Network Community Detection

Gonzalo Mateos
Dept. of Electrical and Computer Engineering
University of Rochester
gmateosb@ece.rochester.edu
http://www.ece.rochester.edu/~gmateosb/

March 20, 2015
The inter-marriage network of Florentine families in the Renaissance

Medicis did not have the greatest wealth or most legislature seats
Yet via marriages they rose to prominence and consolidated power
Community structure in networks

Examples of network communities

Network community detection

Modularity maximization

Spectral graph partitioning
Communities within networks

- Networks play the powerful role of bridging the local and the global
- Explain how processes at node/link level ripple to a population
- We often think of (social) networks as having the following structure

- Can we gain insights behind this conceptualization?
In the 60s., M. Granovetter interviewed people who changed jobs. He asked them about how they discovered their new jobs. Many learned about opportunities through personal contacts. Surprisingly, contacts where often acquaintances rather than friends. Close friends presumably have the most motivation to help you out.

Q: Why do distant acquaintances convey the crucial information?

Granovetter’s answer and impact

- Linked two different perspectives on distant friendships
  - **Structural**: focus on how friendships span parts of the network
  - **Interpersonal**: local consequences of friendship being strong or weak
- Intertwining between structural and informational role of an edge
  - **Structurally-embedded edges** within a community:
    - Tend to be socially strong; and
    - Are highly redundant in terms of information access
  - **Long-range edges** spanning different parts of the network:
    - Tend to be socially weak; and
    - Offer access to useful information (e.g., on a new job)
- The answer transcends the specific setting of job-seeking
- General way of thinking about the architecture of social networks
A basic principle of network formation is that of **triadic closure**

“If two people have a friend in common, then there is an increased likelihood that they will become friends in the future”

Emergent edges in a social network likely to close triangles

More likely to see the **red** edge than the **blue** one

Prevalence of triadic closure measured by the **clustering coefficient**

\[ \text{cl}(v) = 0 \quad \text{cl}(v) = 1/3 \quad \text{cl}(v) = 1 \]
Reasons for triadic closure

Triadic closure is intuitively very natural. Reasons why it operates:

- Increased opportunity for B and C to meet
  - Both spend time with A
- There is a basis for mutual trust among B and C
  - Both have A as a common friend
- A may have an incentive to bring B and C together
  - Lack of friendship may becomes a source of latent stress

Premise based on theories dating to early work in social psychology
Consider the simple social network in the figure:

- A’s links to C, D, and E connect her to a tightly knit group.
  - Reasonable that A, C, D, and E are exposed to similar opinions.
- A’s link to B seems to reach to a different part of the network.
  - Offers her access to views she would otherwise not hear about.

The A-B edge is called a *bridge*, its removal disconnects the networks. Discussion on giant components taught us that *bridges are quite rare*.
Ex: In reality, the social network is larger and may look as

Without A, B realizing it, there may be a longer path among them

The span of \((u, v)\) is the \(u - v\) distance when the edge is removed

Def: A local bridge is an edge with span \(> 2\)

Ex: Edge A-B is a local bridge with span 3

Local bridges with large spans \(\approx\) bridges, but less extreme

Implicit link with triadic closure: local bridges not part of triangles
Strong triadic closure property

- Categorize all edges in the network according to their strength
  - Strong ties corresponding to friendship
  - Weak ties corresponding to acquaintances

- Opportunity, trust, incentive act more powerfully for strong ties
- Suggests a qualitative assumption termed strong triadic closure

  "Two strong ties implies a third edge exists closing the triangle"

- An abstraction to reason about consequences of strong/weak ties
Local bridges and weak ties

- Local, interpersonal distinction between edges ⇒ strong/weak ties
- Global, structural notion ⇒ local bridges present or absent

**Theorem**

*If a node satisfies the strong triadic closure property and is involved in at least two strong ties, then any local bridge incident to it is a weak tie.*

- Links structural and interpersonal perspectives on friendships

- Back to job-seeking, local bridges connect to new information
- Their conceptual span is related to their weakness as social ties
- This surprising dual role suggests a "strength of weak ties"
Proof by contradiction

Proof.

- We will argue by contradiction. Suppose node A has 2 strong ties
- Moreover, suppose A satisfies the strong triadic closure property

Let A-B be a local bridge as well as a strong tie

Then edge B-C must exist by strong triadic closure
- This contradicts A-B is a local bridge (cannot be part of a triangle)
Can one test Granovetter’s theory with real network data?
Elusive for decades. Lack of large-scale social interaction surveys
Now we have “who-calls-whom” networks with both key ingredients
⇒ Network structure of communication among pairs of people
⇒ Total talking time, i.e., a proxy for tie strength
Ex: Cell-phone network spanning ≈ 20% of country’s population
Generalizing weak ties and local bridges

- Model described so far imposes sharp dichotomies on the network.
- Edges are either strong or weak, they are local bridges or not.
- Convenient to have versions exhibiting smoother gradations.
- Numerical tie strength $\Rightarrow$ Minutes spent in phone conversations.
- Order edges by strength, report their percentile occupancy.
- Generalize local bridges $\Rightarrow$ Define neighborhood overlap of edge $(i, j)$

$$O_{ij} = \frac{n(i) \cap n(j)}{n(i) \cup n(i)}; \quad n(i) := \{j \in V : (i, j) \in E\}$$

- Desirable property: $O_{ij} = 0$ if $(i, j)$ is a local bridge.
Empirical results

- **Strength of weak ties prediction:** $O_{ij}$ grows with tie strength
- Dependence borne out very cleanly by the data (○ points)

- Randomly permuted tie strengths, fixed network structure (□ points)
- Effectively removes the coupling between $O_{ij}$ and tie strength
The cell-phone network with color-coded tie strengths

- Stronger ties are more structurally-embedded (within communities)
- Weaker ties correlate with long-range edges joining communities
Randomly permuted tie strengths

- The same cell-phone network with randomly permuted tie strengths

- No apparent link between structural and interpersonal roles of edges
Weak ties linking communities

- **Strength of weak ties prediction:** long-range, weak ties bridge communities

![Graph showing edge removal by strength and overlap](image)

- Delete **decreasingly weaker** (small overlap) edges one at a time
- The giant component shrinks rapidly, eventually disappears
- Repeat with strong-to-weak tie deletions \( \Rightarrow \) slower shrinkage observed
We often think of (social) networks as having the following structure:

- Long-range, weak ties
- Embedded, strong ties

Conceptual picture supported by Granovetter’s *strength of weak ties*
Network communities

Community structure in networks

Examples of network communities

Network community detection

Modularity maximization

Spectral graph partitioning
Communities

- Nodes in real-world networks organize into **communities**
- **Ex:** families, clubs, political organizations, proteins by function, . . .
- Supported by Granovetter’s **strength of weak ties** theory

- Community (a.k.a. group, cluster, module) members are:
  - Well connected among themselves; while they are
  - Relatively well separated from the rest of the nodes
- Exhibit high cohesiveness w.r.t. the underlying relational patterns
Zachary’s karate club

- Social interactions among members of a karate club in the 70s

Zachary witnessed the club split in two during his observation period

- Toy network, yet canonical for community detection algorithms
- Offers “ground truth” community membership (a rare luxury)
Political blogs

- The political blogosphere for the US 2004 presidential election

- Community structure of liberal and conservative blogs is apparent
- People have a stronger tendency to interact with “equals”
High-school students

- Network of social interactions among high-school students

- Strong assortative mixing, with race as latent characteristic
Physicists working on Network Science

- Coauthorship network of physicists publishing networks’ research

- Tightly-knit subgroups are evident from the network structure
College football

- Vertices are NCAA football teams, edges are games during Fall’00

- Communities are the NCAA conferences and independent teams
Facebook friendships

- Facebook egonet with 744 vertices and 30K edges

- Asked “ego” to identify social circles to which friends belong
- Those here are: company, high-school, basketball club, squash club
Network community detection

Community structure in networks

Examples of network communities

Network community detection

Modularity maximization

Spectral graph partitioning
Nodes in real-world networks organize into communities

Ex: families, clubs, political organizations, proteins by function, ...

Community (a.k.a. group, cluster, module) members are:
  \(\Rightarrow\) Well connected among themselves; while they are
  \(\Rightarrow\) Relatively well separated from the rest of the nodes

Exhibit high cohesiveness w.r.t. the underlying relational patterns

Q: How can we automatically identify such cohesive subgropus?
Community detection and graph partitioning

- **Community detection** is a challenging clustering problem
  - No consensus on the structural definition of community
  - Node subset selection often leads to intractable formulations
  - Lack of ground-truth to validate on real data

- Mainly used for exploratory analysis of network data
  - **Ex:** clues about social interactions, content-related web pages

- A related, better defined problem is that of **graph partitioning**
  - Divide $V$ into **given number** of non-overlapping groups of **given sizes**
  - Criterion: number of edges between groups is minimized (more soon)

- **Ex:** task-processor assignment for load balancing

- Number and sizes of groups unspecified in community detection
  - **Goal:** identify the natural fault lines along which a network separates
Graph partitioning is hard

- Consider a graph bisection problem, i.e., partition $V$ into two groups
- Suppose the groups $V_1$ and $V_2$ are non-overlapping
- Suppose groups have equal size, i.e., $|V_1| = |V_2| = N_v/2$
- Minimize edges running between vertices in different groups
- Simple problem to describe, but hard to solve

\[
\text{Number of ways to partition } V : \binom{N_v}{N_v/2} \approx \frac{2^{N_v+1}}{\sqrt{N_v}}
\]

- Used Stirling’s formula $(N_v)! \approx \sqrt{2\pi N_v}(N_v/e)^{N_v}$, dropped constants
- Exhaustive search intractable beyond toy small-sized networks
- No smart (i.e., polynomial time) algorithm, the problem is NP-hard
- Seek good heuristics, e.g., relaxations of natural optimization criteria
Local bridges connect weakly interacting parts of the network.

Q: What about removing those to reveal communities?

Multiple local bridges. Some better that others? Which one first?

There might be no local bridge, yet an apparent natural division.
Edge betweenness centrality

- Idea: look for **high edge betweenness centrality** to identify weak ties

- Edges of high $c_{Be}(e)$ carry large traffic volume over shortest paths

- Suggests a position at the interface between tightly-knit groups

- **Ex:** cell-phone network with colored edge strength and betweenness

![Edge strength](image1)

![Edge betweenness](image2)
The Girvan-Newmann algorithm is extremely simple conceptually.

- Identify and remove “spanning links” between cohesive subgroups.
- Repeat until there are no edges left:
  - Calculate the betweenness centrality $c_{Be}(e)$ of all edges.
  - Remove edge(s) with highest $c_{Be}(e)$.
- The connected components are the communities identified.
- **Divisive method:** network falls apart into pieces as we go.
- **Nested partition:** larger communities potentially host denser groups.
- Recalculate edge betweenness in $O(N_v N_e)$-time per step.
Example: The algorithm in action

Original graph

Step 1

Step 2

Step 3

Nested graph decomposition
Scientific collaboration network

- Coauthorship network of scientists at the Santa Fe Institute

- Communities found can be traced to different disciplines
Hierarchical clustering

- Hierarchical clustering at the heart of many graph partitioning methods
- Greedy approach to iteratively modify successive candidate partitions
- **Agglomerative**: successive coarsening of partitions through merging
- **Divisive**: successive refinement of partitions through splitting
- Per step, partitions are modified in a way that minimizes a cost
- Measures of (dis)similarity $x_{ij}$ between pairs of vertices $v_i$ and $v_j$
- **Ex**: Euclidean distance dissimilarity

$$x_{ij} = \sqrt{\sum_{k \neq i,j} (A_{ik} - A_{jk})^2}$$

- Method returns an entire hierarchy of nested partitions of the graph
- These can range fully from $\{\{v_1\}, \ldots, \{v_{N_v}\}\}$ to $V$
An agglomerative hierarchical clustering algorithm proceeds as follows:

S1: Choose a dissimilarity metric and compute it for all vertex pairs.
S2: Start by assigning each vertex to a group of its own.
S3: Merge the pair of groups with smallest dissimilarity.
S4: Calculate the dissimilarity between the new group and all other groups.
S5: Repeat from S3 until all vertices belong to a single group.

Need to define group dissimilarity from pairwise vertex counterparts.

- **Single linkage:** group dissimilarity \( x^{SL}_{G_i, G_j} \) follows single most dissimilar pair.

\[
  x^{SL}_{G_i, G_j} = \max_{u \in G_i, v \in G_j} x_{uv}
\]

- **Complete linkage:** every vertex pair highly dissimilar to have high \( x^{CL}_{G_i, G_j} \).

\[
  x^{CL}_{G_i, G_j} = \min_{u \in G_i, v \in G_j} x_{uv}
\]
Hierarchical partitions are often represented with a **dendrogram**.

- Shows groups found in the network at all algorithmic steps
  - Split the network at different resolutions

**Ex:** Girvan-Newman’s algorithm for the Zachary’s karate club

**Q:** Which of the divisions is the most useful/optimal in some sense?

**A:** Need to define metrics of graph clustering quality
Modularity maximization

Community structure in networks

Examples of network communities

Network community detection

Modularity maximization

Spectral graph partitioning
Modularity

- Size of communities typically unknown \(\Rightarrow\) Identify automatically
- **Modularity** measures how well a network is partitioned in communities
- **Intuition**: density of edges in communities higher than expected
- Consider a graph \(G\) and a partition into groups \(s \in S\). Modularity:

\[
Q(G, S) \propto \sum_s \left[ (\# \text{ of edges within group } s) - \mathbb{E}[\# \text{ of such edges}] \right]
\]

- Formally, and after normalization such that \(Q(G, S) \in [-1, 1]\)

\[
Q(G, S) = \frac{1}{2N_e} \sum_s \sum_{i,j \in s} \left[ A_{ij} - \frac{d_i d_j}{2N_e} \right]
\]

- **Null model**: random edge placement, preserving degree distribution
Assessing clustering quality

- Can evaluate the modularity of each partition in a dendrogram
- The maximum value gives the “best” community structure
- **Ex:** Girvan-Newman’s algorithm for the Zachary’s karate club

**Q:** Why not optimize $Q(G, S)$ directly over possible partitions $S$?
Modularity revisited

- Recall our definition of modularity

\[ Q(G, S) = \frac{1}{2N_e} \sum_s \sum_{i,j \in s} \left[ A_{ij} - \frac{d_i d_j}{2N_e} \right] \]

- Let \( g_i \) be the group membership of vertex \( i \), and rewrite

\[ Q(G, S) = \frac{1}{2N_e} \sum_{i,j \in V} \left[ A_{ij} - \frac{d_i d_j}{2N_e} \right] \mathbb{1} \{ g_i = g_j \} \]

- Define for convenience the summands \( B_{ij} := A_{ij} - \frac{d_i d_j}{2N_e} \)

- Note that both marginal sums of \( B_{ij} \) vanish, since e.g.,

\[ \sum_j B_{ij} = \sum_j A_{ij} - \frac{d_i}{2N_e} \sum_j d_j = d_i - \frac{d_i}{2N_e} 2N_e = 0 \]
Graph bisection

- Consider (for simplicity) dividing the network in two parts
- Introduce the **community membership variables** per vertex

\[ s_i = \begin{cases} 
  +1, & \text{vertex } i \text{ belongs to group 1} \\
  -1, & \text{vertex } i \text{ belongs to group 2} 
\end{cases} \]

- Using the identity \( \frac{1}{2}(s_is_j + 1) = \mathbb{I}\{g_i = g_j\} \), the modularity is

\[
Q(G, S) = \frac{1}{2N_e} \sum_{i,j \in V} \left[ A_{ij} - \frac{d_i d_j}{2N_e} \right] \mathbb{I}\{g_i = g_j\} \\
= \frac{1}{4N_e} \sum_{i,j \in V} B_{ij}(s_is_j + 1)
\]

- Recall that \( \sum_j B_{ij} = 0 \) to obtain the simpler expression

\[
Q(G, S) = \frac{1}{4N_e} \sum_{i,j \in V} B_{ij} s_is_j
\]
Let $\mathbf{B} \in \mathbb{R}^{N_v \times N_v}$ be the modularity matrix with entries $B_{ij}$.

A given partition $S$ is defined by the vector $\mathbf{s} = [s_1, \ldots, s_{N_v}]^\top$.

The modularity is a quadratic form

$$Q(G, S) = \frac{1}{4N_e} \sum_{i,j \in V} B_{ij} s_i s_j = \frac{1}{4N_e} \mathbf{s}^\top \mathbf{B} \mathbf{s}$$

Modularity as criterion for graph bisection yields the formulation

$$\hat{s} = \arg \max_{\mathbf{s} \in \{\pm 1\}^{N_v}} \mathbf{s}^\top \mathbf{B} \mathbf{s}$$

Nasty binary constraints $\mathbf{s} \in \{\pm 1\}^{N_v}$ (hypercube vertices).

Not surprisingly, modularity optimization is NP-hard [Brandes et al ’06].
Just relax!

- Relax the constraint $s \in \{\pm 1\}^{N_v}$ to $s \in \mathbb{R}^{N_v}$, $\|s\| = 1$
- We are left with the quadratic problem

$$\hat{s} = \arg \max_s s^\top Bs, \quad s. \text{ to } s^\top s = 1$$

- Associate a Lagrange multiplier $\lambda$ to the constraint $s^\top s = 1$
- The optimality conditions yields

$$\nabla_s \left[ s^\top Bs + \lambda(1 - s^\top s) \right] = 0 \Rightarrow Bs = \lambda s$$

- The conclusion is that $s$ is an eigenvector of $B$ with eigenvalue $\lambda$
- **Q:** Which eigenvector should we pick?
- **A:** To maximize modularity pick the dominant eigenvector of $B$
Let $u_1$ be the dominant eigenvector of $B$, with $i$-th entry $[u_1]_i$.

We cannot just set $s = u_1$ because $u_1 \neq \{\pm 1\}^{N_v}$.

**Best effort:** maximize their similarity $s^\top u_1$ which gives

$$s_i = \text{sign}([u_1]_i) := \begin{cases} +1, & [u_1]_i > 0 \\ -1, & [u_1]_i \leq 0 \end{cases}$$

**Spectral modularity maximization algorithm**

- **S1:** Compute the modularity matrix $B$ with entries $B_{ij} = A_{ij} - \frac{d_i d_j}{2N_e}$
- **S2:** Find the dominant eigenvector $u_1$ of $B$ (e.g., power method)
- **S3:** Cluster membership of vertex $i$ is $s_i = \text{sign}([u_1]_i)$

Multiple (> 2) communities through e.g., repeated graph bisection.
Spectral graph partitioning

Community structure in networks

Examples of network communities

Network community detection

Modularity maximization

Spectral graph partitioning
Graph bisection

Consider an undirected graph $G(V, E)$.

Recall the graph bisection problem, i.e., partition $V$ into two groups.

Suppose the groups $V_1$ and $V_2 = V_1^c$ are non-overlapping.

Suppose groups have given size, i.e., $|V_1| = N_1$ and $|V_2| = N_2$.

Q: What is a good criterion to partition the graph?

A: We have already seen modularity. Let’s see a different one.
Graph cut

- **Desiderata**: Community members should be
  - Well connected among themselves; and
  - Relatively well separated from the rest of the nodes

- **Def**: A cut is the number of edges between groups $V_1$ and $V \setminus V_1$
  \[
  R := \text{cut}(V_1, V_2) = \sum_{i \in V_1, j \in V_2} A_{ij}
  \]

- **Natural criterion**: minimize cut, i.e., edges across groups $V_1$ and $V_2$
Recall the community membership variables per vertex

\[ s_i = \begin{cases} 
+1, & \text{vertex } i \text{ belongs to } V_1 \\
-1, & \text{vertex } i \text{ belongs to } V_2 
\end{cases} \]

Let \( g_i \) be the group membership of vertex \( i \), such that

\[ \mathbb{I}\{g_i \neq g_j\} = \frac{1}{2}(1 - s_i s_j) = \begin{cases} 
1, & i \text{ and } j \text{ in different groups} \\
0, & i \text{ and } j \text{ in the same group} 
\end{cases} \]

Thus, the cut is expressible in terms of the variables \( s_i \) as

\[ R = \sum_{i \in V_1, j \in V_2} A_{ij} = \frac{1}{2} \sum_{i,j} A_{ij}(1 - s_i s_j) \]
The first summand in $R = \frac{1}{2} \sum_{i,j} A_{ij} (1 - s_is_j)$ is

$$\sum_{i,j} A_{ij} = \sum_i d_i = \sum_i d_i s_i^2 = \sum_i d_i s_i s_j \mathbb{1} \{i = j\}$$

Used $s_i^2 = 1$ since $s_i \in \{\pm 1\}$. The cut becomes

$$R = \frac{1}{2} \sum_{i,j} (d_i \mathbb{1} \{i = j\} - A_{ij}) s_is_j = \frac{1}{2} \sum_{i,j} L_{ij} s_is_j$$

Cut in terms of $L_{ij}$, entries of the graph Laplacian $L = D - A$, i.e.,

$$R(s) = \frac{1}{2} s^\top L s, \quad s := [s_1, \ldots, s_N]^\top$$

Modularity $Q(s) \propto s^\top B s$ maximization vs. cut $R(s) \propto s^\top L s$ minimization
Laplacian matrix properties revisited

- **Smoothness:** For any vector \( \mathbf{x} \in \mathbb{R}^{N_v} \) of “vertex values”, one has
  \[
  \mathbf{x}^\top \mathbf{L} \mathbf{x} = \sum_{i,j} L_{ij} x_i x_j = \sum_{(i,j) \in E} (x_i - x_j)^2
  \]
  which can be minimized to enforce smoothness of functions on \( G \)

- **Positive semi-definiteness:** Follows since \( \mathbf{x}^\top \mathbf{L} \mathbf{x} \geq 0 \) for all \( \mathbf{x} \in \mathbb{R}^{N_v} \)

- **Spectrum:** All eigenvalues of \( \mathbf{L} \) are real and non-negative
  - The eigenvectors form an orthonormal basis of \( \mathbb{R}^{N_v} \)

- **Rank deficiency:** Since \( \mathbf{L} \mathbf{1} = \mathbf{0} \), \( \mathbf{L} \) is rank deficient

- **Spectrum and connectivity:** The smallest eigenvalue \( \lambda_1 \) of \( \mathbf{L} \) is 0
  - If the second-smallest eigenvalue \( \lambda_2 \neq 0 \), then \( G \) is connected
  - If \( \mathbf{L} \) has \( n \) zero eigenvalues, \( G \) has \( n \) connected components
Since $|V_1| = N_1$ and $|V_2| = N_2 = N - N_1$, we have the constraint

$$\sum_{i} s_i = \sum_{i \in V_1} (+1) + \sum_{i \in V_2} (-1) = N_1 - N_2 \Rightarrow 1^T s = N_1 - N_2$$

Minimum-cut criterion for graph bisection yields the formulation

$$\hat{s} = \arg \min_{s \in \{\pm 1\}^{N_v}} s^T L s, \text{ s. to } 1^T s = N_1 - N_2$$

Again, binary constraints $s \in \{\pm 1\}^{N_v}$ render cut minimization hard

Will relax binary constraints as with modularity maximization
Further intuition

- Since \( s^T L s = \sum_{(i,j) \in E} (s_i - s_j)^2 \), the minimum-cut formulation is

\[
\hat{s} = \arg \min_{s \in \{-1,1\}^{N_v}} \sum_{(i,j) \in E} (s_i - s_j)^2, \quad \text{s. to } 1^T s = N_1 - N_2
\]

- Now, forget for a second that we proved \( R(s) = \frac{1}{2} s^T L s \)
- Does this equivalent cost function make sense? Absolutely!
  - \( \Rightarrow \) Edges joining vertices in the same group do not add to the sum
  - \( \Rightarrow \) Edges joining vertices in different groups add 4 to the sum

- Minimize cut: assign values \( s_i \) to nodes \( i \) such that few edges cross 0
Minimum-cut relaxation

- Relax the constraint $s \in \{\pm 1\}^{N_v}$ to $s \in \mathbb{R}^{N_v}$, $\|s\| = 1$
- We are left with the quadratic problem [Fiedler ’73]

$$\hat{s} = \arg\min_s s^\top L s, \quad \text{s. to } 1^\top s = N_1 - N_2 \text{ and } s^\top s = 1$$

- Straightforward to solve using Lagrange multipliers

- Characterization of the solution $\hat{s}$:

$$\hat{s} = v_2 + \frac{N_1 - N_2}{N_v} 1$$

⇒ The ‘second-smallest’ eigenvector $v_2$ of $L$ satisfies $1^\top v_2 = 0$

⇒ The minimum cut is $R(\hat{s}) = \hat{s}^\top L \hat{s} = v_2^\top L v_2 \propto \lambda_2$

- If the graph $G$ is disconnected then we know $\lambda_2 = 0 = R(\hat{s})$
- If $G$ is amenable to bisection, the cut is small and so is $\lambda_2$
Spectral graph bisection

How do we obtain the binary cluster labels $s \in \{\pm 1\}^{N_v}$ from $\hat{s} \in \mathbb{R}^{N_v}$?

Again we maximize the similarity measure $s^\top \hat{s}$ which gives

$$s_i = \text{sign}([v_2]_i) := \begin{cases} +1, & [v_2]_i > 0 \\ -1, & [v_2]_i \leq 0 \end{cases}$$

Spectral graph bisection algorithm

S1: Compute the Laplacian matrix $L$ with entries $L_{ij} = D_{ij} - A_{ij}$

S2: Find the ‘second smallest’ eigenvector $v_2$ of $L$

S3: Cluster membership of vertex $i$ is $s_i = \text{sign}([v_2]_i)$

Complexity: efficient Lanczos algorithm variant in $O(N_e \frac{N_e}{\lambda_3 - \lambda_2})$ time

Nomenclature: $v_2$ is known as the Fiedler vector

Eigenvalue $\lambda_2$ is the Fiedler value, or the algebraic connectivity of $G$
Suppose $G$ is disconnected and has two connected components.

- $L$ is block diagonal and two smallest eigenvectors indicate groups, i.e.,

\[
v_1 = [1, 1, \ldots, 1, 0, \ldots, 0]^\top \quad \text{and} \quad v_2 = [0, 0, \ldots, 0, 1, \ldots, 1]^\top
\]

- If $G$ is connected but amenable to bisection, $v_1 = 1$ and $\lambda_2 \approx 0$.

- Also, $1^\top v_2 = \sum_i [v_2]_i = 0 \Rightarrow$ Positive and negative entries in $v_2$. 

![Graph and graph signal vector](image.png)
Unknown community sizes

- Consider the graph bisection problem with unknown group sizes
- Minimizing the graph cut may be no longer meaningful!

Cost $R :=\sum_{i\in V_1, j\in V_2} A_{ij}$ agnostic to groups’ internal structure

Better criterion is the ratio cut (a.k.a. conductance) defined as

$$C = \frac{R}{\min(|V_1|, |V_2|)}$$

Balanced partitions: a small community is penalized by the cost
Ratio-cut minimization

- Fix a bisection $S$ of $G$ into groups $V_1$ and $V_2$
- Define $f : f(S) = [f_1, \ldots, f_{N_v}]^T \in \mathbb{R}^{N_v}$ with entries
  \[
  f_i = \begin{cases} 
  \sqrt{\frac{|V_2|}{|V_1|}}, & \text{vertex } i \text{ belongs to } V_1 \\
  -\sqrt{\frac{|V_1|}{|V_2|}}, & \text{vertex } i \text{ belongs to } V_2
  \end{cases}
  \]
- One can establish the following properties:
  - $P1$: $f^T L f = N_v C(S)$;
  - $P2$: $\sum_i f_i = 0$, i.e., $1^T f = 0$; and
  - $P3$: $\|f\|^2 = N_v$
- From $P1$-$P3$ it follows that ratio-cut minimization is equivalent to
  \[
  \min_S f^T L f, \quad \text{s. to } 1^T f = 0 \text{ and } f^T f = N_v
  \]
Ratio cut and spectral graph bisection

- Ratio-cut minimization is also NP-hard. Relax to obtain

\[ \hat{s} = \arg \min_{s \in \mathbb{R}^{N_V}} s^\top L s, \quad \text{s. to } 1^\top s = 0 \text{ and } s^\top s = N_v \]

- Partition \( \hat{S} \) again given by the spectral graph bisection algorithm
- S1: Compute the Laplacian matrix \( L \) with entries \( L_{ij} = D_{ij} - A_{ij} \)
- S2: Find the ‘second smallest’ eigenvector \( v_2 \) of \( L \)
- S3: Cluster membership of vertex \( i \) is \( s_i = \text{sign}([v_2]_i) \)

- Theoretical optimality guarantee:

\[ \frac{\lambda_2}{2} \leq \min_S C(S) \leq C(\hat{S}) \leq \lambda_2 \]

- Multiple (> 2) communities through e.g., repeated graph bisection
- Or cluster rows of multiple eigenvectors of \( L \) with e.g., k-means
Glossary

- Network community
- (Strong) triadic closure
- Clustering coefficient
- Bridges and local bridges
- Tie strength
- Neighborhood overlap
- Strength of weak ties
- Zachary’s karate club
- Community detection
- Graph partitioning and bisection
- Non-overlapping communities
- Edge betweenness centrality
- Girvan-Newmann method
- Hierarchical clustering
- Dendrogram
- Single and complete linkage
- Modularity
- Spectral modularity maximization
- Modularity and Laplacian matrices
- Minimum-cut partitioning
- Fiedler vector and value
- Ratio-cut minimization