“A Unified View of Spectral Clustering” by Desmond J. Higham and Milla Kibble[6]

Teresa Selee

MA 591

Wednesday, March 28, 2007

Outline

- Introduction
  - Problem statement
  - Applications
  - History
  - Notation
  - Example
- The Initial Problem
- Relaxation
- The 3^{rd} eigenvector
- Numerical Examples

Basic idea

- “discrete optimization problem that leads to a simple and informative derivation of a widely used class of spectral clustering algorithms"
- algorithms partition graph into 2 non-empty sets, regardless of the number in each set
- results for normalized and unnormalized Laplacian
- directly explain how Laplacian eigenvectors (after the Fiedler vector) give more clustering information

Problem Statement

cluster points so that within each group objects are similar and objects in different groups are dissimilar

- objects/points are vertices of an undirected graph
- edges are weighted according to symmetric weight matrix W

Applications

- computer imaging
- patter recognition
- high-performance scientific computing
- sparse matrix computations
- multicasting
- graph layout
- data mining
- informatics
- spectral clustering $\longleftarrow$ general focus for this paper

History

- Fiedler vector got its name from key properties identified in [5]
- early reference from numerical analysis/matrix re-ordering perspective used unnormalized Fiedler vectors [2]
- the normalized Laplacian/Fiedler vector was first propsed in [4]
- there are multiple computer science and graph theory resources, including [1, 7, 8, 10, 11]

Notation

- n = number of vertices or objects
- $w_{ij} \equiv$ similarity weight,
  $w_{ij} = w_{ji}, \qquad w_{ii} = 0$
  large weights $\Longrightarrow$ high similarity
- $W \in \mathbb{R}^{n \times n} \equiv$ symmetric weight matrix
- diagonal matrix $D \in \mathbb{R}^{n \times n}$,
  $d_i = \sum_{j=1}^{n} w_{ij}$
- $D - W \equiv$ (unnormalized) Laplacian
- $D^{-1/2}(D - W)D^{-1/2} \equiv$ normalized Laplacian

On the (unnormalized) Laplacian

- symmetric, positive, semi-definite
- constant row sums $= 0$
- smallest eigenvalue $= 0$,
  associated eigenvector $e = \begin{pmatrix} 1 & 1 & \cdots & 1 \end{pmatrix}^T$
- assume connected graph (any pair of vertices is connected by a path along non-zero weighted edges) $\Longrightarrow$ all other eigenvalues are positive [3, 4]
- eigenvalues are ordered, and 0 is assumed unique

$$0 = \lambda_1 < \lambda_2 \leq \lambda_3 \leq \cdots \leq \lambda_n$$

- eigenvalues have corresponding, mutually orthogonal eigenvectors $v_1, v_2, \ldots, v_n$ with $v_1 = e/\sqrt{n}$

On the Normalized Laplacian

- symmetric, positive, semi-definite
- smallest eigenvalue $\mu_1 = 0$,
  associated eigenvector $D^{1/2}e$
- eigenvalues are ordered, and 0 is assumed unique

$$0 = \mu_1 < \mu_2 \leq \mu_3 \leq \cdots \leq \mu_n$$

- eigenvalues satisfy $0 \leq \mu_i \leq 2$, [4]
- eigenvalues have corresponding, mutually orthogonal eigenvectors $w_1, w_2, \ldots, w_n$ with

$$w_1 = \left(D^{1/2}e\right) / \|D^{1/2}e\|_2$$

The Fiedler vector

$$
\begin{aligned}
v_2 &\equiv \text{the Fiedler vector of the Laplacian} \\
&\equiv \text{eigenvector associated with the} \\
&\qquad \text{second smallest eigenvalue of } D - W
\end{aligned}
$$

$$
\begin{aligned}
D^{-1/2} w_2 &\equiv \text{the normalized Fiedler vector of the} \\
&\qquad \text{normalized Laplacian} \\
&\equiv D^{-1/2} \times [\text{eigenvector associated} \\
&\qquad \text{with the } 2^{nd} \text{ smallest eigenvalue of} \\
&\qquad D^{-1/2}(D - W)D^{-1/2}]
\end{aligned}
$$

Clustering

Unnormalized case:
compute 2^{nd} eigenvector of the Laplacian, v_2 and perhaps the next few eigenvectors as well.

Normalized case:
compute $D^{-1/2}w_2, D^{-1/2}w_3, D^{-1/2}w_4, \cdots$ from the normalized Laplacian

- the information in the eigenvectors forms a basis for clustering decisions
- the details vary across the literature [1, 9, 4]

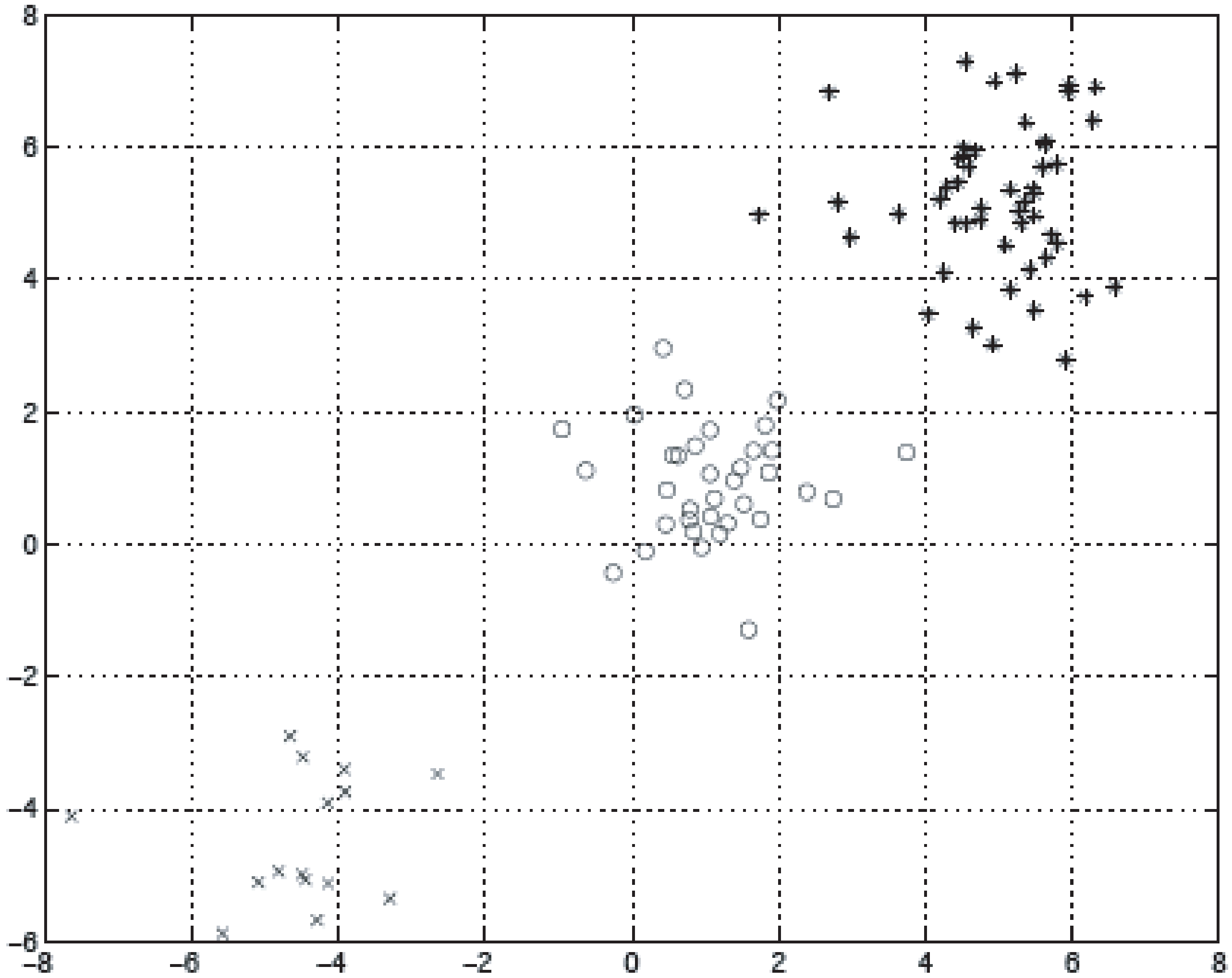
Main idea:

what can eigenvectors tell us in terms of clustering?

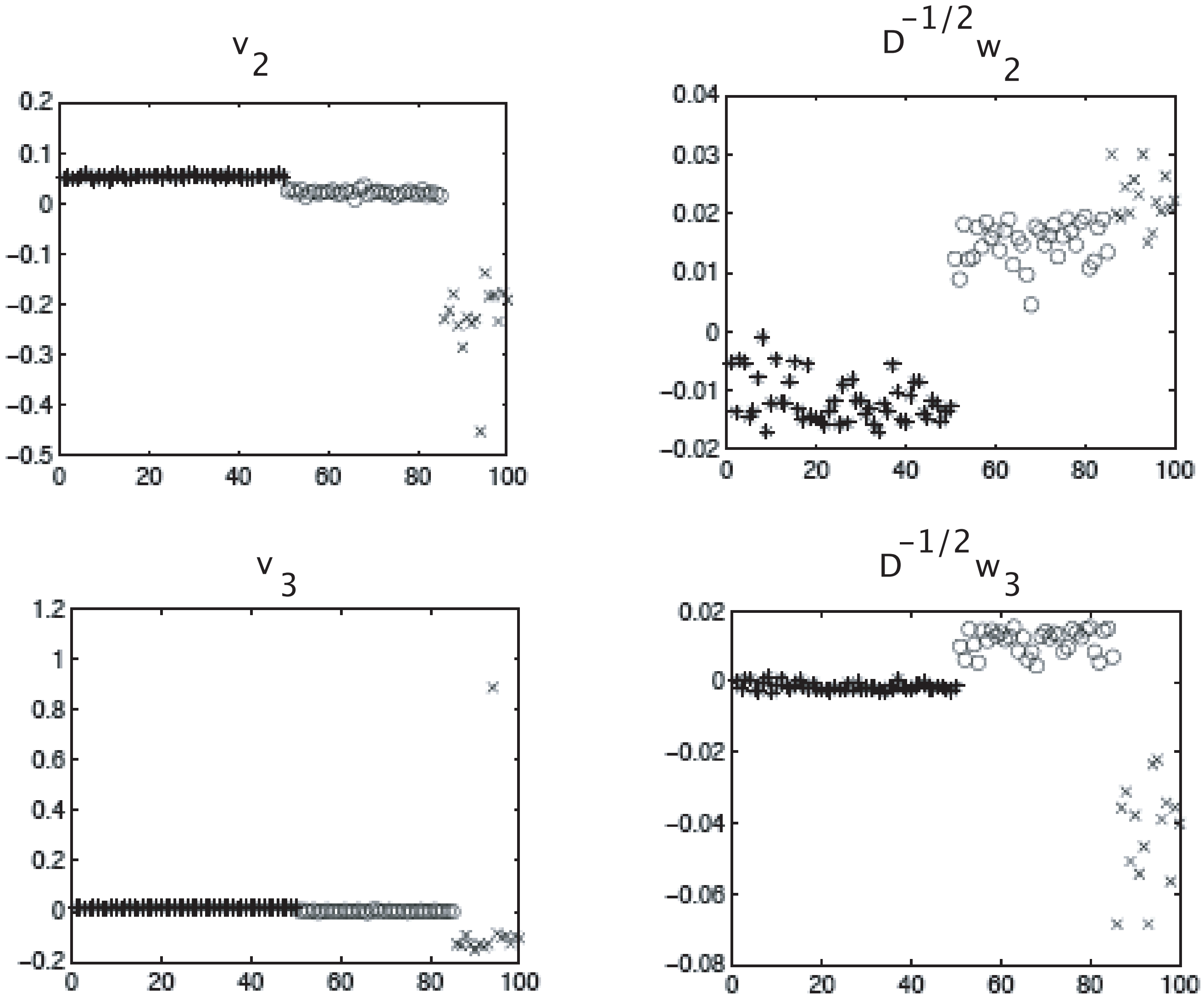
Example

- 100 points in the $x - y$ plane are to be clustered
- $w_{ij} \equiv$ reciprocal of Euclidean distance between i and j
- there are natural clusters in the data
- plot the eigenvectors v_2 and v_3 and the scaled, normalized vectors $D^{-1/2}w_2$ and $D^{-1/2}w_3$
- normalized & unnormalized vectors present information very differently

Data Points



2^{nd} and 3^{rd} eigenvectors



The Initial Partitioning

Begin with a partition of the vertices into 2 distinct sets A and B using

$$y_i = \begin{cases} -\frac{1}{2}, & \text{if vertex } i \text{ is in } A \\ \frac{1}{2}, & \text{if vertex } i \text{ is in } B \end{cases}$$

Initial Problem

Reasonable starting point:

$$\min_{y_i \in \{\pm 1/2\}} \sum_{i,j} \left(y_i - y_j\right)^2 w_{ij}.$$

Contains $2 \times$(total weight of the edges between A and B).

- if y_i and y_j are in the same set, get 0 for that term
- If y_i and y_j are in different sets, get 1 for that term

Notice that

$$(y_i - y_j)^2 w_{ij} + (y_j - y_i)^2 w_{ji} = 2w_{ij} = 2w_{ji}$$

when i and j are in different sets.

Alternative to the original problem

No partitioning occurs with the equation

$$\min_{y_i \in \{\pm 1/2\}} \sum_{i,j} \left(y_i - y_j\right)^2 w_{ij}$$

Minimize by putting all vertices in same group.

Need a modification:

- alter the objective function [9]
- add an extra constant [1, 4, 6]

New idea: use added constant to derive spectral clustering algorithm

Modified Problem

$$\min_{\substack{y_i \in \{\pm 1/2\} \\ |y^T e| \leq \beta}} \sum_{i,j} (y_i - y_j)^2 w_{ij}$$

where $\beta \equiv$ balancing threshold.

Define:
$2y^T e \equiv$ difference b/w number of vertices in each partition

$|y^T e| \leq \beta$ controls how uneven the groups are

Extremes:

- $\beta = 1/2 \longrightarrow$ most even splitting
  (e.g. k vertices in A, $k \pm 1$ vertices in B).
- $\beta = n/2 \longrightarrow$ allows for all bi-partitions, incl. empty set
  Equal with or without modification

Different Modified Problem

Alternate restriction changes

$$|y^T e| \leq \beta \quad \text{to} \quad |y^T D e| \leq \beta$$

$$\min_{\substack{y_i \in \{\pm 1/2\} \\ |y^T D e| \leq \beta}} \sum_{i,j} (y_i - y_j)^2 w_{ij}$$

where
$D \equiv$ controls difference between total weight in each cluster

Can think of D as balancing the center of mass

Complications

Both modifications are very sensitive to choice of β

Fix: exploit natural relaxation tendency of the problem

Result:

- solutions that are extremely insensitive to β
- appropriate balance of nodes in each set
- small number of edges that span the 2 sets

Relaxation

- weaken constraint from $y_i = \{\pm\frac{1}{2}\}$ to $y_i \in \mathbb{R}$
- components of y *should* fall into distinct groups
- scaling issue:
  - could just scale y by ϵ to get the minimum
  - fix: normalize y

Relaxation

With relaxation, the unnormalized problem becomes

$$\min_{\substack{y \in \mathbb{R}^n \\ |y^T e| \leq \beta \\ y^T y = n/4}} \sum_{i,j} (y_i - y_j)^2 w_{ij}$$

where the condition $y^T y = n/4$ comes from the initial $y_i = \pm\frac{1}{2}$.

Simplify further by scaling each y_i by $2/\sqrt{n}$

$$\min_{\substack{y \in \mathbb{R}^n \\ |y^T e| \leq \frac{2\beta}{\sqrt{n}} \\ y^T y = 1}} \sum_{i,j} (y_i - y_j)^2 w_{ij}.$$

Relaxation

With relaxation, the normalized problem becomes

$$\min_{\substack{y \in \mathbb{R}^n \\ |y^T De| \leq \frac{\beta}{\sqrt{\theta n}} \\ y^T Dy = 1}} \sum_{i,j} (y_i - y_j)^2 w_{ij}$$

Should be least affected by poorly calibrated vertices with abnormally large/small weights

Comments on Relaxation

- since $\min \sum (y_i - y_j)^2 w_{ij}$, large y_i largely influences other y_j
- fix $y^T D y = \theta n$ to encourage y_i close to 0 when d_i is large
  - ensures 1 or 2 highly weighted nodes don't throw everything off
  - avoids committing vertex i too strongly to either cluster
- $y^T D y = \theta n$ normalizes the energy

Theorem

Solve relaxed problems with a variation of Rayleigh- Ritz

THEOREM:
Let $A \in \mathbb{R}^{n\times n}$ *be symmetric with*

- *eigenvalues ordered* $\nu_1 < \nu_2 \le \cdots \le \nu_n$
- *associated mutually orthonormal eigenvectors* $x_1, x_2, \ldots, x_n$.

Then for fixed $0 \le \alpha < 1$, *the problem:*

$$\min_{\substack{y\in\mathbb{R}^n \\ |y^T x_1|\le\alpha \\ y^T y=1}} \quad y^T A y$$

is solved by $y = \alpha x_1 + \sqrt{1-\alpha^2}x_2$.

Corollary

For $0 \le \beta < n/2$*, the unnormalized relaxed problem has solution*

$$y = \frac{2\beta}{n\sqrt{n}} e + \sqrt{1 - 4\frac{\beta^2}{n^2}} v_2.$$

For $0 \le \beta \le \sqrt{\theta n}\|D^{1/2}e\|_2$*, the normalized relaxed problem has solution*

$$y = \frac{\beta}{\sqrt{\theta n}\|D^{1/2}e\|_2^2} e + \sqrt{1 - \frac{\beta^2}{\theta n \|D^{1/2}e\|_2^2}} \left(D^{-1/2}w_2\right).$$

Conclusions

Clustering:

- both multiples of e from the corollary do not affect the clustering
- just use v_2 or $D^{-1/2}w_2$, in the unnormalized and normalized cases, respectively

After relaxation, the value of β no longer matters

(as long as it fits within the stated bounds).

The third eigenvector

The problem:

$$\min \sum_{i,j} (y_i - y_j)^2 w_{ij}$$

over $y \in \mathbb{R}^n$ subject to

$$y^T y = 1, \; |y^T v_1| \le \alpha_1, \; |y^T v_2| \le \alpha_2, \; \alpha_1^2 + \alpha_2^2 < 1$$

produces

$$\frac{\alpha_1}{\sqrt{n}} e + \alpha_2 v_2 + \sqrt{1 - \alpha - 1^2 - \alpha - 2^2} v_3.$$

Thus, v_3 is the next best direction after the Fiedler vector v_2.

Lemma

If the weights are altered so that

$$w_{ij} \longmapsto w_{ij} - \nu (v_2)_i (v_2)_j$$

for any fixed $\nu > \lambda_3 - \lambda_2$. Then the new weight matrix has Fiedler vector v_3.

Similarly, if

$$w_{ij} \longmapsto w_{ij} - \nu \sqrt{d_i}\,[(w_2)_i]\,\sqrt{d_j}\,[(w_2)_j]$$

for any fixed $\nu > \mu_3 - \mu_2$, then the new weight matrix has normalized Fiedler vector $D^{-1/2} w_3$.

Interpretation of Lemma

(for first case of lemma)

- recall v_2 attempts to split vertices into 2 sets
- expect $(v_2)_i$ and $(v_2)_j$ to have
  - same sign if i and j are in the same cluster
  - opposite sign if i and j are in different clusters
- then for $w_{ij} \longmapsto w_{ij} - \nu(v_2)_i(v_2)_j$
  - decrease w_{ij} if v_2 assigns i and j to the same cluster
  - increase w_{ij} if v_2 assigns i and j to different clusters

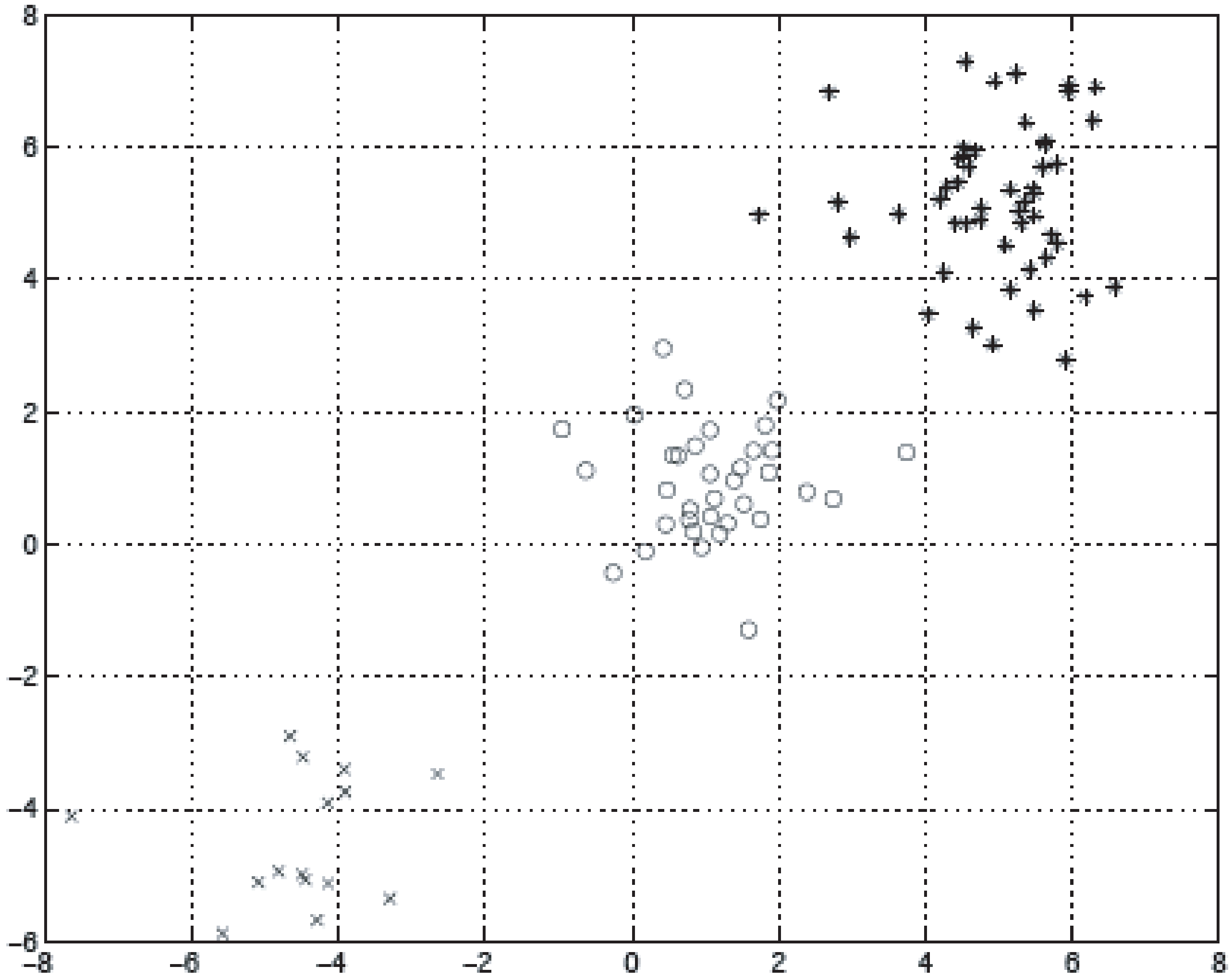
Conclusions

- think of v_3 as solving a new problem
  - try to remove automatic bias toward bi-partitioning
- apply lemma to higher eigenvalues
  - subtract sufficiently large multiples of $v_2 v_2^T, v_3 v_3^T, \ldots, v_k v_k^T$
  - use v_{k+1} as the new Fiedler vector
- new weights in the lemma can produce negative weights
  - the more negative the value of w_{ij}
  - the more dissimilar i and j
  - the further apart y_i and y_j

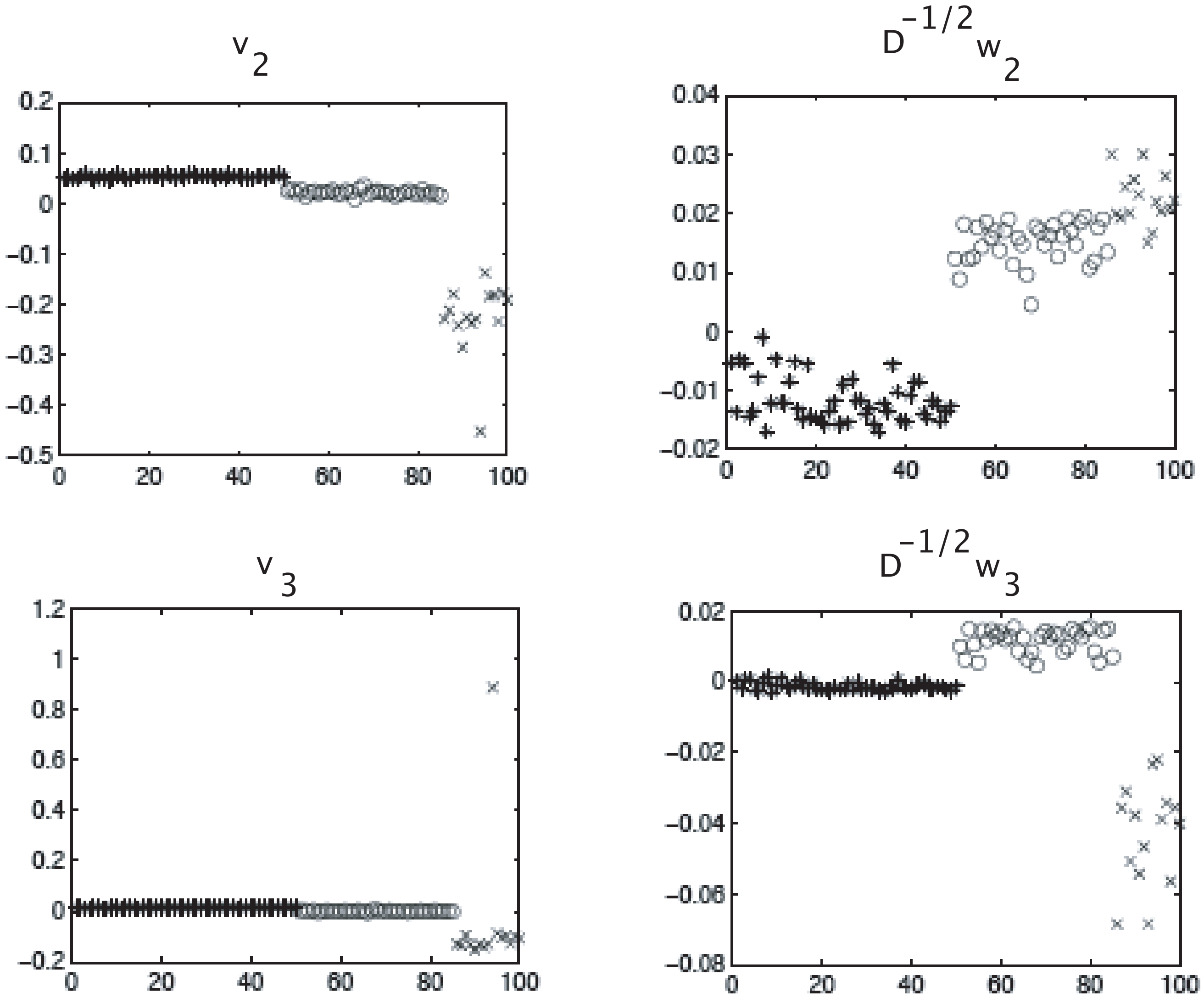
Numerical example

- 100 points in the $x - y$ plane are to be clustered
- $w_{ij} \equiv$ reciprocal of Euclidean distance between points i and j
- there are natural clusters in the data
- plot the eigenvectors v_2 and v_3 and the scaled, normalized vectors $D^{-1/2}w_2$ and $D^{-1/2}w_3$
- normalized & unnormalized vectors present information very differently

Data Points



2^{nd} and 3^{rd} eigenvectors



Fiedler vector, v_2

- identifies 3 natural clusters
- smallest cluster of 15 '$\times$' yields largest $|(v_2)_i|$
  $\longrightarrow$ follows from analysis
- '$\times$' data produces small weights
  - $\min \sum_{i,j} (y_i - y_j)^2 w_{ij}$ subject to
    $y^T y = 1$ and $|y^T e| \le 2\beta/\sqrt{n}$
  - minimized by choosing large values for y_i
  - choose small values with opposite sign for '$\circ$' and '$+$' data points
- note: v_2 notices the natural outlier
- because v_2 separates completely, v_3 can give no more info except for the outlier

Normalized Fiedler vector, $D^{-\frac{1}{2}}w_2$

- clusters data into 2 sets:

  puts '$\times$' and '$\circ$' together to get 2 equal size groups

- why?

  relaxation says this should be insensitive to relatively small weights in '$\times$'

- $D^{-1/2}w_3$ separates '$\times$' and '$\circ$'

Conclusions from 2^{nd} eigenvector

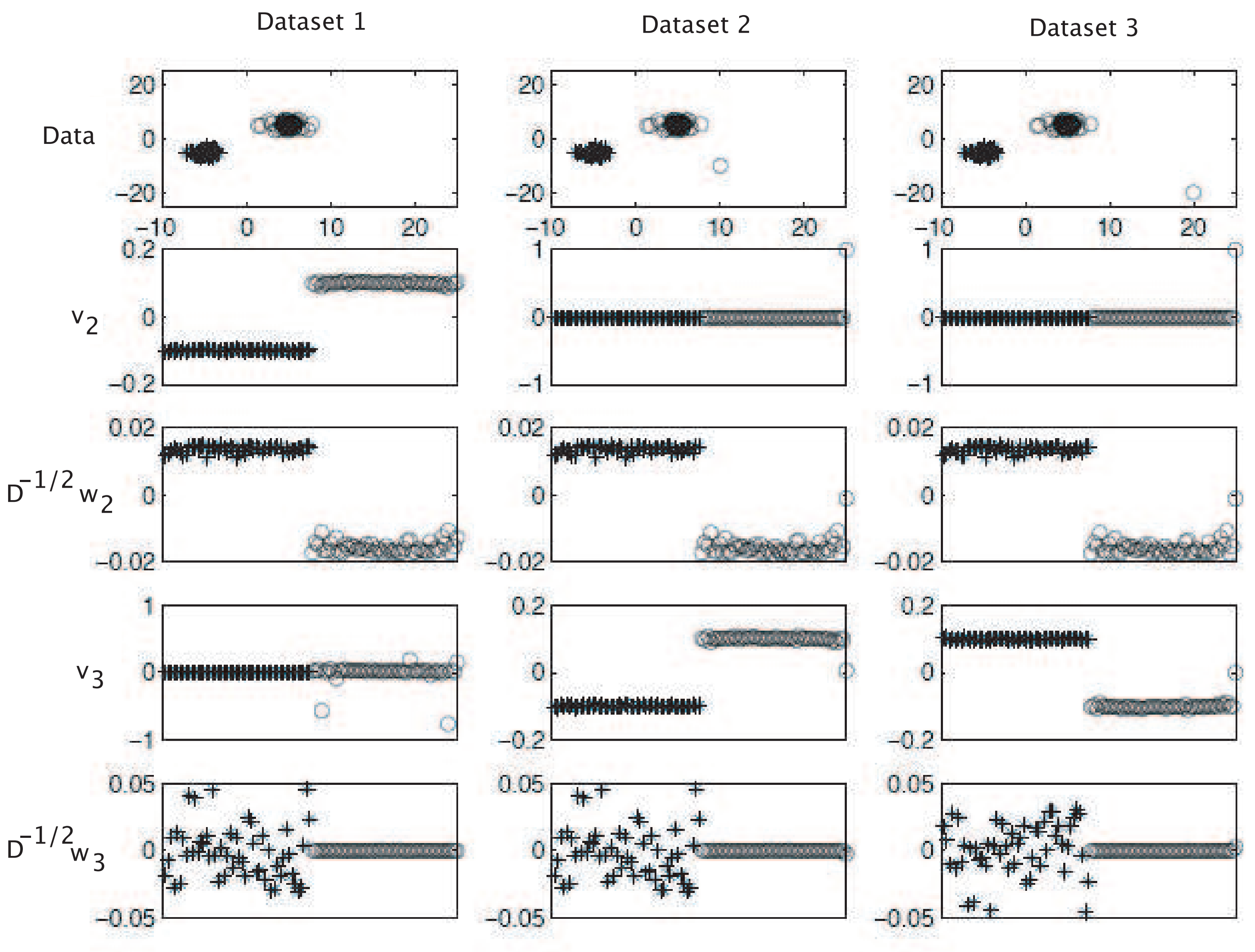
- both unnormalized and normalized separate the data
- only the unnormalized case recognizes the outlier
- Is the outlier important?

  depends on user & context of algorithm

The Effect of Outliers

- similarity weights, $w_{ij} \equiv$ reciprocal of Euclidean distance between i and j
- Data set 1:
  50 clustered '$+$' and 50 clustered '$\circ$'
- Data set 2:
  50 clustered '$+$', 49 clustered '$\circ$', 1 outlier '$\circ$'
- Data set 3:
  50 clustered '$+$', 49 clustered '$\circ$', 1 farther outlier '$\circ$'

Data and Eigenvectors: outliers



Unnormalized Eigenvectors

v_2

- separates the clusters for dataset 1
- separates outlier from remainder of points for datasets 2 & 3

v_3

- does nothing for dataset 1 since already sorted
- separates non-outlier cluster for datasets 2 & 3

Normalized Eigenvectors

Relatively insensitive to presence of outliers due to relaxation

$D^{-1/2}w_2$

- separates the clusters for dataset 1
- separates both partitions, plus outlier for datasets 2 & 3

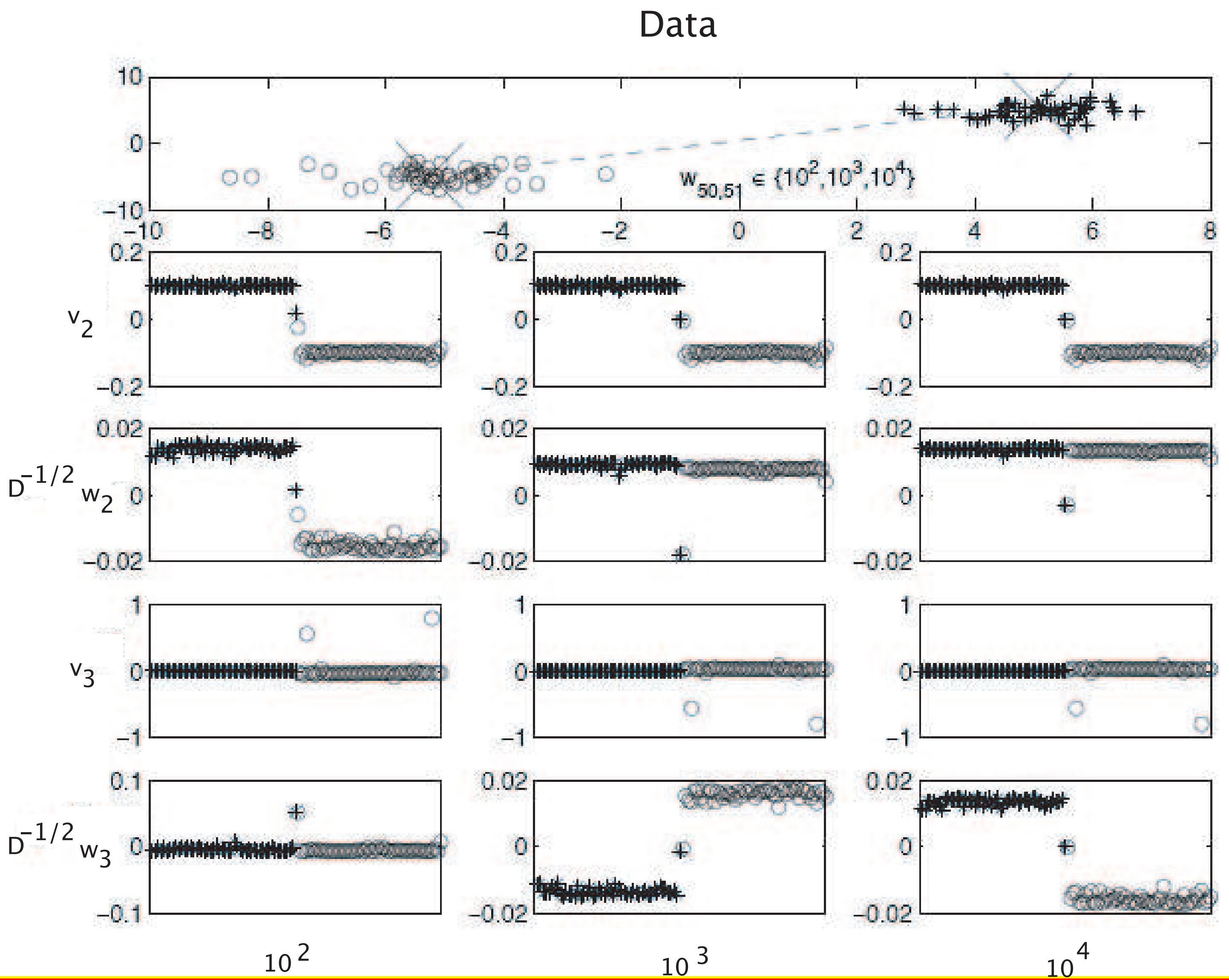
$D^{-1/2}w_3$

- does nothing for datasets 1, 2, & 3 since already sorted

The Effect of a Strong Weight

- 50 clustered '$+$' and 50 clustered '$\circ$'
- similarity weights, $w_{ij} \equiv$ reciprocal of Euclidean distance between i and j
- change weight between points 50 and 51
  (these connect the '$\circ$' and '$+$' clusters)
- three trials for $w_{50,51}$: 10^2, 10^3 and 10^4
- $\max_{i,j} = 12.1$ before change to $w_{50,51}$

Data and e-vectors: strong weight



Unnormalized Eigenvectors

For all values of $w_{50,51} (= 10^2, 10^3,$ and $10^4)$

v_2

- Finds 3 clusters
  - the '$+$' cluster of points 1-49
  - the '$\circ$' cluster of points 52-100
  - a cluster with the 2 strongly connected points 50 & 51

v_3

- does nothing since everything is already sorted

Normalized Eigenvectors

Depends on value of $w_{\mathbf{50,51}}$

$D^{-1/2}w_{\mathbf{2}}$

- $10^2 \longrightarrow$ splits 2 clusters and clusters $(50, 51)$
- $10^3 \longrightarrow$ bi-partition of $(50, 51)$ and all other points
- $10^4 \longrightarrow$ same as for 10^3, but make $(50, 51)$ close to 0

$D^{-1/2}w_{\mathbf{3}}$

- $10^2 \longrightarrow$ does nothing, already clustered
- $10^3 \longrightarrow$ completes partitioning of '$+$' and '$\circ$'
- $10^4 \longrightarrow$ completes partitioning of '$+$' and '$\circ$'

Conclusions

- Problem statement: spectral clustering
- unnormalized and normalized Laplacians & eigenvectors
- use minimization formula
  - improve with β restriction
  - improve further with relaxation
- using $\mathbf{3}^{rd}$ eigenvector for additional clustering
- numerical examples to illustrate points

References

[1] C. J. Alpert and S.-Z. Yao, *Spectral partitioning: The more eigenvectors the better*, in Proceedings of the 32^{nd} Conference on Design Automation, 1995, pp. 195–200.

[2] S. T. Barnard, A. Pothen, and H. D. Simon, *A spectral algorithm for envelope reduction of sparse matrices*, Numerical Linear Algebra with Applications, 2 (1995), pp. 317–334.

[3] I. S. Dhillon, *Co-clustering documents and words using bipartite spectral graph partitioning*, in Proceedings of the Seventh ACM SIGKKD Conference, 2001.

[4] R. V. Driessche and D. Roose, *An improved spectral bisection algorithm and its application to dynamic load balancing*, Parallel Computing, 21 (1995), pp. 29–48.

[5] M. Fiedler, *A property of eigenvectors of nonnegative symmetric matrices and its applications to graph theory*, Czechoslovak Mathematical Journal, 25 (1975), pp. 619–633.

[6] D. J. Higham and M. Kibble, *A unified view of spectral clustering*, Mathematics Research report 02, University of Strathclyde, 2004.

[7] R. KANNAN, S. VEMPALA, AND A. VETTA, *On clusterings: good, bad, and spectral*, in Proceedings of the 41^{st} Foundations of Computer Science (FOCS ’00), 2000.

[8] A. Y. NG, M. I. JORDAN, AND Y. WEISS, *On spectral clustering: Analysis and an algorithm*, NIPS, (2001).

[9] J. SHI AND J. MALIK, *Normalized cuts and image segmentation*, IEEE Transactions on Pattern Analysis and Machine Intelligence, 22 (2000), pp. 888–905.

[10] D. A. SPIELMAN AND S.-H. TENG, *Spectral partitioning works: Planar graphs and finite element meshes*, in Proceedings of the 37^{th} Annual IEEE Conference on Foundations of Computer Science, 1996.

[11] E. P. XING AND R. P. KARP, *Cliff:Clustering of high-dimensional microarray data via iterative feature filtering using normalized cuts*, Bioinformatics (Discovery Note), 1 (2001), pp. 1–9.