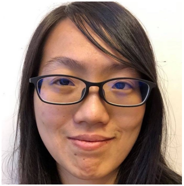
Discovering conflicting groups in signed networks

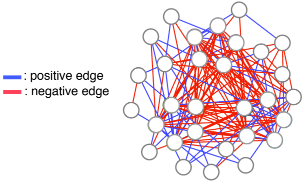
Ruo-Chun Tzeng[1] Bruno Ordozgoiti[2] Aristides Gionis[1,2]



[1]KTH Royal Institute of Technology [2]Aalto University

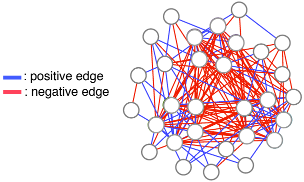
34th Conference on Neural Information Processing Systems (2020)

Motivation



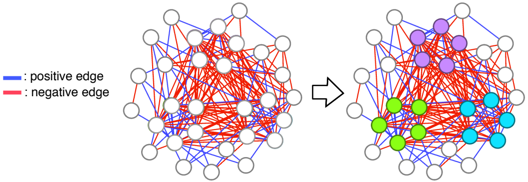
- *Signed networks* can represent opposing social relationships such as friend-foe, trust-distrust, agree-disagree, ... etc.

Motivation



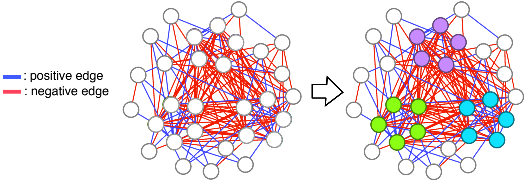
- *Signed networks* can represent opposing social relationships such as friend-foe, trust-distrust, agree-disagree, ... etc.
- Groups are formed of people who interact positively with each other or who have the common enemies.

Motivation



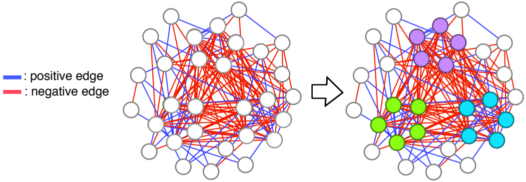
- *Signed networks* can represent opposing social relationships such as friend-foe, trust-distrust, agree-disagree, ... etc.
- Our goal is to detect *conflicting groups* where intra-group edges are mostly positive and inter-group edges are mostly negative.

Motivation



- *Signed networks* can represent opposing social relationships such as friend-foe, trust-distrust, agree-disagree, ... etc.
- Our goal is to detect *conflicting groups* where intra-group edges are mostly positive and inter-group edges are mostly negative.
- REBOUND aims to mitigate the polarization, filter-bubble, and fake news and conflicting groups are closely related to these phenomena.

Motivation



- *Signed networks* can represent opposing social relationships such as friend-foe, trust-distrust, agree-disagree, ... etc.
- Our goal is to detect *conflicting groups* where intra-group edges are mostly positive and inter-group edges are mostly negative.
- Different to signed clustering [5] and correlation clustering [1] that partition the entire network, we allow *neutral* nodes to exist.

2-PC [3]: detecting $k = 2$ conflicting groups

- Given $G = (V, E_+ \cup E_-)$ with unit weight, the objective is to

$$\max_{S_1 \cap S_2 = \emptyset} \frac{\sum_{h \in [2]} (|E_+(S_h)| - |E_-(S_h)|) + \sum_{h \neq l \in [2]} (|E_-(S_h, S_\ell)| - |E_+(S_h, S_\ell)|)}{|\cup_{h \in [2]} S_h|},$$

where $E(S_h, S_\ell) = \{(i, j) \in E : i \in S_h, j \in S_\ell\}$ and $E(S_h) = E(S_h, S_h)$.

2-PC [3]: detecting $k = 2$ conflicting groups

- Given $G = (V, E_+ \cup E_-)$ with unit weight, the objective is to

$$\max_{S_1 \cap S_2 = \emptyset} \frac{\sum_{h \in [2]}(|E_+(S_h)| - |E_-(S_h)|) + \sum_{h \neq l \in [2]}(|E_-(S_h, S_\ell)| - |E_+(S_h, S_\ell)|)}{|\cup_{h \in [2]} S_h|},$$

where $E(S_h, S_\ell) = \{(i,j) \in E : i \in S_h, j \in S_\ell\}$ and $E(S_h) = E(S_h, S_h)$.

- Idea: prefer the S_1, S_2 that
  - have many consistent edges and few inconsistent edges, and
  - the size of $S_1 \cup S_2$ is as small as possible.

2-PC [3]: detecting $k = 2$ conflicting groups

- Given $G = (V, E_+ \cup E_-)$, the objective is to

$$\max_{S_1 \cap S_2 = \emptyset} \frac{\sum_{h \in [2]}(|E_+(S_h)| - |E_-(S_h)|) + \sum_{h \neq l \in [2]}(|E_-(S_h, S_\ell)| - |E_+(S_h, S_\ell)|)}{|\cup_{h \in [2]} S_h|},$$

- Denote $A \in \{0, \pm 1\}^{n \times n}$ the signed adjacency matrix of G.

2-PC [3]: detecting $k = 2$ conflicting groups

- Given $G = (V, E_+ \cup E_-)$, the objective is to

$$\max_{S_1 \cap S_2 = \emptyset} \frac{\sum_{h\in[2]}(|E_+(S_h)| - |E_-(S_h)|) + \sum_{h\neq l\in[2]}(|E_-(S_h, S_\ell)| - |E_+(S_h, S_\ell)|)}{|\cup_{h\in[2]} S_h|},$$

$$= \max_{S_1 \cap S_2 = \emptyset} \frac{\sum_{h\in[2]} \sum_{(i,j)\in E(S_h)} A_{i,j} + \sum_{h\neq \ell\in[2]} \sum_{(i,j)\in E(S_h, S_\ell)}(-A_{i,j})}{|\cup_{h\in[2]} S_h|}$$

$$= \max\{\frac{x^T A x}{x^T x} : x \in \{-1, 0, 1\}^n \backslash \mathbf{0}\}. \quad (1)$$

2-PC [3]: detecting $k = 2$ conflicting groups

- Given $G = (V, E_+ \cup E_-)$, the objective is to

$$\max_{S_1 \cap S_2 = \emptyset} \frac{\sum_{h \in [2]}(|E_+(S_h)| - |E_-(S_h)|) + \sum_{h \neq l \in [2]}(|E_-(S_h, S_\ell)| - |E_+(S_h, S_\ell)|)}{|\cup_{h \in [2]} S_h|},$$

$$= \max_{S_1 \cap S_2 = \emptyset} \frac{\sum_{h \in [2]} \sum_{(i,j) \in E(S_h)} A_{i,j} + \sum_{h \neq \ell \in [2]} \sum_{(i,j) \in E(S_h, S_\ell)} (-A_{i,j})}{|\cup_{h \in [2]} S_h|}$$

$$= \max\{\frac{x^T A x}{x^T x} : x \in \{-1, 0, 1\}^n \backslash \mathbf{0}\}. \quad (1)$$

- Solving Eq (1) is APX-Hard and the current best $\mathcal{O}(n^{1/3})$-approx algorithm [2] is based on SDP.

2-PC [3]: detecting $k = 2$ conflicting groups

- Given $G = (V, E_+ \cup E_-)$, the objective is to

$$\max_{S_1 \cap S_2 = \emptyset} \frac{\sum_{h \in [2]}(|E_+(S_h)| - |E_-(S_h)|) + \sum_{h \neq l \in [2]}(|E_-(S_h, S_\ell)| - |E_+(S_h, S_\ell)|)}{|\cup_{h \in [2]} S_h|},$$

$$= \max_{S_1 \cap S_2 = \emptyset} \frac{\sum_{h \in [2]} \sum_{(i,j) \in E(S_h)} A_{i,j} + \sum_{h \neq \ell \in [2]} \sum_{(i,j) \in E(S_h, S_\ell)} (-A_{i,j})}{|\cup_{h \in [2]} S_h|}$$

$$= \max\{\frac{x^T A x}{x^T x} : x \in \{-1, 0, 1\}^n \backslash \mathbf{0}\}. \quad (1)$$

- 2-PC [3] proposed a more practical $O(n^{1/2})$-approx algorithm by randomized rounding the leading eigenvector of A.

2-PC [3]: detecting $k = 2$ conflicting groups

- Given $G = (V, E_+ \cup E_-)$, the objective is to

$$\max_{S_1 \cap S_2 = \emptyset} \frac{\sum_{h \in [2]}(|E_+(S_h)| - |E_-(S_h)|) + \sum_{h \neq l \in [2]}(|E_-(S_h, S_\ell)| - |E_+(S_h, S_\ell)|)}{|\cup_{h \in [2]} S_h|},$$

$$= \max_{S_1 \cap S_2 = \emptyset} \frac{\sum_{h \in [2]} \sum_{(i,j) \in E(S_h)} A_{i,j} + \sum_{h \neq \ell \in [2]} \sum_{(i,j) \in E(S_h, S_\ell)} (-A_{i,j})}{|\cup_{h \in [2]} S_h|}$$

$$= \max\{\frac{x^T A x}{x^T x} : x \in \{-1, 0, 1\}^n \backslash \mathbf{0}\}. \quad (1)$$

- In this work, we are interested in detecting $k \geq 2$ conflicting groups.

2-PC [3]: detecting $k = 2$ conflicting groups

- Given $G = (V, E_+ \cup E_-)$, the objective is to

$$\max_{S_1 \cap S_2 = \emptyset} \frac{\sum_{h\in[2]}(|E_+(S_h)| - |E_-(S_h)|) + \sum_{h\neq l\in[2]}(|E_-(S_h, S_\ell)| - |E_+(S_h, S_\ell)|)}{|\cup_{h\in[2]} S_h|},$$

$$= \max_{S_1 \cap S_2 = \emptyset} \frac{\sum_{h\in[2]}\sum_{(i,j)\in E(S_h)} A_{i,j} + \sum_{h\neq \ell\in[2]}\sum_{(i,j)\in E(S_h,S_\ell)}(-A_{i,j})}{|\cup_{h\in[2]} S_h|}$$

$$= \max\{\frac{x^T A x}{x^T x} : x \in \{-1, 0, 1\}^n \backslash \mathbf{0}\}. \quad (1)$$

- Observation: Eq (1) limits to detect only 2 conflicting groups, but the idea of the objective does generalize to $k \geq 2$!

Our approach: detecting $k \geq 2$ conflicting groups

- Generalize the objective of 2-PC [3] from

$$\max_{S_1 \cap S_2 = \emptyset} \frac{\sum_{h \in [2]}(|E_+(S_h)| - |E_-(S_h)|) + \sum_{h \neq l \in [2]}(|E_-(S_h, S_\ell)| - |E_+(S_h, S_\ell)|)}{|\cup_{h \in [2]} S_h|}.$$

Our approach: detecting $k \geq 2$ conflicting groups

- Generalize the objective of 2-PC [3] to

$$\max_{S_1,\cdots,S_k} \frac{\sum_{h\in[k]}(|E_+(S_h)| - |E_-(S_h)|) + \frac{1}{k-1}\sum_{h\neq l\in[k]}(|E_-(S_h,S_\ell)| - |E_+(S_h,S_\ell)|)}{|\cup_{h\in[2]} S_h|}.$$

The weighting to prevent the inter-group edges from dominating the objective.

Our approach: detecting $k \geq 2$ conflicting groups

- The generalized objective is equivalent to

$$\max_{S_1,\cdots,S_k} \frac{\sum_{h\in[k]}\sum_{(i,j)\in E(S_h)} A_{i,j} + \frac{1}{k-1}\sum_{h\neq\ell\in[k]}\sum_{(i,j)\in E(S_h,S_\ell)}(-A_{i,j})}{|\cup_{h\in[k]} S_h|}. \quad (2)$$

Our approach: detecting $k \geq 2$ conflicting groups

- The generalized objective is equivalent to

$$\max_{S_1,\cdots,S_k} \frac{\sum_{h\in[k]}\sum_{(i,j)\in E(S_h)} A_{i,j} + \frac{1}{k-1}\sum_{h\neq\ell\in[k]}\sum_{(i,j)\in E(S_h,S_\ell)}(-A_{i,j})}{|\cup_{h\in[k]} S_h|}. \quad (2)$$

- Introducing the indicator matrix $X \in \{0,1\}^{n\times k}$ with $X_{i,:} = I_{j,:}$ if $i \in S_j$.

Our approach: detecting $k \geq 2$ conflicting groups

- The generalized objective is equivalent to

$$\max_{S_1,\cdots,S_k} \frac{\sum_{h\in[k]} \sum_{(i,j)\in E(S_h)} A_{i,j} + \frac{1}{k-1} \sum_{h\neq \ell \in[k]} \sum_{(i,j)\in E(S_h,S_\ell)} (-A_{i,j})}{|\cup_{h\in[k]} S_h|}. \quad (2)$$

- Introducing the indicator matrix $X \in \{0,1\}^{n\times k}$ with $X_{i,:} = I_{j,:}$ if $i \in S_j$.
- The numerator of Eq (2) can be written as

$$\langle A, XX^T\rangle_F - \frac{1}{k-1}\langle A, XJ_kX^T - XX^T\rangle_F = \frac{1}{k-1}\langle A, XL_kX^T\rangle_F,$$

where J_k is the $k \times k$ matrix of all 1s and $L_k = kI - J_k$.

Our approach: detecting $k \geq 2$ conflicting groups

- The generalized objective is equivalent to

$$\max_{S_1,\cdots,S_k} \frac{\sum_{h\in[k]}\sum_{(i,j)\in E(S_h)} A_{i,j} + \frac{1}{k-1}\sum_{h\neq\ell\in[k]}\sum_{(i,j)\in E(S_h,S_\ell)}(-A_{i,j})}{|\cup_{h\in[k]} S_h|}. \quad (2)$$

- The numerator of Eq (2) can be written as

$$\langle A, XX^T\rangle_F - \frac{1}{k-1}\langle A, XJ_kX^T - XX^T\rangle_F = \frac{1}{k-1}\langle A, XL_kX^T\rangle_F,$$

where J_k is the $k \times k$ matrix of all 1s and $L_k = kI - J_k$.

- Observation: the EVD of $L_k = U\text{diag}([0, k \cdots, k])U^T$ is useful if choosing

$$U = \begin{bmatrix} 1/\sqrt{k} & c_1(k-1) & 0 & \cdots & 0 \\ 1/\sqrt{k} & -c_1 & c_2(k-2) & \cdots & 0 \\ \vdots & \vdots & \cdots & & \vdots \\ 1/\sqrt{k} & -c_1 & -c_2 & \cdots & c_{k-1} \\ 1/\sqrt{k} & -c_1 & -c_2 & \cdots & -c_{k-1} \end{bmatrix}.$$

Our approach: detecting $k \geq 2$ conflicting groups

- The generalized objective is equivalent to

$$\max_{S_1,\cdots,S_k} \frac{\sum_{h\in[k]}\sum_{(i,j)\in E(S_h)} A_{i,j} + \frac{1}{k-1}\sum_{h\neq \ell\in[k]}\sum_{(i,j)\in E(S_h,S_\ell)}(-A_{i,j})}{|\cup_{h\in[k]} S_h|}. \quad (2)$$

- With $Y = (XU)_{:,2:}$, the numerator of Eq (2) can be written as

$$\frac{1}{k-1}\langle A, XL_kX^T\rangle_F = \frac{k}{k-1}\langle A, (XU)_{:,2:}((XU)_{:,2:})^T\rangle_F = \frac{k}{k-1} Tr(Y^TAY).$$

- Observation: the EVD of $L_k = U\text{diag}([0, k\cdots, k])U^T$ is useful if choosing

$$U = \begin{bmatrix} 1/\sqrt{k} & c_1(k-1) & 0 & \cdots & 0 \\ 1/\sqrt{k} & -c_1 & c_2(k-2) & \cdots & 0 \\ \vdots & \vdots & \ldots & & \vdots \\ 1/\sqrt{k} & -c_1 & -c_2 & \cdots & c_{k-1} \\ 1/\sqrt{k} & -c_1 & -c_2 & \cdots & -c_{k-1} \end{bmatrix}.$$

Our approach: detecting $k \geq 2$ conflicting groups

- The generalized objective is equivalent to

$$\max_{S_1,\cdots,S_k} \frac{\sum_{h\in[k]}\sum_{(i,j)\in E(S_h)} A_{i,j} + \frac{1}{k-1}\sum_{h\neq\ell\in[k]}\sum_{(i,j)\in E(S_h,S_\ell)}(-A_{i,j})}{|\cup_{h\in[k]} S_h|}. \quad (2)$$

- With $Y = (XU)_{:,2:}$, the numerator of Eq (2) can be written as

$$\frac{1}{k-1}\langle A, XL_kX^T\rangle_F = \frac{k}{k-1}\langle A, (XU)_{:,2:}((XU)_{:,2:})^T\rangle_F = \frac{k}{k-1}Tr(Y^TAY).$$

- The denominator of Eq (2) can be written as

$$|\cup_{h\in[k]} S_h| = Tr(X^TX) = Tr((XU)^T(XU)) = Tr((XU)_{:,1}^T(XU)_{:,1}) + Tr(Y^TY)$$
$$= kTr((XU)_{:,1}^T(XU)_{:,1}) = \frac{k}{k-1}Tr(Y^TY).$$

- Observation: the EVD of $L_k = U\text{diag}([0, k\cdots, k])U^T$ is useful if choosing

$$U = \begin{bmatrix} 1/\sqrt{k} & c_1(k-1) & 0 & \cdots & 0 \\ 1/\sqrt{k} & -c_1 & c_2(k-2) & \cdots & 0 \\ \vdots & \vdots & \ldots & & \vdots \\ 1/\sqrt{k} & -c_1 & -c_2 & \cdots & c_{k-1} \\ 1/\sqrt{k} & -c_1 & -c_2 & \cdots & -c_{k-1} \end{bmatrix}.$$

Our approach: detecting $k \geq 2$ conflicting groups

- The objective becomes $\max_{Y \in \mathbb{R}^{n \times (k-1)} \setminus \{\mathbf{0}\}} \frac{Tr(Y^T A Y)}{Tr(Y^T Y)}$ subject to $Y = (XU)_{:,2:}$ and $X_{i,:} \in \{\mathbf{0}, I_{1,:}, \cdots, I_{k,:}\}$, where

$$U = \begin{bmatrix} 1/\sqrt{k} & c_1(k-1) & 0 & \cdots & 0 \\ 1/\sqrt{k} & -c_1 & c_2(k-2) & \cdots & 0 \\ \vdots & \vdots & \cdots & & \vdots \\ 1/\sqrt{k} & -c_1 & -c_2 & \cdots & c_{k-1} \\ 1/\sqrt{k} & -c_1 & -c_2 & \cdots & -c_{k-1} \end{bmatrix}.$$

Our approach: detecting $k \geq 2$ conflicting groups

- The objective becomes $\max_{Y \in \mathbb{R}^{n \times (k-1)} \setminus \{\mathbf{0}\}} \frac{Tr(Y^T A Y)}{Tr(Y^T Y)}$ subject to $Y = (XU)_{:,2:}$ and $X_{i,:} \in \{\mathbf{0}, I_{1,:}, \cdots, I_{k,:}\}$, where

$$U = \begin{bmatrix} 1/\sqrt{k} & c_1(k-1) & 0 & \cdots & 0 \\ 1/\sqrt{k} & -c_1 & c_2(k-2) & \cdots & 0 \\ \vdots & \vdots & \cdots & & \vdots \\ 1/\sqrt{k} & -c_1 & -c_2 & \cdots & c_{k-1} \\ 1/\sqrt{k} & -c_1 & -c_2 & \cdots & -c_{k-1} \end{bmatrix}.$$

- The constraint is equivalent to requiring $\forall j \in [k-1]$,
  1. $Y_{:,j} \in \{c_j(k-j), 0, -c_j\}^n$ and
  2. $Y_{i,\ell} = c_\ell(k-\ell)$ implies $Y_{i,j} = 0, \forall j > \ell$.

Our approach: detecting $k \geq 2$ conflicting groups

- The objective becomes $\max_{Y \in \mathbb{R}^{n \times (k-1)} \setminus \{\mathbf{0}\}} \frac{Tr(Y^T A Y)}{Tr(Y^T Y)}$ subject to $Y = (XU)_{:,2:}$ and $X_{i,:} \in \{\mathbf{0}, I_{1,:}, \cdots, I_{k,:}\}$, where

$$U = \begin{bmatrix} 1/\sqrt{k} & c_1(k-1) & 0 & \cdots & 0 \\ 1/\sqrt{k} & -c_1 & c_2(k-2) & \cdots & 0 \\ \vdots & \vdots & \cdots & & \vdots \\ 1/\sqrt{k} & -c_1 & -c_2 & \cdots & c_{k-1} \\ 1/\sqrt{k} & -c_1 & -c_2 & \cdots & -c_{k-1} \end{bmatrix}.$$

- The constraint is equivalent to requiring $\forall j \in [k-1]$,
  1. $Y_{:,j} \in \{c_j(k-j), 0, -c_j\}^n$ and
  2. $Y_{i,\ell} = c_\ell(k-\ell)$ implies $Y_{i,j} = 0, \forall j > \ell$.
- This suggests an algorithm that decides $Y_{:,1}, \cdots, Y_{:,k-1}$ sequentially!

Our approach: detecting $k \geq 2$ conflicting groups

- The objective becomes $\max_{Y \in \mathbb{R}^{n\times(k-1)}\setminus\{\mathbf{0}\}} \frac{Tr(Y^T A Y)}{Tr(Y^T Y)}$ subject to $Y = (XU)_{:,2:}$ and $X_{i,:} \in \{\mathbf{0}, I_{1,:}, \cdots, I_{k,:}\}$, where

$$U = \begin{bmatrix} 1/\sqrt{k} & c_1(k-1) & 0 & \cdots & 0 \\ 1/\sqrt{k} & -c_1 & c_2(k-2) & \cdots & 0 \\ \vdots & \vdots & \cdots & & \vdots \\ 1/\sqrt{k} & -c_1 & -c_2 & \cdots & c_{k-1} \\ 1/\sqrt{k} & -c_1 & -c_2 & \cdots & -c_{k-1} \end{bmatrix}.$$

- The constraint is equivalent to requiring $\forall j \in [k-1]$,

  1. $Y_{:,j} \in \{c_j(k-j), 0, -c_j\}^n$ and
  2. $Y_{i,\ell} = c_\ell(k-\ell)$ implies $Y_{i,j} = 0, \forall j > \ell$.

- Let's finish the rewriting by combining with the meaning that:

$$Y_{i,j} = \begin{cases} c_j(k-j) \text{ implies } i \in S_j, & \forall j \in [k-1] \\ -c_j \text{ implies } i \in S_k, & \text{if } j = k \end{cases}.$$

Our approach: **S**pectral **C**onflicting **G**roups

- The objective becomes $\max_{Y \in \mathbb{R}^{n \times (k-1)} \setminus \{\mathbf{0}\}} \dfrac{Tr(Y^T A Y)}{Tr(Y^T Y)}$ subject to

$$\forall i \in [n], Y_{i,j} = \begin{cases} c_j(k-j), & \text{if } i \in S_j \\ 0, & \text{if } i \in \cup_{h=1}^{j-1} S_h \text{ or } i \notin \cup_{h \in [k]} S_h \\ -c_j, & \text{if } i \in \cup_{h=j+1}^{k} S_h \end{cases}, \forall j \in [k-1],$$

Our approach: **S**pectral **C**onflicting **G**roups

- The objective becomes $\max_{Y \in \mathbb{R}^{n\times(k-1)}\setminus\{\mathbf{0}\}} \frac{Tr(Y^T AY)}{Tr(Y^T Y)}$ subject to

$$\forall i \in [n], Y_{i,j} = \begin{cases} c_j(k-j), & \text{if } i \in S_j \\ 0, & \text{if } i \in \cup_{h=1}^{j-1} S_h \text{ or } i \notin \cup_{h\in[k]} S_h \\ -c_j, & \text{if } i \in \cup_{h=j+1}^{k} S_h \end{cases}, \forall j \in [k-1],$$

- Main Idea: suppose $S_1, \cdots, S_{j-1}$ are determined, find S_j by solving

$$x^* = \operatorname*{argmax}_{x\in\{k-j,0,-1\}^n} \frac{x^T A^{(j-1)} x}{x^T x}. \tag{3}$$

  - Let $A^{(0)} = A$ and $A^{(j-1)}$ results after removing $\cup_{h\in[j-1]} S_h$ from G.
  - After Eq (3) is solved, we know $S_j = \{i : x_i^* = k-j\}$.
  - Repeat the same process to decide the remaining $S_{j+1}, \cdots, S_k$.

Our approach: **S**pectral **C**onflicting **G**roups

- The objective becomes $\max_{Y \in \mathbb{R}^{n \times (k-1)} \setminus \{\mathbf{0}\}} \frac{Tr(Y^T A Y)}{Tr(Y^T Y)}$ subject to

$$\forall i \in [n], Y_{i,j} = \begin{cases} c_j(k-j), & \text{if } i \in S_j \\ 0, & \text{if } i \in \cup_{h=1}^{j-1} S_h \text{ or } i \notin \cup_{h \in [k]} S_h \\ -c_j, & \text{if } i \in \cup_{h=j+1}^{k} S_h \end{cases}, \forall j \in [k-1],$$

- Main Idea: suppose $S_1, \cdots, S_{j-1}$ are determined, find S_j by solving

$$x^* = \underset{x \in \{k-j, 0, -1\}^n}{\operatorname{argmax}} \frac{x^T A^{(j-1)} x}{x^T x}. \tag{3}$$

  - Let $A^{(0)} = A$ and $A^{(j-1)}$ results after removing $\cup_{h \in [j-1]} S_h$ from G.
  - After Eq (3) is solved, we know $S_j = \{i : x_i^* = k - j\}$.
  - Repeat the same process to decide the remaining $S_{j+1}, \cdots, S_k$.

Our approach: **S**pectral **C**onflicting **G**roups

- The objective becomes $\max_{Y \in \mathbb{R}^{n \times (k-1)} \setminus \{\mathbf{0}\}} \dfrac{Tr(Y^T A Y)}{Tr(Y^T Y)}$ subject to

$$\forall i \in [n], Y_{i,j} = \begin{cases} c_j(k-j), & \text{if } i \in S_j \\ 0, & \text{if } i \in \cup_{h=1}^{j-1} S_h \text{ or } i \notin \cup_{h \in [k]} S_h \\ -c_j, & \text{if } i \in \cup_{h=j+1}^{k} S_h \end{cases}, \forall j \in [k-1],$$

- Main Idea: suppose $S_1, \cdots, S_{j-1}$ are determined, find S_j by solving

$$x^* = \operatorname*{argmax}_{x \in \{k-j, 0, -1\}^n} \frac{x^T A^{(j-1)} x}{x^T x}. \quad (3)$$

  - Let $A^{(0)} = A$ and $A^{(j-1)}$ results after removing $\cup_{h \in [j-1]} S_h$ from G.
  - After Eq (3) is solved, we know $S_j = \{i : x_i^* = k - j\}$.
  - Repeat the same process to decide the remaining $S_{j+1}, \cdots, S_k$.

Our approach: **S**pectral **C**onflicting **G**roups

- The objective becomes $\max_{Y \in \mathbb{R}^{n\times(k-1)} \setminus \{\mathbf{0}\}} \dfrac{Tr(Y^T A Y)}{Tr(Y^T Y)}$ subject to

$$\forall i \in [n], Y_{i,j} = \begin{cases} c_j(k-j), & \text{if } i \in S_j \\ 0, & \text{if } i \in \cup_{h=1}^{j-1} S_h \text{ or } i \notin \cup_{h\in[k]} S_h \\ -c_j, & \text{if } i \in \cup_{h=j+1}^{k} S_h \end{cases}, \forall j \in [k-1],$$

- Main Idea: suppose $S_1, \cdots, S_{j-1}$ are determined, find S_j by solving

$$x^* = \operatorname*{argmax}_{x\in\{k-j,0,-1\}^n} \frac{x^T A^{(j-1)} x}{x^T x}. \quad (3)$$

  - Let $A^{(0)} = A$ and $A^{(j-1)}$ results after removing $\cup_{h\in[j-1]} S_h$ from G.
  - After Eq (3) is solved, we know $S_j = \{i : x_i^* = k-j\}$.
  - Repeat the same process to decide the remaining $S_{j+1}, \cdots, S_k$.

Our approach: **S**pectral **C**onflicting **G**roups

Algorithm 1: SCG(A, k)

$A^{(0)} \leftarrow A$;

for $t = 1, \cdots, k-1$ **do**

  $r^{(t)} \leftarrow$ Solve-Max-DRQ($A^{(t-1)}, k-t$) **if** $t < k-1$ **then**

    $S_t \leftarrow \{i \notin \cup_{j=1}^{t-1} S_j : |r_i^{(t)}| = (k-t)\}$;

    $A^{(t)} \leftarrow A^{(t-1)}$;

    $A_{i,:}^{(t)} \leftarrow 0_{1\times n}$ and $A_{:,i}^{(t)} \leftarrow 0_{n\times 1}$ for all $i \in S_t$

  else $S_{k-1} \leftarrow \{i \notin \cup_{j=1}^{t-1} S_j : r_i^{(t)} = 1\}$ and

   $S_k \leftarrow \{i \notin \cup_{j=1}^{t-1} S_j : r_i^{(t)} = -1\}$;

end

return $S_1, \ldots, S_k$;

Our approach: solving Max-DRQ problem

$$x^* = \underset{x \in \{k-j, 0, -1\}^n}{\operatorname{argmax}} \frac{x^T A^{(j-1)} x}{x^T x}. \tag{3}$$

- APX-Hard [2] for $k = 2$ and practical $\mathcal{O}(n^{1/2})$-approx by 2-PC [3].

Our approach: solving Max-DRQ problem

$$x^* = \underset{x \in \{k-j,0,-1\}^n}{\operatorname{argmax}} \frac{x^T A^{(j-1)} x}{x^T x}. \tag{3}$$

- APX-Hard [2] for $k = 2$ and practical $\mathcal{O}(n^{1/2})$-approx by 2-PC [3].
- Our approach is based on rounding the leading eigenvector of $A^{(j-1)}$ to a vector in $\{k - j, 0, -1\}^n$.

Algorithm 1: Solve-Max-DRQ(A, q)

Input : Square and symmetric matrix A, and positive integer q.
Output: The rounded vector $r \in \{0, -1, q\}^n$.

```
v ← the leading eigenvector of A;
(d1, r1) ← Round(v, q) ;        // d1 = sin θ(v, r1)
(d2, r2) ← Round(−v, q) ;       // d2 = sin θ(v, r2)
if d1 ≤ d2 then r ← r1;
else r ← r2;
return r ;
```

Deterministic Rounding: Minimum Angle (MA)

- Rounding v to $r^* \in \operatorname{argmin}_{u \in \{q,0,-1\}^n} \sin\theta(v,u)$ takes $\mathcal{O}(n^2)$ time.
- For practical consideration, an $\mathcal{O}(n)$ algorithm is implemented.

Algorithm 2: MA(v, q)

$\{i_k\}_{k=1}^n \leftarrow$ Sort v and return the indexes such that $v_{i_1} \geq \cdots \geq v_{i_n}$;
$(d, u^*) \leftarrow (\infty, 0)$;
$(k_1, k_2) \leftarrow (0, n+1)$;
while $k_1 < k_2$ **do**
  $u_1 \leftarrow$ set the i_{k_1+1}-th element of u^* to q;
  $u_2 \leftarrow$ set the i_{k_2-1}-th element of u^* to -1;
  if $\min\{\sin\theta(v,u_1), \sin\theta(v,u_2)\} \geq d$ **then** break;
  if $\sin\theta(v,u_1) < \sin\theta(v,u_2)$ **then**
   $(k_1, d, u^*) \leftarrow (k_1+1, \sin\theta(v,u_1), u_1$;
  else $(k_2, d, u^*) \leftarrow (k_2-1, \sin\theta(v,u_2), u_2)$;
end
return (d, u^*);

Randomized Rounding (R)

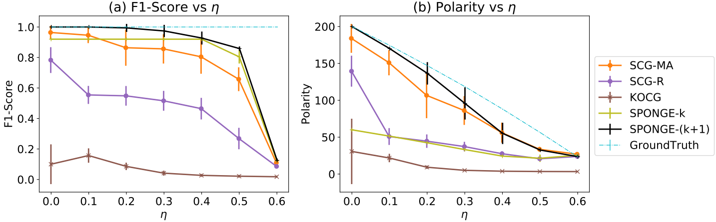
- Generalize the randomized approach of 2-PC [3].
- Round v to r by setting $r_i = \begin{cases} q, & \text{w.p. } |v_i|/q \\ -1, & \text{w.p. } |v_i| \end{cases}$.
- It gives a $\mathcal{O}(qn^{1/2})$-approx to the Max-DRQ problem, which is tight upto a factor of q.

Experiment Results

- Real-world networks:

	Bitcoin	WikiVote	Referendum	Slashdot	WikiConflict	Epinions	Wikipolitics
$\|V\|$	5 881	7 115	10 884	82 140	116 717	131 580	138 587
$\|E\|$	21 492	100 693	251 406	500 481	2 026 646	711 210	715 883
$\|E_-\|/\|E\|$	0.2	0.2	0.1	0.2	0.6	0.2	0.1
SCG-MA	**14.6**	**45.5**	**84.9**	**37.8**	**102.6**	**88.8**	**57.5**
SCG-R	5.0	9.7	39.8	7.3	16.2	39.4	5.5
KOCG [4]	4.4	5.5	8.8	2.6	4.5	8.7	4.8
SPONGE-k [5]	5.0	15.8	41.5	—	—	—	—
SPONGE-(k+1) [5]	0.8	1.0	1.0	—	—	—	—

- Synthetic:



Summary

- Contributions:
  - Connecting the EVD of $L_k = U\text{diag}([0, k, \cdots, k])U^T$ to the characterization of the conflicting groups in signed network.
  - Generalizing 2-PC[3] with provable guarantee to $k \geq 2$.
- Future works:
  - Is it possible to improve $\mathcal{O}(n^{1/2})$-approx by other approach?
  - Detecting conflicting groups in memory-limited setting or dynamic networks.

Reference I

Nikhil Bansal, Avrim Blum, and Shuchi Chawla.
Correlation clustering.
Machine learning, 2004.

Aditya Bhaskara, Moses Charikar, Rajsekar Manokaran, and Aravindan Vijayaraghavan.
On quadratic programming with a ratio objective.
In *Proc. of ICALP*. Springer, 2012.

Francesco Bonchi, Edoardo Galimberti, Aristides Gionis, Bruno Ordozgoiti, and Giancarlo Ruffo.
Discovering polarized communities in signed networks.
In *Proc. of CIKM*, 2019.

Lingyang Chu, Zhefeng Wang, Jian Pei, Jiannan Wang, Zijin Zhao, and Enhong Chen.
Finding gangs in war from signed networks.
In *Proc. of KDD*, 2016.

Reference II

Mihai Cucuringu, Peter Davies, Aldo Glielmo, and Hemant Tyagi.
Sponge: A generalized eigenproblem for clustering signed networks.
In *Proc. of AISTATS*, 2019.