Communities, Spectral Clustering, and Random Walks

David Bindel

Department of Computer Science
Cornell University

13 Jul 2012

Spectral clustering recipe

Ingredients:

1. A subspace basis with useful information
   (eigenvectors of L, $\bar{L}$, B, etc; NMF factors)
2. A way to extract clusters from the basis
   (sign patterns; (scaled) latent coordinates)

What works for small communities? With overlap?

Classic spectral bisection

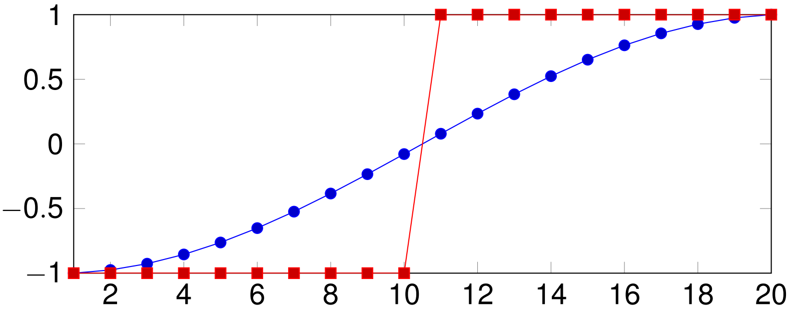
Goal: Split graph into equal parts, minimizing edges cut.

Strategy: turn into a constrained integer QP

- Identify partition with label vector $\bar{s} \in \{\pm 1\}^n$.
- Objective: $\bar{s}^T L \bar{s} = 4 \times |\text{edges cut}|$
- Constraint: $e^T \bar{s} = 0$

This is NP hard, but...

Classical spectral bisection



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. $e^T v = 0$, $v \in \mathbb{R}^n$, $\|v\|^2 = n$.

v is the *Fiedler vector* (second eigenvector of L)

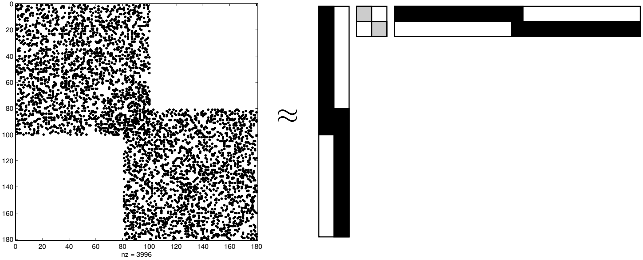
Three clusters?

Take three eigenvectors as latent coordinates + cluster:

$$V \approx \left[\begin{array}{c} r_1 \\ \vdots \\ r_1 \\ \hline r_2 \\ \vdots \\ r_2 \\ \hline r_3 \\ \vdots \\ r_3 \end{array}\right] = \begin{bmatrix} 1 & & \\ \vdots & & \\ 1 & & \\ & 1 & \\ & \vdots & \\ & 1 & \\ & & 1 \\ & & \vdots \\ & & 1 \end{bmatrix} \begin{bmatrix} r_1 \\ r_2 \\ r_3 \end{bmatrix} = SR.$$

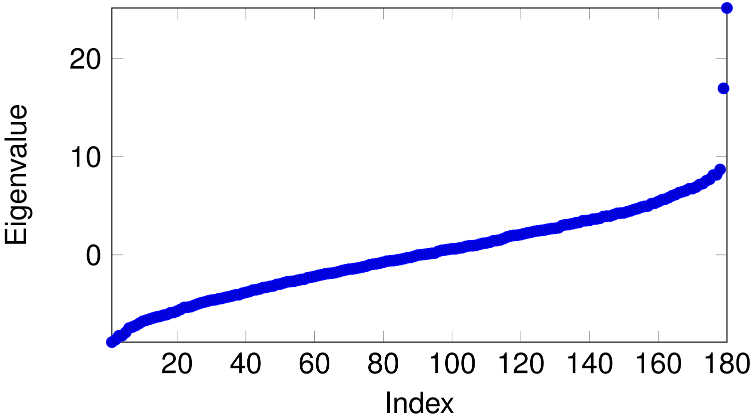
Equivalent to a matrix factorization!

Overlapping communities



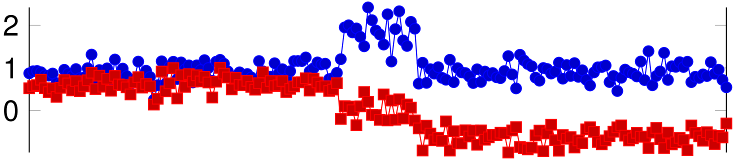
$$A \approx S\,\mathrm{diag}(\beta)S^T, \qquad S \in \{0,1\}^{n \times c}$$

Spectrum for *A* with overlap

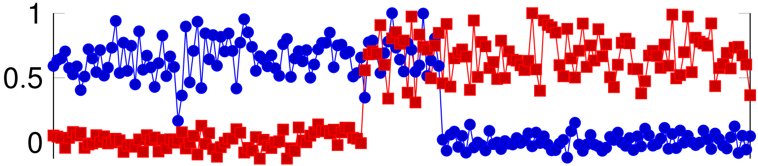


Clustering and overlap

Dominant eigenvectors for A:



Alternate basis for the space:



How do we get the latter basis?

Recovering indicators

$$
\begin{array}{rlr}
\text{minimize} & \|\tilde{s}\|_1 & \text{(proxy for sparsity of } \tilde{s}) \\
\text{s.t.} & \tilde{s} = Uy & (\tilde{s} \text{ in the right space}) \\
 & \tilde{s}_i \geq 1 & \text{(“seed” constraint)} \\
 & \tilde{s} \geq 0 & \text{(componentwise nonnegativity)}
\end{array}
$$

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
- Some known “seeds” are in the community

Indicators from subspaces: LP edition

$$\begin{array}{rll} \text{minimize} & \|\tilde{s}\|_1 & \text{(proxy for sparsity of } \tilde{s}) \\ \text{s.t.} & \tilde{s} = Uy & (\tilde{s} \text{ in the right space}) \\ & \tilde{s}_i \geq 1 & \text{(“seed” constraint)} \\ & \tilde{s} \geq 0 & \text{(componentwise nonnegativity)} \end{array}$$

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.)

Got noise? Need thresholding!

Alas

The method is willing, but the space is weak:

- Eigenvectors of A may not be ideal – what about L, $\bar{L}$, transition matrices for random walks?
- Many small communities $\implies$ many eigenvectors?

So consider:

- Why eigenvectors?
  - Optimization connection
  - Dynamics connection
- What might serve better?

Optimization and quadratic forms

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

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

Rayleigh quotients

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

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

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

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

$$\frac{s^T B s}{s^T s} = \text{mean ``surprising'' internal degree in subgraph}$$

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

Rayleigh quotients and eigenvalues

Basic connection (M spd):

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

Easy despite lack of convexity.

Rayleigh quotients big and small

Decompose:

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

For any $x \neq 0$, $x^T M x = 1$, have

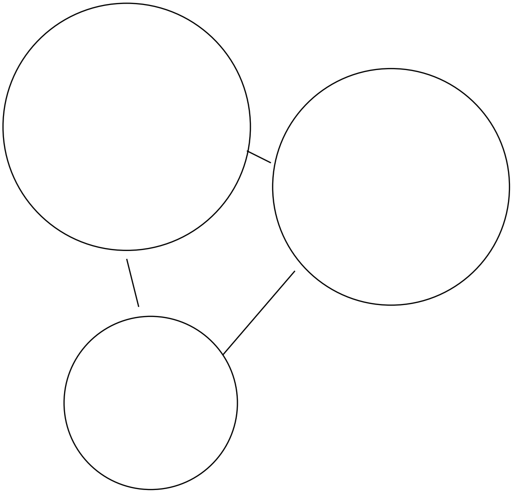
$$x = \sum_j w_j z_j$$
$$x^T K x = \sum_j \lambda_j z_j^2$$

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 dynamics perspective



The random walker

Basic idea: extract structure from random walk.

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?

Or 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

$$S = (\alpha I + AD^{-1})/(1+\alpha).$$

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

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

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

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

Notes:

- If the graph is undirected, $S^T = D^{-1} S D$.
- If the graph is also regular, $S^T = S$.

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.

Spectral Simon-Ando picture

Exactly decoupled case (c decoupled chains):

- Eigenvalue one has multiplicity c.
- Eigenvectors of S are local equilibria.
- Eigenvectors of S^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 S are combinations of local equilibria.
- Eigenvectors of S^T are combinations of indicators.
- Large gap between λ_c and λ_{c+1}.

Beyond eigenvectors

Connections that suggest using dominant eigenvectors:

- Optimization of quadratic forms
- Asymptotic dynamics of random walks

Works well for a few big communities – what about local ones?

Ritz vectors

Variational characterization of eigenpairs:

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

Rayleigh-Ritz approximation: $x = Vy$

$$\frac{(Vy)^T K (Vy)}{(Vy)^T M (Vy)} \text{ stationary at } y \quad \Longleftrightarrow \quad (V^T K V) y = \hat{\lambda} (V^T M V) y$$

Krylov + Rayleigh-Ritz

Usual approach to large-scale symmetric eigenproblems:

1. Generate a basis for a *Krylov subspace*

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

2. Use Rayleigh-Ritz on subpace
3. Re-start if subspace gets too large

Krylov + Rayleigh-Ritz + dynamics

Eigendecomposition picture:

$$p_k = \sum_{j=1}^{N} \alpha_j \lambda_j^k v_j$$

Krylov + Rayleigh-Ritz:

$$\{(\mu_j, w_j)\}_{j=1}^{r} = \text{Ritz pairs from } \mathcal{K}_r(S, p_0)$$

$$p_k = \sum_{j=1}^{r} \beta_j \mu_j^k w_j, \quad k < r$$

The idea: Ritz subspaces

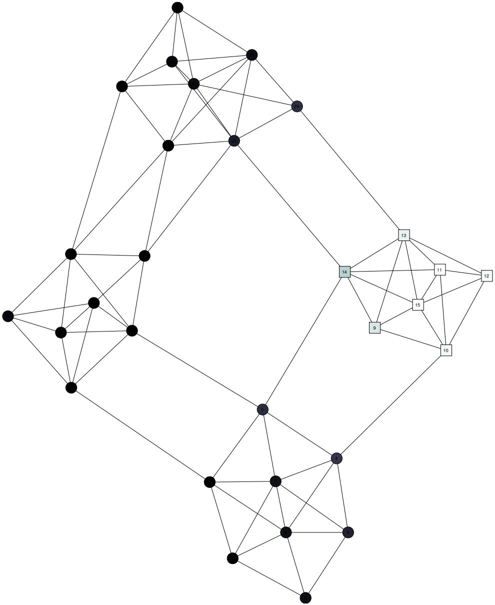
Idea: Use *unconverged* subspace computations

Invariant subspace	Ritz subspace
Long random walks	Short random walks
Global information	Local information
Potentially expensive	Cheap to compute
May need large space	Small space okay

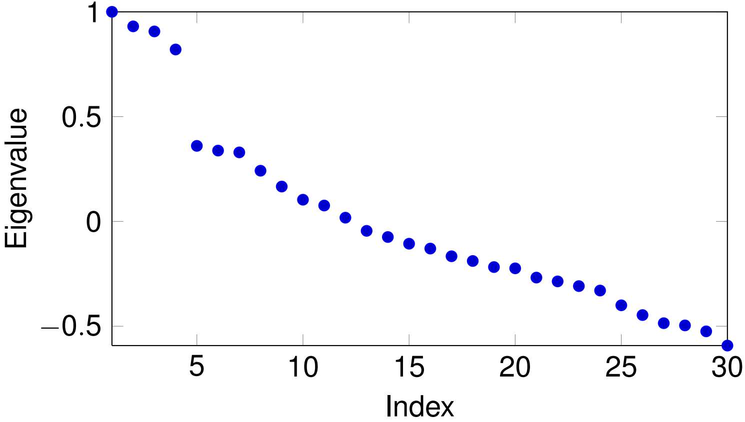
Current favorite method

1. Pick "seed" nodes $j_1, j_2, \ldots$
2. Take short random walks (length k) from each seed
3. Extract few Ritz vectors from span$\{q_0, q_1, \ldots, q_{k-1}\}$.
4. Approx indicators in span of all Ritz vectors.
5. Possibly add more seeds and return to step 1.
6. Convert raw "score" vector to a $\{0, 1\}$ indicator.

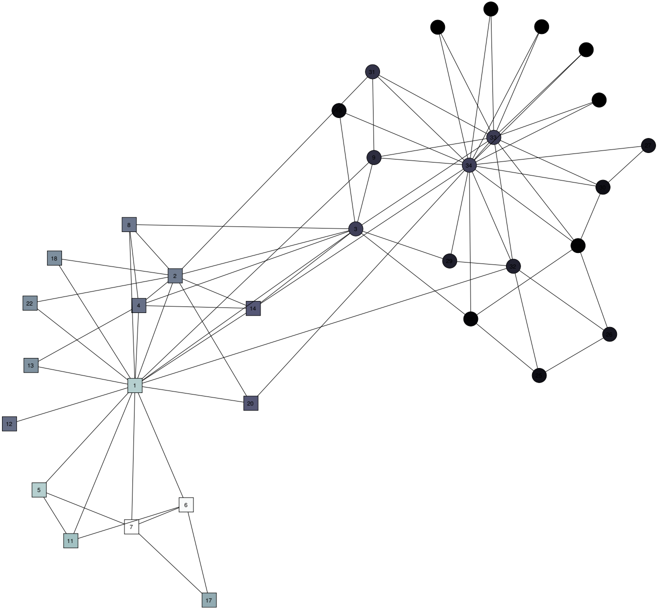
Wang test graph



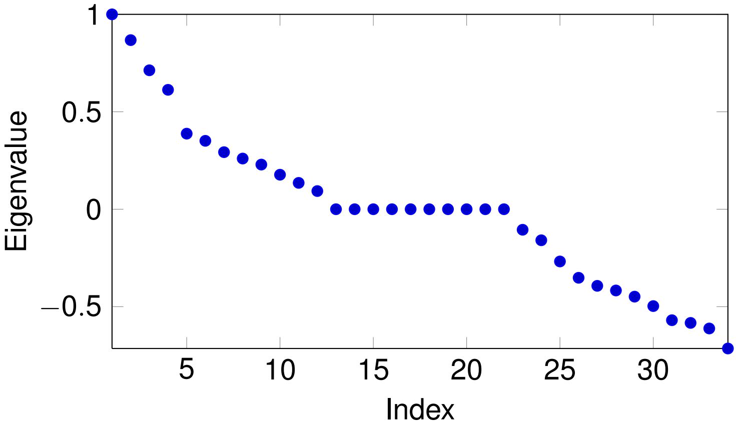
Spectrum for Wang test graph



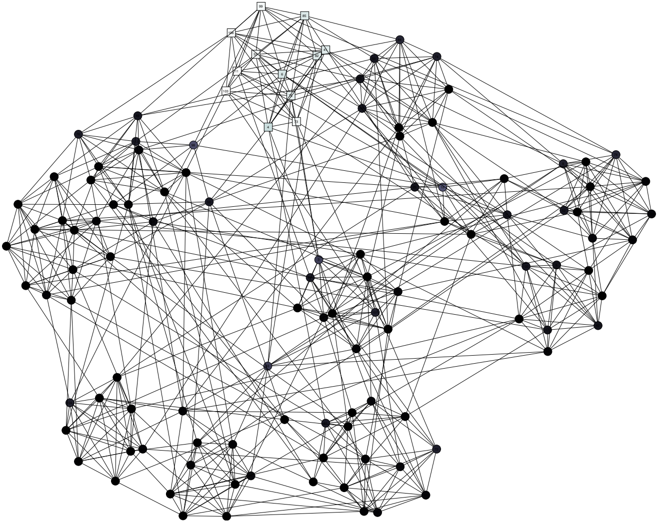
Zachary Karate graph



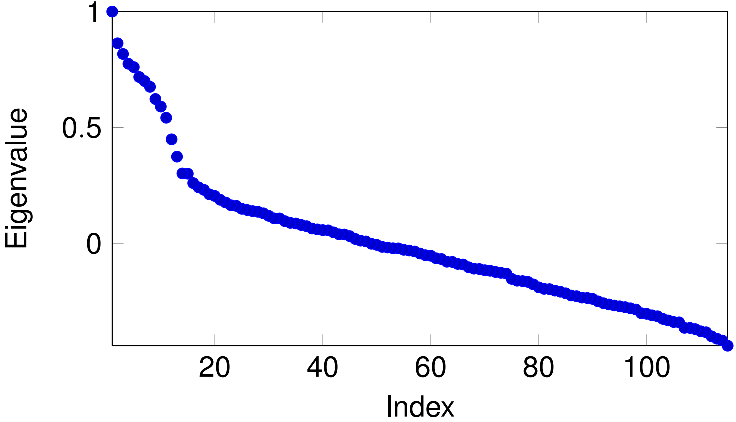
Spectrum for Karate



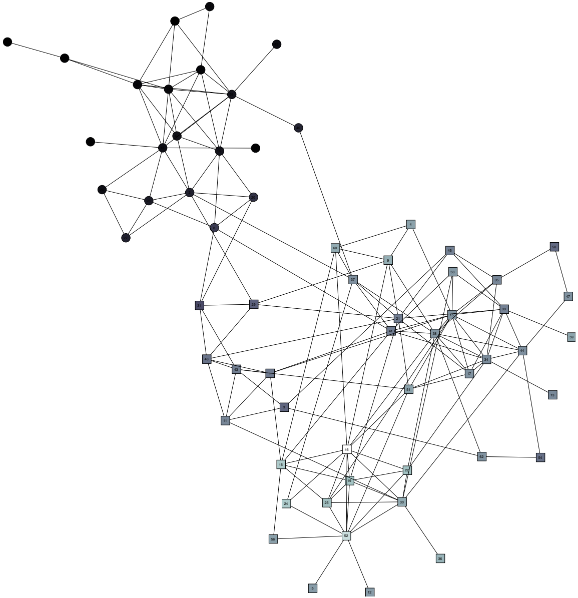
Football graph



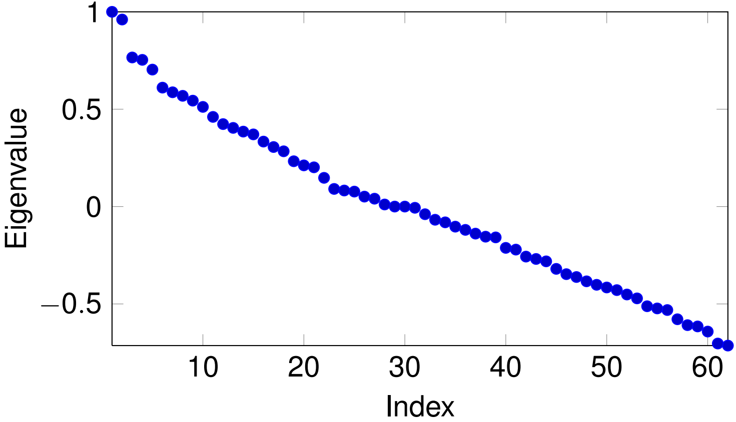
Spectrum for Football



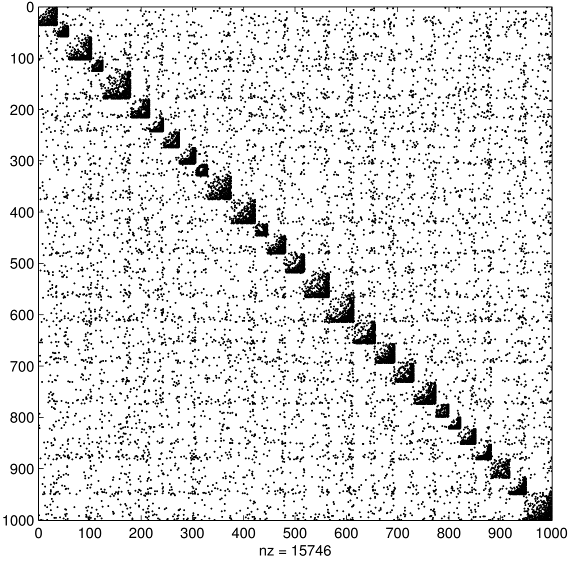
Dolphin graph



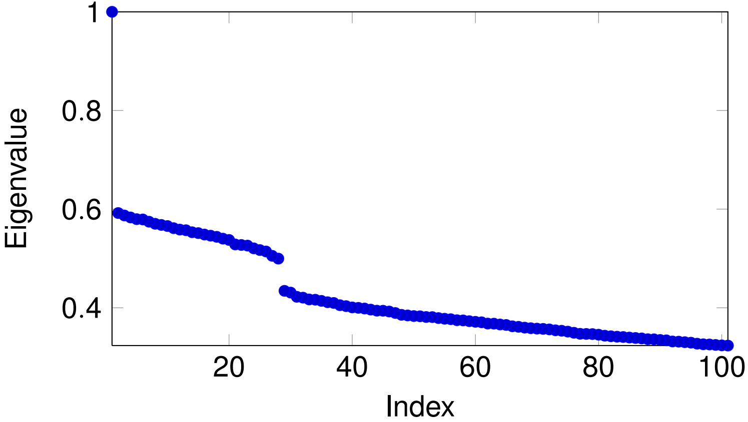
Spectrum for Dolphin



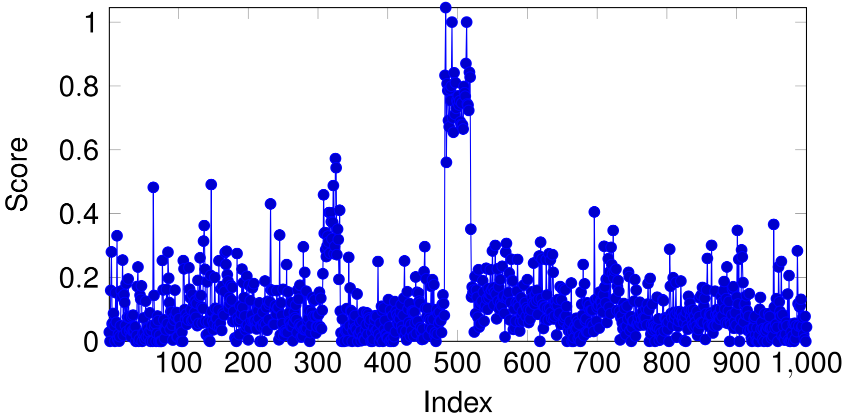
Non-overlapping synthetic benchmark ($\mu = 0.5$)



Spectrum for synthetic benchmark

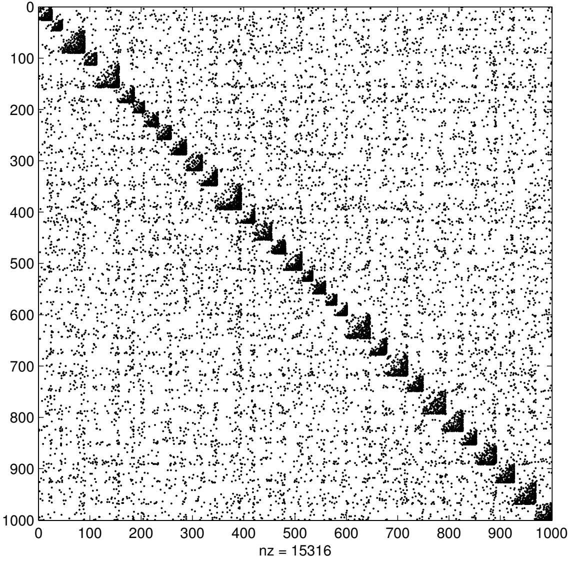


Score vector

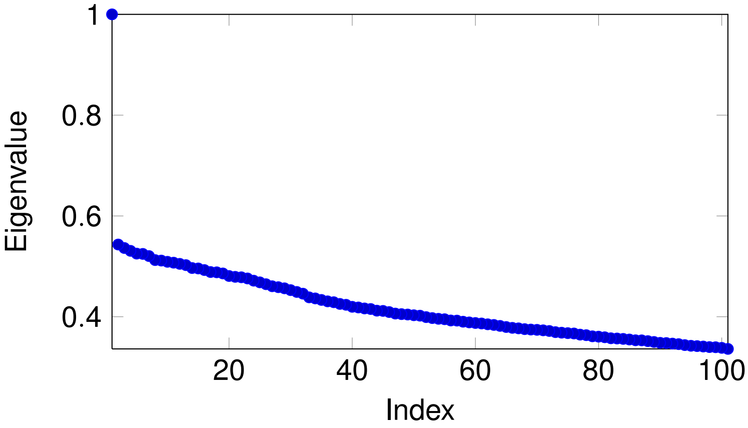


Score vector for the two-node seed of 492 and 513 in the first LFR benchmark graph. Ten steps, three Ritz vectors.

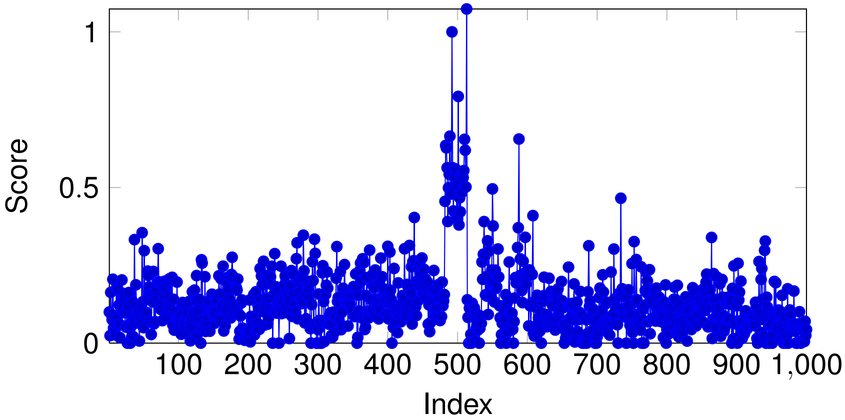
Non-overlapping synthetic benchmark ($\mu = 0.6$)



Spectrum for synthetic benchmark



Score vector

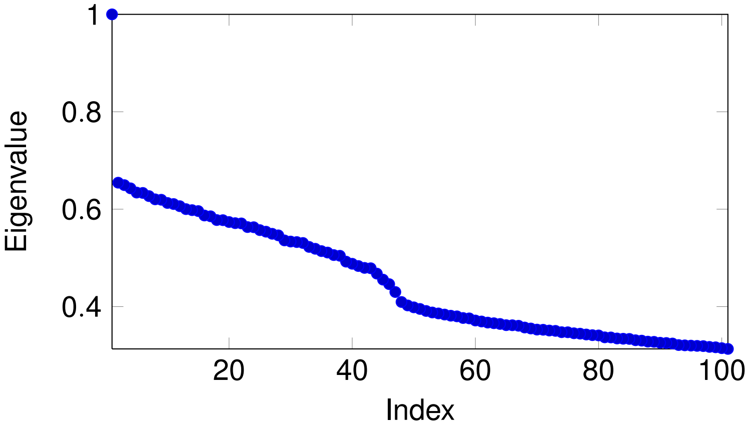


Score vector for the two-node seed of 492 and 513 in the first LFR benchmark graph. Ten steps, three Ritz vectors.

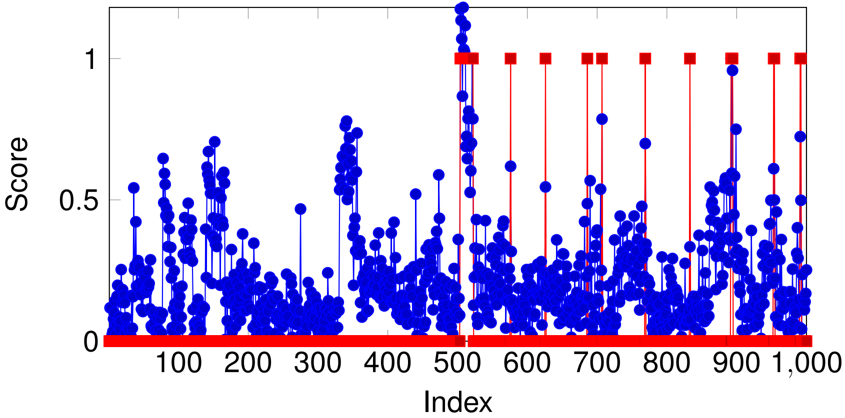
Overlapping synthetic benchmark ($\mu = 0.3$)

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

Spectrum for synthetic benchmark

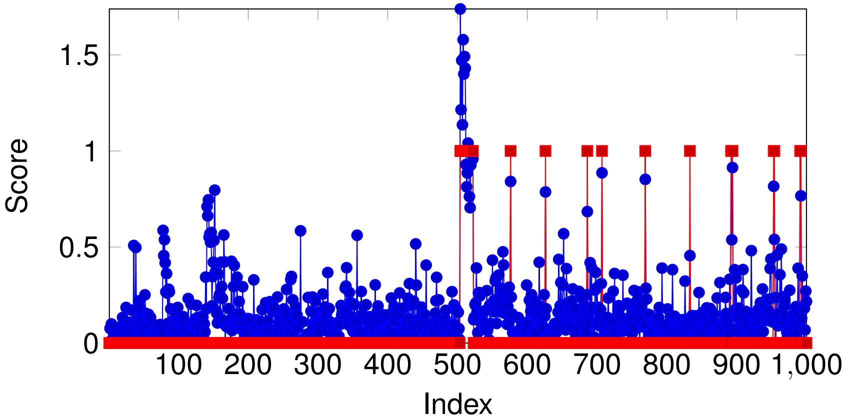


Score vector



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 cluster, 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

Still very much need:

- Better theoretical grounding
- Better ideas for thresholding
- Testing on large-scale examples