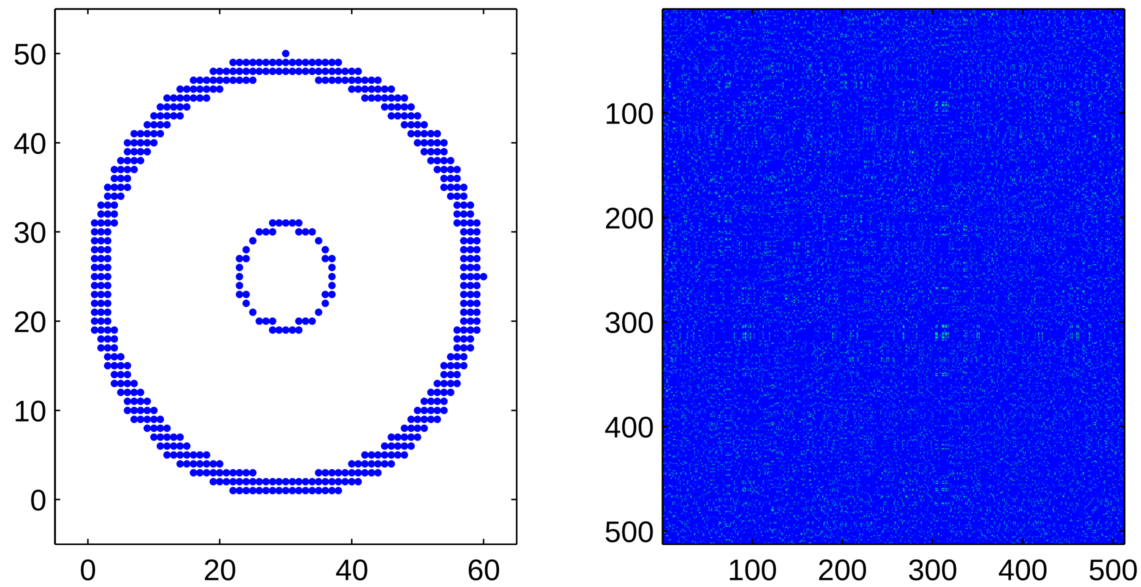
On Spectral Bipartite Clustering Algorithm and Automatic Determination of the Number of Clusters

by Ivančica Mirošević and Nevena Jakovčević Stor

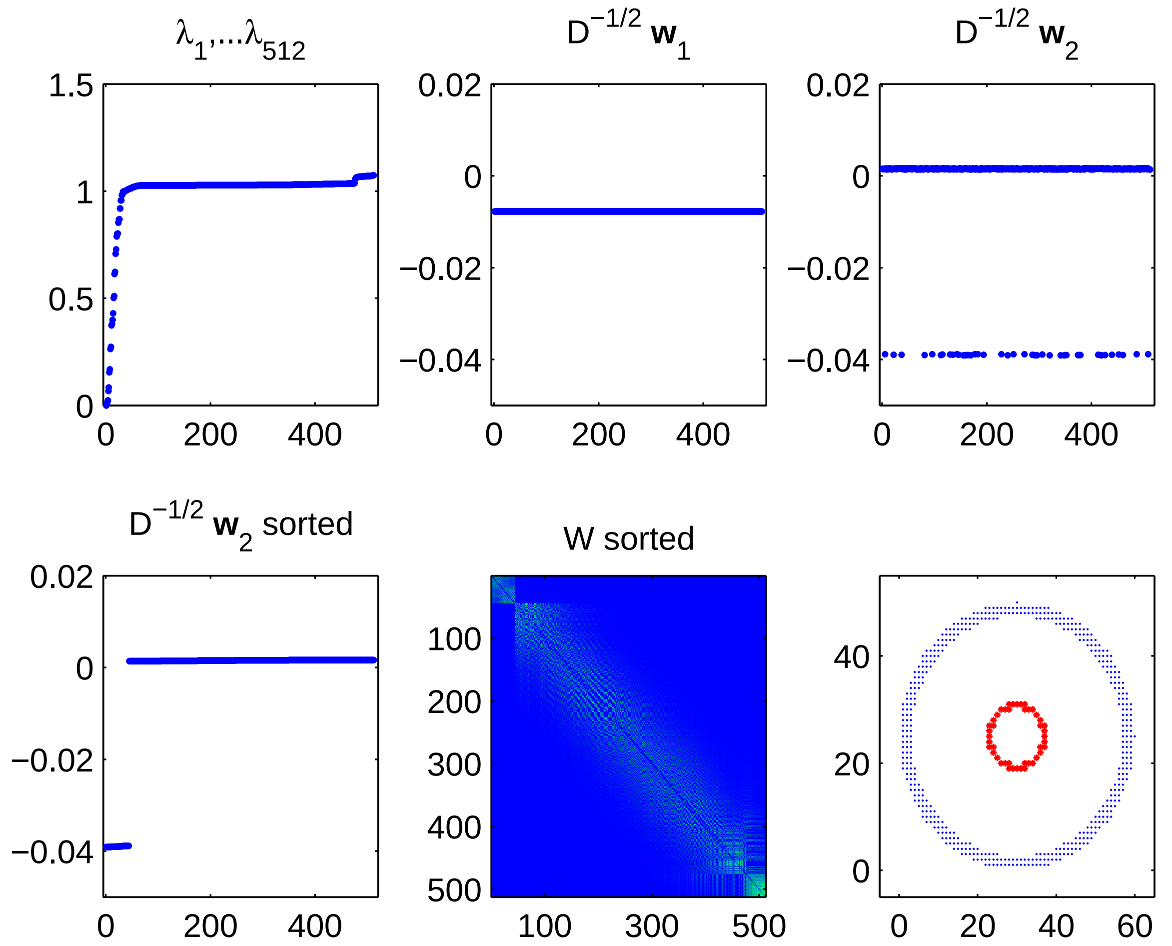
Faculty of EE, ME and Naval Arch.

University of Split

The clustering problem



Recovering the structure



Graph notation

$G = (V, E)$ is a simple, finite, undirected, weighted graph where:

$V = \{1, 2, 3, ..., n\}$ is a set of vertices and

E is a set of edges $\{i, j\}$, $i, j \in V$, with weights $w_{ij} \in \mathbb{R}^+$.

The weighted adjacency matrix of G is a $n \times n$ matrix

$$W = [w_{ij}].$$

$w_{ij} = 0$ means vertices i and j are not connected by an edge.

Graph cut

Let $V_1, V_2 \subset V$, $V_1, V_2 \neq \emptyset$. We define

$$\mathrm{cut}(V_1, V_2) = \sum_{i \in V_1, j \in V_2} w_{ij},$$

$$d_i = \sum_{j=1}^{n} w_{ij},$$

$$\mathrm{vol}(V_l) = \sum_{i \in V_l} d_i = \sum_{i \in V_l} \sum_{j \in V} w_{ij} = \mathrm{cut}(V_l, V \setminus V_l) + \mathrm{within}(V_l).$$

Partitioning functions

Let $\pi = \{V_1, V_2\}$ be the partition of V.

Ratio cut*

$$R(V_1, V_2) = \frac{\operatorname{cut}(V_1, V_2)}{|V_1|} + \frac{\operatorname{cut}(V_1, V_2)}{|V_2|}$$

favors partitions into sets with equal number of vertices.

Normalized cut**

$$N(V_1, V_2) = \frac{\operatorname{cut}(V_1, V_2)}{\operatorname{vol}(V_1)} + \frac{\operatorname{cut}(V_1, V_2)}{\operatorname{vol}(V_2)}$$

favors partitions into sets with equal weights of edges within subsets.

---

* Hagen and Kahng, 1992
** Shi and Malik, 2000

Graph Laplacian

$L = [l_{ij}]$ is a real $n \times n$ matrix, s.t.

$$l_{ij} = \begin{cases} \sum_{k=1}^{n} w_{ik} & , \quad i = j \\ -w_{ij} & , \quad i \neq j \end{cases}.$$

The matrix L satisfies the following properties:

- → $L = D - W$, where D is a degree matrix (diagonal matrix with degrees $d_1, d_2, \cdots, d_n$ on its diagonal);
- → L is symmetric and positive semi-definite;
- → $L\mathbf{1} = 0$ for $\mathbf{1} = [1, ..., 1]^T$;
- → The multiplicity k of the eigenvalue 0 equals the number of connected components in the graph;
- → L has n real-valued eigenvalues $0 = \lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_n$.

Normalized graph Laplacian

$L_n = [l_{n_{ij}}]$ is a $n \times n$ matrix, s.t.

$$(l_n)_{ij} = \begin{cases} 1 \quad , & i = j \\ -\frac{w_{ij}}{\sqrt{d_i}\sqrt{d_j}} \quad , & i \neq j \end{cases}.$$

In other words,

$$L_n = D^{-1/2}(D - W)D^{-1/2}.$$

L_n is symmetric and positive semi-definite matrix with the smallest eigenvalue 0 and corresponding eigenvector $D^{\frac{1}{2}}\mathbf{1}$.

Relaxation of the discrete problem

The partition $\pi = \{V_1, V_2\}$ of V is determined by a vector $\mathbf{y}$ s.t.

$$y_i = \begin{cases} \frac{1}{2} & \text{za } i \in V_1 \\ -\frac{1}{2} & \text{za } i \in V_2 \end{cases}$$

The Ratio cut problem:

$$\min_{\substack{y_i \in \{-\frac{1}{2}, \frac{1}{2}\} \\ |\mathbf{y}^T \mathbf{1}| \leq \beta}} \frac{1}{2} \sum_{i,j} \left(y_i - y_j\right)^2 w_{ij}$$

The Normalized cut problem:

$$\min_{\substack{y_i \in \{-\frac{1}{2}, \frac{1}{2}\} \\ |\mathbf{y}^T D \mathbf{1}| \leq \beta}} \frac{1}{2} \sum_{i,j} \left(y_i - y_j\right)^2 w_{ij}$$

Relaxation of the discrete problem

The partition $\pi = \{V_1, V_2\}$ of V is determined by a vector $\mathbf{y}$ s.t.

$$y_i = \begin{cases} \frac{1}{2} & \text{za } i \in V_1 \\ -\frac{1}{2} & \text{za } i \in V_2 \end{cases}$$

The Ratio cut problem:

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

The Normalized cut problem:

$$\min_{\substack{y_i \in \{-\frac{1}{2}, \frac{1}{2}\} \\ |\mathbf{y}^T D \mathbf{1}| \leq \beta}} \frac{1}{2} \sum_{i,j} \left(y_i - y_j\right)^2 w_{ij}$$

Relaxation of the discrete problem

The partition $\pi = \{V_1, V_2\}$ of V is determined by a vector $\mathbf{y}$ s.t.

$$y_i = \begin{cases} \frac{1}{2} & \text{za } i \in V_1 \\ -\frac{1}{2} & \text{za } i \in V_2 \end{cases}$$

The Ratio cut problem:

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

The Normalized cut problem:

$$\min_{\substack{y_i \in \{-\frac{1}{2}, \frac{1}{2}\} \\ |\mathbf{y}^T D \mathbf{1}| \leq \beta}} \frac{1}{2} \sum_{i,j} \left(y_i - y_j\right)^2 w_{ij} \qquad \Rightarrow \qquad \min_{\substack{y \in \mathbb{R}^n \\ |\mathbf{y}^T D \mathbf{1}| \leq \frac{\beta}{\sqrt{\theta n}} \\ \mathbf{y}^T D \mathbf{y} = 1}} \mathbf{y}^T L \mathbf{y}$$

The Theorem

Theorem 1 *Let $A \in \mathbb{R}^{n\times n}$ be a symmetric matrix with eigenvalues $\lambda_1 < \lambda_2 < \lambda_3 \leq \cdots \leq \lambda_n$ and eigenvectors $\mathbf{v}_1, \mathbf{v}_2, ..., \mathbf{v}_n$. For a fixed $0 \leq \alpha < 1$, the problem*

$$\min_{\substack{\mathbf{y}\in\mathbb{R}^n \\ |\mathbf{y}^T\mathbf{v}^{[1]}|\leq\alpha \\ \mathbf{y}^T\mathbf{y}=1}} \mathbf{y}^T A \mathbf{y}$$

has the solution $y = \pm\alpha\mathbf{v}_1 \pm \sqrt{1-\alpha^2}\mathbf{v}_2$.

The solution (1)

Corollary 1 *For* $0 \leq \beta < \frac{n}{2}$ *the relaxed Ratio Cut problem*

$$\min_{\substack{y \in \mathbb{R}^n \\ |\mathbf{y}^T \mathbf{1}| \leq \frac{2\beta}{\sqrt{n}} \\ \mathbf{y}^T \mathbf{y} = 1}} \mathbf{y}^T L \mathbf{y}$$

has the solution

$$\mathbf{y} = \pm \frac{2\beta}{\sqrt{n}} \mathbf{1} \pm \sqrt{1 - 4\frac{\beta^2}{n^2}} \mathbf{v}_2.$$

$\mathbf{v}_2$ is the Fiedler vector of the Laplacian L.

The solution (2)

Corollary 2 *For* $0 \leq \beta < \sqrt{\theta n} \left\| D^{\frac{1}{2}} \mathbf{1} \right\|_2$ *the relaxed normalized cut problem*

$$\min_{\substack{y \in \mathbb{R}^n \\ |\mathbf{y}^T D \mathbf{1}| \leq \frac{\beta}{\sqrt{\theta n}} \\ \mathbf{y}^T D \mathbf{y} = 1}} \quad \mathbf{y}^T L \mathbf{y}$$

has the solution

$$\mathbf{y} = \pm \frac{\beta}{\sqrt{\theta n} \left\| D^{\frac{1}{2}} \mathbf{1} \right\|_2^2} \mathbf{1} \pm \sqrt{1 - \frac{\beta^2}{\theta n \left\| D^{\frac{1}{2}} \mathbf{1} \right\|_2^2}} D^{-\frac{1}{2}} \mathbf{w}_2,$$

$D^{-\frac{1}{2}} \mathbf{w}_2$ is the normalized Fiedler vector (of the normalized Laplacian).

Constructing the partition

According to the definition, the sets V_1 and V_2 are determined by

$$V_1 = \{i : \mathbf{v}_2(i) < 0\}, \quad V_2 = \{i : \mathbf{v}_2(i) \geq 0\},$$

for the Ratio Cut, and

$$V_1 = \{i : D^{-\frac{1}{2}}\mathbf{w}_2(i) < 0\}, \quad V_2 = \{i : D^{-\frac{1}{2}}\mathbf{w}_2(i) \geq 0\}$$

for the Normalized Cut.

Recursive bipartitioning - the algorithm

Input: Adjacency matrix W, number of clusters k to construct.
Output: Cluster indicator vector $\mathbf{y}$.

1. Bipartition V. Set counter $k_c = 2$.
2. **If** $k_c < k$ **then**
3. For each subset of V compute the optimal bipartition.
4. Within all $(k_c + 1)$-partitions, choose the one with the smallest value of the partitioning function.
5. Set $k_c = k_c + 1$ and proceed recursively with step 2.
6. Stop.

Bipartite graph

Undirected bipartite graph G is a triplet $G = (T, D, E)$.
$T = \{t_1, \cdots, t_m\}$ and $D = \{d_1, ..., d_n\}$ are two sets of vertices and

$$E = \{\{t_i, d_j\} : t_i \in T, d_j \in D\}$$

is a set of edges.

For example, D is a set of documents, T is a set of terms and edge $\{t_i, d_j\}$ exists if document d_j contains term t_i.

The adjacency matrix has the form

$$W = \begin{bmatrix} 0 & A \\ A^T & 0 \end{bmatrix},$$

where A is the *term* × *document* matrix.

Connection to SVD*

Let

$$D = \begin{bmatrix} D_1 & 0 \\ 0 & D_2 \end{bmatrix}, \quad L = \begin{bmatrix} D_1 & -A \\ -A^T & D_2 \end{bmatrix}.$$

Then

$$L_n = D^{-\frac{1}{2}} \begin{bmatrix} D_1 & -A \\ -A^T & D_2 \end{bmatrix} D^{-\frac{1}{2}} = \begin{bmatrix} I & -D_1^{-\frac{1}{2}} A D_2^{-\frac{1}{2}} \\ -D_2^{-\frac{1}{2}} A D_1^{-\frac{1}{2}} & I \end{bmatrix}.$$

*Inderjit Dhillon, 2001

Connection to SVD (2)

Let

$$\mathbf{w} = \begin{bmatrix} \mathbf{u} \\ \mathbf{v} \end{bmatrix}, \quad \mathbf{u} \in \mathbb{R}^m, \quad \mathbf{v} \in \mathbb{R}^n,$$

be an eigenvector of the <u>normalized</u> Laplacian,

$$D^{-\frac{1}{2}} L D^{-\frac{1}{2}} \mathbf{w} = \lambda \mathbf{w}.$$

Then

$$\begin{aligned} D_1^{-\frac{1}{2}} A D_2^{-\frac{1}{2}} \mathbf{v} &= (1-\lambda)\mathbf{u}, \\ D_2^{-\frac{1}{2}} A^T D_1^{-\frac{1}{2}} \mathbf{u} &= (1-\lambda)\mathbf{v}. \end{aligned}$$

Connection to SVD (3)

Instead of computing the Fiedler vector of L_n, we compute the left and right singular vector of the normalized matrix

$$A_n = D_1^{-\frac{1}{2}} A D_2^{-\frac{1}{2}}$$

which correspond to the second largest singular value,

$$A_n \mathbf{v}_2 = \sigma_2 \mathbf{u}_2,$$

where $\sigma_2 = 1 - \lambda_2$

This is more stable!

$D_1^{-\frac{1}{2}} \mathbf{u}_2$ partitions terms and $D_2^{-\frac{1}{2}} \mathbf{v}_2$ partitions documents!

Advantages and disadvantages of SC

- Does not make assumptions on the form of the clusters (k-means can recognize only convex sets);
- Possible to implement efficiently for large data sets as long as the similarity graph is sparse;
- There are no issues of getting stuck in local minima or restarting the algorithm for several times with different initializations;
- Only two singular vectors need to be calculated.

But

- The number of clusters has to be predefined;
- The choice of similarity function and its parameters can affect the results of clustering a lot;
- Cannot serve as a "black box algorithm" which automatically detects the correct clusters in any given set.

The coupling matrix

In [1] the authors introduce a spectral clustering algorithm for calculating the number of metastable states of Markov chains based on a concept of block diagonal dominance.

Definition 1 *Let* $\pi = \{V_1, \cdots, V_k\}$ *be the partition of* V. *Let* $W_{ml} = W(V_m, V_l)$, $m, l \in \{1 \cdots, k\}$, *be blocks of the corresponding block decomposition of stochastic matrix W. The coupling matrix of the decomposition is matrix* B *defined by*

$$B_{ml} = \|W_{ml}\|_{\mathbf{1}} = \frac{1}{|V_m|} \sum_{i \in V_m, j \in V_l} |w_{ij}|.$$

[1] An SVD Approach to Identifying Metastable States of Markov Chains, D. Fritzsche, V. Mehrmann, D. Szyld, E. Virnik

Identification of nearly decoupled bloks - the INDB algorithm

Input: Stochastic matrix W, treshold *thr* $(= 1 - \delta)$.

Output: Number k and sizes n_i, $i = 1, \cdots, k$, of identified blocks in W, a permutation matrix P such that PWP^T is bd-dominant.

1. Compute the second left and right singular vectors of W, $\mathbf{u}_2$ and $\mathbf{v}_2$.
2. Sort it and use the resulting permutation P to permute the matrix W.
3. Identify two potential blocks W_{11} and W_{22} by using the change in sign in $\mathbf{u}_2$ and $\mathbf{v}_2$.
4. The hight of the first (second) block is the number of negative (positive) values in $\mathbf{u}_2$, the width of the first (second) block is the number of negative (positive) values in $\mathbf{v}_2$.
5. **if** *The norm of the diagonal blocks is larger than thr* **then**
6. Separate two found blocks.
7. Proceed recursively with step 1. applied to each of the blocks.
8. **else** The current block cannot be further reduced.

Automatization

If the noise is not too large, a block diagonally dominant structure can be successfully recovered by proposed algorithm.

But, there is still a parametar *thr* to be predetermined. Variation of the idea would be to stop with recursive normalized spectral bipartitioning when the corresponding coupling matrix fails to be diagonally dominant.

Definition 2 *The matrix B is (strictly) diagonally dominant if*

$$|b_{ii}| > \sum_{j, j \neq i} |b_{ij}|$$

for all i.

Where to stop?

In examples with very small between-cluster similarity we needed even stronger criteria - we stopped when the matrix failed to satisfy

$$|b_{ii}| > \sum_{\substack{j,k \\ j \neq k}} |b_{jk}|.$$

We also observed diagonal dominance of the scaled coupling matrices

$$B_r = D^{-1}B,$$

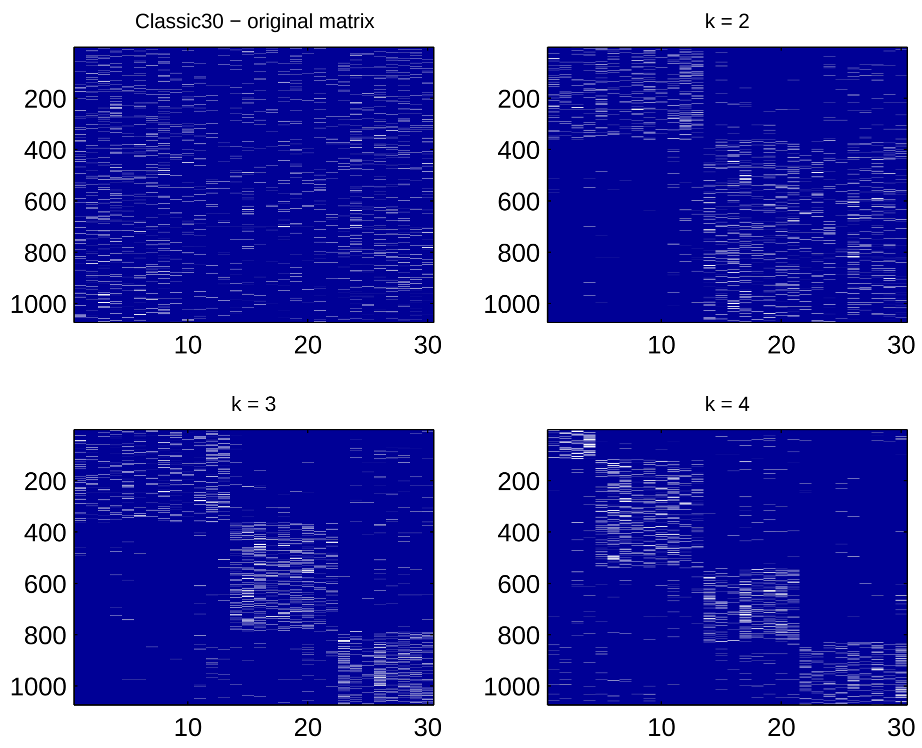
$$B_c = BD^{-1},$$

$$B_a = D^{-1/2}BD^{-1/2},$$

where $D = \operatorname{diag} B$.

Example - full matrices

The *Classic** collection of abstracts (Medline - 1033 medical abstracts, Cranfield - 1400 aeronautical systems abstracts and Cisi - 1460 information retrieval abstracts) is naturally divided in three clusters. The data is clustered by the recursive normalized spectral clustering algorithm for $k = 2$, $k = 3$ and $k = 4$.



Example - full matrices (2)

In the following table we give a number of clusters found by different criteria:

	B		B_{scal} (dds)			B (thr)				
	dd	dds	r	c	a	0.5	0.6	0.7	0.8	0.9
$Classic30$ (1073×30)	18	3	5	5	5	21	21	21	17	2
$Classic150$ (3652×150)	10	3	4	4	4	105	103	73	5	1
$Classic300$ (5577×300)	7	3	3	3	3	200	98	29	1	1
$Random4$ (770×795)	7	4	4	4	4	6	4	3	3	1
$Random8$ (930×880)	10	8	8	8	8	8	6	6	5	2
$Mat1$ (3213×146)	6	1	1	3	2	2	1	1	1	1

dd - diagonal dominance

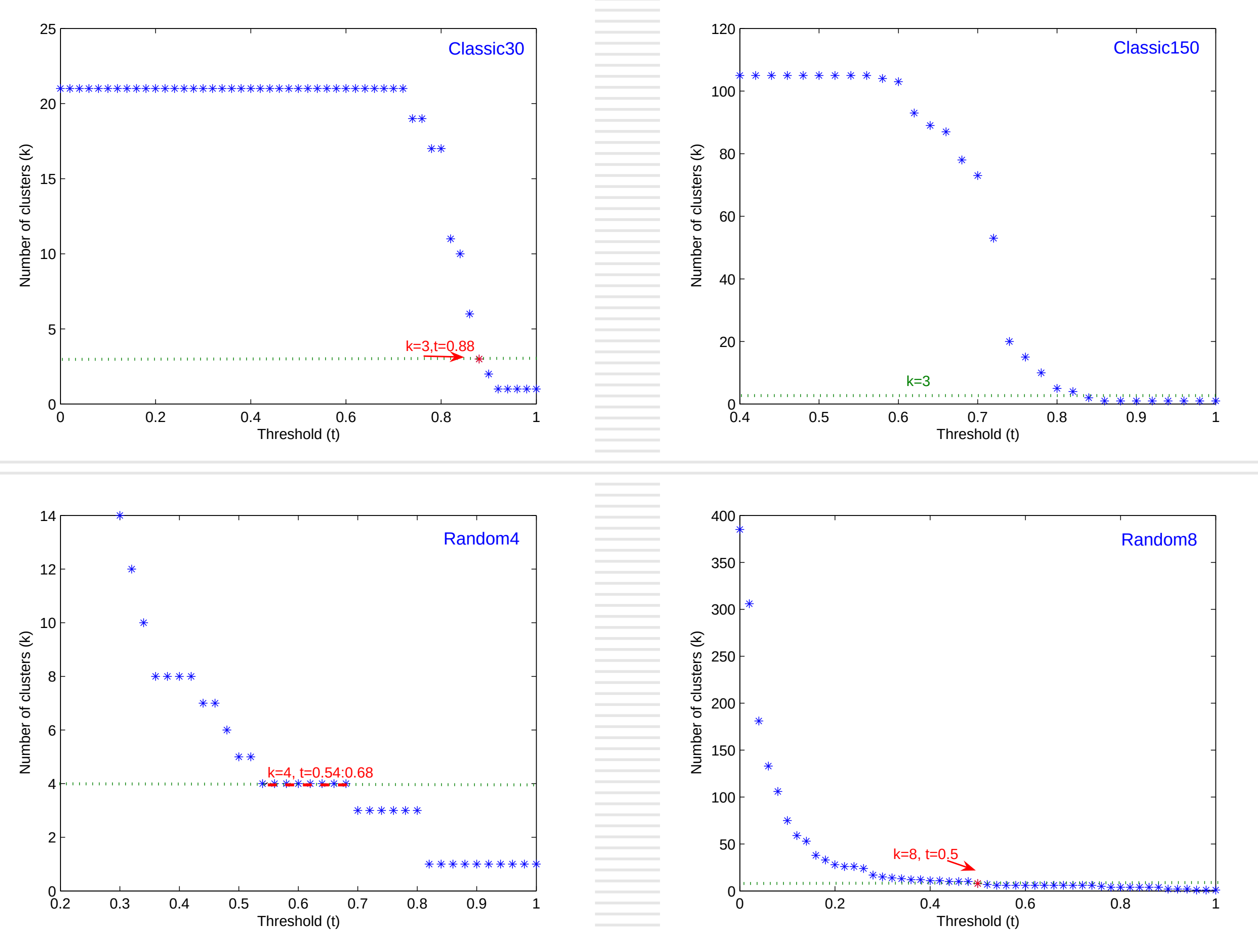
dds - diagonal dominance (stronger criteria)

B_{scal} - scaled matrix (r - row, c - column, a - all)

thr - threshold as a measure of diagonal dominance (INDB algorithm)

Example - full matrices(3)

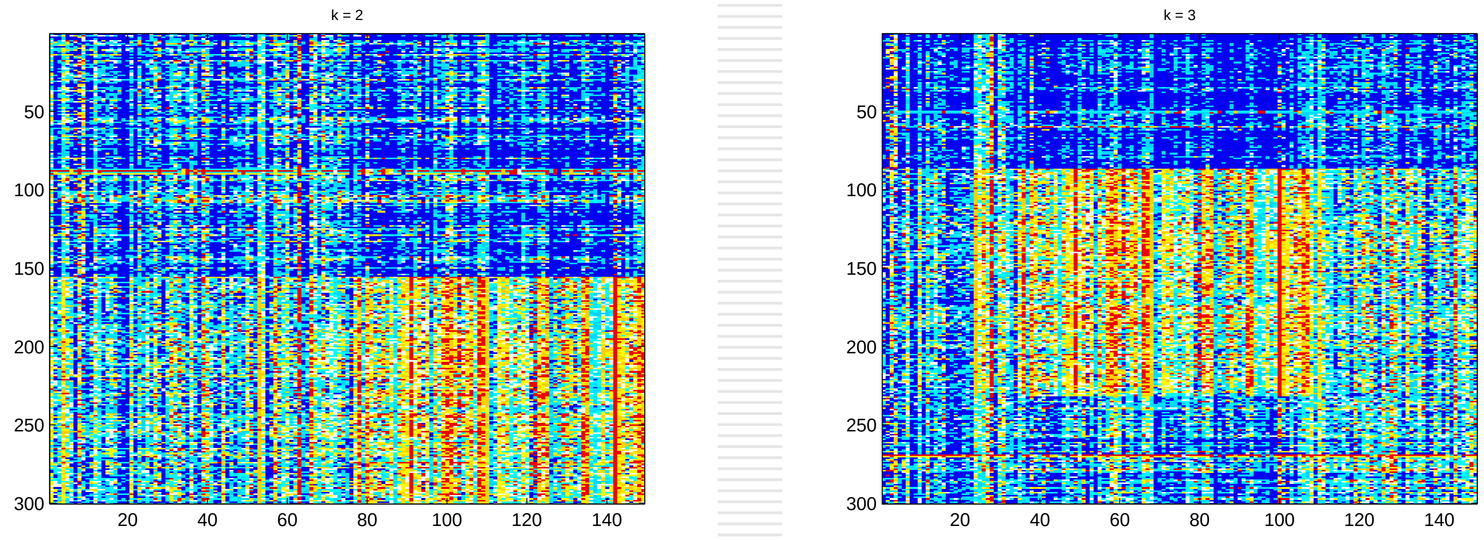
Number of clusters as a result of different thresholds.



Example Imaginable

An example* of 300 words graded by 149 people on how imaginable these words are. Grades are scaled from 1 to 7.

The data is clustered by the recursive normalized spectral clustering algorithm for $k = 2$ and $k = 3$.



* The data was provided by Fekete Istvan, BME, Department of Cognitive Science

Example Imaginable (2)

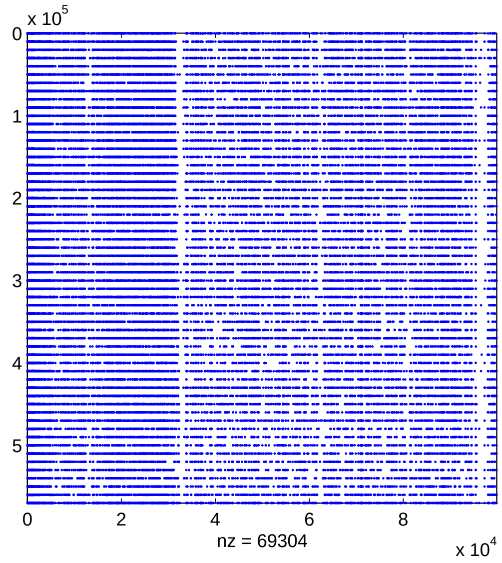
The result of 3-clustering (representative words):

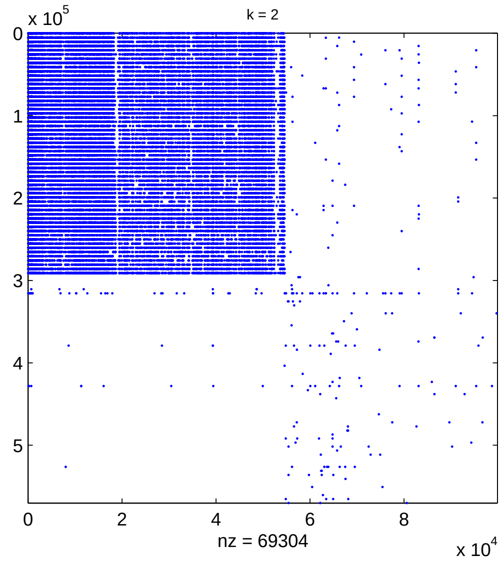
cluster 1 (146)	**cluster 2** (69)	**cluster 3** (85)
coffee	*dance*	*faith*
apple	*dream*	*fate*
table	*cinema*	*love*
dog	*clothes*	*hope*
computer	*congressman*	*freedom*
bird	*coastline*	*time*
honey	*night*	*responsibility*
tea	*decrease*	*beauty*
book	*proposal*	*friendship*
bed	*line*	*culture*
⋮	⋮	⋮

	B		B_{scal} (dds)			B (thr)			
	dd	dds	r	c	a	0.1	0.2	0.3	0.6
$Imaginable(300 \times 149)$	2	1	1	1	1	9	5	2	1

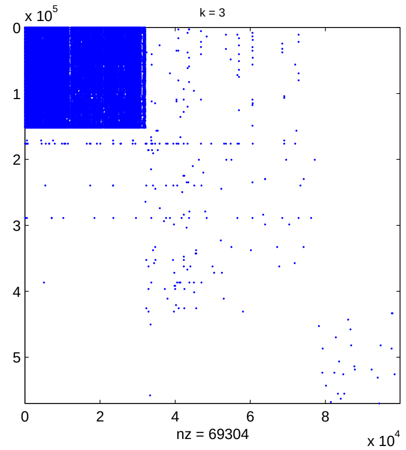
Example - sparse matrix

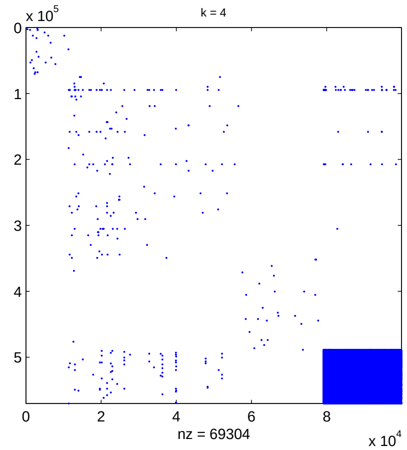
Matrix (570166×99899) represents big store receipts registered during one month.





Example - sparse matrix (2)





	B		$B_{scal}\ (dds)$			$B(thr)$	
	dd	dds	r	c	a	0.01	0.005
(570166×99899)	> 8	3	> 8	> 8	> 8	1	1

Conclusion

- Clustering according to normalized singular vectors of normalized similarity matrix outperforms the unnormalized version of spectral clustering.
- It can be hard to predict the treshold value in INDB algorithm that will result with good number of clusters.
- Diagonal dominance as a stoping criteria in recursive normalised spectral clustering works if the perturbation is not too large.
- In examples with very small between-cluster similarity we needed stronger criteria.
- Scaling the coupling matrix showed no effect on determination of the number of clusters.