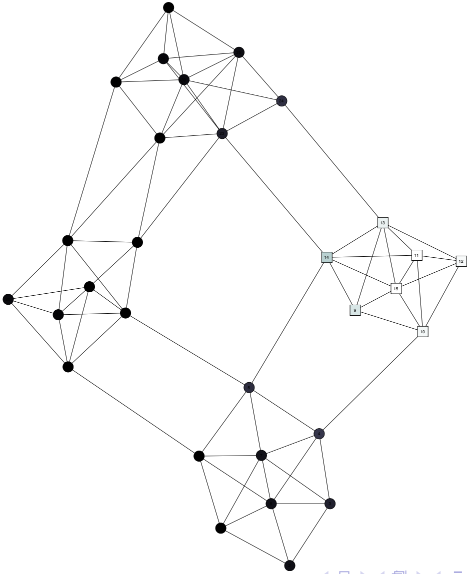
Communities, Spectral Clustering, and Random Walks

David Bindel

Department of Computer Science
Cornell University

4 Nov 2011


13
11
12
14
15
9
10

Basic setting

Informal: Community = “unusually tight” node group?

Formal: Graph $G = (V, E)$, seek subgraph $G' = (V', E')$:

1. By model fitting
2. By optimization of some metric
3. By random walks on G

Unified by linear algebra!

Plan for today

- Three routes to an invariant subspace
- How to mine a subspace for information
- From eigenvectors to Ritz vectors
- Some examples

Notation

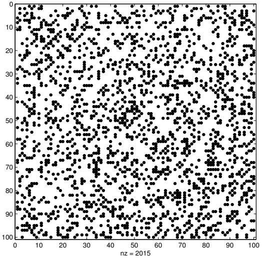
Adjacency matrix $A \in \{0, 1\}^{n \times n}$ is

$$A_{ij} = \begin{cases} 1, & (i,j) \in E \\ 0, & \text{otherwise} \end{cases}$$

Also define

$$\begin{aligned} e &= \text{vector of } n \text{ ones} \\ d &= Ae = \text{degree vector} \\ D &= \text{diag}(d) \\ L &= D - A = \text{graph Laplacian} \\ B &= A - \frac{dd^T}{m} = \text{modularity matrix} \end{aligned}$$

Spectrum for a random graph

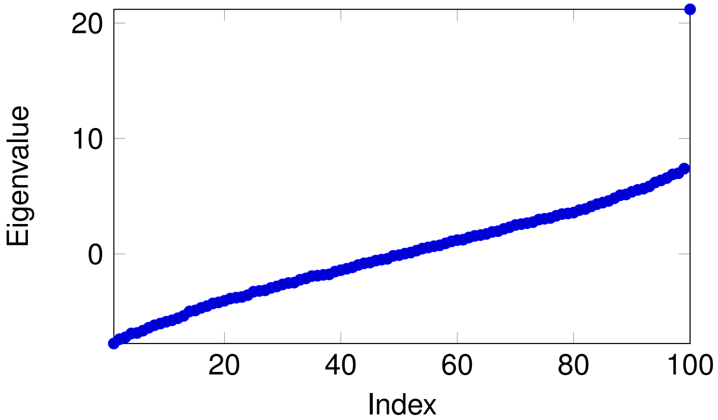


$\approx$

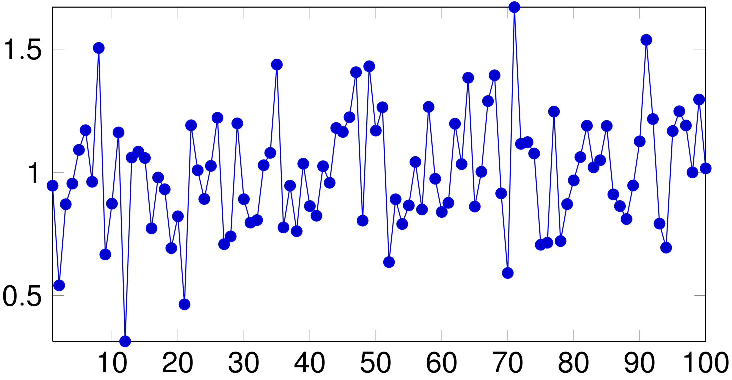
Spectrum of a $G_{n,p}$ graph:

- One large eigenvalue $\approx np$
- Other eigs between $\approx \pm\sqrt{np(1-p)/4}$
- Adjacency matrix = $pee^T +$ "noise"

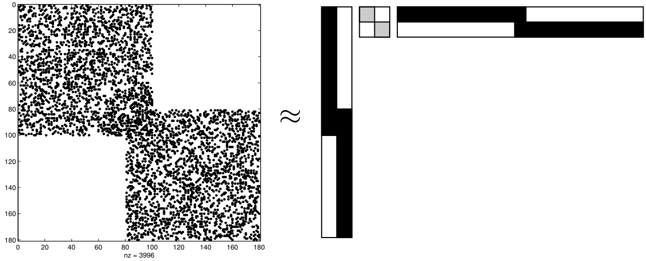
Spectrum for a $G_{100,0.2}$ sample



Perron vector for a $G_{100,0.2}$ sample



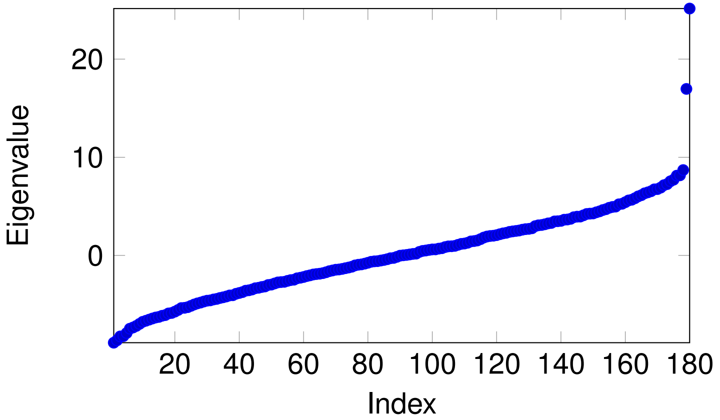
Block model approach



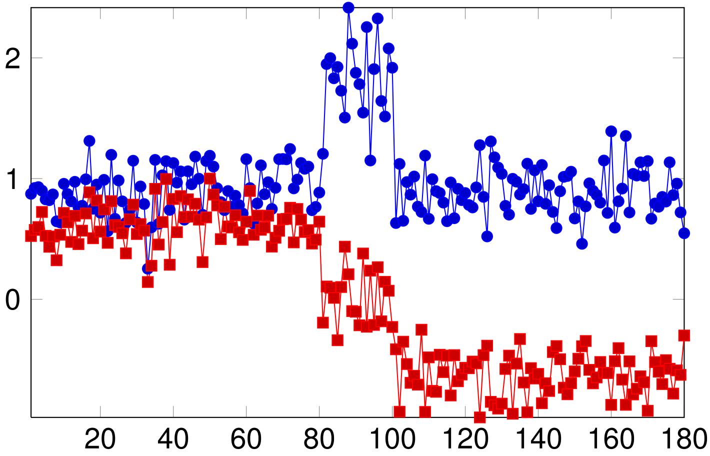
Composite model: $A \approx S \operatorname{diag}(\beta) S^T$, $S \in \{0, 1\}^{n \times c}$

- Motivation: possibly-overlapping random graphs
- Columns of S are one basis for range space
- Want to go from some general basis back to S

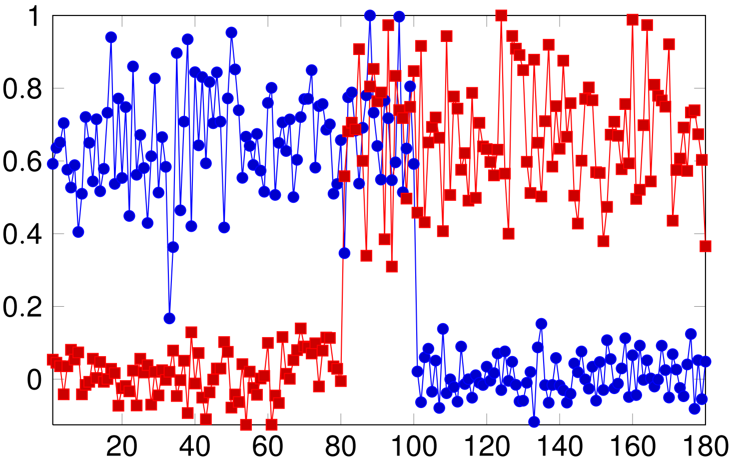
Spectrum for a block model sample



Dominant vectors for a block model example



Same space, different basis



Questions

- What about different matrices (e.g. L)?
- What about more interesting graph structures?
- How do we find the “right” subspace basis?

Measurement by quadratic forms

Indicate $V' \subseteq V$ by $s \in \{0, 1\}^n$. Measure subgraph:

$$
\begin{aligned}
s^T A s &= |E'| = \text{internal edges} \\
s^T D s &= \text{edges incident on subgraph} \\
s^T L s &= \text{edges between } V' \text{ and } \bar{V}' \\
s^T B s &= \text{“surprising” internal edges}
\end{aligned}
$$

Graph bisection

Idea: Find $s \in \{0, 1\}^n$ such that $e^T s = n/2$ to

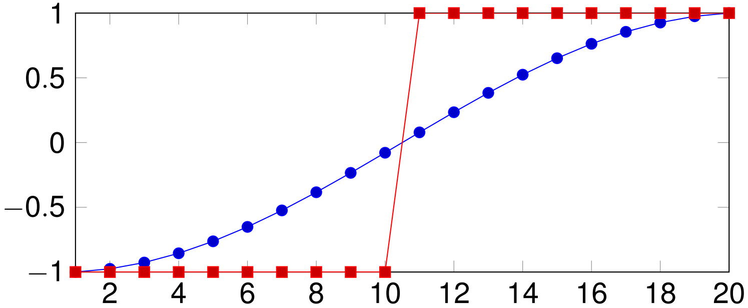
- minimize $s^T Ls$ (min cut)
- maximize $s^T Bs$ (max modularity)

Equivalently: Find $\bar{s} \in \{\pm 1\}^n$ such that $e^T \bar{s} = 0$ to

- minimize $\bar{s}^T L\bar{s} = s^T Ls$ or
- maximize $\bar{s}^T B\bar{s} = s^T Bs$

Oops — NP hard!

Relax!



Hard: $\min \bar{s}^T L \bar{s}$ s.t. $e^T \bar{s} = 0$, $\bar{s} \in \{\pm 1\}^n$.

Easy: $\min v^T L v$ s.t. $e^T v = 0$, $v \in \mathbb{R}^n$, $\|v\|^2 = n$.

Rayleigh quotients

$$\frac{s^T A s}{s^T s} = \text{mean internal degree in subgraph}$$

$$\frac{s^T L s}{s^T s} = \text{edges cut between } V' \text{ and } \bar{V}'$$

$$\frac{s^T A s}{s^T D s} = \text{fraction of incident edges internal to } V'$$

$$\frac{s^T L s}{s^T D s} = \text{fraction of incident edges cut}$$

$$\frac{s^T B s}{s^T s} = \text{mean “surprising” internal degree in subgraph}$$

$$\frac{s^T B s}{s^T D s} = \text{mean fraction of internal degree that is surprising}$$

Rayleigh quotients and eigenvalues

Basic connection (M spd):

$$\frac{x^T K x}{x^T M x} \text{ stationary at x} \quad \Longleftrightarrow \quad Kx = \lambda Mx$$

Easy despite lack of convexity.

Limits of Rayleigh quotients

But small variations kill us:

$$\max_{x \neq 0} \frac{x^T A x}{\|x\|_2^2} = \lambda_{\max}(A), \text{ but}$$
$$\max_{x \neq 0} \frac{x^T A x}{\|x\|_1^2} = 1 - \omega^{-1}$$

where ω is the max clique size (Motzkin-Strauss).

Rayleigh quotients and eigenproblems

Decompose:

$$W^T M W = I \text{ and } W^T K W = \Lambda = \operatorname{diag}(\lambda_1, \ldots, \lambda_n).$$

For any $x \neq 0$,

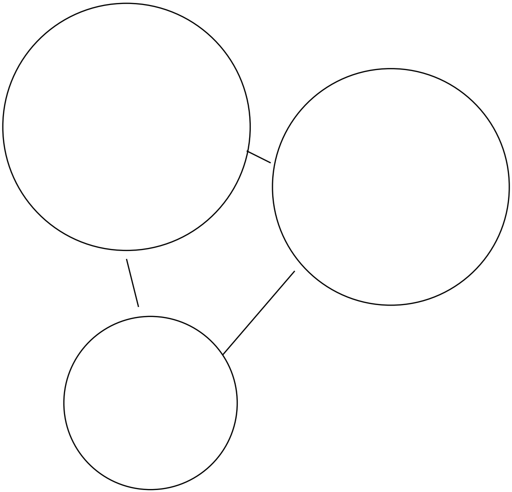
$$\frac{x^T K x}{x^T M x} = \sum_{j=1}^{n} \lambda_j z_j^2, \text{ where } z = \frac{W^{-1} x}{\|W^{-1} x\|_2}.$$

So

$$\frac{s^T K s}{s^T M s} \approx \lambda_{\max} \implies s \approx \sum_{\lambda_j \approx \lambda_{\max}} w_j z_j.$$

So look at invariant subspaces for extreme eigenvalues.

The random walker



The random walker

Basic idea: extract structure from random walk.

Old: start at seed and walk forward

- Day 1: I came up with a funny joke!
- Day 2: I tell everyone in my family
- Day 3: My mother tells a friend?

New: look at how quickly source is forgotten

- Day 1: David came up with a funny joke!
- Day 2: There's a joke going around Cornell CS.
- Day 3: I read this bad joke on the web...

The random walker

Lazy random walk with transition matrix $T = \frac{1}{2}(I + AD^{-1})$.

1. Start at p_0, take k steps. Distribution:

$$p_k = T^k p_0 \quad (\rightarrow d/m \text{ as } k \rightarrow \infty)$$

2. End at q_0 after k steps. Conditional distribution on start:

$$q_k \propto \left(T^T\right)^k q_0 \quad (\rightarrow e/n \text{ as } k \rightarrow \infty)$$

Note: If the graph is undirected, $T^T = D^{-1} T D$.

Simon-Ando theory

Markov chain with loosely-coupled subchains:

- Rapid *local* mixing: after a few steps

$$p_k \approx \sum_{j=1}^{c} \alpha_{j,k} p_\infty^{(j)}$$

  where $p_\infty^{(j)}$ is a local equilibrium for the jth subchain
- Slow equilibration: $\alpha_{j,k} \to \alpha_{j,\infty}$.

Alternately, rapid local mixing looks like:

$$q_k \approx \sum_{j=1}^{c} \gamma_{j,k} s_j$$

where s_j is an indicator for nodes in one subchain.

Simon-Ando theory

In chemistry:
transitions among metastable states.

In network analysis:
transitions among communities?

Spectral Simon-Ando picture

Exactly decoupled case (c decoupled chains):

- Eigenvalue one has multiplicity c.
- Eigenvectors of T are local equilibria.
- Eigenvectors of T^T are indicators for chains.
- Rapid mixing $\implies$ large gap to λ_{c+1}.

Weakly coupled case:

- Cluster of c eigenvalues near 1.
- Eigenvectors of T are combinations of local equilibria.
- Eigenvectors of T^T are combinations of indicators.
- Large gap between λ_c and λ_{c+1}.

Summary so far

Indicator vectors approximately in invariant subspaces

- Several possible motivations
- Several possible matrices (I like T^T)

But how do we go from the subspace to the indicators?

Indicators from subspaces: spectral clustering

U spans a small subspace (e.g. an invariant subspace)

1. $\text{range}(U) \approx \text{range}(S)$, S indicates a partition. Rows of U in the same partition are identical.

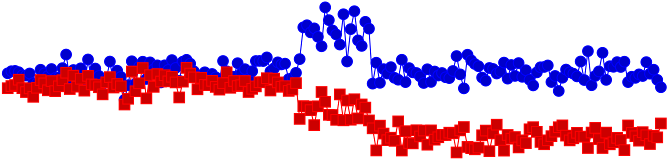
   Idea: Treat rows of U are *latent coordinates*. Cluster.

2. Suppose some indicator $s \approx Uy$. Then row $U(j,:)$
   - forms an acute angle with y when $s_j = 1$
   - is almost normal to y when $s_j = 0$.

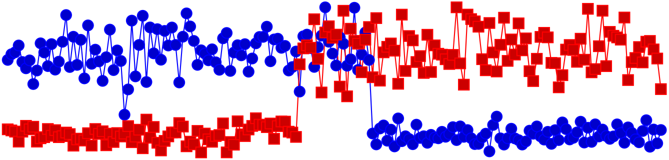
   Clustering? What if sets overlap?

Clustering and overlap

Dominant eigenvectors for *A*:



Alternate basis for the space:



How do we get the latter basis?

Desiderata

Given a basis U, want to extract a vector $\tilde{s}$ s.t.

- $\tilde{s}$ lies close to the span of U
- $\tilde{s}$ is almost an indicator for a community
  - Maybe nonnegative?
  - Not too many ones?

Indicators from subspaces: LP version

Suppose $s \approx Uy$ for some y, $s_i = 1$. Want to find s.
Try optimization (a linear program):

$$\begin{aligned} \text{minimize} \quad & \|\tilde{s}\|_1 && \text{(proxy for sparsity of } \tilde{s}) \\ \text{s.t.} \quad & \tilde{s} = Uy && (\tilde{s} \text{ in the right space}) \\ & \tilde{s}_i \geq 1 && \text{(``seed'' constraint)} \\ & \tilde{s} \geq 0 && \text{(componentwise nonnegativity)} \end{aligned}$$

Recovers smallest set containing node i if

- $U = SY^{-1}$ exactly.
- Each set contains at least one element only in that set. (Frequently works if there is not "too much" overlap.)

What about noise? Generally need a thresholding strategy.

Indicators from subspaces: QP version

Alternate optimization (box-constrained quadratic program):

$$\begin{aligned} \text{minimize} \quad & \tfrac{1}{2}\tilde{s}^T P \tilde{s} + \tau \|\tilde{s}\|_1 \\ \text{s.t.} \quad & \tilde{s}_i \geq 1 \\ & \tilde{s} \geq 0 \end{aligned}$$

Recover LP with $P = I - UU^T$ and $\tau \to 0$ (for $U^T U = I$).

- Can let P be general semidefinite matrix (e.g. $P = L$)
- Size of τ controls sparsity (can automate choice)

Summary so far

Two pieces to spectral community detection:

- Pull out an invariant subspace
- Mine the subspace for community structure

Motivation: optimization or random walk dynamics.

But...

- What about when n and c are both large?
- What if there is no clear spectral gap?

Would like an alternative to invariant subspaces!

Eigenvectors to Ritz vectors

Eigenvectors are stationary points of Rayleigh quotients.
Find stationary points in a subspace $\implies$ *Ritz* vectors.

Usual approach to large-scale eigenproblems:

1. Generate a basis for a *Krylov subspace*

$$\mathcal{K}_k(A, x_0) = \operatorname{span}\{x_0, Ax_0, A^2x_0, \ldots, A^{k-1}x_0\}$$

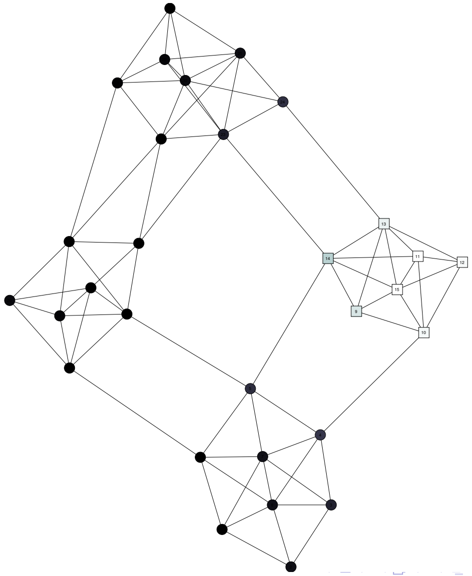
2. Ritz values rapidly approximate extreme eigenvalues
3. Ritz vectors approximate extreme eigenvectors

Idea: Instead of searching invariant subspace, search in a space spanned by a few scaled Ritz vectors. Pulls out dynamics of *short* random walks (vs long).

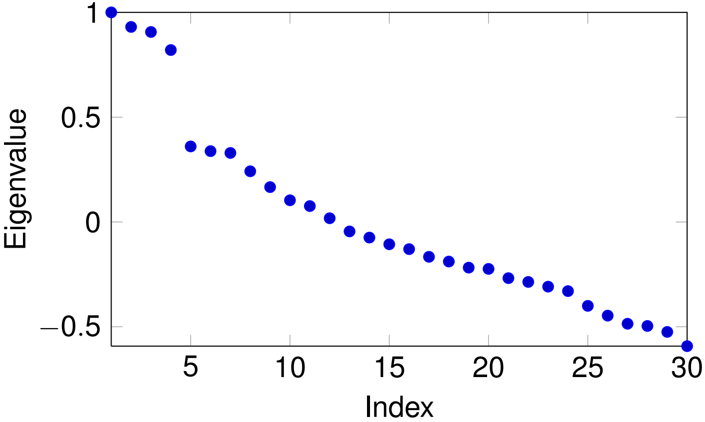
Current favorite method

1. Pick “seed” nodes $j_1, j_2, \ldots$
2. Take short random walks (length k) from each seed
3. Extract a few Ritz vectors (fewer than k) from $\text{span}\{\phi_0, \phi_1, \ldots, \phi_{k-1}\}$.
4. Use quadratic programming to find approximate indicators in subspace space spanned by all Ritz vectors.
5. Possibly add more seeds and return to step 1.
6. Threshold to get initial indicator approximation.
7. Greedily optimize angle between indicator and space.

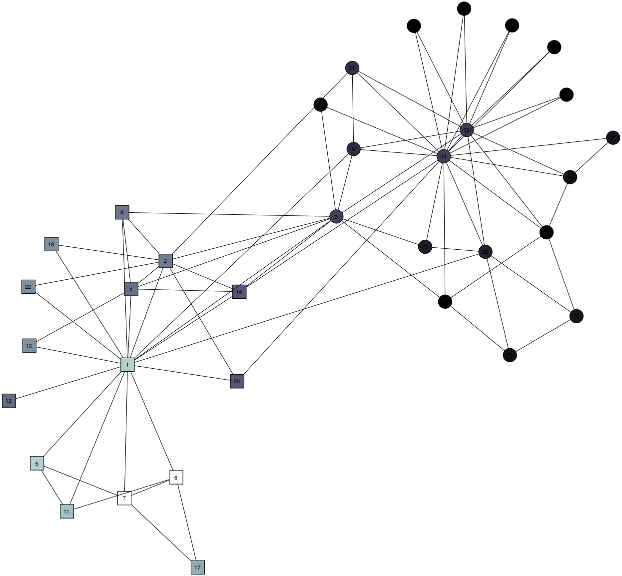
Wang test graph



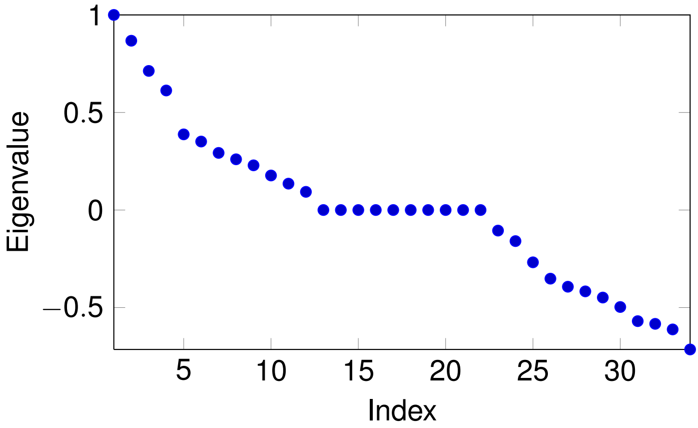
Spectrum for Wang test graph



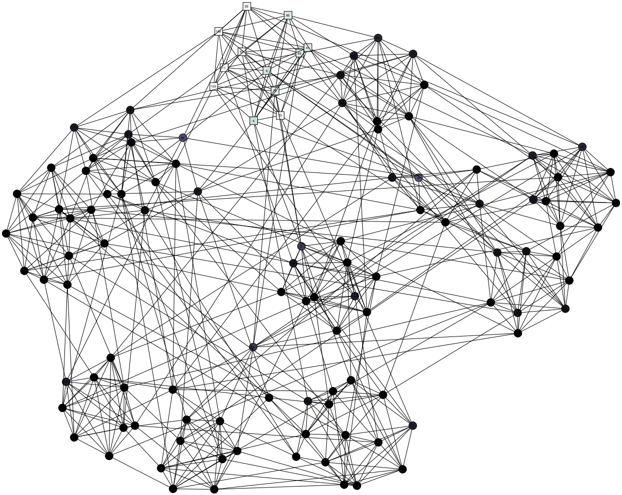
Zachary Karate graph



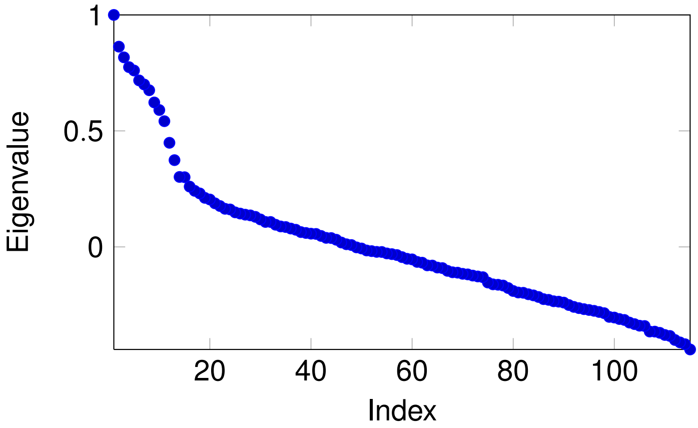
Spectrum for Karate



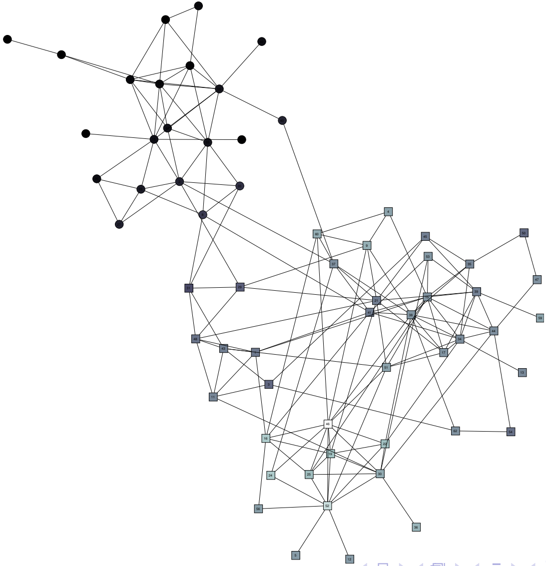
Football graph



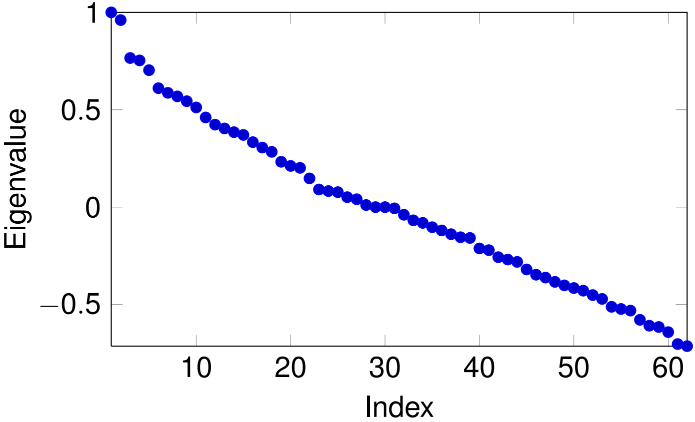
Spectrum for Football



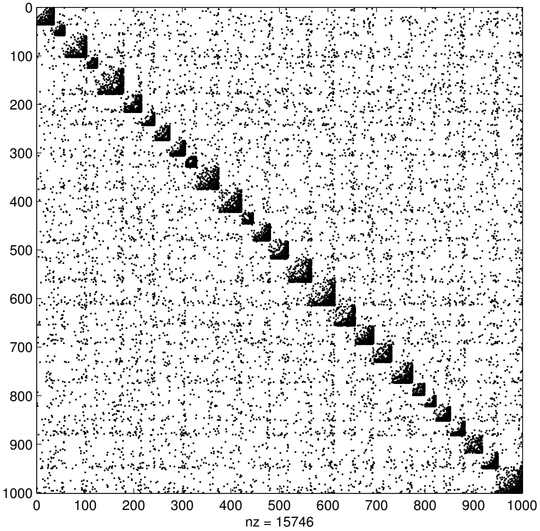
Dolphin graph



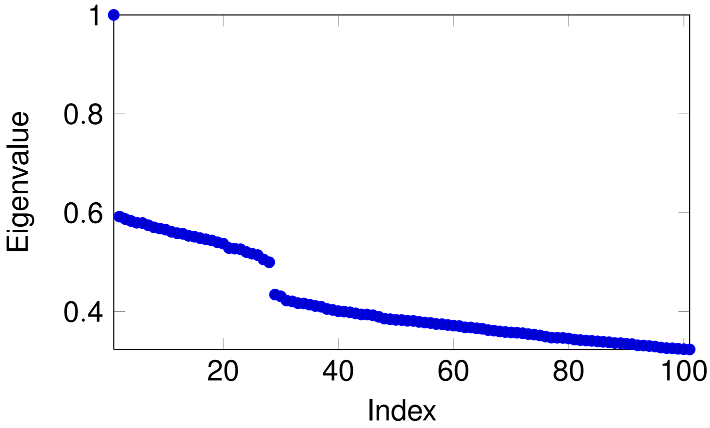
Spectrum for Dolphin



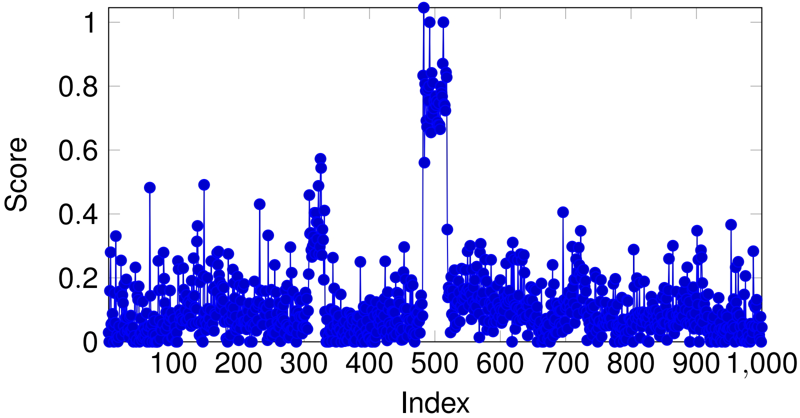
Non-overlapping synthetic benchmark ($\mu = 0.5$)



Spectrum for synthetic benchmark

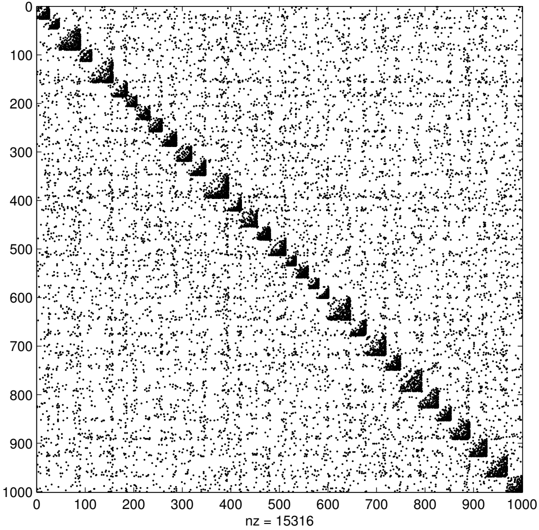


Score vector

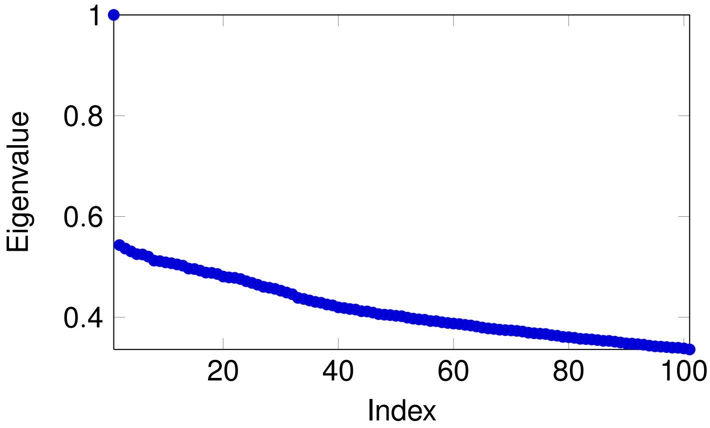


Score vector for the two-node seed of 492 and 513 in the first LFR benchmark graph. Ten steps, three Ritz vectors.

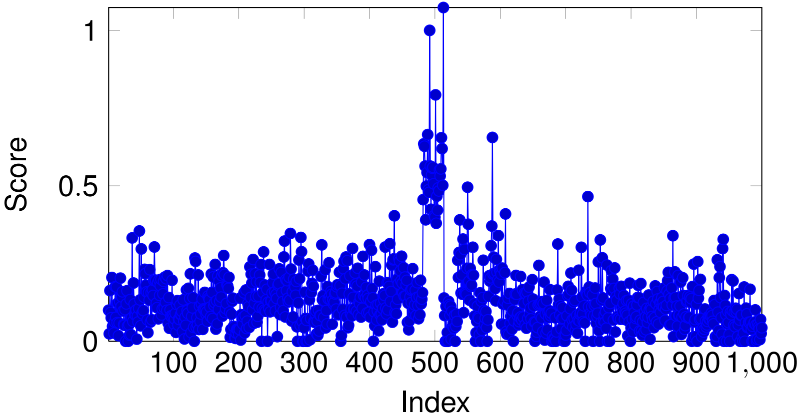
Non-overlapping synthetic benchmark ($\mu = 0.6$)



Spectrum for synthetic benchmark



Score vector

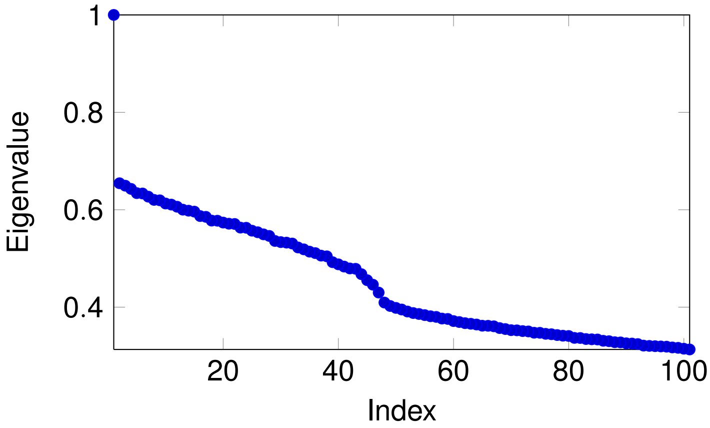


Score vector for the two-node seed of 492 and 513 in the first LFR benchmark graph. Ten steps, three Ritz vectors.

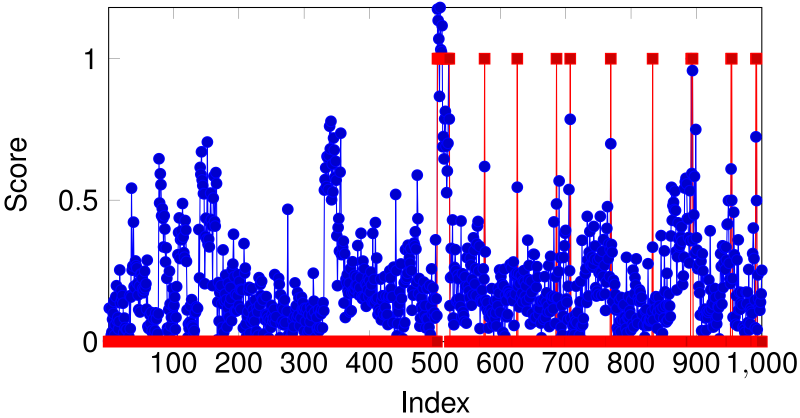
Overlapping synthetic benchmark ($\mu = 0.3$)

- 1000 nodes
- 47 communities
- 500 nodes belong to two communities

Spectrum for synthetic benchmark

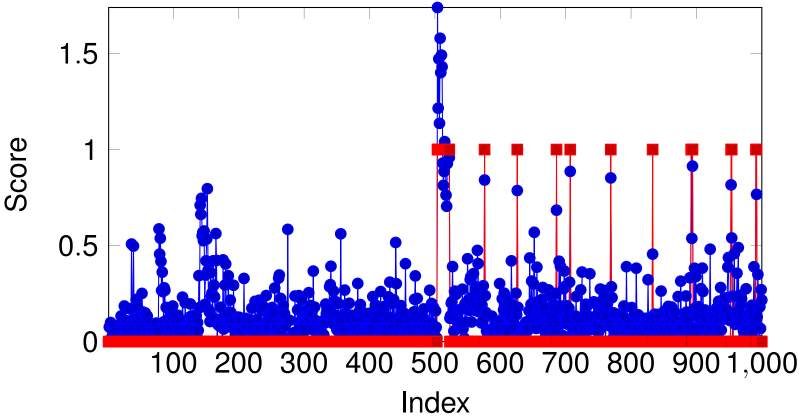


Score vector



Score vector for the two-node seed of 521 and 892.
The desired indicator is in red.

Score vector



Score vector for the two-node seed of 521 and 892 + twelve reseeds. The desired indicator is in red.

Conclusions

Classic spectral methods use eigenvectors to find communities, but:

- We don't need to stop at partitioning!
  - Overlap is okay
  - Key is how we mine the subspace
- We don't need to stop at eigenvectors!
  - Can also use *Ritz* vectors
  - Computation is cheap: short random walks