - y_{kct}}{n}$$

- Two unbiased estimators of SATE_{st} are the difference-in-means estimator and the the Horvitz-Thompson estimator.

$$\hat{\delta}_{st,\text{diff}} \equiv \sum_{c=1}^{b} \frac{n_c}{n} \sum_{k=1}^{n_c} \left(\frac{y_{kcs} T_{kcs}}{\# T_{cs}} - \frac{y_{kct} T_{kct}}{\# T_{ct}} \right),$$

$$\hat{\delta}_{st,\text{HT}} \equiv \sum_{c=1}^{b} \frac{n_c}{n} \sum_{k=1}^{n_c} \left(\frac{y_{kcs} T_{kcs}}{n_c/r} - \frac{y_{kct} T_{kct}}{n_c/r} \right).$$

- Assume complete randomization of treatment, r divides n_c.

Variance of estimators

$$\begin{aligned} & \mathrm{Var}(\hat{\delta}_{st,\mathrm{diff}}) = \mathrm{Var}(\hat{\delta}_{st,\mathrm{HT}}) \\ = & \sum_{c=1}^{b} \frac{n_c^2}{n^2} \left(\frac{r-1}{n_c - 1}(\sigma_{cs}^2 + \sigma_{ct}^2) + 2\frac{\gamma_{cst}}{n_c - 1} \right) \end{aligned}$$

$$\begin{aligned} \mu_{cs} &= \frac{1}{n_c} \sum_{k=1}^{n_c} y_{kcs} \\ \sigma_{cs}^2 &= \frac{1}{n_c} \sum_{k=1}^{n_c} (y_{kcs} - \mu_{cs})^2 \\ \gamma_{cst} &= \frac{1}{n_c} \sum_{k=1}^{n_c} (y_{kcs} - \mu_{cs})(y_{kct} - \mu_{ct}) \end{aligned}$$

Variance of estimators

$$\begin{aligned} \mathrm{Var}(\hat{\delta}_{st,\mathrm{diff}}) &= \mathrm{Var}(\hat{\delta}_{st,\mathrm{HT}}) \\ &= \sum_{c=1}^{b} \frac{n_c^2}{n^2} \left(\frac{r-1}{n_c - 1}(\sigma_{cs}^2 + \sigma_{ct}^2) + 2\frac{\gamma_{cst}}{n_c - 1} \right) \end{aligned}$$

- Note: σ_{cs}^2 and σ_{ct}^2 are estimable, γ_{cst} not directly estimable.
- Conservative estimate:

$$\widehat{\mathrm{Var}} = \sum_{c=1}^{b} \frac{n_c^2}{n^2} \left(\frac{2(r-1)}{n_c - 1}(\hat{\sigma}_{cs}^2 + \hat{\sigma}_{ct}^2) \right)$$

- Small differences for more general treatment assignments.

When does blocking help?

- Blocking vs. completely randomized treatment assignment (no blocking): which estimates of $SATE_{st}$ have lower variance?
- Blocking helps if and only if:

$$\sum_{c=1}^{b} n_c^2 \left[\left(\frac{(r-1)(\sigma_s^2 + \sigma_t^2) + 2\gamma_{st}}{\sum n_c^2 (n-1)} \right) - \left(\frac{(r-1)(\sigma_{cs}^2 + \sigma_{ct}^2) + 2\gamma_{cst}}{n^2 (n_c - 1)} \right) \right] \geq 0$$

- Intuitive to make $\sigma_{cs}^2, \sigma_{ct}^2$ small w.r.t. σ_s^2, σ_t^2, but other blocking designs may also improve treatment effect estimates.

Can blocking hurt?

- When blocking is completely randomized:

$$\mathbb{E}\left[\sum_{c=1}^{b} n_c^2 \left(\frac{(r-1)(\sigma_{cs}^2+\sigma_{ct}^2)+2\gamma_{cst}}{n^2(n_c-1)}\right)\right]$$

$$= \sum_{c=1}^{b} n_c^2 \left(\frac{(r-1)(\sigma_s^2+\sigma_t^2)+2\gamma_{st}}{\sum n_c^2(n-1)}\right)$$

  Blocked variance = Completely randomized variance

- Any improvement to completely random blocking $\rightarrow$ Reduced variance in treatment effect estimates.

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.