Recovering Social Networks by Observing Votes

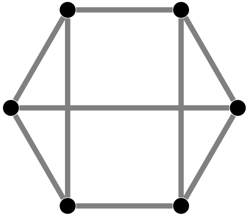
Lev Reyzin
work joint with Ben Fish and Yi Huang
UIC, Department of Mathematics

talk at ITA
paper to appear in AAMAS 2016

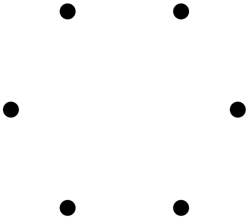
February 2016

Set-up

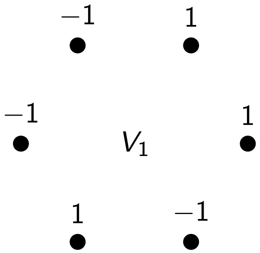
Hidden G:



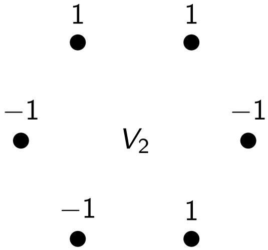
Set-up



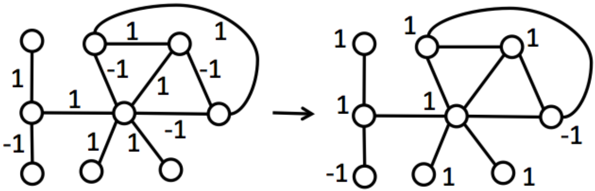
Set-up



Set-up



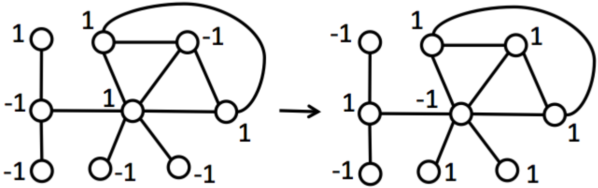
Models: independent conversations with neighbors



- *Independent conversation model (Conitzer, 2013):* Initial preferences are on the edges, final votes are the majority of preferences.

Each initial preference is 1 with probability some p and -1 with probability $1 - p$.

Models: taking majority of neighbors' preferences



- *Common neighbor model (Goles and Olivos, 1980, etc.):* Initial preferences are on the vertices, final votes are the majority of preferences.

Each initial preference is 1 with probability some p and -1 with probability $1 - p$.

Learning objectives

For a hidden graph G on n voters,

Definition (Exact learning)

Produce a graph G' in poly(n) time using only a poly(n) number of votes from G such that w.h.p. $G' = G$.

Definition (MLE learning)

Given m rounds of votes (which may not be from G), produce a graph G' in poly(n, m) time such that G' is the maximum likelihood graph for the votes.

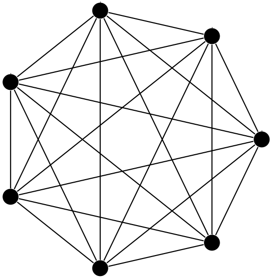
Independent conversation model, exact learning

Result (Upper bound)

When $p = 1/2$, there is an efficient algorithm for exact learning.

Observation (Lower bound)

For any constant $p \neq 1/2$, an exponential number of votes are required for exact learning.



Algorithm for $p = 1/2$

Result (Exact learning)

When $p = 1/2$, there is an efficient algorithm for exact learning.

Theorem

For $p = 1/2$, if (u, v) is an edge of G, then the covariance

$$Cov(X_u, X_v) \geq \frac{1}{2\pi n}$$

.

If (u, v) is not an edge of G, then the covariance

$$Cov(X_u, X_v) = 0.$$

Independent conversation model, MLE learning

Result (Upper bound)

Given m rouds of votes from G, when $p = 1/2$, for m sufficiently large, there is an efficient algorithm for finding the MLE w.h.p.

Result (Lower bound)

There is a randomized polynomial-time oracle reduction from computing the likelihood of the MLE from a sequence of votes w.h.p. to a #P*-hard problem.*

Moving from exact learning to MLE

Result (Upper bound)

Given m rounds of votes from G, when $p = 1/2$, for m sufficiently large, there is an efficient algorithm for finding the MLE w.h.p.

Theorem

For a vote sequence of length $\Omega(n^4)$ drawn from G, w.h.p. the MLE for that vote sequence is G.

Randomized reduction

Result (Lower bound)

There is a randomized polynomial-time oracle reduction from computing the likelihood of the MLE from a sequence of votes w.h.p. to a #P*-hard problem.*

Theorem (Conitzer, 2013)

Given a graph G and a set of votes V^* on G, computing $P(V^*|G)$ is #P*-hard (assuming $p = 1/2$).*

Reduction: Draw votes $V_1, \ldots, V_m$ from G, and then compute

$$P(V^*|G) = \frac{P(V_1, \ldots, V_m, V^*|G)}{P(V_1, \ldots, V_m|G)}.$$

Theorem

For a vote sequence of length $\Omega(n^4)$ drawn from G, w.h.p. the MLE for that vote sequence is G.

Common neighbor model, exact learning

Result (Lower bound)

No algorithm can exactly learn the unknown graph, regardless of the number of votes.

Common neighbor model, exact learning, lower bounds cont.

Definition (Covariance matching)

Find a graph G' such that for every pair of vertices in G', the expected covariance of the votes of the pair is close to the expected covariance of that pair in G.

Result (Covariance matching)

Finding a graph with given expected covariances between votes is at least as hard as recovering an adjacency matrix from its square.

Reduction from squared adjacency problem

Definition (Squared adjacency problem)

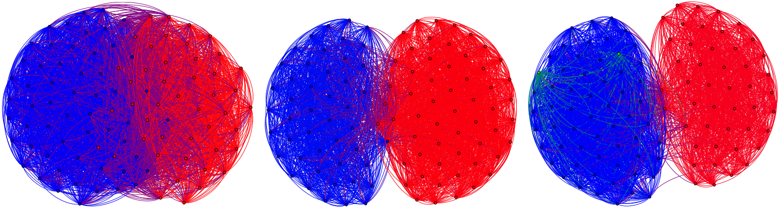
The input is a matrix B, and the decision problem asks if there is an adjacency matrix A of a simple graph such that $A^2 = B$.

Observation

The covariance between two vertices' votes is solely a function of the number of common neighbors between the two and the number of neighbors they do not share.

(We also show a more general version of the covariance matching problem to be NP-hard.)

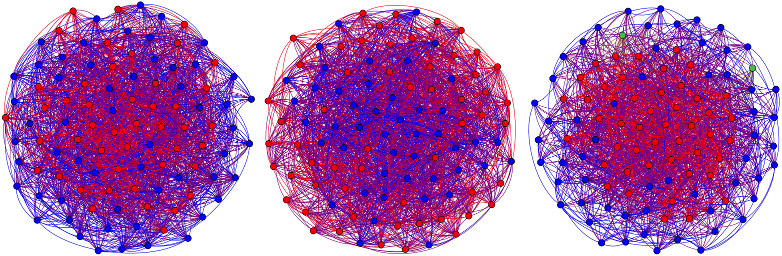
Independent conversation model, experiments



(a) 1989-1990 (b) 1999-2000 (c) 2013-2014

Figure: US Senate voting networks. Blue represents Democrats, red represents Republicans.

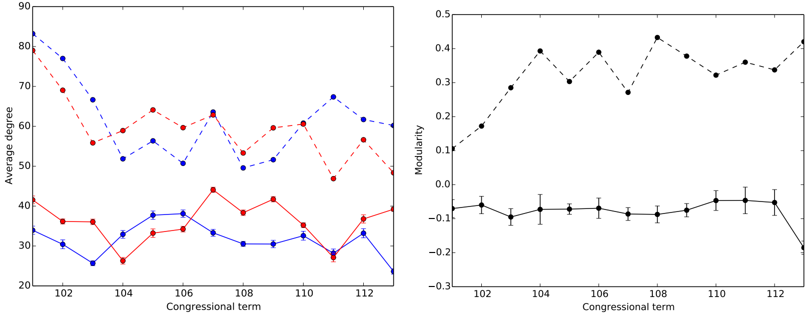
Common neighbor model, experiments with heuristic



(a) 1989-1990 (b) 1999-2000 (c) 2013-2014

Figure: US Senate voting networks. Blue represents Democrats, red represents Republicans.

Comparison



(a) Average degree (b) Modularity

Figure: Dashed and solid lines are statistics for the independent conversation model and common neighbor model, respectively.

Conclusion

Great care needs to be taken when choosing a model of voting:

- The resulting graphs are very different from each other
- Hardness of learning is very different from each other

Future work

Independent conversation model:

- When is MLE learning possible for $p \neq 1/2$ or when $p = 1/2$ but there are very few rounds of voting?
- Is there a PRAS for calculating the MLE?

Common neighbor model:

- For what kinds of graphs is exact learning possible?
- What about MLE learning?
- Is the squared adjacency problem hard?