Semidefinite relaxations of the clustering program and first-order methods for their solution

Brendan Ames

Department of Mathematics
The University of Alabama

AN70 Workshop on Modern Convex Optimization
The Fields Institute, University of Toronto
Friday July 7, 2017

Agenda

Present a semidefinite relaxation for the graph clustering problem based on decomposition of graph into densest union of disjoint subgraphs.

Give a probabilistic model for "clusterable" data and graphs, and theoretical recovery guarantees.

Propose an ADMM algorithm for solving this relaxation.

Open problems.

Joint with Aleksis Pirinen, Lund University.

Clustering

Clustering: partition data so that items in each cluster are similar to each other and items not in the same cluster are dissimilar.

Fundamental problem in statistics and machine learning:

- pattern recognition, computational biology, image processing/computer vison, network analysis.

No consensus on what constitutes a *good* clustering; depends heavily on application.

Intractable: usually modeled as some NP-hard problem (e.g. clique, normalized cut, k-means).

A sanity check

Clustering seems to be a very difficult/ill-posed problem.

Many heuristics seem to work well in practice.

Question: can we show that we can cluster “clusterable” data?
How do we model clusterable data?

Graph clustering

Similarity Graph: represent data set as a graph

- items = nodes
- edges indicate similarity

Cluster the data set by dividing the graph into dense subgraphs.

Dense = large average degree

The Weighted Similarity Graph

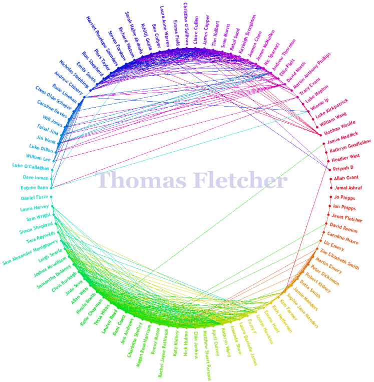
Given data and affinity function f indicating similarity between any two items.

Can model the data as weighted similarity graph $G_S = (V, E, \boldsymbol{W})$ as follows:

- Each item is represented by a node in V.
- We add an edge between each pair of two nodes i, j with edge weight $w_{ij} = f(i, j)$.
- w_{ij} is large if i and j are highly similar.

Example: Communities in Social Networks

- Nodes = users
- Edges = “friendship”.
- Densely connected groups = communities

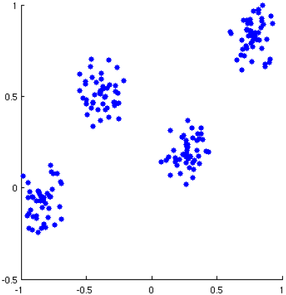


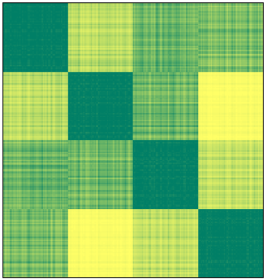
Example: Clustered Euclidean data

Suppose each data point in the ith cluster C_i is placed uniformly at random in a ball centered at $c_i \in \mathbf{R}^d$.

Distance within clusters will be small compared to the distance between clusters if centers are well-separated.

Choose $w_{ij} = \exp(-\|\boldsymbol{x}^i - \boldsymbol{x}^j\|^2)$.



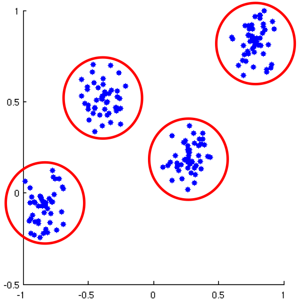


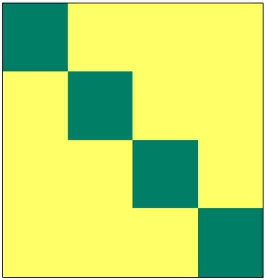
Example: Clustered Euclidean data

Suppose each data point in the ith cluster C_i is placed uniformly at random in a ball centered at $c_i \in \mathbf{R}^d$.

Distance within clusters will be small compared to the distance between clusters if centers are well-separated.

Choose $w_{ij} = \exp(-\|\boldsymbol{x}^i - \boldsymbol{x}^j\|^2)$.





The Densest k-Disjoint Clique Problem

To cluster the data we want to partition the graph into cliques with heavy support.

A **k-disjoint-clique subgraph** of a graph G is a subgraph of G induced by k disjoint cliques.

Densest k-disjoint-clique problem (KDC): find a k-disjoint-clique subgraph such that the sum of the densities of the k complete subgraphs induced by the cliques is maximized.

Density of complete subgraph induced by C:

$$d(C) = \frac{1}{|C|} \sum_{i \in C} \sum_{j \in C} w_{ij} = \frac{\mathbf{v}^T \mathbf{W} \mathbf{v}}{\mathbf{v}^T \mathbf{v}}$$

where $\mathbf{v}$ is the characteristic vector of C.

Lifting procedure for KDC

Let $\{C_1, \ldots, C_k\}$ define a k-disjoint-clique subgraph with characteristic vectors $\{\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_k\}$

Lift the k characteristic vectors $\{\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_k\}$ to the rank-k matrix variable $\boldsymbol{X}$:

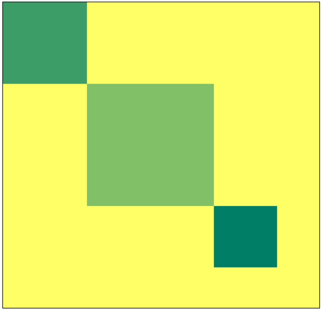
$$\boldsymbol{X} = \sum_{i=1}^{k} \frac{\boldsymbol{v}_i \boldsymbol{v}_i^T}{\|\boldsymbol{v}_i\|^2} = \sum_{i=1}^{k} \frac{\boldsymbol{v}_i \boldsymbol{v}_i^T}{|C_i|}$$

Want to find $\boldsymbol{X}$ that maximizes

$$\mathrm{Tr}(\boldsymbol{W}\boldsymbol{X}) = \sum_{i=1}^{k} \frac{\boldsymbol{v}_i^T \boldsymbol{W} \boldsymbol{v}_i}{\|\boldsymbol{v}_i\|^2} = \sum_{i=1}^{k} d(C_i)$$

Lifted solutions

Lifted solution $\boldsymbol{X}$ must satisfy:



Inlier rows sum to 1. Outlier rows equal 0: $\boldsymbol{X}\boldsymbol{e} \leq \boldsymbol{e}$

$\boldsymbol{X}$ is symmetric doubly nonnegative: $\boldsymbol{X} \geq \mathbf{0}, \quad \boldsymbol{X} \succeq 0$

$\mathrm{rank}(\boldsymbol{X}) = \mathrm{Tr}(\boldsymbol{X}) = k$

plus other combinatorial constraints

SDP Relaxation

Ignoring rank constraint and relaxing combinatorial constraints on $\boldsymbol{X}$ gives the semidefinite program:

$$\begin{aligned} \max \quad & \mathrm{Tr}(\boldsymbol{W}\boldsymbol{X}) \\ \mathrm{st} \quad & \boldsymbol{X}\boldsymbol{e} \le \boldsymbol{e} \\ & \mathrm{Tr}(\boldsymbol{X}) = k \\ & \boldsymbol{X} \ge \mathbf{0}, \quad \boldsymbol{X} \succeq 0. \end{aligned}$$

Question: When does the optimal solution of this relaxation recover underlying cluster structure in similarity graph?

The Stochastic Block Model

Stochastic Block Model (SBM): generate random graph containing k clusters of size r, where edges within-clusters are added with probability p and edges between-clusters are added with probability $q < p$.

Chen/Xu (2014): characterize when graphs sampled from the SBM are easy to cluster (have polynomial-time algorithm), hard to cluster (via max likelihood), and impossible to cluster. In particular, n-node graph from SBM is **easy** to cluster if

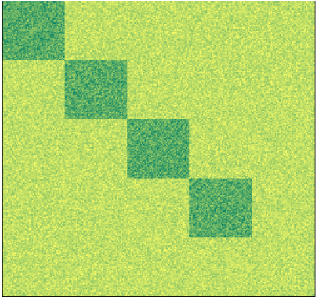
$$\frac{(p-q)^2}{q(1-q)} = \Omega\left(\frac{n}{r^2}\right).$$

Many other papers establish similar results for different classes algorithms, e.g., spectral clustering, convex/semidefinite relaxation, etc., as well as variants of the SBM with unbalanced clusters, outliers, heterogeneous edge probabilities, etc.

The Planted cluster model

Randomly generate weights $\boldsymbol{W} \in [0,1]^{n\times n}$ according to the following model:

- Start with clusters $C_1, \dots, C_k$ of sizes $r_1, \dots, r_k$. plus outlier set C_{k+1} of size r_{k+1}.
- Sample entries of $\boldsymbol{W}(C_i, C_i)$ i.i.d. from probability distribution Ω_1 with mean $\alpha(n)$ and variance $\sigma_1^2(n)$.
- Sample remaining entries of $\boldsymbol{W}$ i.i.d. from distribution Ω_2 with mean $\beta(n) < \alpha(n)$ and variance $\sigma_2^2(n)$.



Question: under what conditions on $\alpha, \beta, \sigma_1, \sigma_2, \boldsymbol{r}, n, k$ do we have perfect recovery of the clusters $C_1, \dots, C_k$?

Guaranteed Recovery: Notation

Suppose $\boldsymbol{W} \in \Sigma^n$ is sampled from the planted cluster model.

Let $\boldsymbol{X}^* = \sum_{i=1}^{k} \frac{\boldsymbol{v}_i \boldsymbol{v}_i^T}{r_i}$ denote the cluster matrix corresponding to the planted clusters $C_1, \dots, C_k$.

Let $\hat{r} = \min_{i=1,\dots,k} r_i$. and $\tilde{r} = \max_{i=1,\dots,k} r_i$.

Let $\tilde{\sigma}^2 := \max\{\sigma_1^2, \sigma_2^2\}$.

Guaranteed Recovery: Phase Transition

There exists constants $c_1, \ldots, c_5 > 0$ (independent of $\alpha, \beta, \hat{r}, n$) such that if

1. the gap assumption

$$\alpha - \beta \geq c_5 \max\left\{\sqrt{\frac{\tilde{\sigma}^2 \log n}{\hat{r}}}, \frac{\log n}{\hat{r}}\right\}$$

is satisfied, and

2. $\hat{r}$ satisfies

$$(\alpha - \beta)\hat{r} \geq c_1 \max\left\{\sigma_2\sqrt{n}, \sqrt{\log n}\right\} + c_2 \max\left\{\sigma_1\sqrt{\tilde{r}}, \sqrt{\log n}\right\} + c_3\left(\max\left\{\sigma_1^2, \frac{\log n}{\hat{r}}\right\} k r_{k+1}\right)^{1/2} + c_4 \beta r_{k+1}$$

then $\{C_1, \ldots, C_k\}$ is the unique densest k-disjoint-clique subgraph and $\boldsymbol{X}^*$ is the unique optimal solution of the SDP relaxation with high probability

Signal-to-noise ratio

This suggests that we can recover the planted clusters w.h.p. provided that

$$\frac{(\alpha-\beta)^2}{\tilde{\sigma}^2} = \Omega\left(\frac{n}{\hat{r}^2}\right).$$

The left-hand side acts as a signal-to-noise ratio: ratio of difference between expected edge weights to noise variance.

This agrees with/generalizes the easy regime for cluster recovery proposed by Chen and Xu (2014), and Jalali et al. (2015).

The relaxation is mostly parameter free: SDP needs number of clusters k but doesn't need estimate of cluster sizes r_i, gap statistic $\alpha - \beta$, etc., seen in similar theoretical guarantees.

Special Case: Stochastic Block Models

Suppose Ω_1 and Ω_2 are Bernoulli distributions with probability of adding an edge p and q respectively ($p > q$) with no outliers ($r_{k+1} = 0$).

Dense case: p, q constant (independent of n).

Have exact recovery w.h.p. if $\hat{r} \geq \hat{c}\sqrt{n}$ for some scalar $\hat{c}$ (depending on p, q).

Sparse case: p constant, $q \leq \frac{\log n}{n}$.

Have exact recovery w.h.p. if $\hat{r} \geq \tilde{c} \log n$ for some constant $\tilde{c}$.

Sensitivity to outliers

SBM with $k = 1$ specializes to the **planted clique model.** (Graph consists of a single complete subgraph obscured by noise).

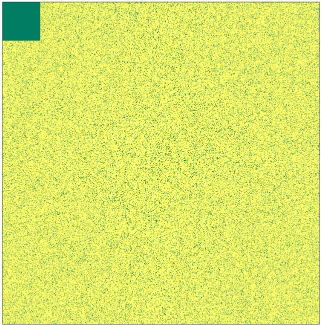
Theorem suggests exact recovery with $\hat{r} = \Omega(r_{k+1}) = \Omega(n)$.

This far exceeds the standard planted clique recovery guarantee of $\hat{r} = \Omega(\sqrt{n})$ (in dense case).

Unfortunately this bound is tight.

- Expected value of $\boldsymbol{X}^*$ is $p\hat{r}$.
- Expected value of $\frac{1}{n}\operatorname{Tr}(\boldsymbol{W}\boldsymbol{e}\boldsymbol{e}^T) = \Omega(qn)$. Therefore, $\boldsymbol{X}^*$ is suboptimal unless
$$\hat{r} = \Omega\left(\frac{p}{q}n\right).$$

Sensitivity to outliers (2)



$n = 10000$

$\hat{r} = 1200 = 12\sqrt{n}$.

$\mathrm{Tr}(\boldsymbol{W}\boldsymbol{X}^*) = 1200$

$< \frac{1}{n}\,\mathrm{Tr}(\boldsymbol{W}\boldsymbol{e}\boldsymbol{e}^T) \approx 1375$

Proof Outline

Can construct a choice of dual variables using KKT conditions.

Have a dual certificate when $\boldsymbol{W} = \boldsymbol{E}[\boldsymbol{W}]$.

Use concentration inequalities to show that this choice of dual variables is feasible w.h.p. when gap assumption and $\hat{r}$ bound are met.

- Establish nonnegativity using Bernstein inequality.
- Establish semidefiniteness using the Matrix Bernstein Inequality (Bandeira and van Handel 2016).

Solving the SDP

Clustering SDP has $n \times n$ semidefinite variable and $m = O(n^2)$ (in)equality constraints.

Can be (approximately) solved in polynomial-time using interior point methods.

Costs $O(m^3) = O(n^6)$ flops per Newton iteration.

- Prohibitively expensive for large graphs/data.

Alternating Direction Method of Multipliers

Let $\Xi := \left\{ \boldsymbol{X} \in \Sigma^V : \boldsymbol{X}\boldsymbol{e} \leq \boldsymbol{e},\ \boldsymbol{X} \geq \boldsymbol{0} \right\}.$

Let $\Omega := \left\{ \boldsymbol{X} \in \Sigma^V : \mathrm{Tr}(\boldsymbol{X}) = k, \boldsymbol{X} \in \Sigma^V_+ \right\}$.

Can rewrite Cluster SDP as:

$$\max_{\boldsymbol{X},\boldsymbol{Y}} \left\{ \mathrm{Tr}(\boldsymbol{W}\boldsymbol{Y}) : \boldsymbol{X} - \boldsymbol{Y} = \boldsymbol{0},\ \boldsymbol{X} \in \Xi,\ \boldsymbol{Y} \in \Omega \right\}.$$

Augmented Lagrangian is

$$L_\rho(\boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{U}) = \mathrm{Tr}(\boldsymbol{W}\boldsymbol{Y}) - \mathrm{Tr}(\boldsymbol{U}(\boldsymbol{X} - \boldsymbol{Y})) + \frac{\rho}{2}\|\boldsymbol{X} - \boldsymbol{Y}\|_F^2.$$

Solve using **ADMM**: update iterate $(\boldsymbol{X}^k, \boldsymbol{Y}^k, \boldsymbol{U}^k)$ by

$$\begin{aligned}
\boldsymbol{Y}^{k+1} &= \operatorname*{arg\,min}_{\boldsymbol{Y}\in\Omega} L_\rho(\boldsymbol{X}^k, \boldsymbol{Y}, \boldsymbol{U}^k) \\
\boldsymbol{X}^{k+1} &= \operatorname*{arg\,min}_{\boldsymbol{X}\in\Xi} L_\rho(\boldsymbol{X}, \boldsymbol{Y}^{k+1}, \boldsymbol{U}^k) \\
\boldsymbol{U}^{k+1} &= \boldsymbol{U}^k - \rho(\boldsymbol{X}^{k+1} - \boldsymbol{Y}^{k+1}).
\end{aligned}$$

Updating $\boldsymbol{Y}$

$\boldsymbol{Y}^{k+1}$ is a minimizer of the subproblem

$$\min_{\boldsymbol{Y}\in\Omega} \frac{1}{2} \left\| \boldsymbol{Y} - \left(\boldsymbol{X}^k - \frac{1}{\rho}\left(\boldsymbol{W} + \boldsymbol{U}^k\right)\right)\right\|_F^2 .$$

Let $\boldsymbol{X}^k - \frac{1}{\rho}\left(\boldsymbol{W} + \boldsymbol{U}^k\right)$ have eigenvalue decomposition $\boldsymbol{V}\,\mathrm{Diag}(\boldsymbol{v}^k)\boldsymbol{V}^T$.

Then $\boldsymbol{Y}^{k+1} = \boldsymbol{V}\,\mathrm{Diag}(\boldsymbol{y}^*)\boldsymbol{V}^T$ where $\boldsymbol{y}^*$ is the projection of $\boldsymbol{v}^k$ onto the simplex

$$\left\{\boldsymbol{y} : \boldsymbol{e}^T\boldsymbol{y} = k,\ \boldsymbol{y} \geq \boldsymbol{0}\right\}.$$

Updating $\boldsymbol{X}$

$\boldsymbol{X}^{k+1}$ is a minimizer of the subproblem

$$\min_{\boldsymbol{X}\in\Xi} \frac{1}{2}\left\|\boldsymbol{X} - \left(\boldsymbol{Y}^{k+1} + \frac{1}{\rho}\boldsymbol{U}^k\right)\right\|_F^2.$$

Has dual problem

$$\min_{\boldsymbol{z}\geq\boldsymbol{0}} \frac{1}{2}\left\|\left[\left(\boldsymbol{Y}^{k+1} + \frac{\boldsymbol{U}^k}{\rho}\right) - \frac{\boldsymbol{z}\boldsymbol{e}^T + \boldsymbol{e}\boldsymbol{z}^T}{2}\right]_+\right\|_F^2 + \boldsymbol{z}^T\boldsymbol{e}$$

can be efficiently solved for $\boldsymbol{z}^*$ using the spectral projected gradient method of Birgin et al. 2000.

Update $\boldsymbol{X}$ by

$$\boldsymbol{X}^{k+1} = \left[\left(\boldsymbol{Y}^{k+1} + \frac{1}{\rho}\boldsymbol{U}^k\right) - \frac{\boldsymbol{z}^*\boldsymbol{e}^T + \boldsymbol{e}(\boldsymbol{z}^*)T}{2}\right]_+;$$

here $([\boldsymbol{Z}]_+)_{ij} = \max\{0, z_{ij}\}$.

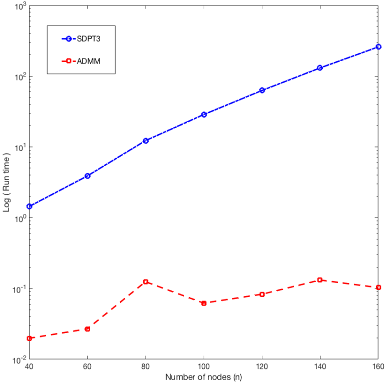
Numerical results

Add within-cluster edges independently with prob $p = 0.75$ and between-cluster edges with prob $q = 0.25$.

Fix $k = 4$ equal sized clusers so that $n = k\hat{r}$; *l*et $\hat{r}$ vary from 10 to 40.

Solve with SDPT3 (using CVX) and our ADMM algorithm.

Numerical results (2)



n	**SDPT3**	**ADMM**
40	1.4385	0.0198
60	3.8845	0.0269
80	12.2463	0.1245
100	28.6202	0.0622
120	62.6146	0.0829
140	130.4715	0.1316
160	258.5871	0.1029

Future work: low-rank factorization

Have an alternate nonconvex relaxation:

$$\max_{\boldsymbol{Y}\in\mathbf{R}^{n\times k}} \left\{ \mathrm{Tr}(\boldsymbol{Y}^T\boldsymbol{W}\boldsymbol{Y}) : \boldsymbol{Y}\boldsymbol{Y}^T\boldsymbol{e} \le \boldsymbol{e},\ \mathrm{Tr}(\boldsymbol{Y}\boldsymbol{Y}^T) = \|\boldsymbol{Y}\|_F^2 = k,\ \ \boldsymbol{Y} \ge \mathbf{0} \right\}$$

Burer and Monteiro 2003, 2005 propose this approach and related augmented Lagrangian method.

Wainwright and Chen 2016 analyze gradient methods for similar low-rank factored problems.

Evaluation of the augmented Lagrangian and its gradient cost $O(n^2 k)$ flops.

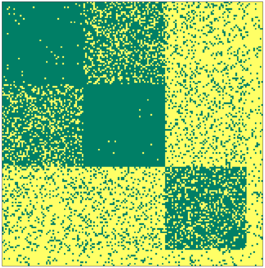
Much further work is needed to show that overall run-time is competitive with ADMM and characterize recovery properties.

Future work: heterogeneous distributions

Conjecture: can strengthen recovery guarantee to following case:

- If u, v cluster C_i then $\boldsymbol{E}[w_{uv}] = \alpha_i$.
- If $u \in C_i$, $v \in C_j$, $i \neq j$ then $\boldsymbol{E}[w_{uv}] = \beta_{ij}$.
- **Weak assortativity:** Replace $\alpha - \beta$ with

$$\min_i \left(\alpha_i - \max_j \beta_{ij}. \right)$$

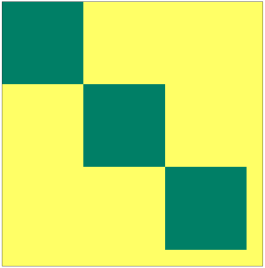


Future work: heterogeneous distributions

Conjecture: can strengthen recovery guarantee to following case:

- If u, v cluster C_i then $\boldsymbol{E}[w_{uv}] = \alpha_i$.
- If $u \in C_i$, $v \in C_j$, $i \neq j$ then $\boldsymbol{E}[w_{uv}] = \beta_{ij}$.
- **Weak assortativity:** Replace $\alpha - \beta$ with

$$\min_i \left(\alpha_i - \max_j \beta_{ij}. \right)$$



Thank you!

Matlab implementations available from
bpames.people.ua.edu/software

Preprint: arxiv.org/abs/1603.05296

Thanks: B. Ames supported in part by University of Alabama RGC grant RG14678.