Mining signed networks: theory and applications

Lecture in course “Social networks and online markets”

Sapienza, Wednesday, April 8, 2024

Aristides Gionis, KTH Royal Institute of Technology, Sweden

argioni@kth.se

outline

introduction

theory of signed networks

problems and applications

- subgraph mining
- correlation clustering

conclusions

introduction

signed networks

graphs with edge signs

either *positive* or *negative*

signed networks: motivation

human interactions

friendly or antagonistic

Image source: pxfuel.com

signed networks: motivation

online social media

- X, facebook, etc.
- users may like or dislike the content of each other
- can be used to study online polarization

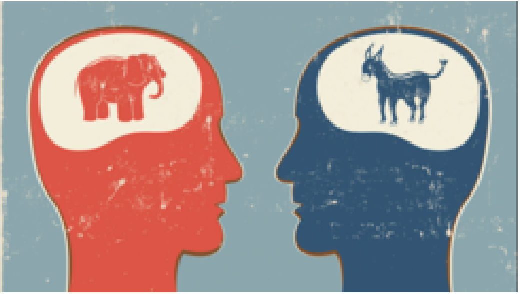
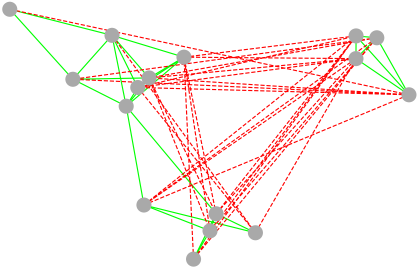


Image source: iStockphoto.com

signed networks: motivation

groups of humans

- examples: tribes, political parties, countries, etc.
- relations of countries during war



New Guinea highland tribes graph Read (1954)

signed networks: motivation

human language

- graph between words that captures synonyms / antonyms

"happy"

Image source: thesaurus.com

signed networks: motivation

molecular biology

- graph between proteins
- one protein activates or inhibits the function of another

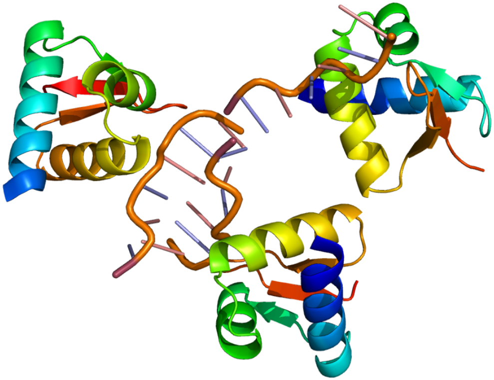


Image source: commons.wikimedia.org

signed networks: motivation

finance

- graph between securities (tradable assets)
- a security *correlates* positively / negatively with another
- “correlate” means the joint movement of price

Image source: vecteezy.com

theory of signed networks

outline

we will discuss:

- balance
- spectrum

signed networks

signed networks (or graphs): each edge labeled $+$ or $-$

definitions:

- $G = (V, E^+, E^-)$,
- $G = (V, E, \sigma), \quad \sigma : E \to \{-, +\}$

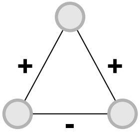
signed networks

signed networks (or graphs): each edge labeled $+$ or $-$

definitions:

- $G = (V, E^+, E^-)$,
- $G = (V, E, \sigma), \quad \sigma : E \to \{-, +\}$

adjacency matrix: $A = A_{E^+} - A_{E^-}$

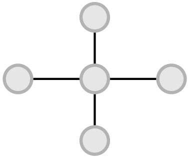


$$\mapsto \begin{pmatrix} 0 & 1 & 1 \\ 1 & 0 & -1 \\ 1 & -1 & 0 \end{pmatrix}$$

expressiveness of signed graphs

signed networks can be quite expressive

example: star graph



- number of possible graphs: $2^{|E|}$
- number of non-isomorphic graphs: $|E|$

differences in signed networks

shortest paths

signed networks can be quite different . . . consider e.g., shortest paths;

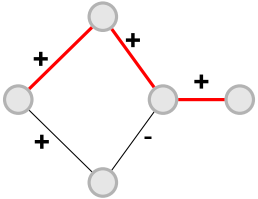
how do we even define path length in signed networks?

differences in signed networks

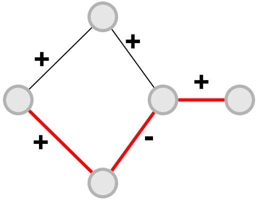
shortest paths

signed networks can be quite different . . . consider e.g., shortest paths;
how do we even define path length in signed networks?

proposal: distinguish positive and negative paths (by product of edge signs)



positive path



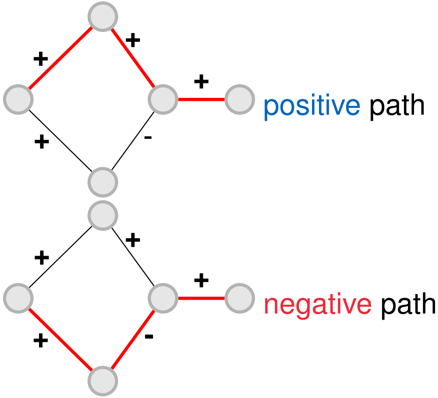
negative path

differences in signed networks

shortest paths

signed networks can be quite different . . . consider e.g., shortest paths;
how do we even define path length in signed networks?

proposal: distinguish positive and negative paths (by product of edge signs)



finding shortest simple signed paths between a source and all other vertices is **NP**-complete problem

if repetitions are allowed, problem can be solved in time $\mathcal{O}(|E| \log \log \frac{D}{d})$

(Hansen, 1984)

differences in signed networks

densest subgraph

densest subgraph problem in unsigned graphs:

$$\max_{x \in \{0,1\}^n} \frac{x^T A x}{x^T x}$$

polynomial-time solvable (Goldberg, 1984)

differences in signed networks

densest subgraph

densest subgraph problem in unsigned graphs:

$$\max_{x \in \{0,1\}^n} \frac{x^T A x}{x^T x}$$

polynomial-time solvable (Goldberg, 1984)

densest subgraph problem in signed graphs:

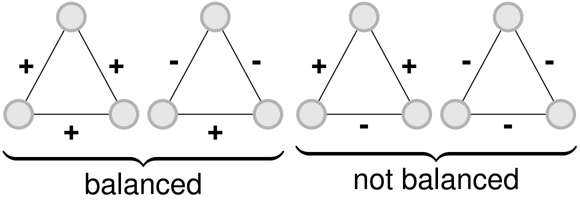
$$\max_{x \in \{-1,0,1\}^n} \frac{x^T A x}{x^T x}$$

NP-hard ! (Bonchi et al., 2019; Tsourakakis et al., 2019) balance

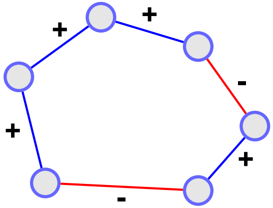
motivation

balance in social networks (Harary, 1953)

"*The friend of a friend is a friend*" (or "*the enemy of a friend is an enemy*").



the four possible non-isomorphic signed triangles balance applies to cycles of any length



$$(+) \times (-) \times (+) \times (-) \times (+) \times (+) = +$$

definition of balanced cycle

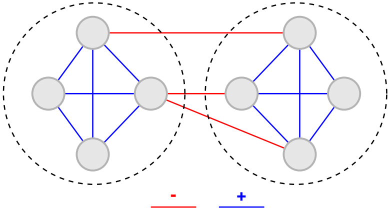
a cycle is balanced if the product of its signs is positive

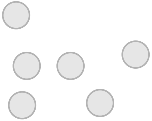
characterizations of balance

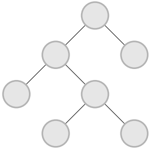
a graph G is balanced if and only if

- there are no negative (unbalanced) cycles

some balanced graphs





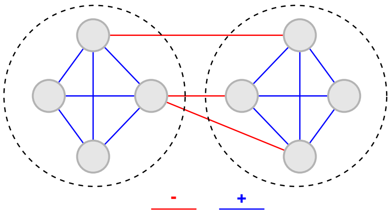


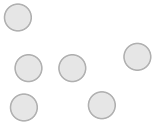
characterizations of balance

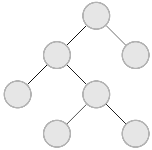
a graph G is balanced if and only if

- there are no negative (unbalanced) cycles
- there exists a sign-compliant partition: $V = V_1 \cup V_2$ such that all + edges are within sets and all - edges are between sets

some balanced graphs





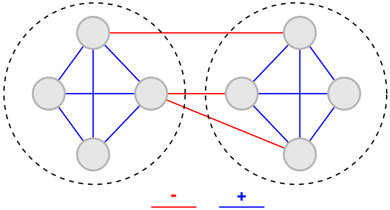


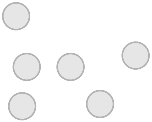
characterizations of balance

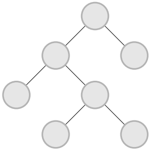
a graph G is balanced if and only if

- there are no negative (unbalanced) cycles
- there exists a sign-compliant partition: $V = V_1 \cup V_2$ such that all + edges are within sets and all - edges are between sets
- all paths between any pair u, v have same sign

some balanced graphs







measures of partial balance

how can we measure partial balance?

- fraction of balanced cycles
  (Cartwright and Harary, 1956; Giscard et al., 2017)
  - fraction of balanced triangles
    (Terzi and Winkler, 2011) (example in next slide)
- spectral methods (discussed later on)

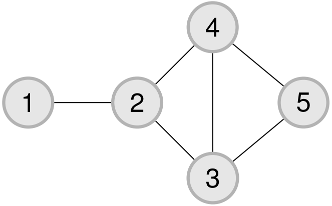
check Aref and Wilson (2018) for an overview of partial measures of balance

measures of partial balance

example: fraction of balanced triangles

reminder: counting triangles in unsigned graphs

$$A = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 \end{pmatrix}$$

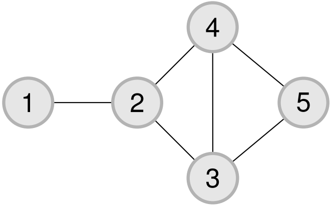


measures of partial balance

example: fraction of balanced triangles

reminder: counting triangles in unsigned graphs

$$A = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 \end{pmatrix}, \; A^2 = \begin{pmatrix} 1 & 0 & 1 & 1 & 0 \\ 0 & 3 & 1 & 1 & 2 \\ 1 & 1 & 3 & 2 & 1 \\ 1 & 1 & 2 & 3 & 1 \\ 0 & 2 & 1 & 1 & 2 \end{pmatrix}$$

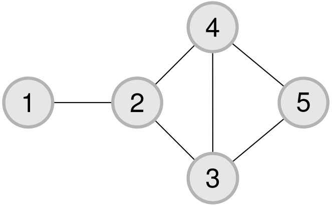


measures of partial balance

example: fraction of balanced triangles

reminder: counting triangles in unsigned graphs

$$A = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 \end{pmatrix}, \; A^2 = \begin{pmatrix} 1 & 0 & 1 & 1 & 0 \\ 0 & 3 & 1 & 1 & 2 \\ 1 & 1 & 3 & 2 & 1 \\ 1 & 1 & 2 & 3 & 1 \\ 0 & 2 & 1 & 1 & 2 \end{pmatrix}, \; A^3 = \begin{pmatrix} 0 & 3 & 1 & 1 & 2 \\ 3 & 2 & 6 & 6 & 2 \\ 1 & 6 & 4 & 5 & 5 \\ 1 & 6 & 5 & 4 & 5 \\ 2 & 2 & 5 & 5 & 2 \end{pmatrix}$$

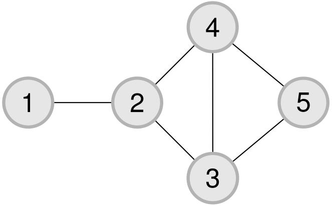


measures of partial balance

example: fraction of balanced triangles

reminder: counting triangles in unsigned graphs

$$A = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 \end{pmatrix}, \; A^2 = \begin{pmatrix} 1 & 0 & 1 & 1 & 0 \\ 0 & 3 & 1 & 1 & 2 \\ 1 & 1 & 3 & 2 & 1 \\ 1 & 1 & 2 & 3 & 1 \\ 0 & 2 & 1 & 1 & 2 \end{pmatrix}, \; A^3 = \begin{pmatrix} 0 & 3 & 1 & 1 & 2 \\ 3 & 2 & 6 & 6 & 2 \\ 1 & 6 & 4 & 5 & 5 \\ 1 & 6 & 5 & 4 & 5 \\ 2 & 2 & 5 & 5 & 2 \end{pmatrix}$$



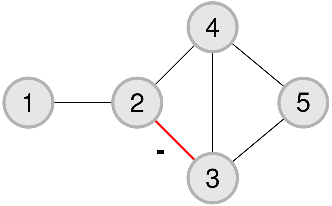
$A^3_{ii} = 2 \times \#(\text{3-cycles adjacent to vertex } i)$

neasures of partial balance

example: fraction of balanced triangles

counting triangles in signed graphs:

$$A = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & -1 & 1 & 0 \\ 0 & -1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 \end{pmatrix}$$

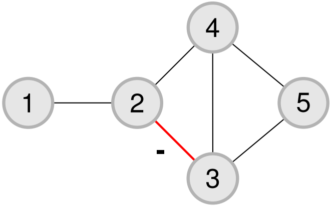


neasures of partial balance

example: fraction of balanced triangles

counting triangles in signed graphs:

$$A = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & -1 & 1 & 0 \\ 0 & -1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 \end{pmatrix}, \; A^2 = \begin{pmatrix} 1 & 0 & -1 & 1 & 0 \\ 0 & 3 & 1 & -1 & 0 \\ -1 & 1 & 3 & 0 & 1 \\ 1 & -1 & 0 & 3 & 1 \\ 0 & 0 & 1 & 1 & 2 \end{pmatrix}$$

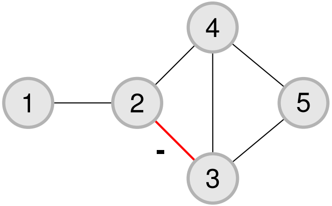


neasures of partial balance

example: fraction of balanced triangles

counting triangles in signed graphs:

$$A = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & -1 & 1 & 0 \\ 0 & -1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 \end{pmatrix}, \ A^2 = \begin{pmatrix} 1 & 0 & -1 & 1 & 0 \\ 0 & 3 & 1 & -1 & 0 \\ -1 & 1 & 3 & 0 & 1 \\ 1 & -1 & 0 & 3 & 1 \\ 0 & 0 & 1 & 1 & 2 \end{pmatrix}, \ A^3 = \begin{pmatrix} 0 & 3 & 1 & -1 & 0 \\ 3 & -2 & -4 & 4 & 0 \\ 1 & -4 & 0 & 5 & 3 \\ -1 & 4 & 5 & 0 & 3 \\ 0 & 0 & 3 & 3 & 2 \end{pmatrix}$$

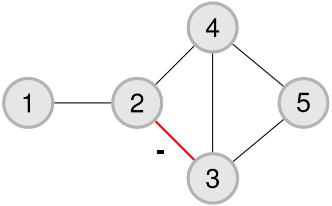


neasures of partial balance

example: fraction of balanced triangles

counting triangles in signed graphs:

$$A = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & -1 & 1 & 0 \\ 0 & -1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 \end{pmatrix}, \quad A^2 = \begin{pmatrix} 1 & 0 & -1 & 1 & 0 \\ 0 & 3 & 1 & -1 & 0 \\ -1 & 1 & 3 & 0 & 1 \\ 1 & -1 & 0 & 3 & 1 \\ 0 & 0 & 1 & 1 & 2 \end{pmatrix}, \quad A^3 = \begin{pmatrix} 0 & 3 & 1 & -1 & 0 \\ 3 & -2 & -4 & 4 & 0 \\ 1 & -4 & 0 & 5 & 3 \\ -1 & 4 & 5 & 0 & 3 \\ 0 & 0 & 3 & 3 & 2 \end{pmatrix}$$



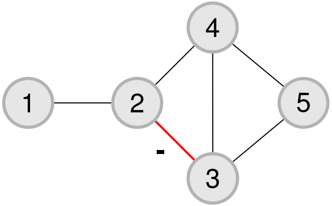
$A^3_{ii} = 2 \times (\#\text{balanced 3-cycles} - \#\text{unbalanced 3-cyles}),$

neasures of partial balance

example: fraction of balanced triangles

counting triangles in signed graphs:

$$A = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & -1 & 1 & 0 \\ 0 & -1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 \end{pmatrix}, \quad A^2 = \begin{pmatrix} 1 & 0 & -1 & 1 & 0 \\ 0 & 3 & 1 & -1 & 0 \\ -1 & 1 & 3 & 0 & 1 \\ 1 & -1 & 0 & 3 & 1 \\ 0 & 0 & 1 & 1 & 2 \end{pmatrix}, \quad A^3 = \begin{pmatrix} 0 & 3 & 1 & -1 & 0 \\ 3 & -2 & -4 & 4 & 0 \\ 1 & -4 & 0 & 5 & 3 \\ -1 & 4 & 5 & 0 & 3 \\ 0 & 0 & 3 & 3 & 2 \end{pmatrix}$$



$A^3_{ii} = 2 \times (\#\text{balanced 3-cycles} - \#\text{unbalanced 3-cyles})$, thus,

$$\frac{Tr(A^3) + Tr(|A|^3)}{2\,Tr(|A|^3)} = \text{fraction of balanced triangles}$$

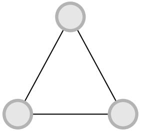
(Terzi and Winkler, 2011)

note: $|A|$ is the adj. matrix of the *underlying* (unsigned) graph spectrum

spectral theory

review of unsigned spectral theory:

Laplacian: $L = D - A$

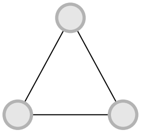


$$L = \begin{pmatrix} 2 & -1 & -1 \\ -1 & 2 & -1 \\ -1 & -1 & 2 \end{pmatrix}$$

spectral theory

review of unsigned spectral theory:

Laplacian: $L = D - A$



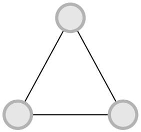
$$L\mathbf{v}_1 = \begin{pmatrix} 2 & -1 & -1 \\ -1 & 2 & -1 \\ -1 & -1 & 2 \end{pmatrix} \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} = 0$$

- $\lambda_{min}(L) = 0$ (multiplicity of 0 = number of connected components)

spectral theory

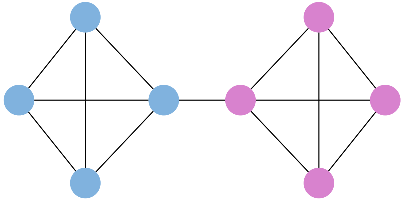
review of unsigned spectral theory:

Laplacian: $L = D - A$



$$L\mathbf{v}_1 = \begin{pmatrix} 2 & -1 & -1 \\ -1 & 2 & -1 \\ -1 & -1 & 2 \end{pmatrix} \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} = 0$$

- $\lambda_{min}(L) = 0$ (multiplicity of 0 = number of connected components)
- eigenvector $\mathbf{v}_2$ gives a “good” partition   (Cheeger inequality)



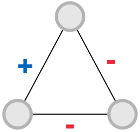
$$\mathbf{v}_2 \approx \begin{pmatrix} -0.38 \\ -0.38 \\ -0.38 \\ -0.25 \\ 0.25 \\ 0.38 \\ 0.38 \\ 0.38 \end{pmatrix}, \quad \lambda_2(L) \approx 0.35.$$

spectral theory

signed spectral theory:

Laplacian: $L = D - A$

unsigned	signed
L is positive semidefinite	
$D_{ii} = \sum_j A_{ij}$	$D_{ii} = \sum_j \lvert A_{ij} \rvert$
$\lambda_{min}(L) = 0$	$\lambda_{min}(L) \geq 0$



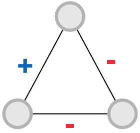
$$L = \begin{pmatrix} 2 & -1 & 1 \\ -1 & 2 & 1 \\ 1 & 1 & 2 \end{pmatrix}$$

spectral theory

signed spectral theory:

Laplacian: $L = D - A$

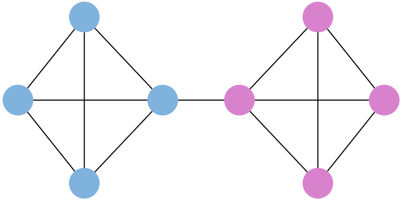
unsigned	signed
L is positive semidefinite	
$D_{ii} = \sum_j A_{ij}$	$D_{ii} = \sum_j \lvert A_{ij} \rvert$
$\lambda_{min}(L) = 0$	$\lambda_{min}(L) \geq 0$



$$L\mathbf{v}_1 = \begin{pmatrix} 2 & -1 & 1 \\ -1 & 2 & 1 \\ 1 & 1 & 2 \end{pmatrix} \begin{pmatrix} 1 \\ 1 \\ -1 \end{pmatrix} = 0$$

spectral theory

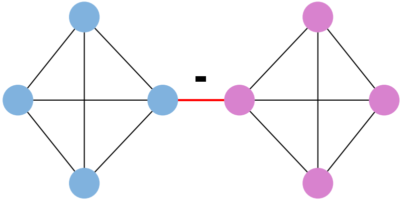
consider previous graph;



$$\mathbf{v}_1 = \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{pmatrix} \qquad \lambda_{min}(L) = 0$$

spectral theory

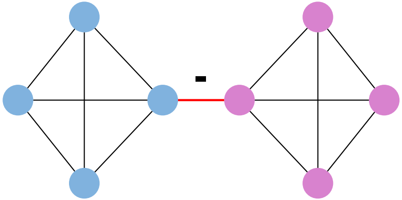
consider previous graph; flip sign of one edge:



$$\mathbf{v}_1 = \begin{pmatrix} -1 \\ -1 \\ -1 \\ -1 \\ 1 \\ 1 \\ 1 \\ 1 \end{pmatrix} \qquad \lambda_{min}(L) = 0$$

spectral theory

consider previous graph; flip sign of one edge:



$$\mathbf{v}_1 = \begin{pmatrix} -1 \\ -1 \\ -1 \\ -1 \\ 1 \\ 1 \\ 1 \\ 1 \end{pmatrix} \qquad \lambda_{min}(L) = 0$$

This graph is balanced !

spectral characterizations of balance

1. connected and $\lambda_{min} = 0$ (or one zero-eigenvalue per connected component)

spectral theory

a taste of spectral analysis:

lemma (Hou et al., 2003)

$\lambda_{max}(L(G)) \leq \lambda_{max}(L(G^-))$, where G^- is the all-negative graph

spectral theory

a taste of spectral analysis:

lemma (Hou et al., 2003)

$\lambda_{max}(L(G)) \leq \lambda_{max}(L(G^-))$, where G^- is the all-negative graph

the Laplacian $L(G^-)$ has all non-negative entries; so,

$\mathbf{x}^T L(G)\, \mathbf{x} =$

spectral theory

a taste of spectral analysis:

lemma (Hou et al., 2003)

$\lambda_{max}(L(G)) \leq \lambda_{max}(L(G^-))$, where G^- is the all-negative graph

the Laplacian $L(G^-)$ has all non-negative entries; so,

$$\mathbf{x}^T L(G)\,\mathbf{x} = \sum_{(v_i,v_j)\in E}(x_i - \sigma(v_i, v_j)x_j)^2$$

spectral theory

a taste of spectral analysis:

lemma (Hou et al., 2003)

$\lambda_{max}(L(G)) \leq \lambda_{max}(L(G^-))$, where G^- is the all-negative graph

the Laplacian $L(G^-)$ has all non-negative entries; so,

$$\mathbf{x}^T L(G)\,\mathbf{x} = \sum_{(v_i,v_j)\in E}(x_i - \sigma(v_i,v_j)x_j)^2 \leq \sum_{(v_i,v_j)\in E}(|x_i| + |x_j|)^2 = \mathbf{x}^T L(G^-)\,\mathbf{x}$$

spectral theory

a taste of spectral analysis:

lemma (Hou et al., 2003)

$\lambda_{max}(L(G)) \leq \lambda_{max}(L(G^-))$, where G^- is the all-negative graph

the Laplacian $L(G^-)$ has all non-negative entries; so,

$$\mathbf{x}^T L(G)\,\mathbf{x} = \sum_{(v_i,v_j)\in E}(x_i - \sigma(v_i, v_j)x_j)^2 \leq \sum_{(v_i,v_j)\in E}(|x_i| + |x_j|)^2 = \mathbf{x}^T L(G^-)\,\mathbf{x}$$

lemma (Hou et al., 2003)

$\lambda_{max}(L(G)) \leq 2(n-1)$, where n is the number of vertices

spectral theory

a taste of spectral analysis:

lemma (Hou et al., 2003)

$\lambda_{max}(L(G)) \leq \lambda_{max}(L(G^-))$, where G^- is the all-negative graph

the Laplacian $L(G^-)$ has all non-negative entries; so,

$$\mathbf{x}^T L(G)\,\mathbf{x} = \sum_{(v_i,v_j)\in E}(x_i - \sigma(v_i, v_j)x_j)^2 \leq \sum_{(v_i,v_j)\in E}(|x_i| + |x_j|)^2 = \mathbf{x}^T L(G^-)\,\mathbf{x}$$

lemma (Hou et al., 2003)

$\lambda_{max}(L(G)) \leq 2(n-1)$, where n is the number of vertices

$$\lambda_{max}(G) = \lambda_{max}(D - A) \leq \lambda_{max}(D_G) + \lambda_{max}(-A_G) \leq n - 1 + n - 1$$

problems and applications

outline

subgraph mining

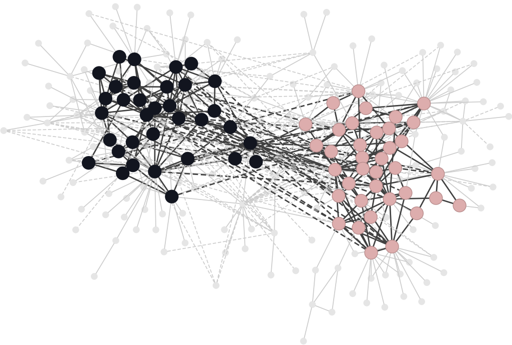
subgraph mining

goal

find interesting subgraphs in a signed networks

some definitions of "interesting":

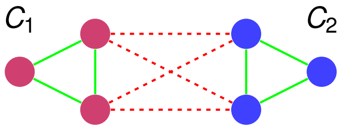
- balanced subgraph
- polarized subgraph



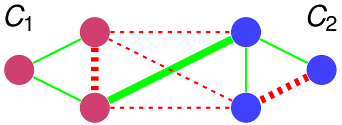
US Congress network (Bonchi et al., 2019)

subgraph mining: balanced graphs vs. polarized graphs

balanced graphs

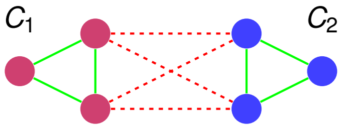


polarized graphs:
"noisy" edges are allowed

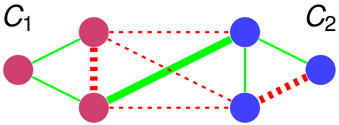


subgraph mining: balanced graphs vs. polarized graphs

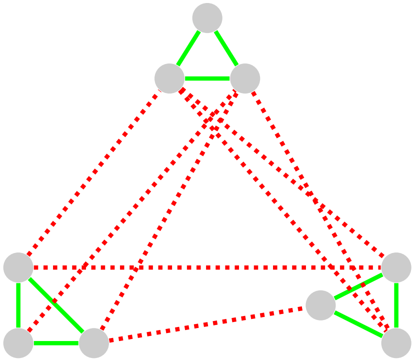
balanced graphs



polarized graphs:
"noisy" edges are allowed



polarized graphs:
more than two groups

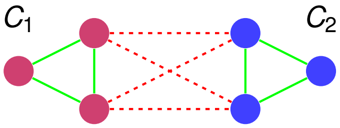


maximum balanced subgraph (MBS) problem

problem definition

input: a signed graph $G = (V, E^+, E^-)$

output: a maximum-cardinality vertex subset $U \subseteq V$ such that $G(U)$ is balanced

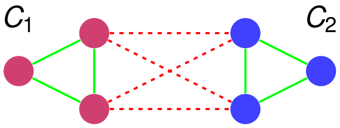


a balanced graph

maximum balanced subgraph (MBS) problem

problem definition

input: a signed graph $G = (V, E^+, E^-)$
output: a maximum-cardinality vertex subset $U \subseteq V$ such that $G(U)$ is balanced



a balanced graph

- an equivalent problem: remove the minimum number of vertices such that the remaining graph is balanced
- solution size of MBS $=$ frustration index
- edge-version of MBS: a balanced subgraph with maximum number of edges
- all these problems are **NP**-hard

spanning-tree heuristic for MBS

notation

- negative graph G^-: induced subgraph on the negative edges in G
- positive graph G^+: induced subgraph on the positive edges in G
- $I(G)$: any maximal independent set of G

spanning-tree heuristic for MBS

notation

- negative graph G^-: induced subgraph on the negative edges in G
- positive graph G^+: induced subgraph on the positive edges in G
- $I(G)$: any maximal independent set of G

high-level idea (Gülpinar et al., 2004)

1. find a spanning tree T on G
2. find a switch W such that T^W is all positive
3. switch G by W, yielding G^W
4. return $I\left(G^W\right)^-$

spanning-tree heuristic: maximal independent set on G^-

intuition 1

any maximal independent set on G^- is balanced in G

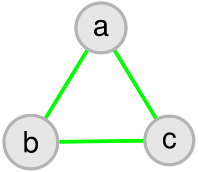
G G^- $\mathrm{I}(G^-)$

spanning-tree heuristic: maximal independent set on G^-

intuition 1

any maximal independent set on G^- is balanced in G

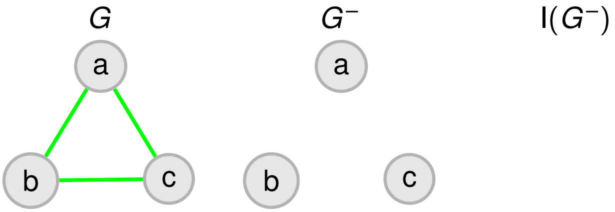
G $\qquad G^- \qquad I(G^-)$



spanning-tree heuristic: maximal independent set on G^-

intuition 1

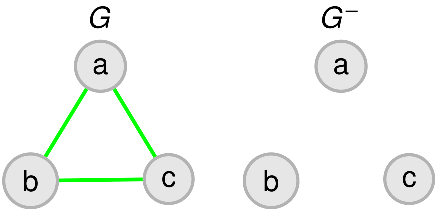
any maximal independent set on G^- is balanced in G



spanning-tree heuristic: maximal independent set on G^-

intuition 1

any maximal independent set on G^- is balanced in G



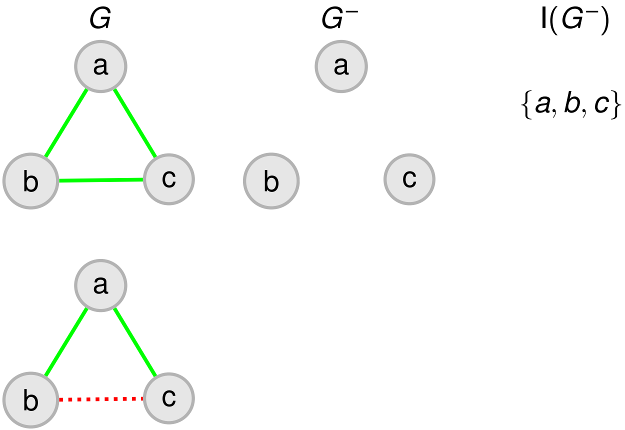
$I(G^-)$

$\{a, b, c\}$

spanning-tree heuristic: maximal independent set on G^-

intuition 1

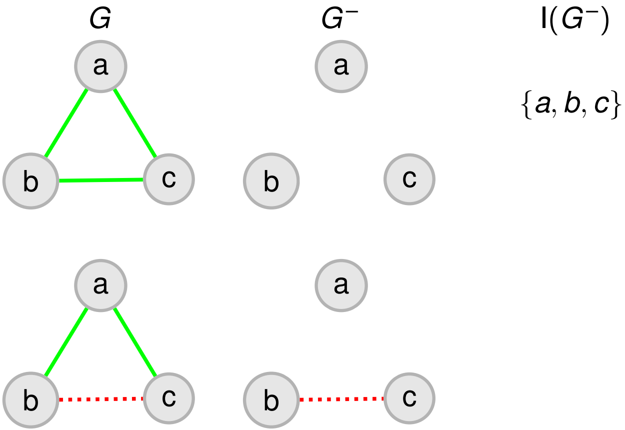
any maximal independent set on G^- is balanced in G



spanning-tree heuristic: maximal independent set on G^-

intuition 1

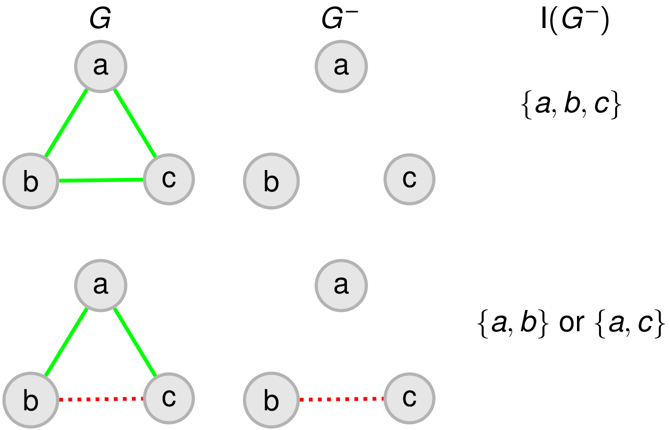
any maximal independent set on G^- is balanced in G



spanning-tree heuristic: maximal independent set on G^-

intuition 1

any maximal independent set on G^- is balanced in G



spanning-tree heuristic: maximal independent set on G^-

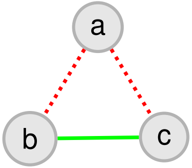
quiz: can we solve MBS optimally by maximizing $|I(G^-)|$?

spanning-tree heuristic: maximal independent set on G^-

quiz: can we solve MBS optimally by maximizing $|I(G^-)|$?

no! a counter-example:

G $\quad$ G^- $\quad$ $I(G^-)$ of maximum size

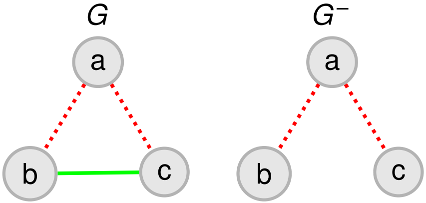


Expected solution: $\{a, b, c\}$

spanning-tree heuristic: maximal independent set on G^-

quiz: can we solve MBS optimally by maximizing $|I(G^-)|$?

no! a counter-example:



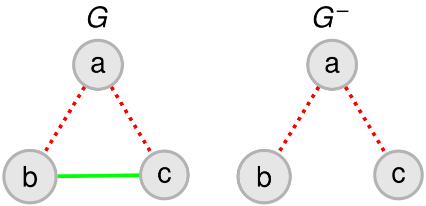
$I(G^-)$ of maximum size

Expected solution: $\{a, b, c\}$

spanning-tree heuristic: maximal independent set on G^-

quiz: can we solve MBS optimally by maximizing $|I(G^-)|$?

no! a counter-example:



I(G^-) of maximum size

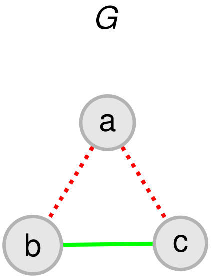
$\{b, c\}$ ✗

Expected solution: $\{a, b, c\}$

spanning-tree heuristic: switch

intuition 2

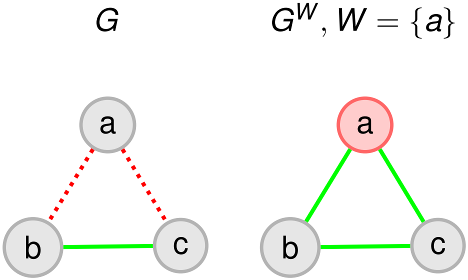
switch G to expand size of $I(G^{-})$



spanning-tree heuristic: switch

intuition 2

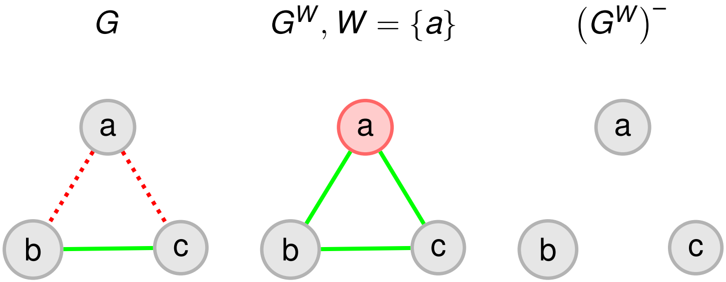
switch G to expand size of $I(G^-)$



spanning-tree heuristic: switch

intuition 2

switch G to expand size of $I(G^-)$

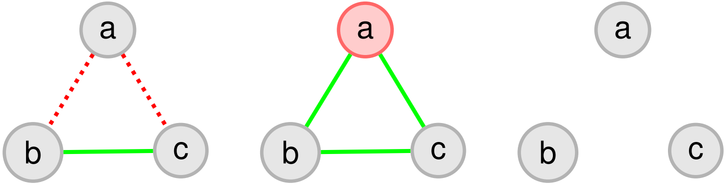


spanning-tree heuristic: switch

intuition 2

switch G to expand size of $I(G^-)$

G | G^W, $W = \{a\}$ | $(G^W)^-$ | $I(G^-)$ of maximum size



$\{a, b, c\}$ ✓

spanning-tree heuristic: combining the previous ideas

an equivalent form of MBS

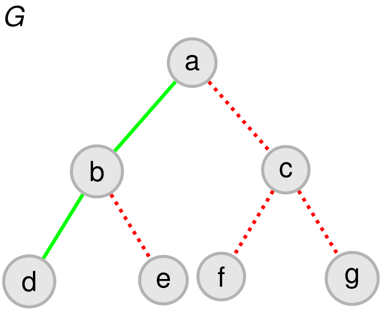
find a switch W sutch that $\left|I\left(\left(G^W\right)^-\right)\right|$ is maximized   an **NP**-hard problem

spanning-tree heuristic: combining the previous ideas

an equivalent form of MBS

find a switch W sutch that $\left|I\left(\left(G^W\right)^-\right)\right|$ is maximized — an **NP**-hard problem

a tree is always balanced, i.e., there exists some W such that T^W is all positive



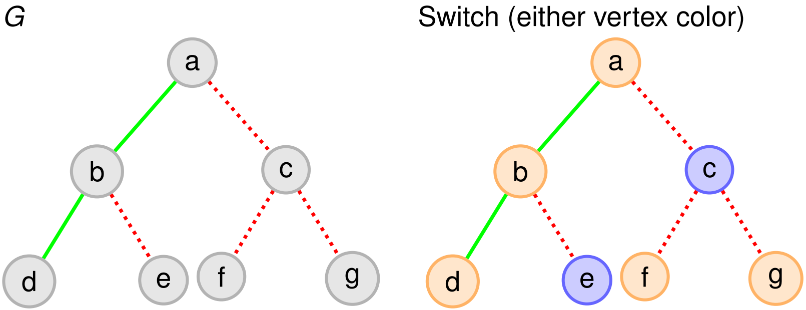
quiz: How to find a switch that makes a tree all positive? Hint: use BFS

spanning-tree heuristic: combining the previous ideas

an equivalent form of MBS

find a switch W sutch that $\left|I\left(\left(G^W\right)^-\right)\right|$ is maximized — an **NP**-hard problem

a tree is always balanced, i.e., there exists some W such that T^W is all positive



quiz: How to find a switch that makes a tree all positive? Hint: use BFS

spanning-tree heuristic for MBS

algorithm (Gülpinar et al., 2004)

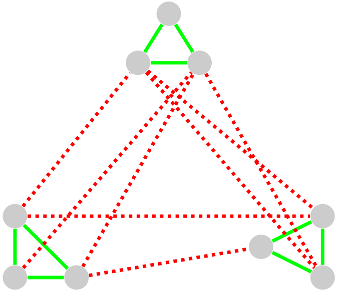
1. find a spanning tree T on G — # a tree is an easy case to solve
2. find a switch W that makes T^W all positive — # expands the solution size
3. use W to switch G, yielding G^W
4. return maximal independent set on $\left(G^W\right)^-$ — # $I\left(G^W\right)^-$ is balanced

polarized subgraph detection

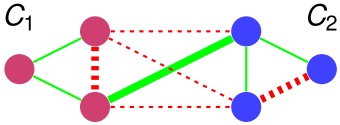
polarized subgraphs as an extension of balanced subgraphs

- can have more than two components
- permits the presence of noisy edges:

  positive edges between C_1 and C_2
  negative edges within C_1 or C_2



more than two components



with "noisy" edges (drawn in thick lines)

polarized subgraph detection: problem dimensions

- what measure of polarization?
- how many groups inside a polarized subgraph?
  - 2-way or *k*-way polarized subgraph?
- how many polarized subgraphs to find: one or multiple?
- are seed nodes given? local or global community detection?

polarized subgraph detection: problem dimensions

- what measure of polarization?
- how many groups inside a polarized subgraph?
  - 2-way or k-way polarized subgraph?
- how many polarized subgraphs to find: one or multiple?
- are seed nodes given? local or global community detection?

Paper	num. groups	num. subgraphs	local / global	approximation guarantee
Chu et al. (2016)	k	≥ 1	global	-
Bonchi et al. (2019)	2	1	global	$\sqrt{n}$
Xiao et al. (2020)	2	≥ 1	local	$\sqrt{OPT}$

polarized subgraph detection: single 2-way subgraph

discovering polarized communities in signed networks (Bonchi et al., 2019)

- intuition of the polarization measure:
  1. in each group, many positive edges
  2. between two groups, many negative edges
  3. the subgraph is dense in terms of the number of nodes

polarized subgraph detection: single 2-way subgraph

discovering polarized communities in signed networks (Bonchi et al., 2019)

- intuition of the polarization measure:
  1. in each group, many positive edges
  2. between two groups, many negative edges
  3. the subgraph is dense in terms of the number of nodes

- objective in matrix form:

$$\max_{\mathbf{x}} \frac{\mathbf{x}^T A \mathbf{x}}{\mathbf{x}^T \mathbf{x}}$$

(**NP**-hard problem)

where $\mathbf{x} \in \{-1, 0, 1\}^n$ is used to encode the subgraph

polarized subgraph detection: single 2-way subgraph

discovering polarized communities in signed networks (Bonchi et al., 2019)

- intuition of the polarization measure:
  1. in each group, many positive edges
  2. between two groups, many negative edges
  3. the subgraph is dense in terms of the number of nodes
- objective in matrix form:

$$\max_{\mathbf{x}} \frac{\mathbf{x}^T A \mathbf{x}}{\mathbf{x}^T \mathbf{x}}$$

(**NP**-hard problem)

  where $\mathbf{x} \in \{-1, 0, 1\}^n$ is used to encode the subgraph
- spectral algorithm:
  - relax x to be continuous
  - the relaxed problem is solved by finding the leading eigenvector
  - randomized $\sqrt{n}$-approximation based on rounding the leading eigenvector

outline

correlation clustering

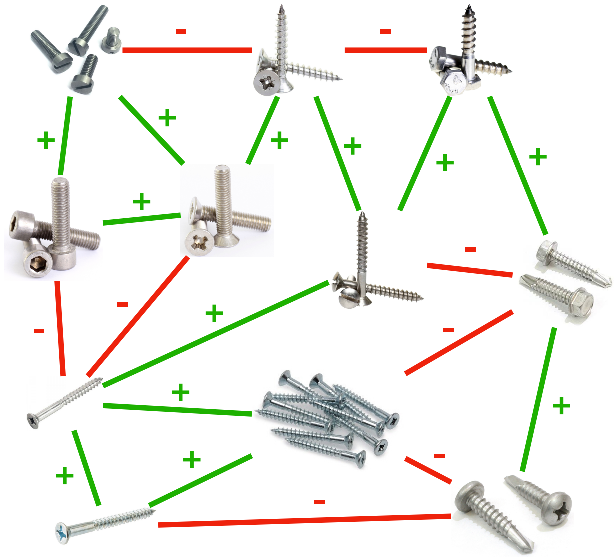
data clustering — background

- data clustering: a fundamental problem in machine learning
- intuitively: we want to partition a dataset into clusters so that similar objects are assigned to the same cluster
- extensively-studied problem, many different settings, objectives, applications
- euclidean setting: data are represented as Euclidean points
  - minimize an objective function such as k-means ($\sum_i \min_j ||x_i - c_j||_2^2$), k-median ($\sum_i \min_j ||x_i - c_j||_2$) or k-center ($\max_i \min_j ||x_i - c_j||_2$)
- graph setting: data are represented as a graph
  - edges represent affinity, e.g., friends in a social network
  - often a similarity value is available, e.g., connection strength
  - optimize an objective function such as normalized edge cut across clusters (minimize) or edge density within clusters (maximize)

correlation clustering — motivation

- in the graph setting described above, edges are positive
  - presence of an edge suggests that nodes should be clustered together
  - absence of an edge suggests that nodes should be assigned to different clusters
- in some cases, we may have a local prediction whether two objects should be assigned to the same cluster or not
  - positive edge : the two objects should be clustered together
  - negative edge : the two objects should be assigned to different clusters
  - no edge : no information
- we obtain a signed network !

correlation clustering — motivation

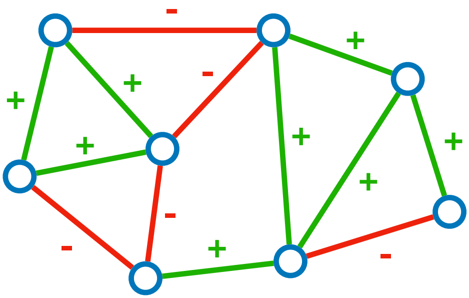


- example: a dataset of images, e.g., screws of different types
- a machine-learning program, which, given two images, outputs whether the images depict the same type of screws
- we obtain a signed network
- we want to cluster the images so that same-type screws are assigned in the same cluster

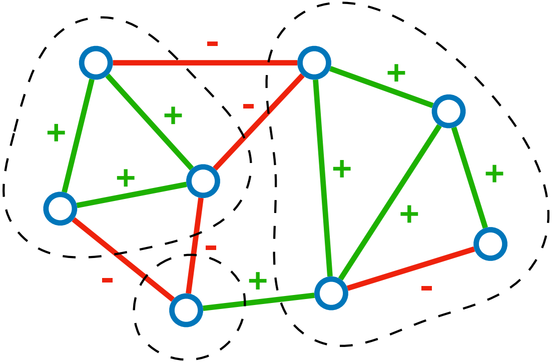
correlation clustering — motivation

- due to noise in the data and classification errors in the network construction, we cannot expect to achieve perfect agreement
- we need an objective function to capture the consistency of the resulting clustering with the input signed network

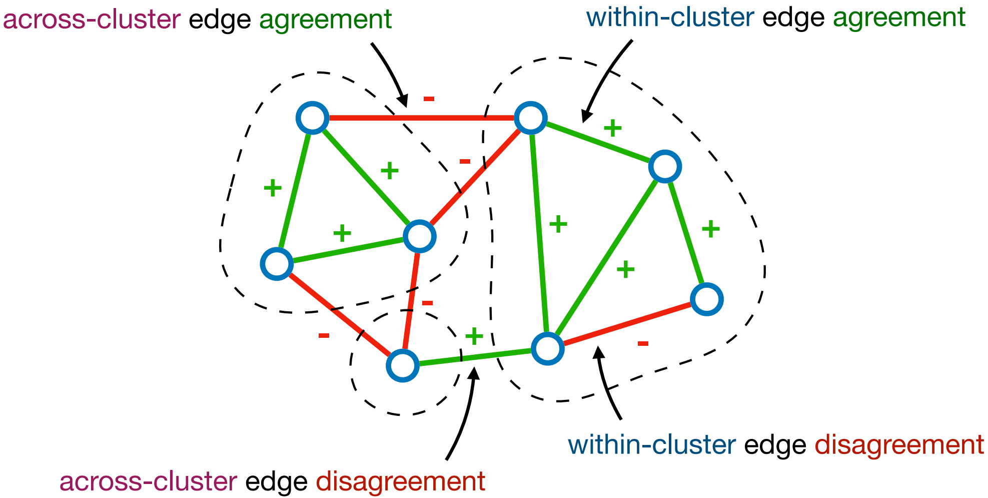
correlation clustering — edge agreements and disagreements



correlation clustering — edge agreements and disagreements



correlation clustering — edge agreements and disagreements



correlation clustering — problem formulation

given a signed network $G = (V, E^+, E^-)$, find a partitioning $\mathcal{C} = \{C_1, \ldots, C_k\}$ of the graph vertices (i.e., $\bigcup_{i=1}^k C_i = V$ and $C_i \cap C_j = \varnothing$, for all $i \neq j$), so as to

variant 1 : [maximize agreements]

$$\max \quad a(\mathcal{C}) = \sum_{i,j} \mathbb{I}\{(i,j) \in E^+\}\,\mathbb{I}\{c(i) = c(j)\} + \sum_{i,j} \mathbb{I}\{(i,j) \in E^-\}\,\mathbb{I}\{c(i) \neq c(j)\}$$

variant 2 : [minimize disagreements]

$$\min \quad d(\mathcal{C}) = \sum_{i,j} \mathbb{I}\{(i,j) \in E^+\}\,\mathbb{I}\{c(i) \neq c(j)\} + \sum_{i,j} \mathbb{I}\{(i,j) \in E^-\}\,\mathbb{I}\{c(i) = c(j)\}$$

correlation clustering — problem formulation

given a signed network $G = (V, E^+, E^-)$, find a partitioning $\mathcal{C} = \{C_1, \ldots, C_k\}$ of the graph vertices (i.e., $\bigcup_{i=1}^k C_i = V$ and $C_i \cap C_j = \varnothing$, for all $i \neq j$), so as to

variant 1 : [maximize agreements]

$$\max \quad a(\mathcal{C}) = \sum_{i,j} \mathbb{I}\{(i,j) \in E^+\}\,\mathbb{I}\{c(i) = c(j)\} + \sum_{i,j} \mathbb{I}\{(i,j) \in E^-\}\,\mathbb{I}\{c(i) \neq c(j)\}$$

variant 2 : [minimize disagreements]

$$\min \quad d(\mathcal{C}) = \sum_{i,j} \mathbb{I}\{(i,j) \in E^+\}\,\mathbb{I}\{c(i) \neq c(j)\} + \sum_{i,j} \mathbb{I}\{(i,j) \in E^-\}\,\mathbb{I}\{c(i) = c(j)\}$$

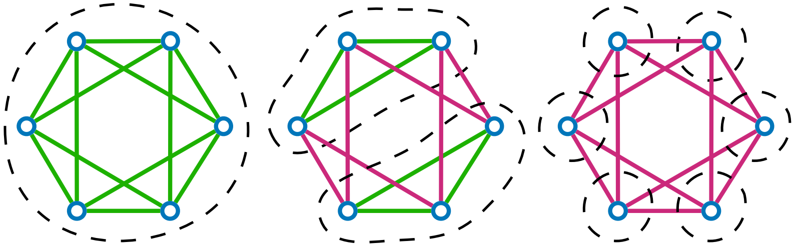
majority of research focuses on the minimization variant

correlation clustering — number of clusters

an important observation

- the problem formulation does not (need to) specify the number of clusters
- optimal k depends on input network, and does not have trivial minimizers

  e.g.,



$k = 1$ $k = 2$ $k = n$

- the optimal solution in each of the above cases is the most intuitive one

correlation clustering — hardness

both formulations (max-agree and min-disagree) are **NP**-hard

the min-disagree problem is

- **NP**-hard for complete unweighted graphs — Bansal et al. (2004)
  reduction from "partition into triangles"
- **APX**-hard for general (un)weighted graphs — Demaine et al. (2006)
  reduction from multiway cut

correlation clustering — existing approximation algorithms

overview of results for the min-disagree problem

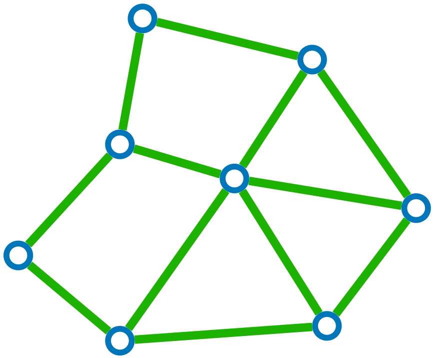
paper	graph type	approximation ratio	deterministic /randomized	running time
Bansal et al. (2004)	complete	large constant	deterministic	$\mathcal{O}(n^2)$
Demaine et al. (2006)	general	$\mathcal{O}(\log n)$	deterministic	LP
Ailon et al. (2005)	complete	2.5	randomized	LP
Ailon et al. (2005)	complete	3	randomized	$\mathcal{O}(m)$
Chawla et al. (2015)	complete	$2.06 - \epsilon$	deterministic	LP
Giotis and Guruswami (2005)[1]	complete	PTAS	randomized	combinatorial
Coleman et al. (2008)[2]	complete[3]	2	deterministic	combinatorial

[1] for fixed k; recall that the problem is **APX**-hard when k is not fixed

[2] for $k = 2$ (2-correlation-clustering)

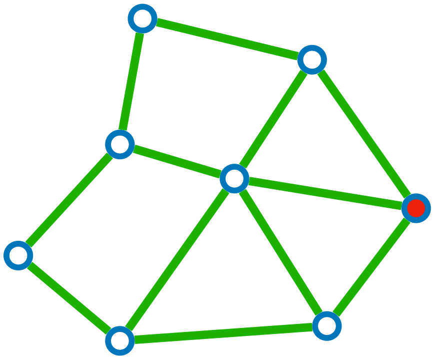
[3] algorithm applicable to general graphs, but analysis for complete graphs

the PIVOT algorithm — example



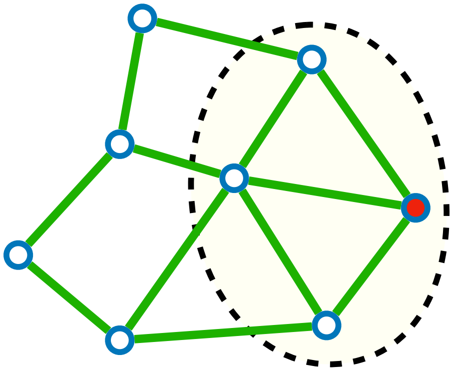
a complete graph: positive edges shown, negative edges not shown

The PIVOT algorithm — example



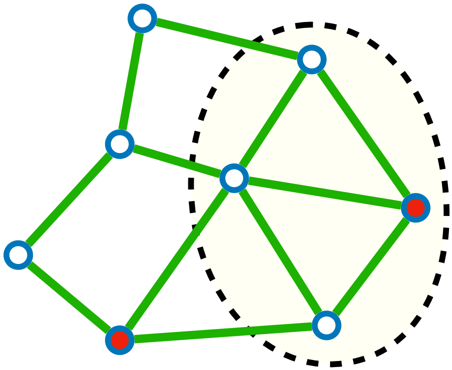
a pivot is selected uniformly at random

The PIVOT algorithm — example



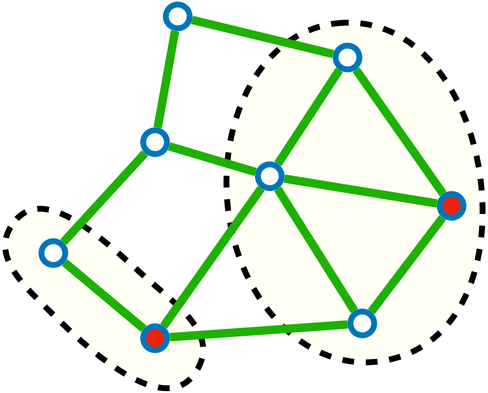
a cluster is formed with the pivot and all its positive neighbors

The PIVOT algorithm — example



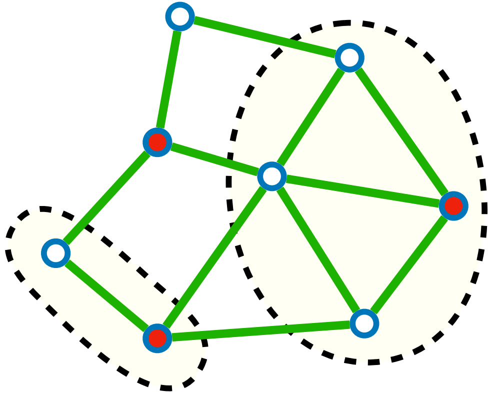
a new pivot is selected from the remaining of the graph vertices

The PIVOT algorithm — example



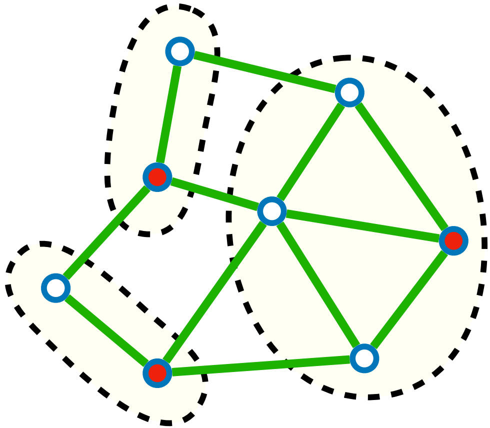
a second cluster is formed with the pivot and all its positive neighbors

The Pivot algorithm — example



and the process continues . . .

The PIVOT algorithm — example



... until the whole graph is consumed.

correlation clustering — the KwikCluster (or Pivot) algorithm

KwikCluster$(G = (V, E^+, E^-))$

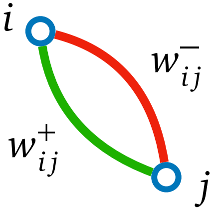
```
If V = ∅ then return ∅
Pick random pivot i ∈ V.
Set C = {i}, V' = ∅.

For all j ∈ V, j ≠ i:
    If (i, j) ∈ E+ then
        Add j to C
    Else (If (i, j) ∈ E−)
        Add j to V'

Let G' be the subgraph induced by V'.

Return C ∪ KwikCluster(G') .
```

- the Pivot algorithm

(Ailon et al., 2005)

- \+ an elegant randomized algorithm
- \+ approximation ratio 3
- \+ running time $\mathcal{O}(m)$
- – it assumes a complete graph
- – it assumes an unweighted graph

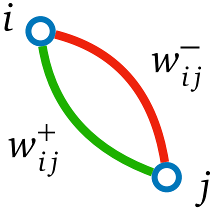
weighted signed networks



we want to extend the methods to weighted signed networks
$G = (V, w^+, w^-)$

- w^+_{ij} : weight of positive edge (i, j)
- w^-_{ij} : weight of negative edge (i, j)
- unweighted case : $w^+_{ij}, w^-_{ij} \in \{0, 1\}$
- weighted case : $w^+_{ij}, w^-_{ij} \in \mathbb{R}_{\geq 0}$

weighted signed networks



we want to extend the methods to weighted signed networks $G = (V, w^+, w^-)$

- w^+_{ij} : weight of positive edge (i, j)
- w^-_{ij} : weight of negative edge (i, j)
- unweighted case : $w^+_{ij}, w^-_{ij} \in \{0, 1\}$
- weighted case : $w^+_{ij}, w^-_{ij} \in \mathbb{R}_{\geq 0}$

interesting cases :

- probability constraints : $w^+_{ij} + w^-_{ij} = 1$, for all $i, j \in V$
- triangle inequality : $w^-_{ik} \leq w^-_{ij} + w^-_{jk}$, for all $i, j, k \in V$

the PIVOT algorithm on weighted signed networks

1. consider a weighted signed networks $G = (V, w^+, w^-)$
2. assume probability constraints $w_{ij}^+ + w_{ij}^- = 1$, for all $i, j \in V$
3. form unweigted $G_u = (V, E^+, E^-)$ by taking “majority” on each edge
4. apply PIVOT on G_u
5. return solution of PIVOT on G_u, as the solution for G

theoretical properties of the above algorithm

- 5 approximation, with probability constraints
- 2 approximation, with probability constraints and triangle inequality

using PIVOT for LP rounding

LP relaxation

(Ailon et al., 2005)

$$
\begin{aligned}
\text{maximize} \quad & \sum_{ij} \left(x_{ij}^{+} w_{ij}^{-} + x_{ij}^{-} w_{ij}^{+} \right) \\
\text{such that} \quad & x_{ik}^{-} \leq x_{ij}^{-} + x_{jk}^{-}, \text{ for all } i, j, k \in V \\
& x_{ij}^{+} + x_{ij}^{-} = 1, \text{ for all } i, j \in V \\
& x_{ij}^{+}, x_{ij}^{-} \geq 0, \text{ for all } i, j \in V
\end{aligned}
$$

- notice that if $x_{ij}^{-} \in \{0, 1\}$, then x_{ij}^{-} define an equivalence class (clustering)

Using PIVOT for LP rounding

LP-KWIKCLUSTER(V, x^+, x^-)

A recursive algorithm for rounding the LP for weighted CORRELATION-CLUSTERING. *Given an LP solution* $x^+ = \{x^+_{ij}\}_{i<j}$, $x^- = \{x^-_{ij}\}_{i<j}$, *returns a clustering of the vertices*

```
If V = ∅ then return ∅
Pick random pivot i ∈ V.
Set C = {i}, V' = ∅.

For all j ∈ V, j ≠ i :
    With probability x+_ij
        Add j to C.
    Else (With probability x-_ij = 1 - x+_ij)
        Add j to V'.

Return clustering
  {C} ∪ LP-KwikCluster(V', x+, x-).
```

(Ailon et al., 2005)

1. solve the LP relaxation
2. use the PIVOT for randomized rounding of the LP solution

- ▶ 2.5-approximation, with probability constraints
- ▶ 2-approximation, with probability & triangle inequality constraints
- – expensive; requires solving an LP

correlation clustering — summary

- signed graphs have been studied in theoretical computer science in the context of correlation clustering
- a wealth of theoretical results for different problem settings
- several applications, e.g., clustering aggregation
- many other problem variants not discussed here

  overlapping, on-line, bipartite, chromatic, local, ...

conclusions

conclusions

- signed networks differ in terms of basic concepts, properties and present unique computational challenges
- in this lecture we gave an overview of mining signed networks
  - we discussed some theoretical concepts
  - we discussed some common applications

many topics not discussed

- graph partitioning and community detection
- link prediction
- network dynamics
- graph embedding and representation learning
- node ranking

References I

Ailon, N., Charikar, M., and Newman, A. (2005). Aggregating inconsistent information: ranking and clustering. In *Proceedings of the thirty-seventh annual ACM symposium on Theory of computing*, pages 684–693.

Aref, S. and Wilson, M. C. (2018). Measuring partial balance in signed networks. *Journal of Complex Networks*, 6(4):566–595.

Bansal, N., Blum, A., and Chawla, S. (2004). Correlation clustering. *Machine learning*, 56(1-3):89–113.

Bonchi, F., Galimberti, E., Gionis, A., Ordozgoiti, B., and Ruffo, G. (2019). Discovering polarized communities in signed networks. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, pages 961–970.

Cartwright, D. and Harary, F. (1956). Structural balance: a generalization of heider's theory. *Psychological review*, 63(5):277.

References II

Chawla, S., Makarychev, K., Schramm, T., and Yaroslavtsev, G. (2015). Near optimal LP-rounding algorithm for correlation clustering on complete and complete k-partite graphs. In *Proceedings of the forty-seventh annual ACM symposium on Theory of computing*, pages 219–228.

Chu, L., Wang, Z., Pei, J., Wang, J., Zhao, Z., and Chen, E. (2016). Finding gangs in war from signed networks. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1505–1514. ACM.

Coleman, T., Saunderson, J., and Wirth, A. (2008). A local-search 2-approximation for 2-correlation-clustering. In *European Symposium on Algorithms*, pages 308–319.

Demaine, E. D., Emanuel, D., Fiat, A., and Immorlica, N. (2006). Correlation clustering in general weighted graphs. *Theoretical Computer Science*, 361(2-3):172–187.

Giotis, I. and Guruswami, V. (2005). Correlation clustering with a fixed number of clusters. *arXiv preprint cs/0504023*.

References III

Giscard, P.-L., Rochet, P., and Wilson, R. C. (2017). Evaluating balance on social networks from their simple cycles. *Journal of Complex Networks*, 5(5):750–775.

Goldberg, A. V. (1984). *Finding a maximum density subgraph*. University of California Berkeley.

Gülpinar, N., Gutin, G., Mitra, G., and Zverovitch, A. (2004). Extracting pure network submatrices in linear programs using signed graphs. *Discrete Applied Mathematics*, 137(3):359–372.

Hansen, P. (1984). Shortest paths in signed graphs. In *North-Holland mathematics studies*, volume 95, pages 201–214. Elsevier.

Harary, F. (1953). On the notion of balance of a signed graph. *The Michigan Mathematical Journal*, 2(2):143–146.

Hou, Y., Li, J., and Pan, Y. (2003). On the Laplacian eigenvalues of signed graphs. *Linear and Multilinear Algebra*, 51(1):21–30.

References IV

Read, K. E. (1954). Cultures of the central highlands, new guinea. *Southwestern Journal of Anthropology*, 10(1):1–43.

Terzi, E. and Winkler, M. (2011). A spectral algorithm for computing social balance. In *International Workshop on Algorithms and Models for the Web-Graph*, pages 1–13.

Tsourakakis, C. E., Chen, T., Kakimura, N., and Pachocki, J. (2019). Novel dense subgraph discovery primitives: Risk aversion and exclusion queries. *arXiv preprint arXiv:1904.08178*.

Xiao, H., Ordozgoiti, B., and Gionis, A. (2020). Searching for polarization in signed graphs: a local spectral approach. *arXiv preprint arXiv:2001.09410*.