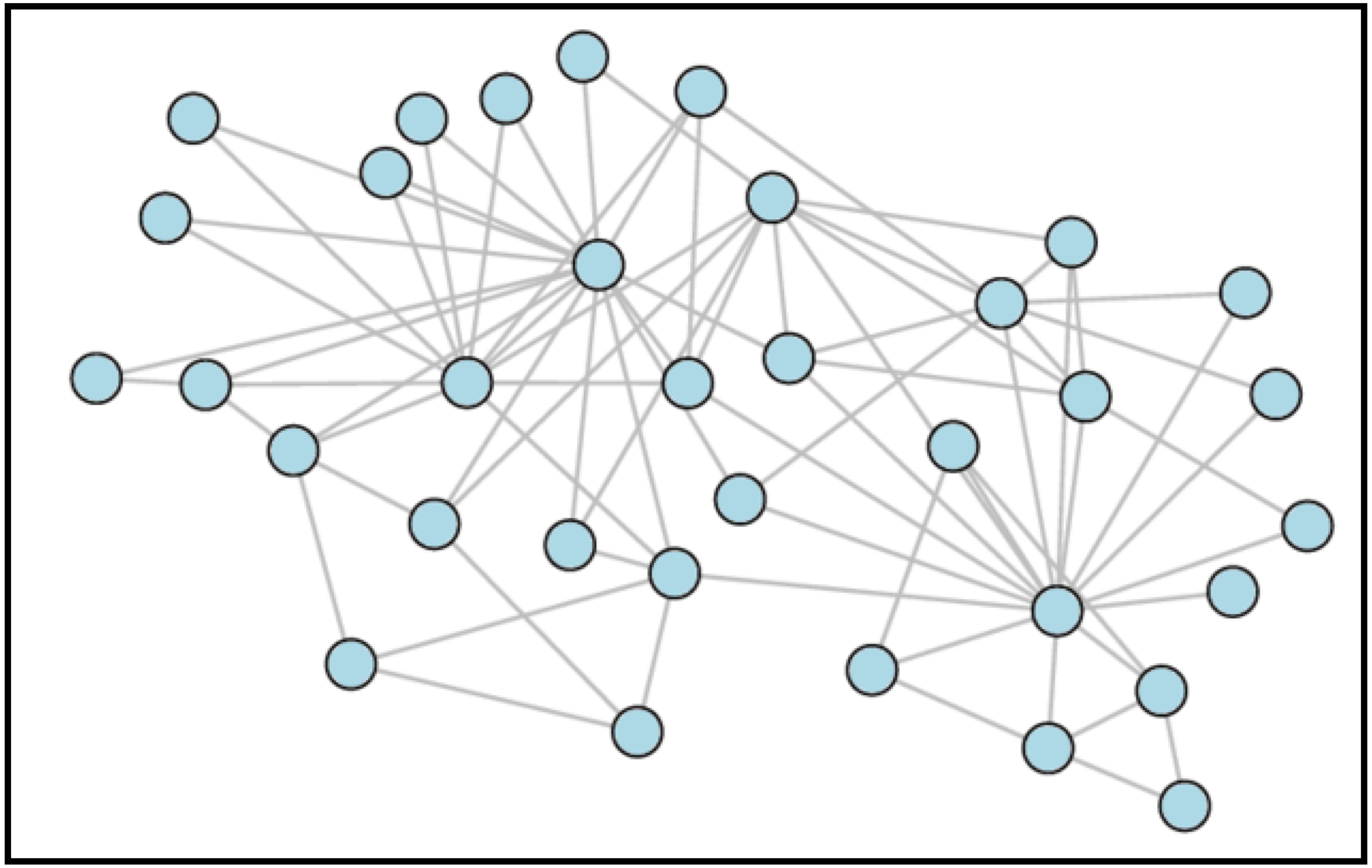
Community Detection and Sampling on Networks

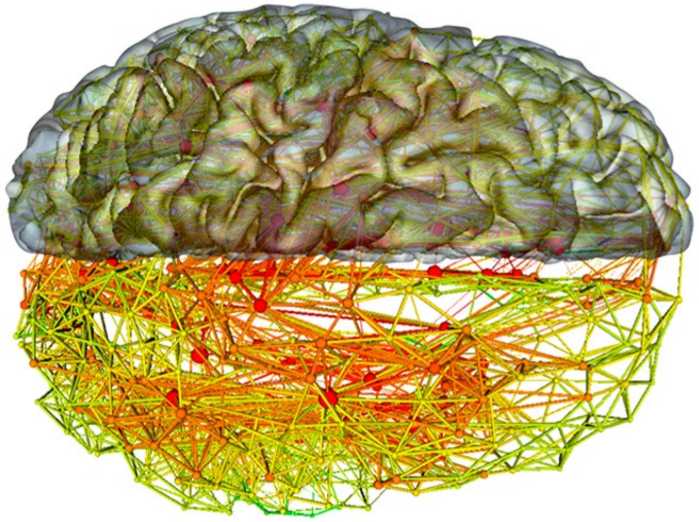
Yilin Zhang

Facebook

Jan 30 2020

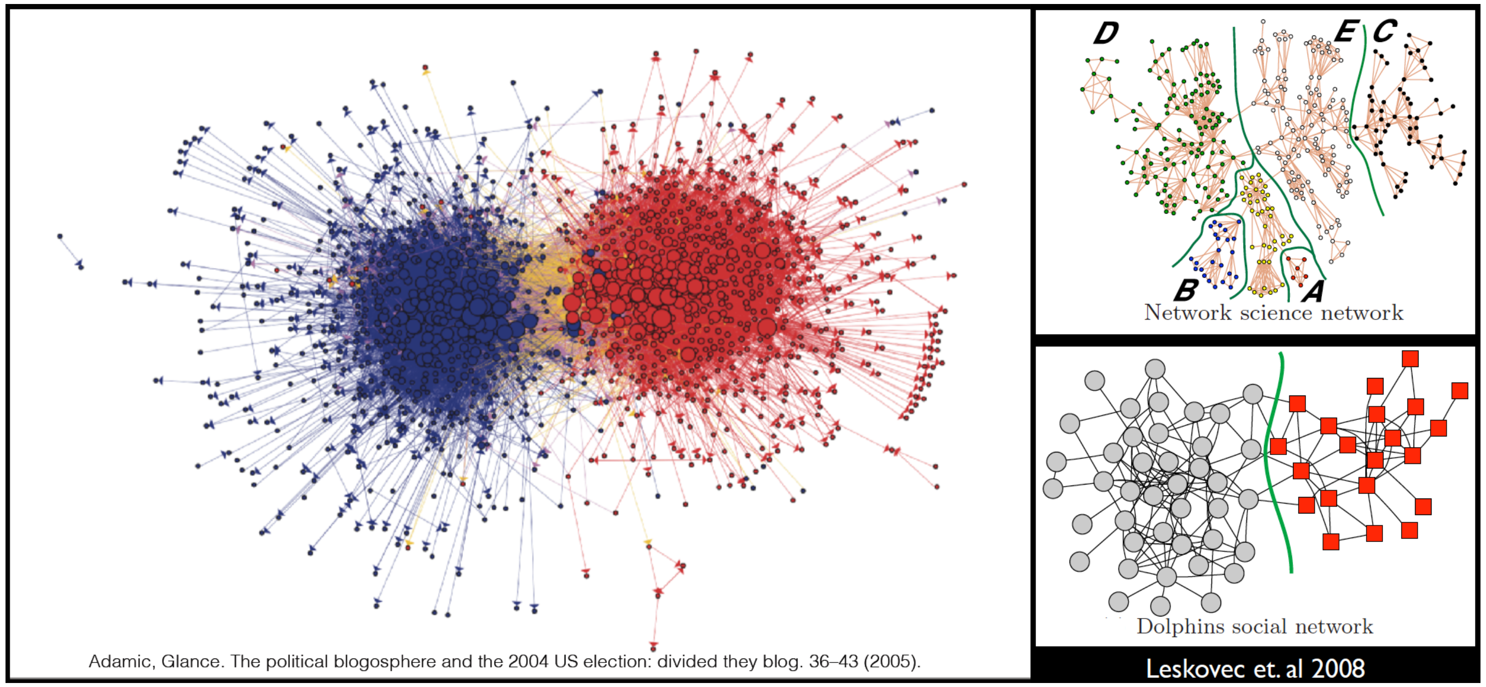
What are networks / graphs?





There are many network-related problems.

How to find communities?

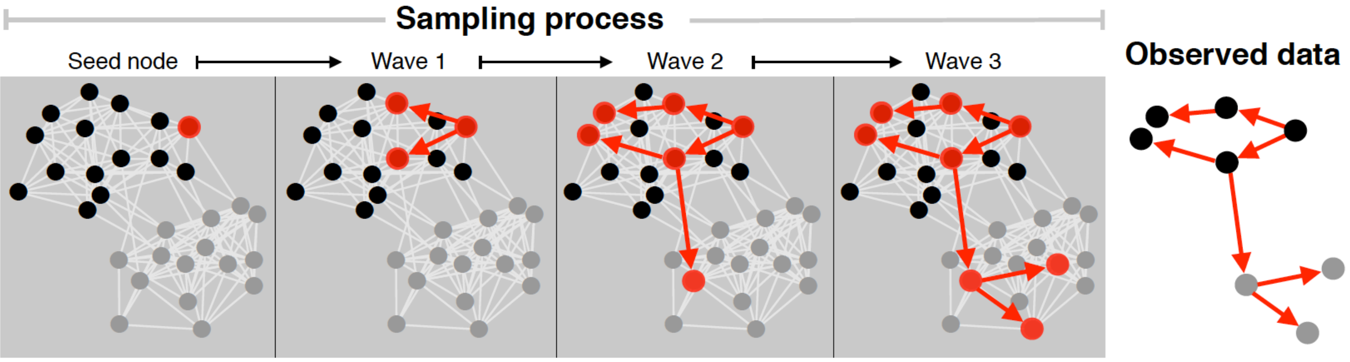


Adamic, Glance. The political blogosphere and the 2004 US election: divided they blog. 36–43 (2005).

Leskovec et. al 2008

How to simultaneously analyze different sources of data (e.g. graph and text)?

How to sample on a network?
How to estimate on the connected samples?



Rohe 2015

My research discussed some Network-related problems.

Network related work	How to find communities?	Regularized Spectral Clustering	**Y. Zhang**, K. Rohe. Understanding Regularized Spectral Clustering via Graph Conductance. *NeurIPS, 2018*.
	How to simultaneously analyze different sources of data (e.g. graph and text)?	Graph Contextualization with PairGraphText	**Y. Zhang**, M. Berthe, C. Wells, K. Michalska, K. Rohe. Discovering Political Topics in Social Network Discussion threads with Graph Contextualization. *The Annals of Applied Statistics, 12(2), 1096-1123, 2018*.
	How to sample on a network?	Respondent Driven Sampling	**Y. Zhang**, K. Rohe, S. Roch. Reducing Seed Bias in Respondent-Driven Sampling by Estimating Block Transition Probabilities. *under revision at The Annals of Statistics*.

Other Work	Domain Adaptation	H. Zhou, **Y. Zhang**, V. Ithapu, S. Johnson, G. Wahba, V. Singh. When can Multi-Site Datasets be Pooled for Regression: Hypothesis Tests, l2-consistency and Neuroscience Applications. *ICML, 2017*.
	Election Polls	**Y. Zhang**, Q. Li, F. Charles, K. Rohe. Direct Evidence for Null Volatility in Election Polling. *in preparation*.
	Gravitational Waves	X. Zhu, L. Wen, G. Hobbs, **Y. Zhang**, et al. Detection and localization of single-source gravitational waves with pulsar timing arrays. *MNRAS, 449:16501663, 2015*.

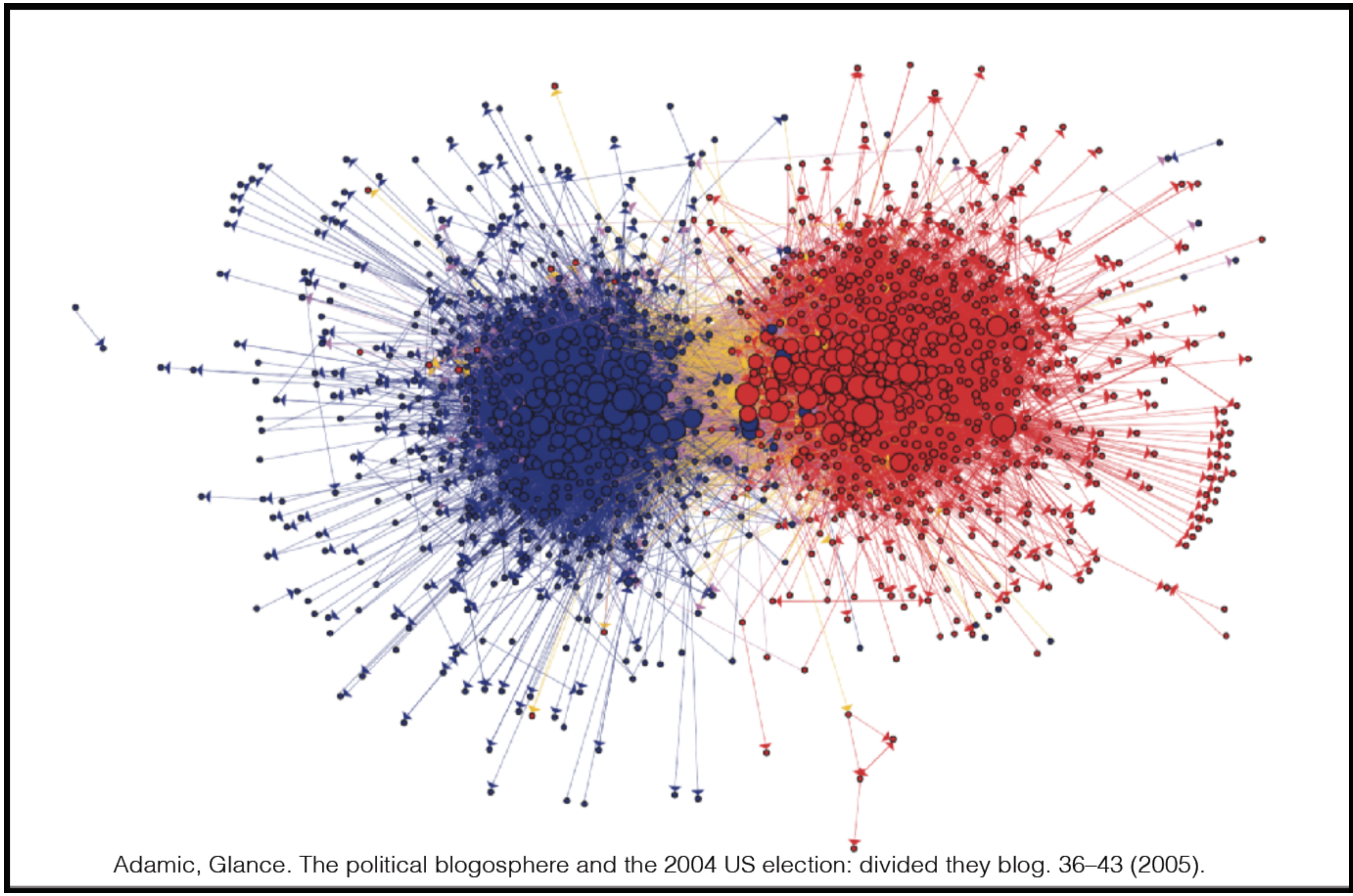
Understanding Regularized Spectral Clustering via Graph Conductance

Yilin Zhang & Karl Rohe

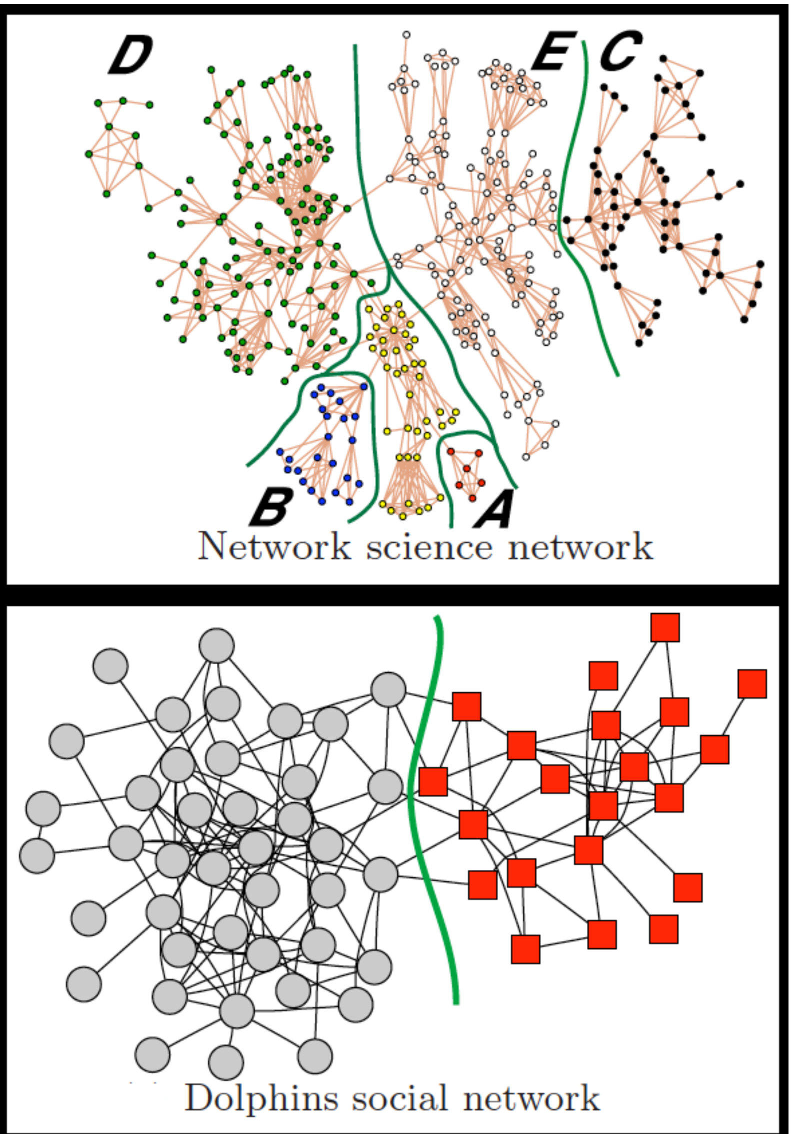
University of Wisconsin-Madison

NeurIPS 2018

Given a graph, one key problem is to find communities.



Adamic, Glance. The political blogosphere and the 2004 US election: divided they blog. 36–43 (2005).



Network science network

Dolphins social network

Leskovec et. al 2008

Spectral Clustering is one popular approach.

Graph $G = (V, E)$

Adjacency Matrix $A \in \{0,1\}^{N \times N}$, where $A_{ij} = \begin{cases} 1 & \text{if } i \sim j \\ 0 & \text{o.w.} \end{cases}$

Graph Laplacian $L = I - D^{-1/2} A D^{-1/2}$, where $D_{ii} = \sum_j A_{ij}$ is degree of node i.

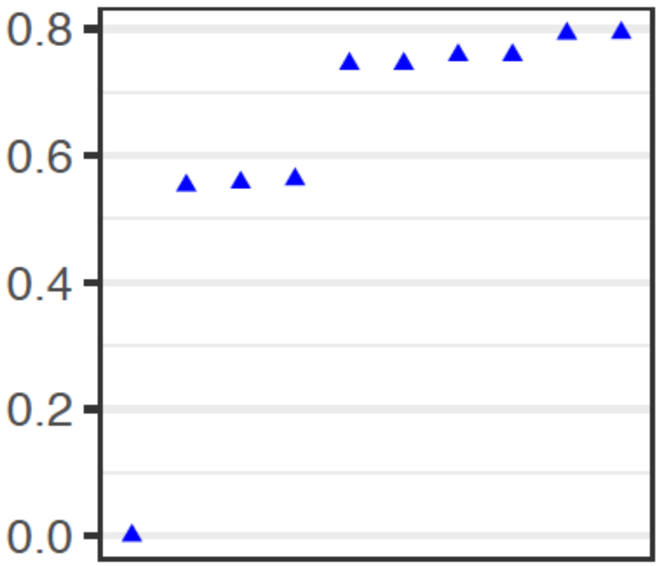
Spectral Clustering partitions the graph based on **top eigenvectors** of L.

an example

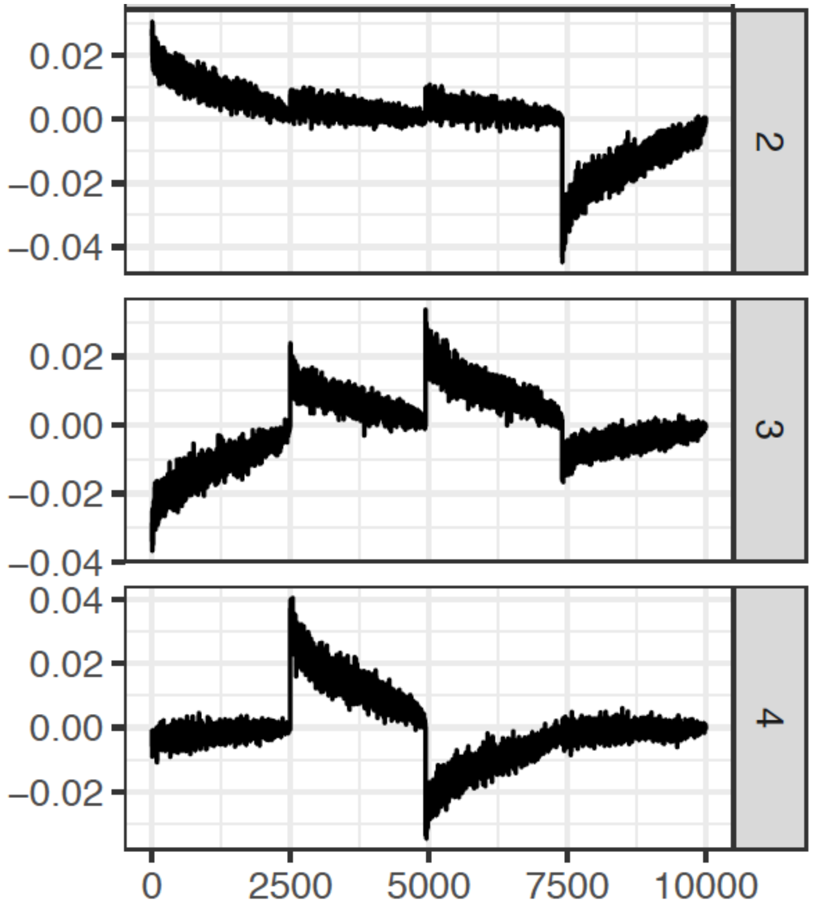
A simulated network with
N = 10000 nodes
K = 4 equal-size blocks

Two nodes in the same block?	Yes	No
connect (edge) probability	0.8	0.2

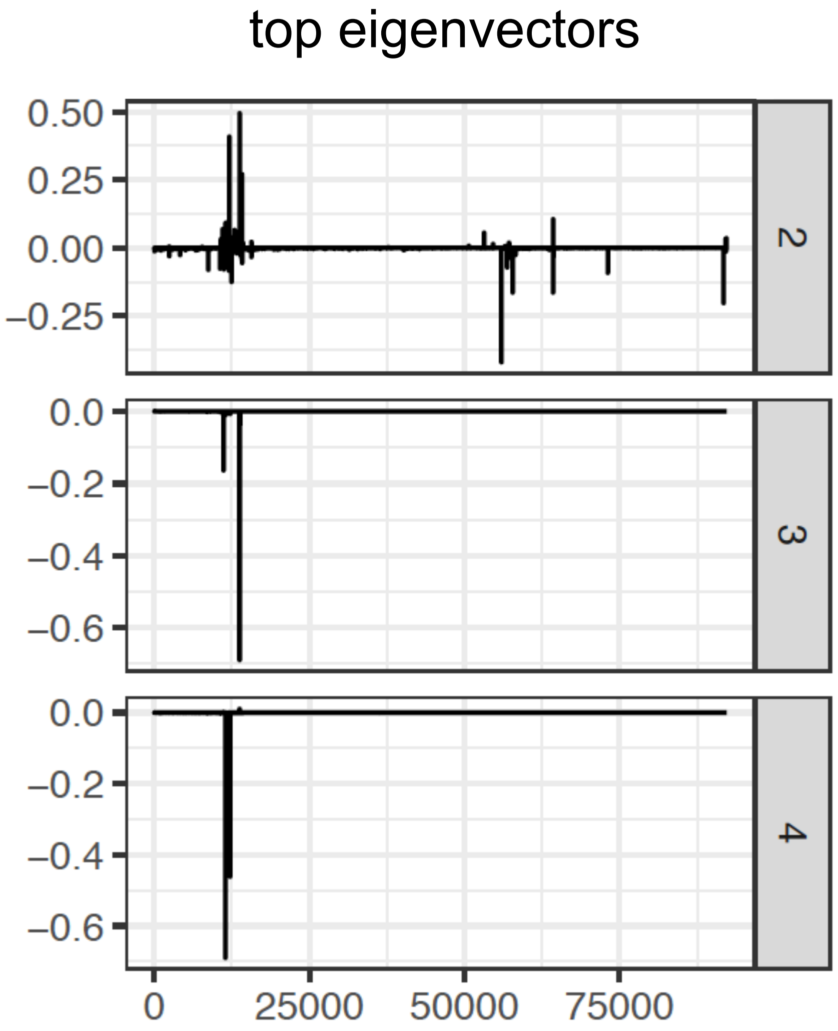
top (smallest) 10 eigenvalues



top 2 - 4 eigenvectors



However, in practice, Spectral Clustering always **fails**.



Social network
[Y. Zhang, et al. 2018]

In practice, eigenvectors always **localize** on just several nodes.

This leads to highly **unbalanced** clusters.

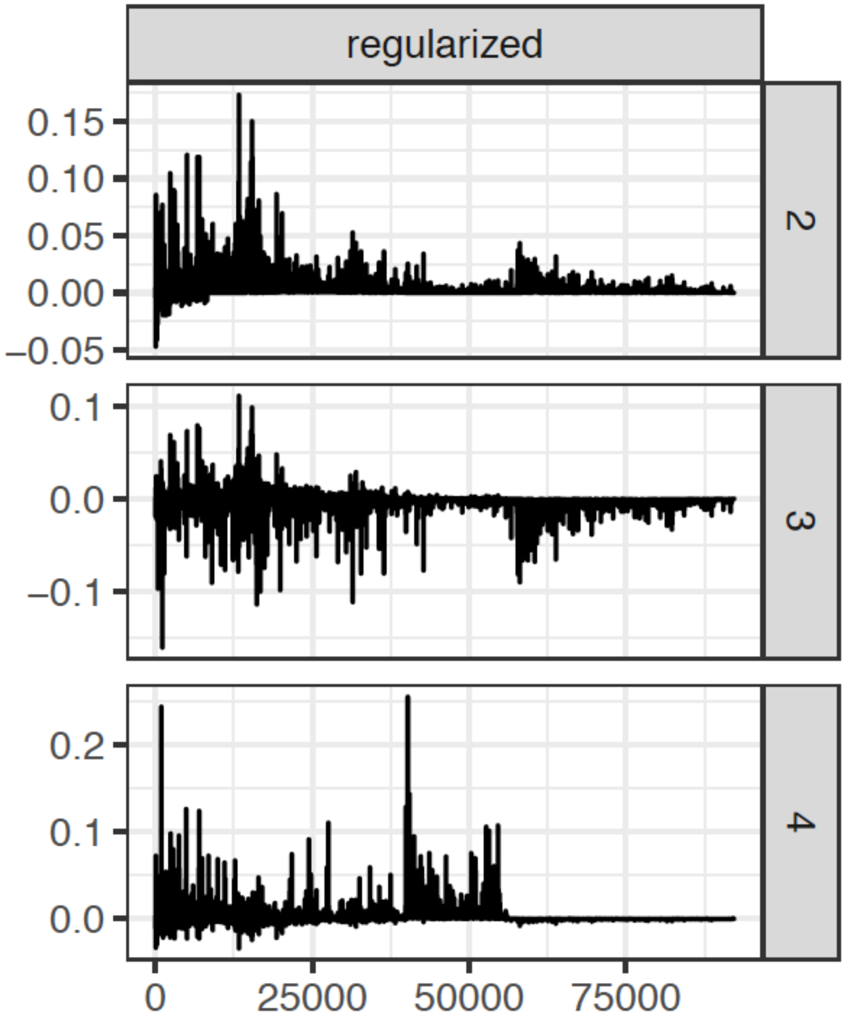
Regularization solves the problem.

Regularization adds a tiny edge on every pair of nodes. i.e. replace G by G_τ.

Regularized Adjacency Matrix $A_\tau = A + \frac{\tau}{N}J$, where J is an all-one matrix.

Regularized Graph Laplacian $L_\tau = I - D_\tau^{-1/2} A_\tau D_\tau^{-1/2}$, where $D_\tau = D + \tau I$.

[T. Qin et al. 2013, A. Amini et al. 2013, K. Chaudhuri et al. 2012,]



Social network
[Y. Zhang, et al. 2018]

In practice, regularization **delocalizes** eigenvectors.

This leads to more **balanced** clusters.

Why?

Why Spectral Clustering fails? graph conductance & noises

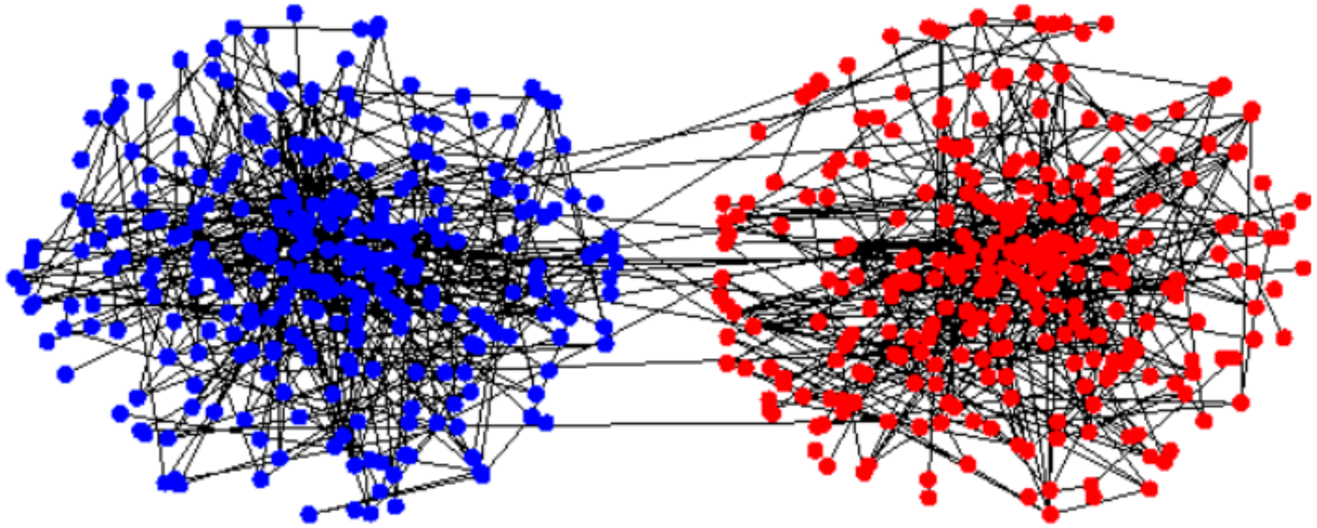
Spectral Clustering likes sets with small conductance!

Given a node set S with $vol(S) \leq vol(S^c)$, its graph conductance is:

$$\phi(S) = \frac{cut(S)}{vol(S)} = \frac{\text{num of edges cut}}{\text{sum of node degrees}}$$

$\min_S \phi(S)$ relaxes to Spectral Clustering. [U. Luxburg, 2007]

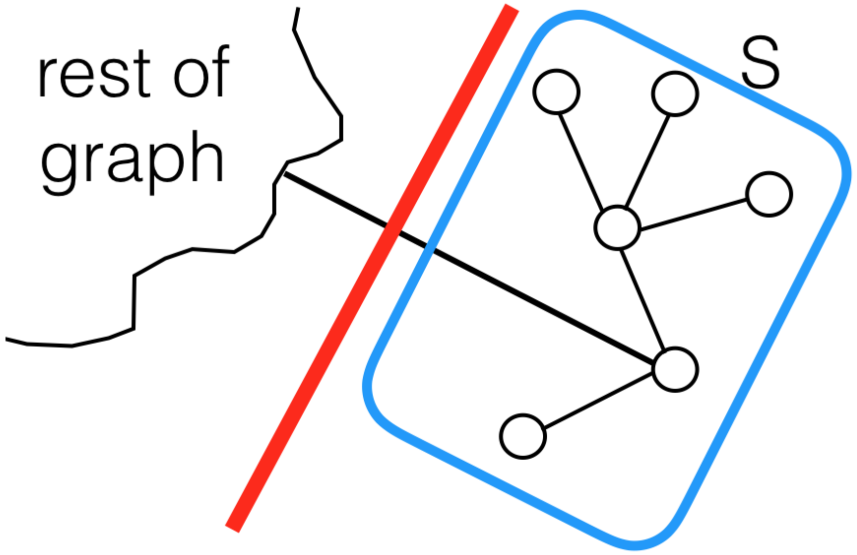
Then, what kinds of sets have small conductance?



g-dangling sets have small conductance!

S is g-**dangling** $\iff$
(1) $|S| = g$, (2) S is a tree, (3) $cut(S) = 1$.

Fact 1: g-dangling set has small conductance $1/(2g-1)$.



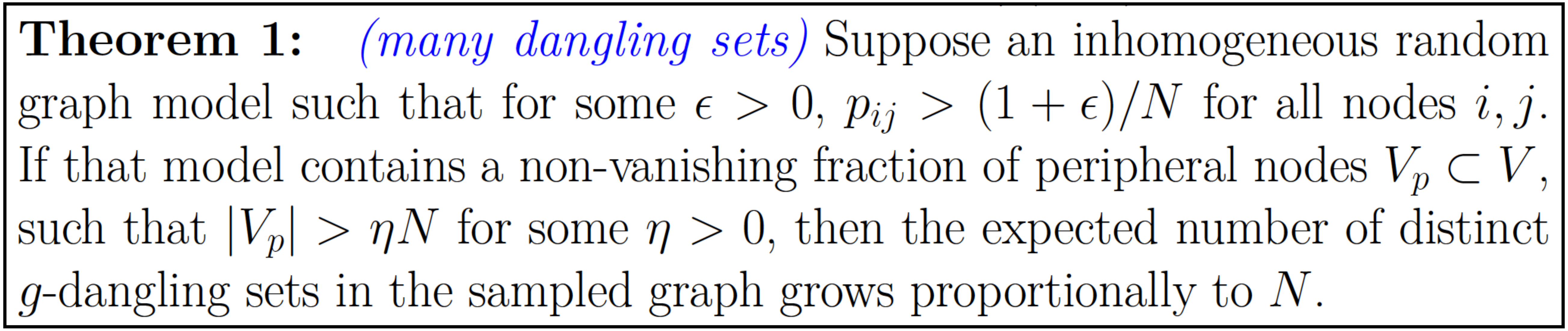
a 6-dangling set

There are many g-dangling sets!

inhomogeneous random graph model: independent edges (e.g. erdös-Rény, SBM)

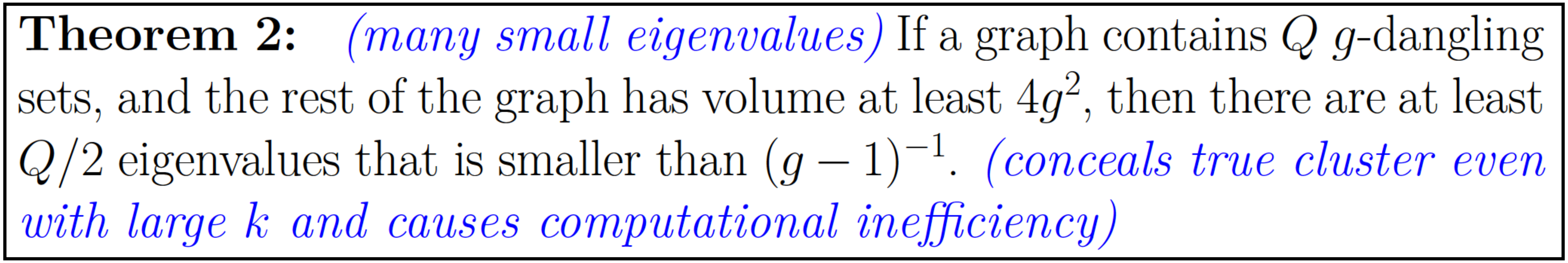
peripheral node: O(1) expected degree

(edge probability with any node is $< b/N$ for some constant b)



Theorem 1: *(many dangling sets)* Suppose an inhomogeneous random graph model such that for some $\epsilon > 0$, $p_{ij} > (1+\epsilon)/N$ for all nodes i, j. If that model contains a non-vanishing fraction of peripheral nodes $V_p \subset V$, such that $|V_p| > \eta N$ for some $\eta > 0$, then the expected number of distinct g-dangling sets in the sampled graph grows proportionally to N.

Graphs with many g-dangling sets have many small eigenvalues.



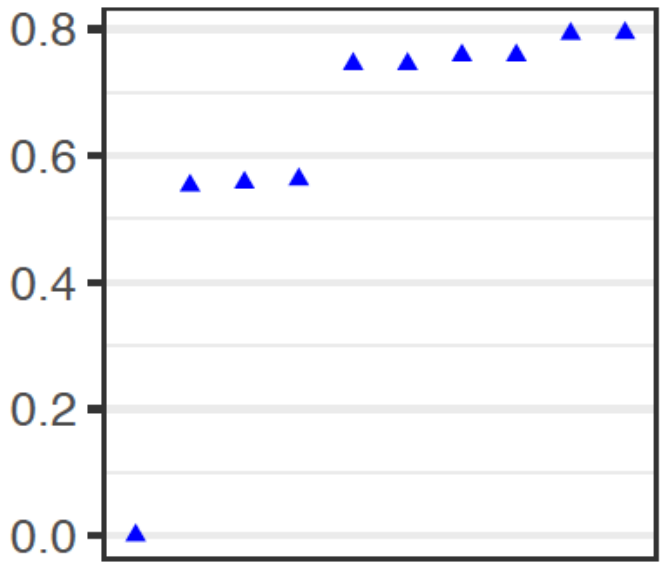
Theorem 2: *(many small eigenvalues)* If a graph contains Q g-dangling sets, and the rest of the graph has volume at least $4g^2$, then there are at least $Q/2$ eigenvalues that is smaller than $(g-1)^{-1}$. *(conceals true cluster even with large k and causes computational inefficiency)*

an example (recall)

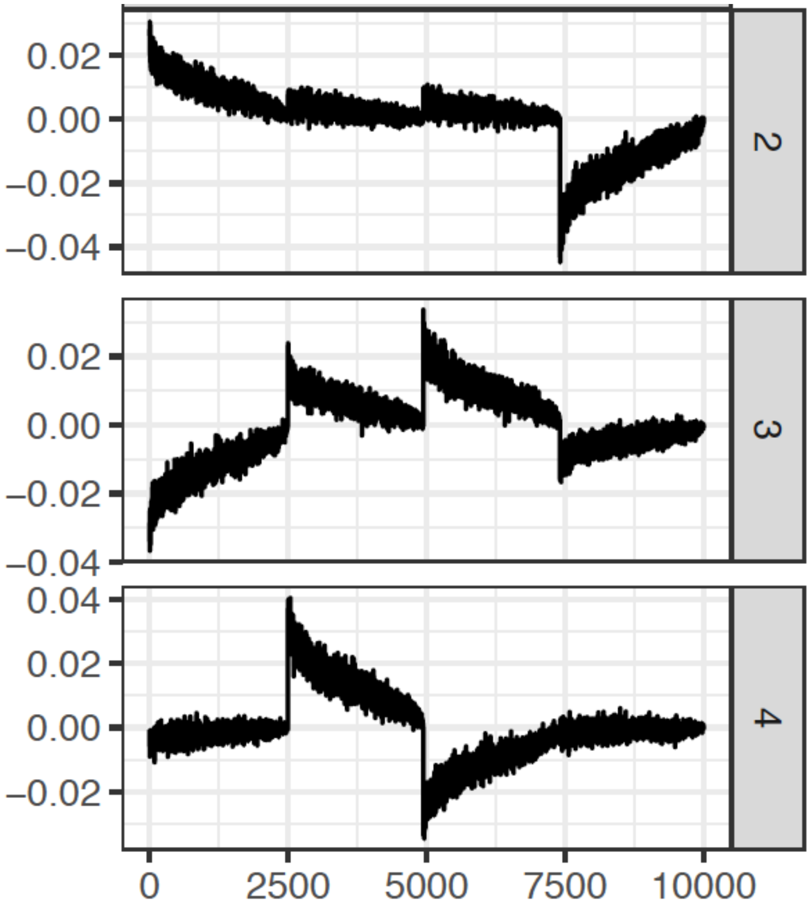
A simulated network with
$N = 10000$ nodes
$K = 4$ equal-size blocks

Two nodes in the same block?	Yes	No
connect (edge) probability	0.8	0.2

top (smallest) 10 eigenvalues



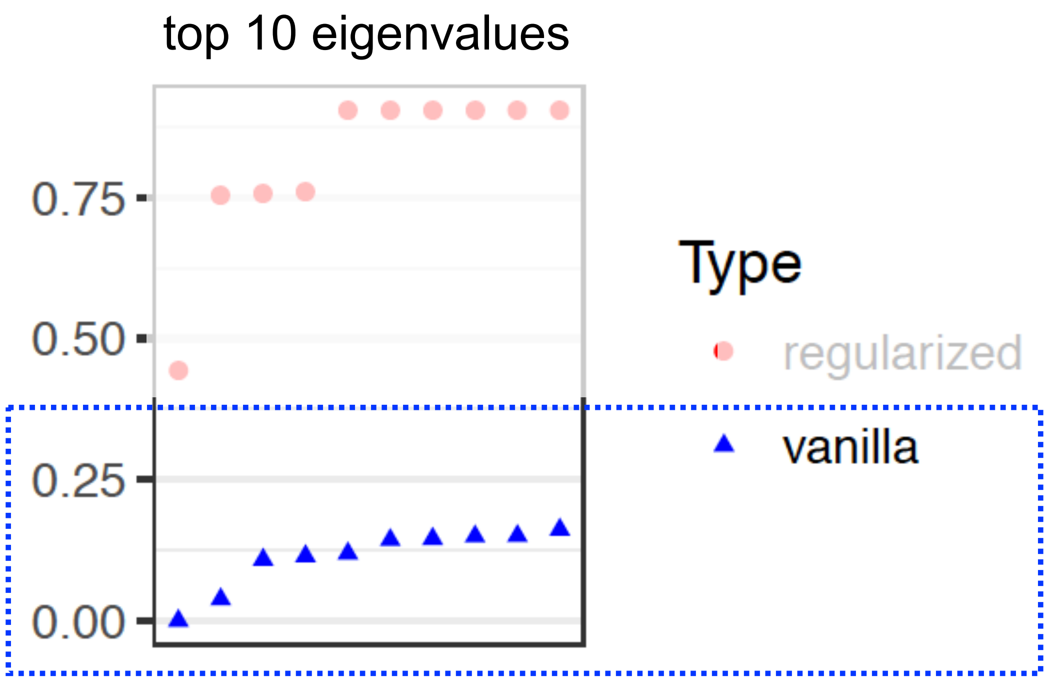
top 2 - 4 eigenvectors

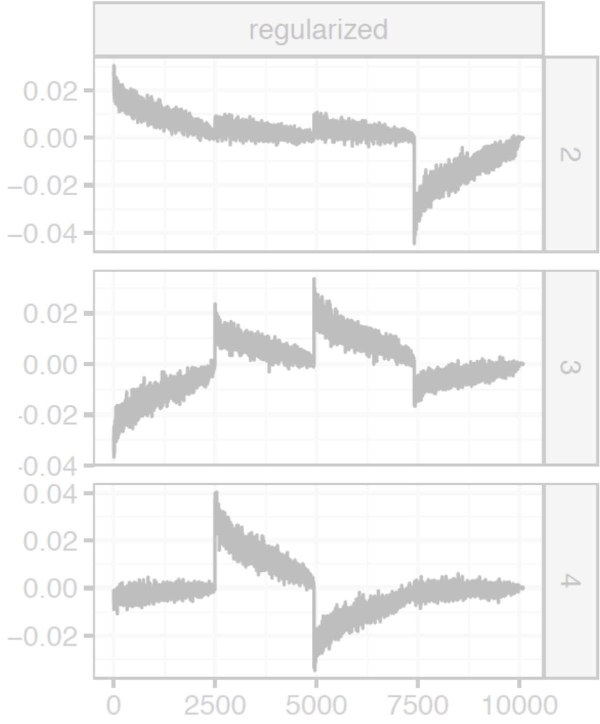


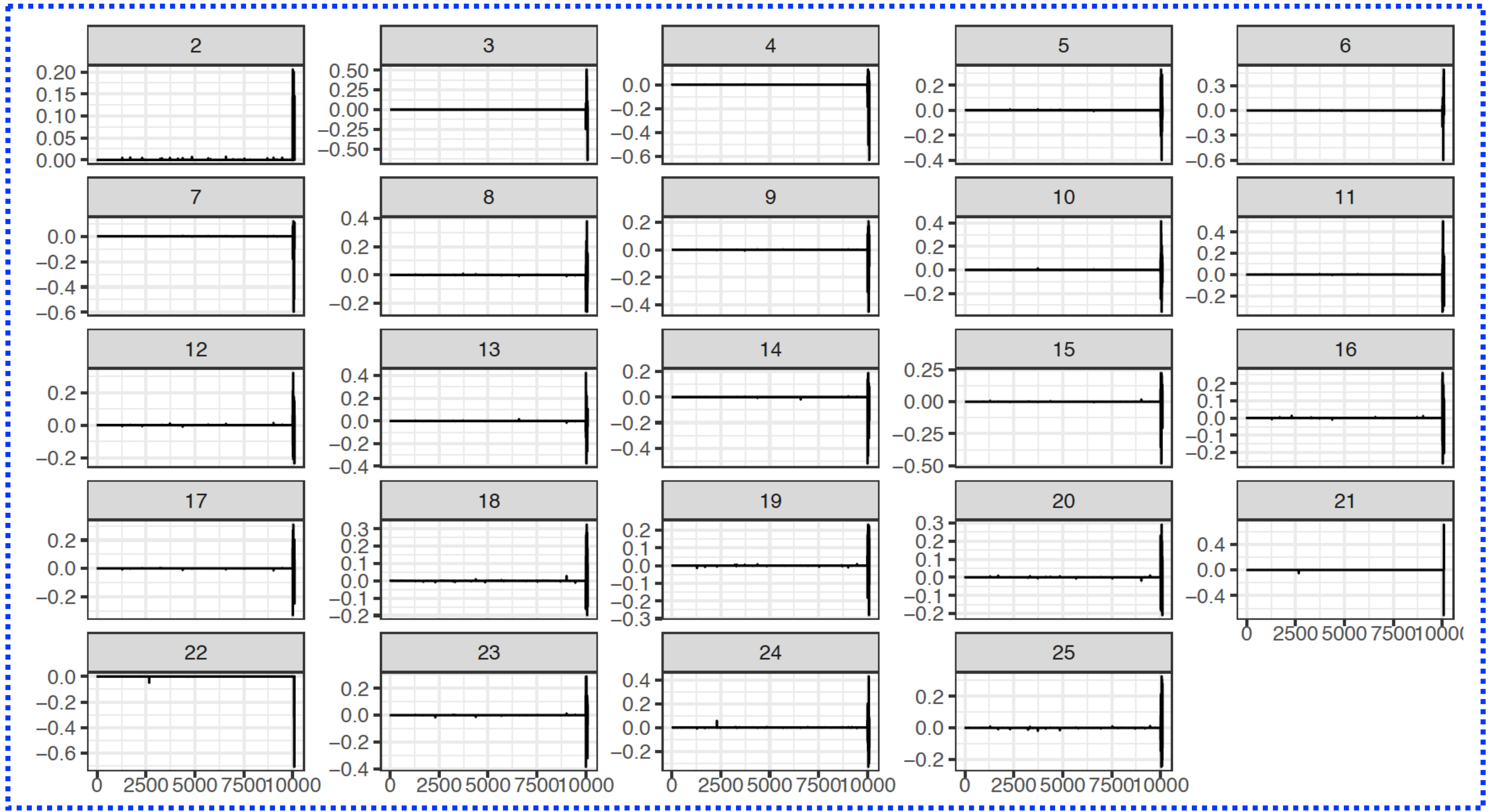
the updated example

A simulated network with
10000 **core** nodes
4 equal-size **core** blocks
100 **peripheral** nodes

Two **core** nodes in the same block?	Yes	No
connect prob	0.8	0.2
peripheral nodes connect prob	0.01	
peripheral - core connect prob	2.5E-05	



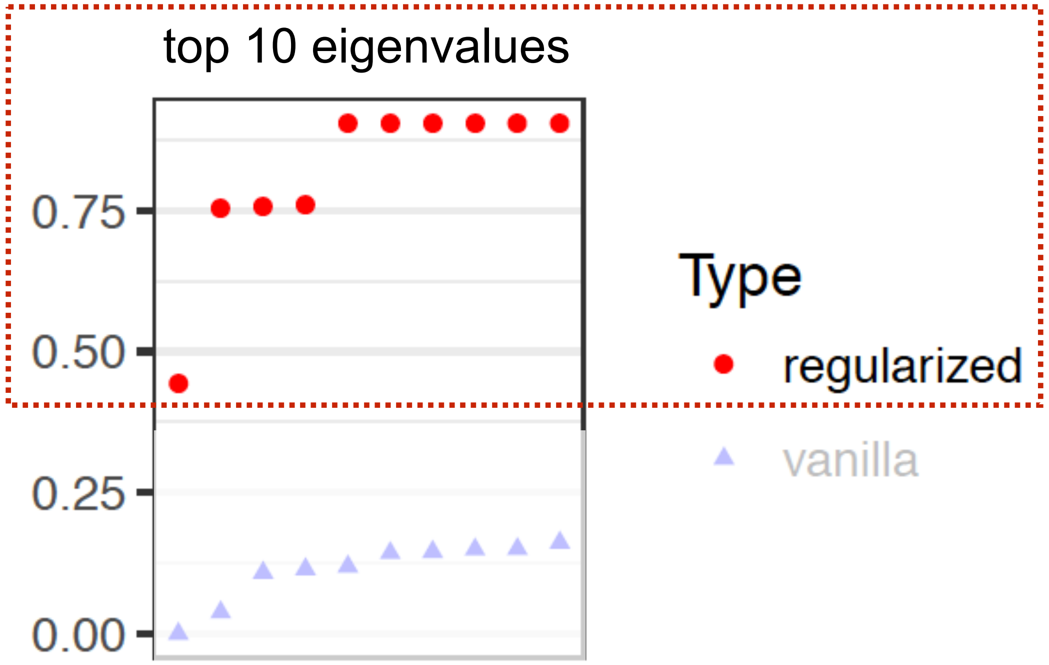


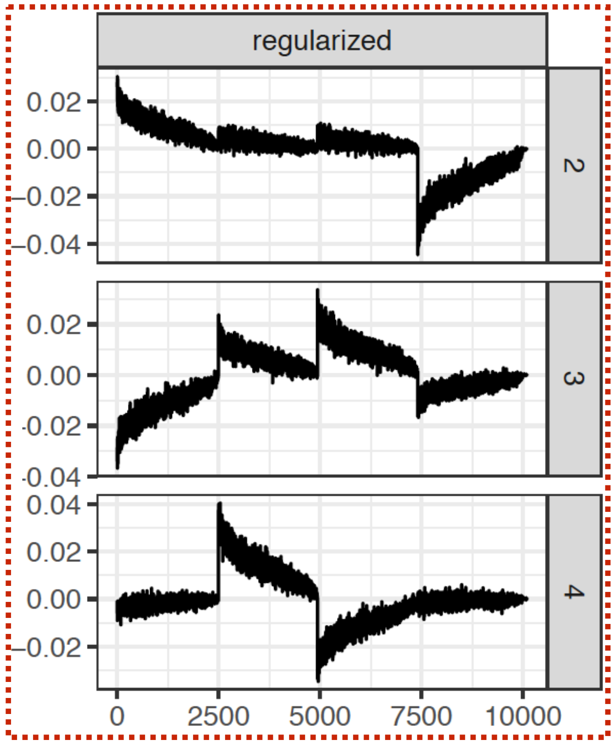


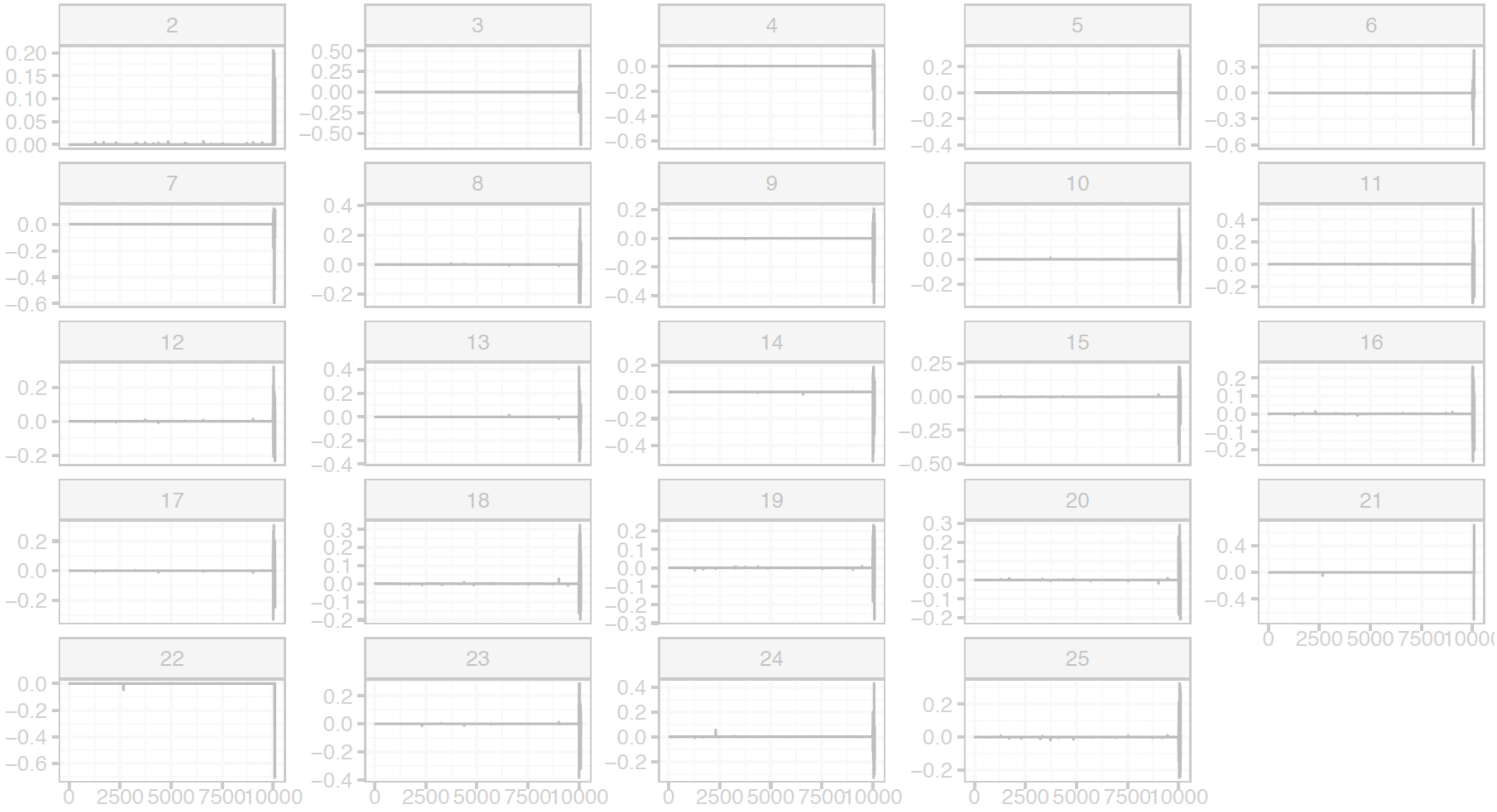
the updated example

A simulated network with
10000 **core** nodes
4 equal-size **core** blocks
100 **peripheral** nodes

Two **core** nodes in the same block?	Yes	No
connect prob	0.8	0.2
peripheral nodes connect prob	0.01	
peripheral - core connect prob	2.5E-05	



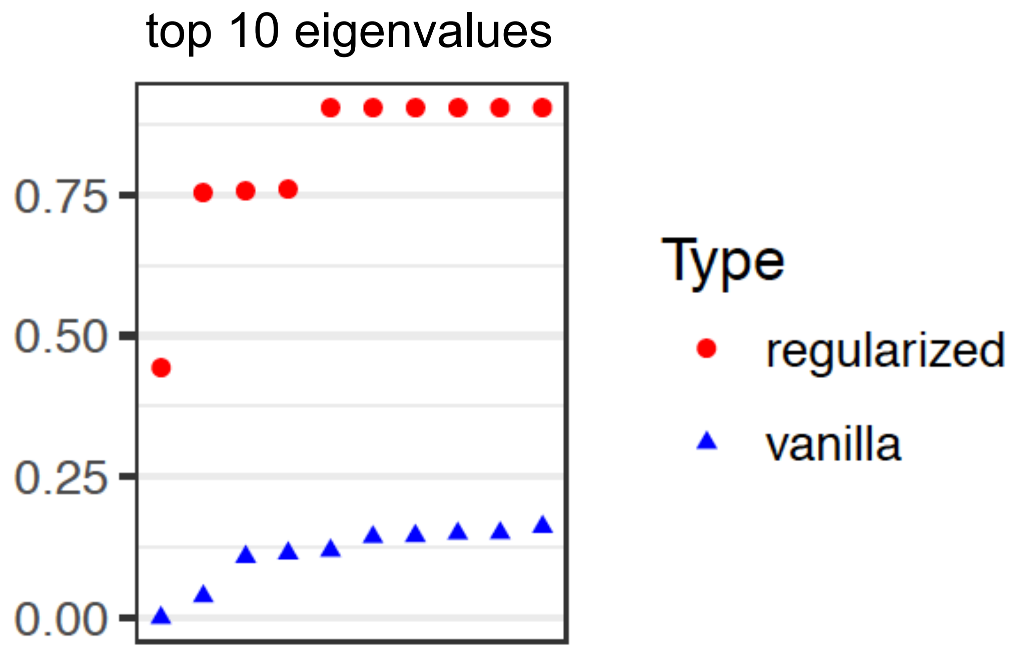


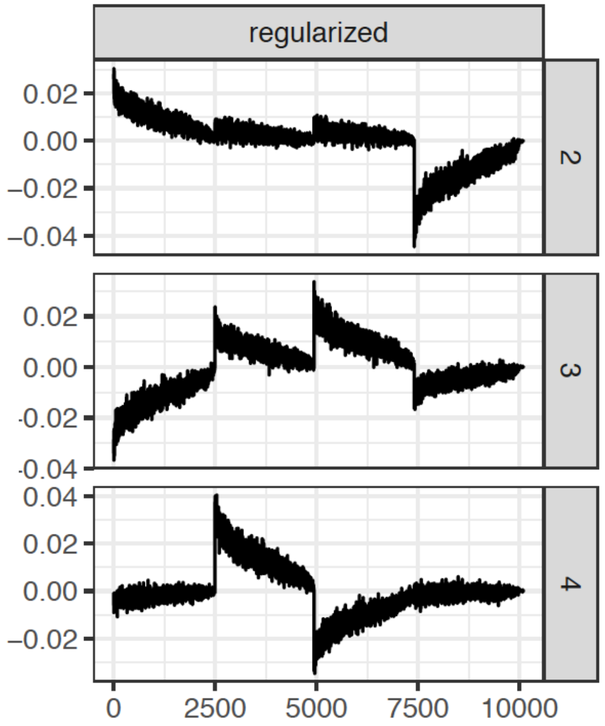


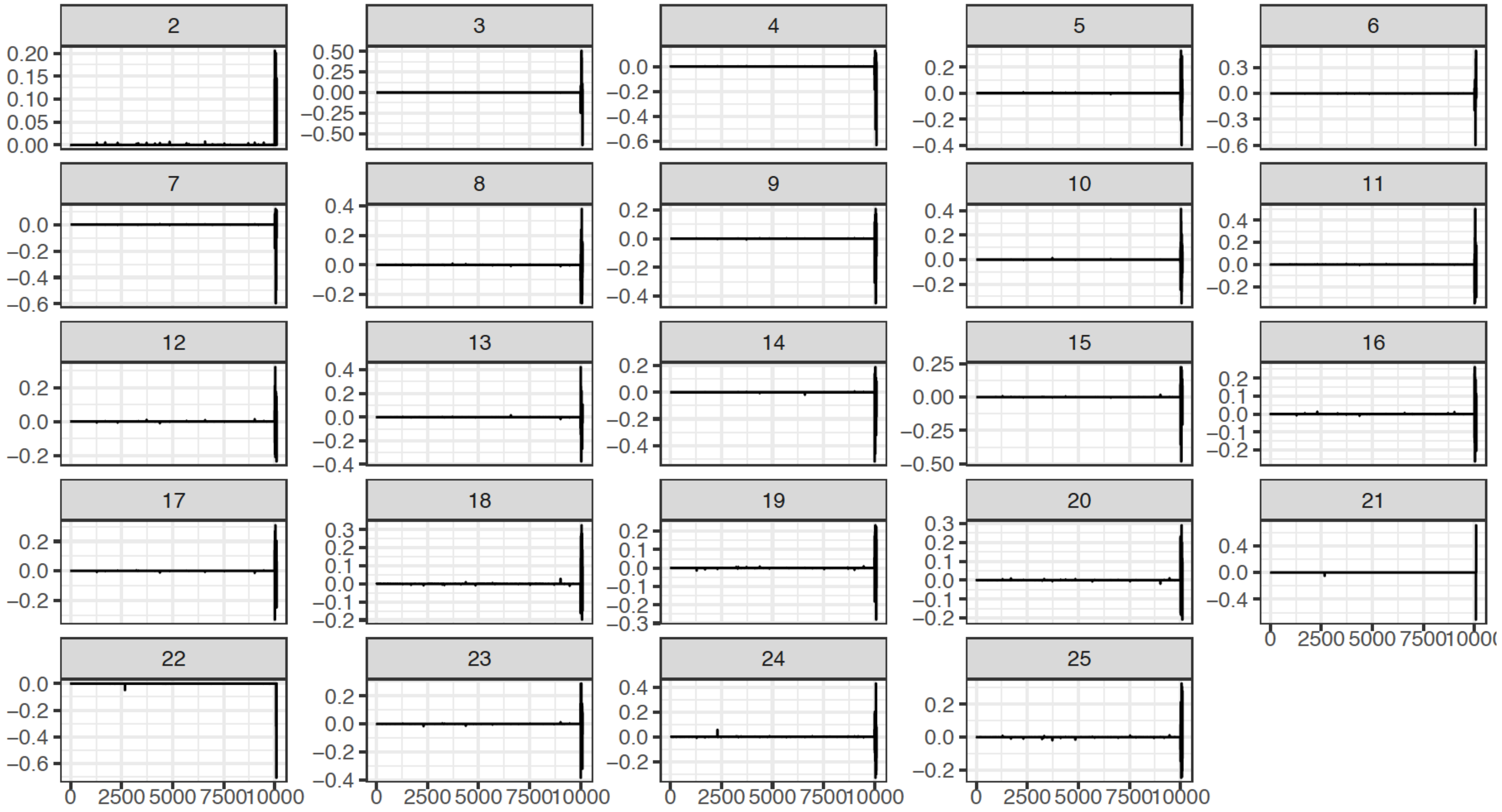
the updated example

A simulated network with
10000 **core** nodes
4 equal-size **core** blocks
100 **peripheral** nodes

Two **core** nodes in the same block?	Yes	No
connect prob	0.8	0.2
peripheral nodes connect prob	0.01	
peripheral - core connect prob	2.5E-05	







Recall: Why Spectral Clustering fails?

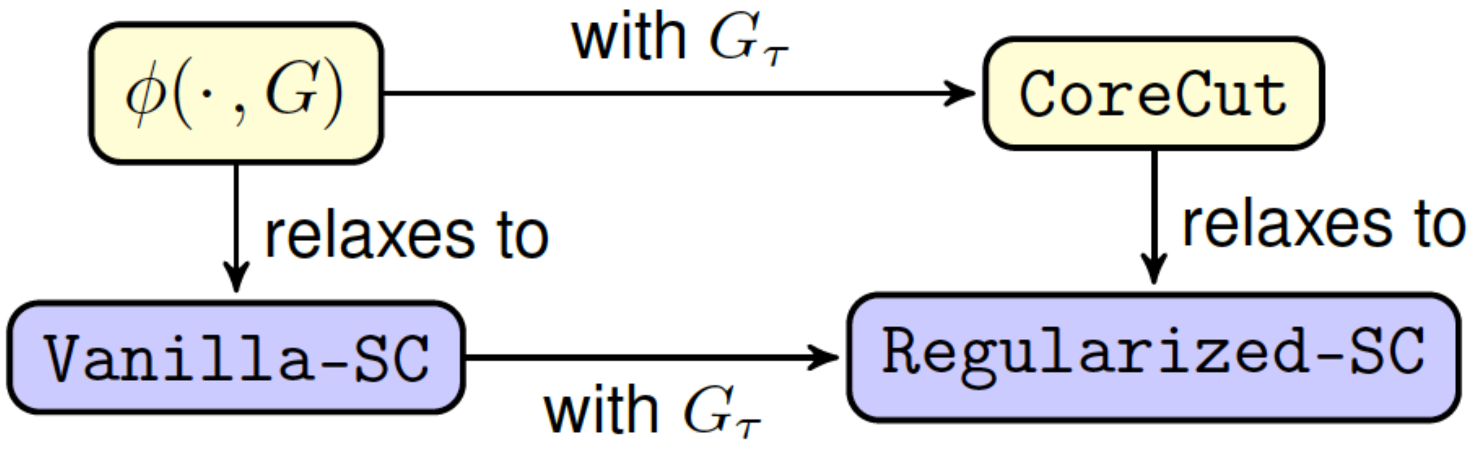
- Spectral Clustering likes sets with small conductance.
- Dangling sets have small conductance.
- There are lots of dangling sets, which lead to lots of small eigenvalues. This conceals true clusters even with large k.

Why does regularization fix this?
Regularization changes the graph conductance.

Regularized SC likes sets with small CoreCut!

CoreCut: (conductance on the regularized graph)

$$\text{CoreCut}_\tau(S) = \frac{cut(S) + \frac{\tau}{N}|S||S^c|}{vol(S) + \tau|S|}$$



$\phi(\cdot\,, G)$

with G_τ

CoreCut

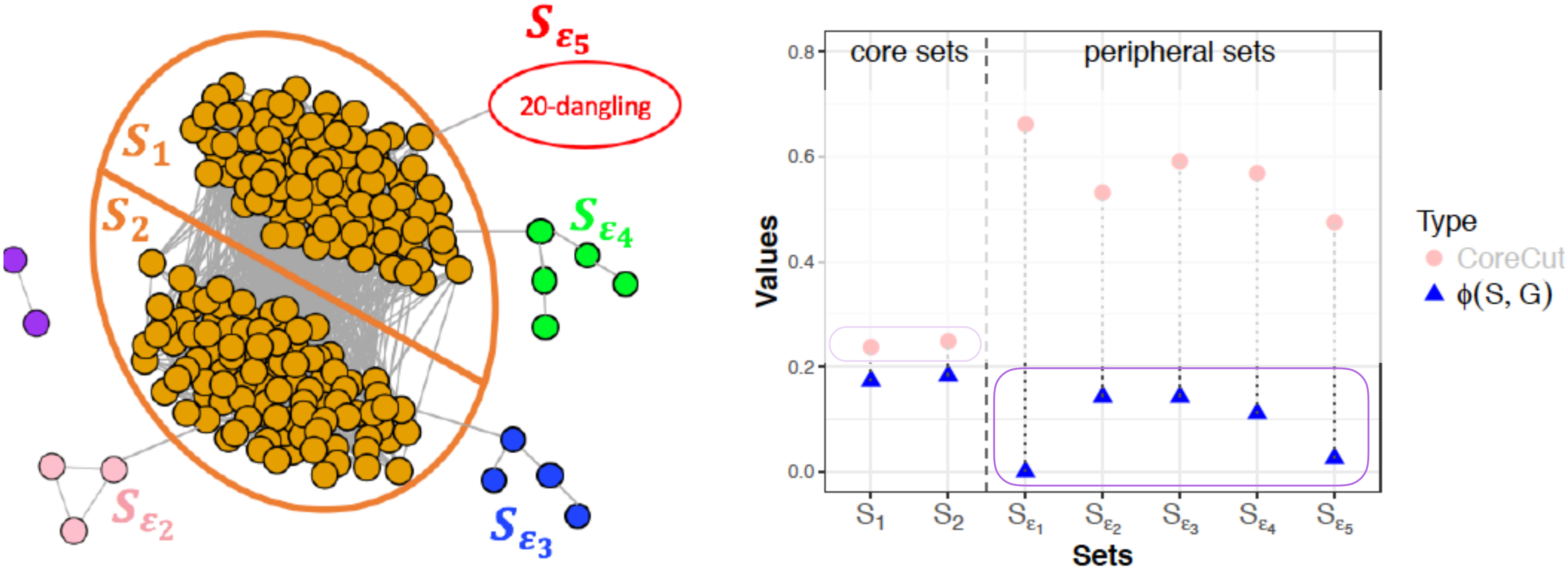
relaxes to

Vanilla-SC

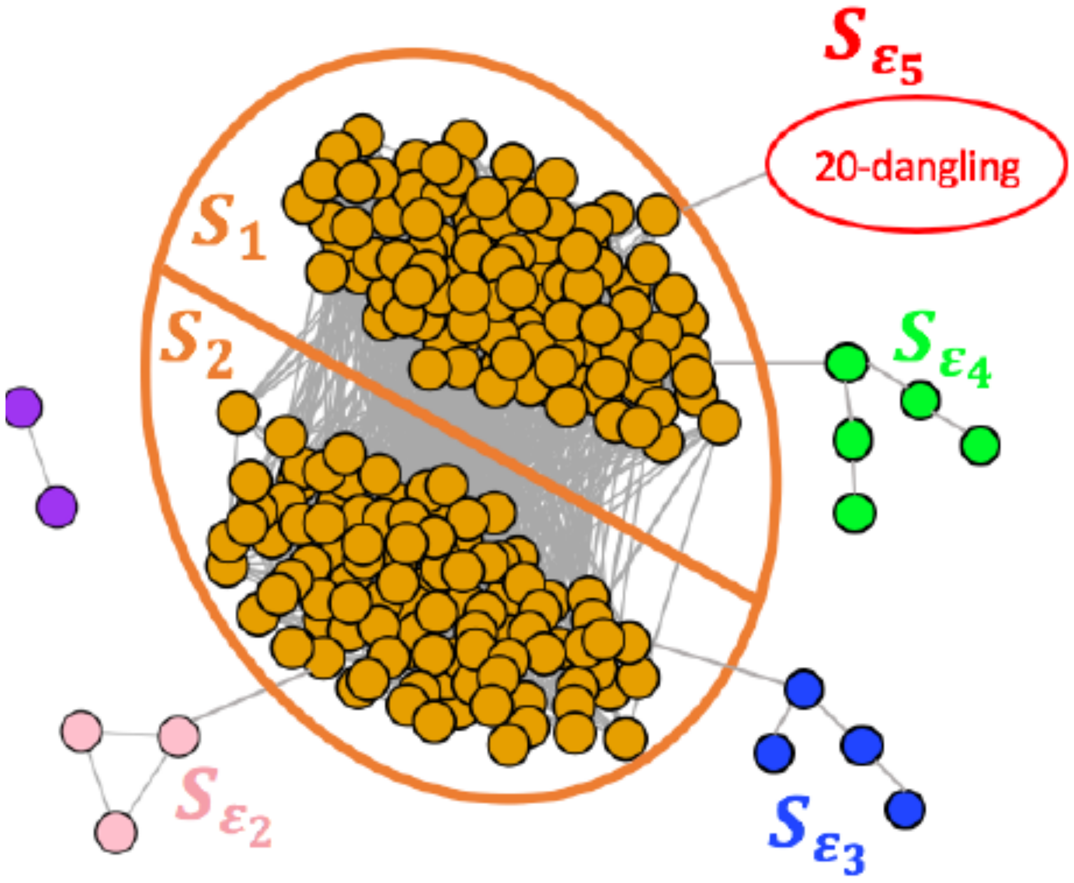
with G_τ

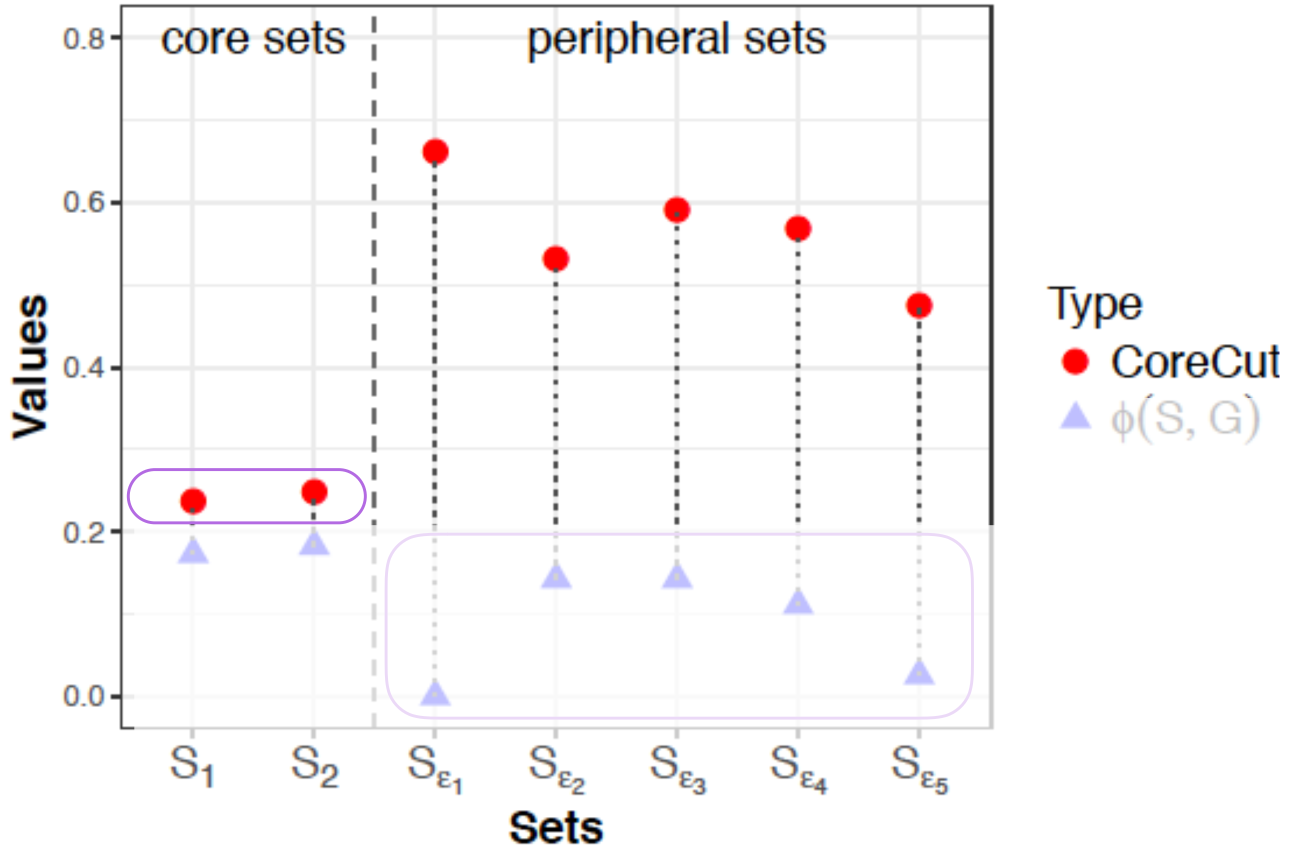
Regularized-SC

relaxes to



Conductance finds peripheral sets.

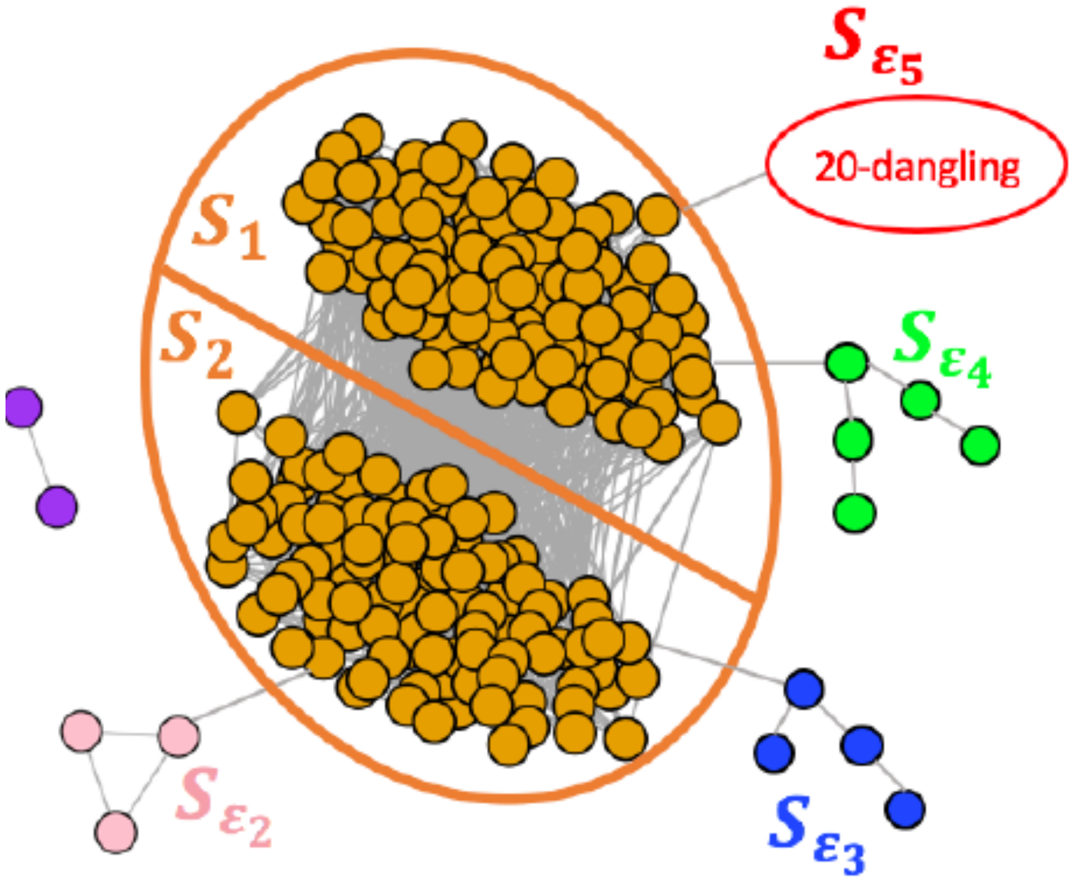


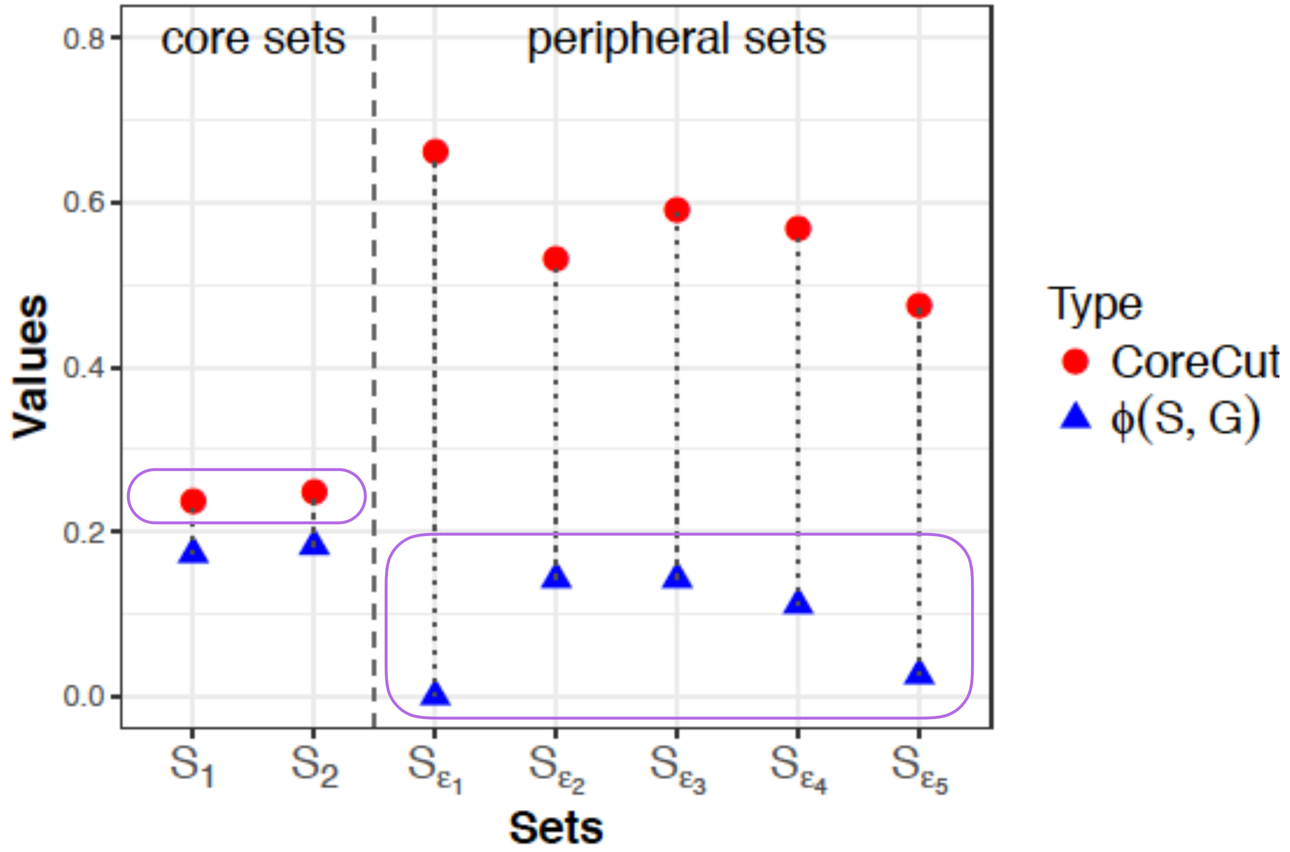


Conductance finds peripheral sets.

CoreCut finds core sets.

CoreCut ignores peripheral sets and focuses on the core!





Conductance finds peripheral sets.

CoreCut finds core sets.

Regularization increases conductance for peripheral sets significantly, but does not affect core sets that much.

CoreCut succeeds when the regularizer overwhelms peripheries but is negligible to the core.

Assumptions:

For a graph G and subsets S, S_ϵ, there exists ϵ, α, s.t.

1. $|S_\epsilon| < \epsilon|V|$ and $vol(S_\epsilon) < \epsilon vol(V)$, S_ϵ is small enough.
2. $\bar{d}(S_\epsilon) < \dfrac{1-\epsilon}{2(1+\alpha)}\bar{d}(S)$, S is dense enough.
3. $\phi(S) < \dfrac{\alpha(1-\epsilon)}{1+\alpha}$. S is a good cut.

Propositions:

Under Assumption 1, if we choose $\tau \geq \alpha\bar{d}(S_\epsilon)$, then $\mathrm{CoreCut}_\tau(S_\epsilon) > \dfrac{\alpha(1-\epsilon)}{1+\alpha}$.

If we choose $\tau < \delta\bar{d}(S)$ for some $\delta > 0$, then $\mathrm{CoreCut}_\tau(S) < \phi(S) + \delta$.

Corollary:

Under Assumptions 1-3, if we choose τ s.t.

$$\alpha\bar{d}(S_\epsilon) \leq \tau \leq \delta\bar{d}(S),$$

where $\delta = \alpha(1-\epsilon)/(1+\alpha) - \phi(S)$, then

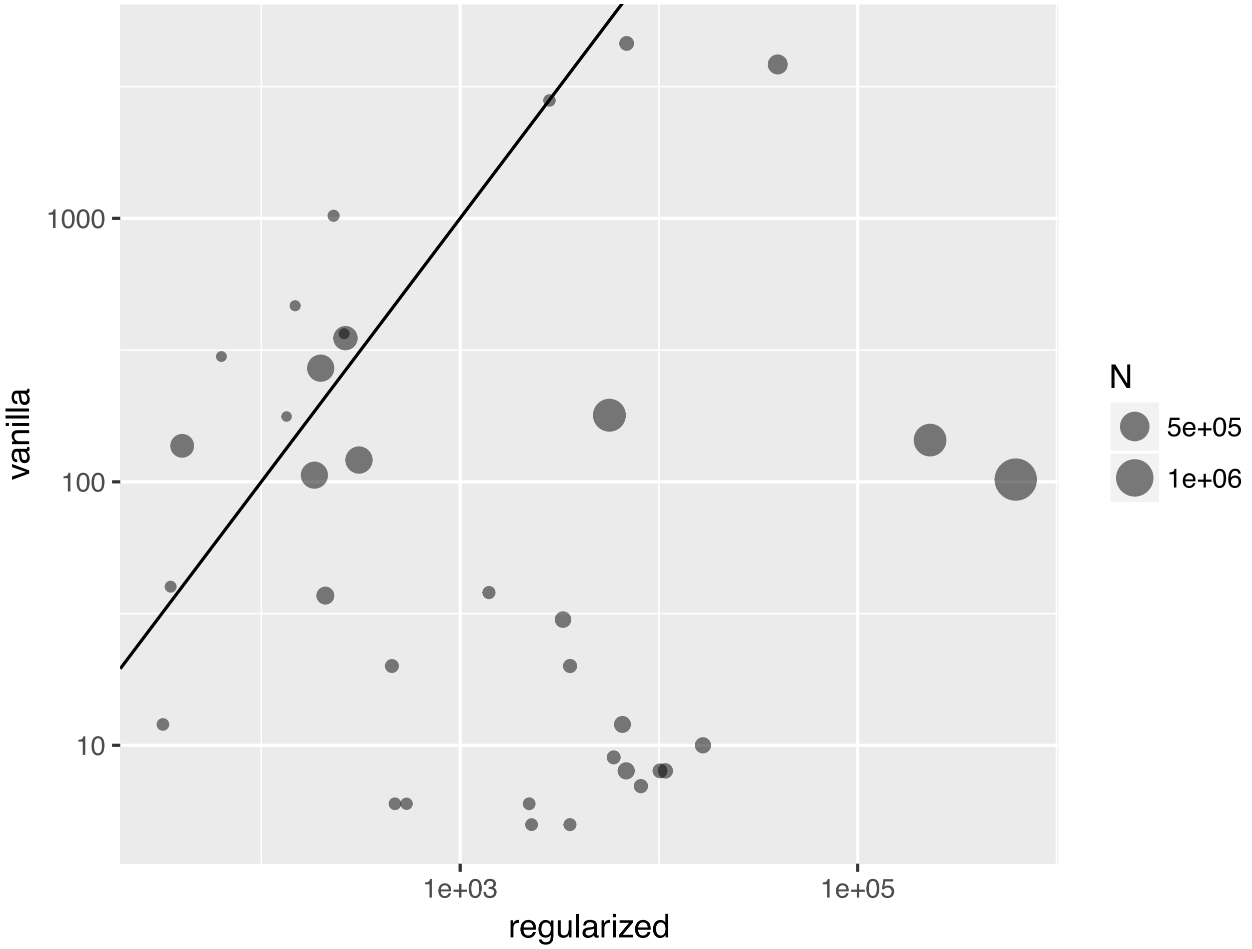
$$\mathrm{CoreCut}_\tau(S) < \mathrm{CoreCut}_\tau(S_\epsilon).$$

My suggestion in practice:
Set τ to be (square root of) average degree of G.

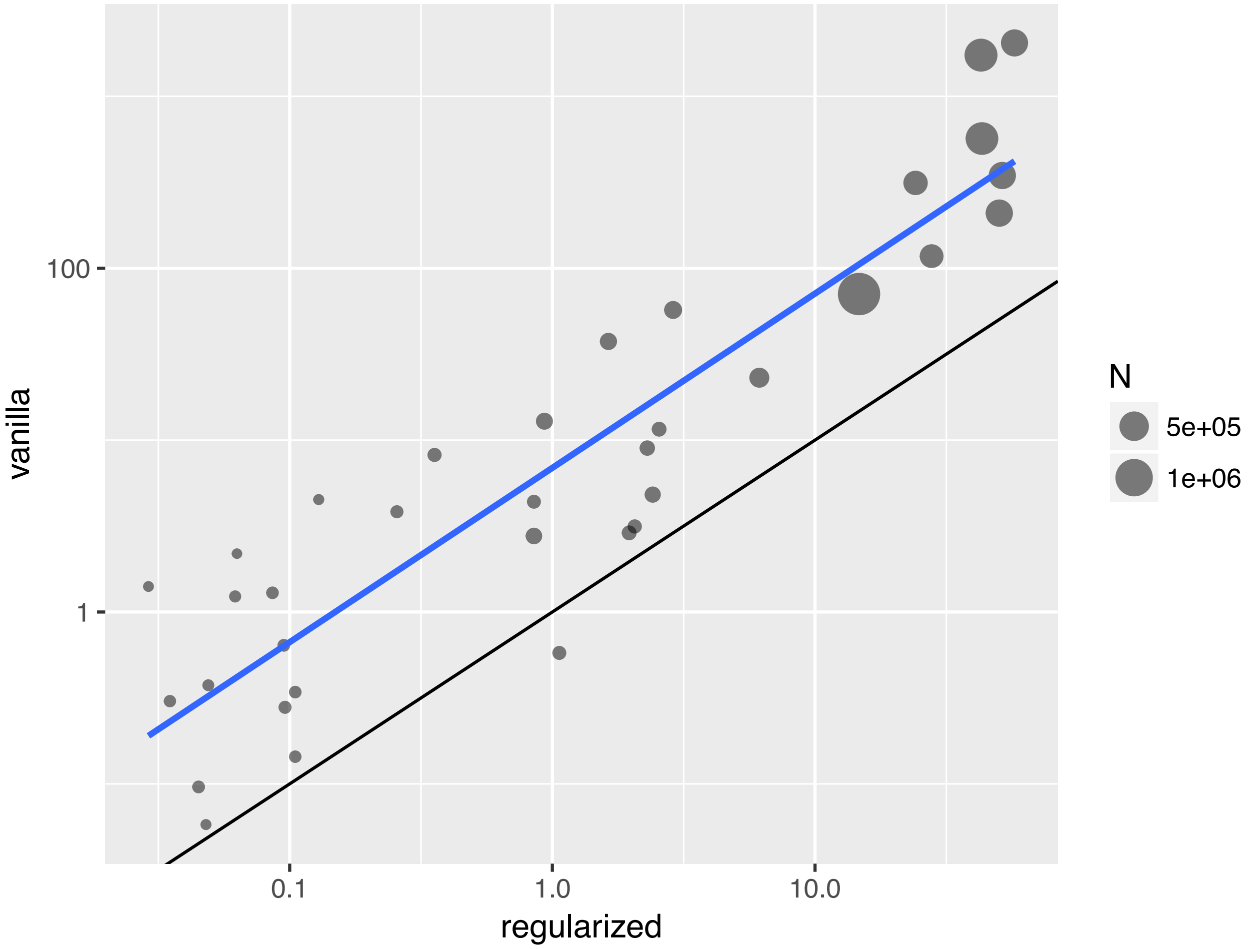
Real data examples.

37 networks from http://snap.stanford.edu/data

Balance vs balance.
Regularization increases balance.



Running time in seconds.
Regularized runs ~8x faster.



more balanced

faster

Summary

Why Spectral Clustering fails?

Spectral Clustering likes sets with small conductance.

Noises such as g-dangling sets have small conductance.

Why Regularization fixes?

Regularized SC likes sets with small CoreCut.

CoreCut focuses on the core!

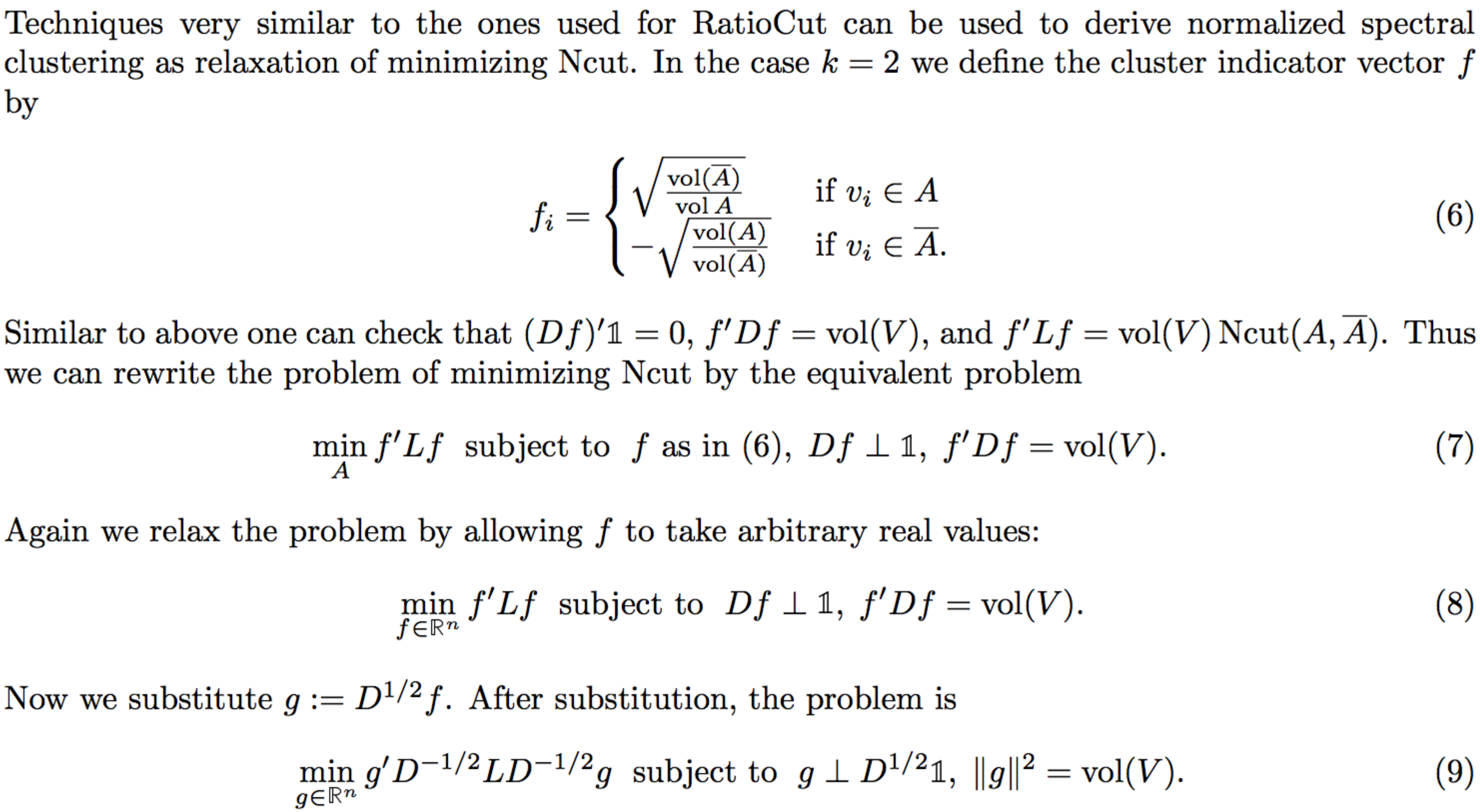
Regularized Spectral Clustering
is more balanced & faster!

Thank you!

Any questions?

Appendix slides…

conductance relaxes to Vanilla SC



Techniques very similar to the ones used for RatioCut can be used to derive normalized spectral clustering as relaxation of minimizing Ncut. In the case $k = 2$ we define the cluster indicator vector f by

$$f_i = \begin{cases} \sqrt{\frac{\operatorname{vol}(\overline{A})}{\operatorname{vol} A}} & \text{if } v_i \in A \\ -\sqrt{\frac{\operatorname{vol}(A)}{\operatorname{vol}(\overline{A})}} & \text{if } v_i \in \overline{A}. \end{cases} \tag{6}$$

Similar to above one can check that $(Df)'\mathbb{1} = 0$, $f'Df = \operatorname{vol}(V)$, and $f'Lf = \operatorname{vol}(V)\operatorname{Ncut}(A, \overline{A})$. Thus we can rewrite the problem of minimizing Ncut by the equivalent problem

$$\min_{A} f'Lf \;\text{ subject to }\; f \text{ as in (6)},\; Df \perp \mathbb{1},\; f'Df = \operatorname{vol}(V). \tag{7}$$

Again we relax the problem by allowing f to take arbitrary real values:

$$\min_{f \in \mathbb{R}^n} f'Lf \;\text{ subject to }\; Df \perp \mathbb{1},\; f'Df = \operatorname{vol}(V). \tag{8}$$

Now we substitute $g := D^{1/2}f$. After substitution, the problem is

$$\min_{g \in \mathbb{R}^n} g'D^{-1/2}LD^{-1/2}g \;\text{ subject to }\; g \perp D^{1/2}\mathbb{1},\; \|g\|^2 = \operatorname{vol}(V). \tag{9}$$

[Luxburg et al. 2007]

Spectral Clustering is fooled by randomness!

Think regression: What do you say if the model perfectly interpolates the data (MSE = 0)? Overfitting!

Spectral clustering overfits to conductance!

Don’t be fooled by randomness!

Vanilla SC overfits!

$$\boxed{f(\hat{\theta}, \mathrm{TrainingData}) < f(\theta^*, \mathrm{TrainingData}) < f(\hat{\theta}, \mathrm{TestData})}$$

data: graph (e.g. SBM)

f: conductance (or Rayleigh Quotient)

θ: partition (eigenvector)

$$\hat{\theta} = \arg\min_{\theta} f(\theta, \mathrm{TrainingData})$$

Think regression: What do you say if the model perfectly interpolates the data (MSE = 0)? Overfitting!

Vanilla SC overfits to conductance. It's fooled by randomness.

In statistical learning theory, there are tools to address overfitting.

Regularize to prevent overfitting.

Cross Validation to

- measure overfitting
- assess the regularization

We can also do this in graphs!