Graph Clustering and Co-clustering

Inderjit S. Dhillon
University of Texas at Austin

IPAM
Nov 5, 2007

Joint work with A. Banerjee, J. Ghosh, Y. Guan, B. Kulis, S. Merugu & D. Modha

Outline

- Clustering Graphs: Spectral Clustering & A Surprising Equivalence
- Matrix Co-Clustering

Clustering

Partition objects into groups so that

- objects within the same group are **similar** to each other
- objects in different groups are **dissimilar** to each other

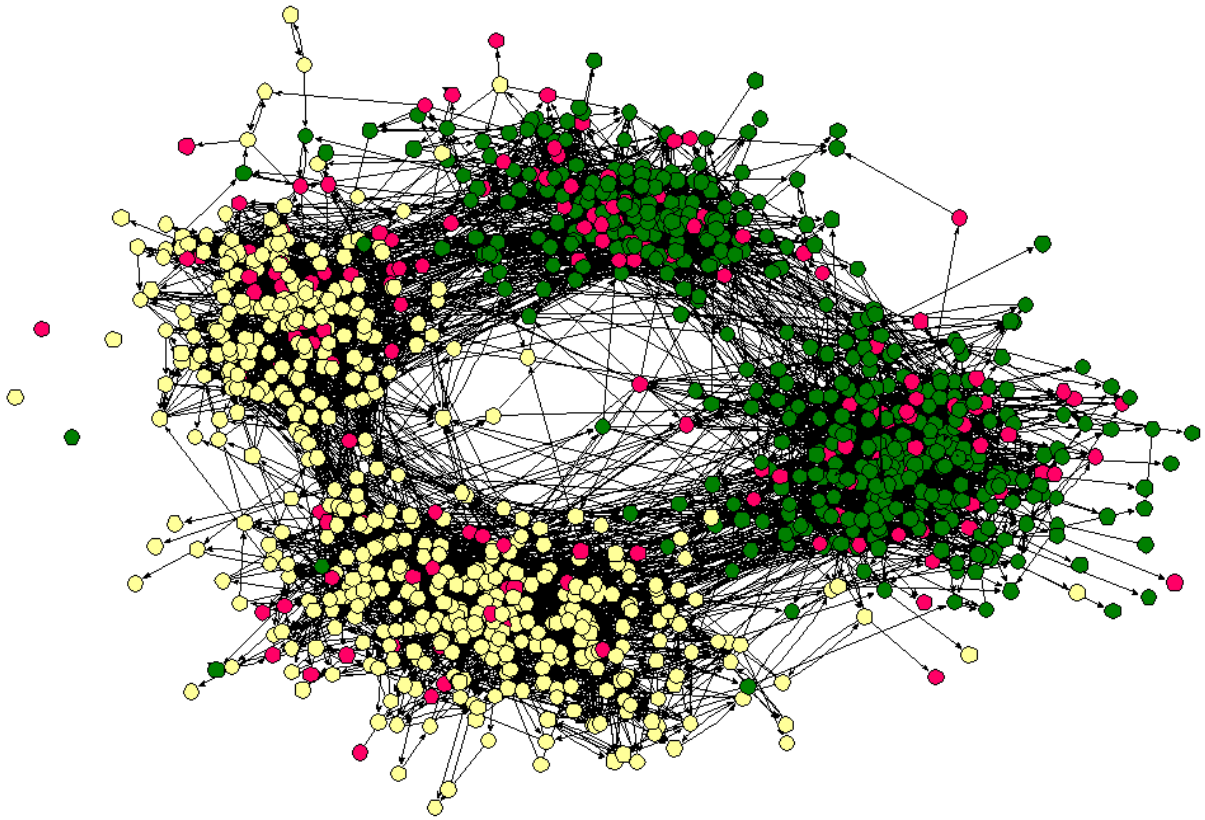
Examples:

- Bioinformatics: Identifying similar genes
- Text Mining: Organizing document collections
- Image/Audio Analysis: Image and Speech segmentation
- Web Search: Clustering web search results
- Social Network Analysis: Identifying social groups

Clustering Graphs

Graph Partitioning/Clustering

- In many applications, goal is to partition/cluster nodes of a graph:

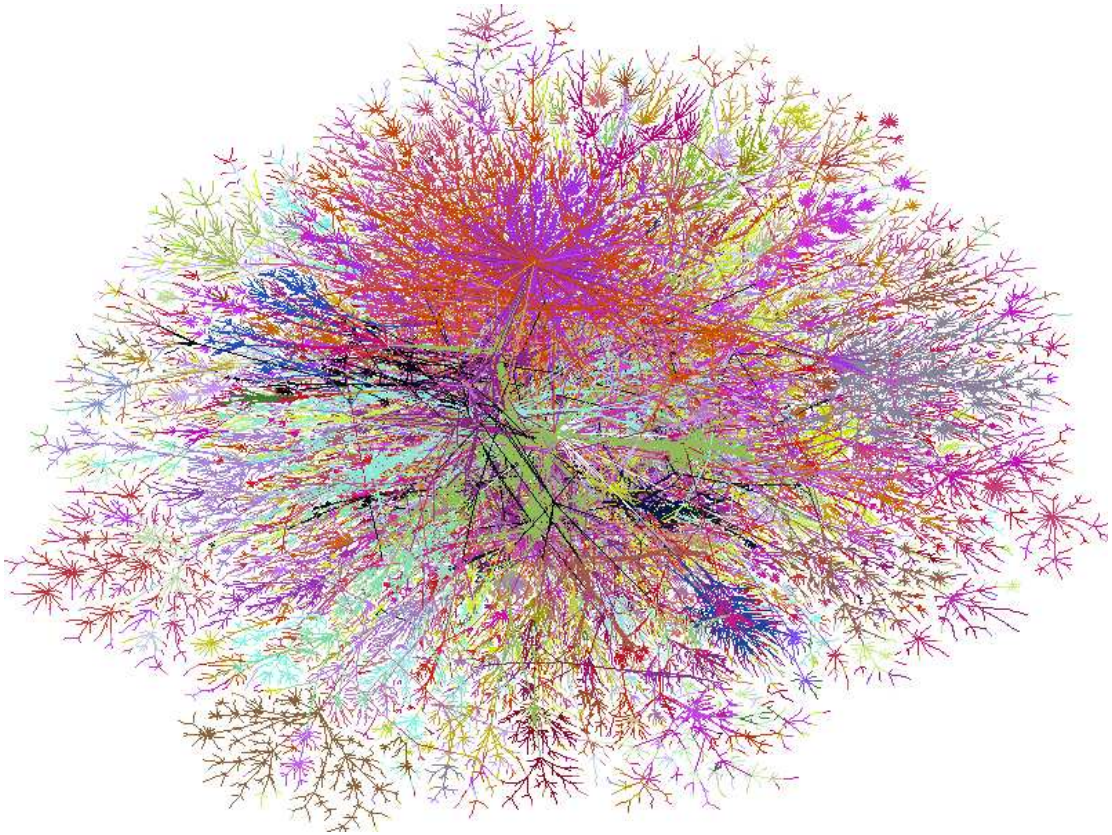


High School Friendship Network

[James Moody. American Journal of Sociology, 2001]

Graph Partitioning/Clustering

- In many applications, goal is to partition/cluster nodes of a graph:



The Internet

[The Internet Mapping Project, Hal Burch and Bill Cheswick, Lumeta Corp, 1999]

Graph Clustering Objectives

- How do we measure the quality of a graph clustering?
- Could simply minimize the *edge-cut* in the graph
  - Can lead to clusters that are highly unbalanced in size
- Could minimize the *edge-cut* in the graph while constraining the clusters to be equal in size
  - Not a natural restriction in data analysis

- Popular objectives include normalized cut, ratio cut and ratio association

$$\text{Normalized Cut:} \qquad \text{minimize} \quad \sum_{i=1}^{c} \frac{\text{links}(\mathcal{V}_i, \mathcal{V} \setminus \mathcal{V}_i)}{\text{degree}(\mathcal{V}_i)}$$

$$\text{Ratio Cut:} \qquad \text{minimize} \quad \sum_{i=1}^{c} \frac{\text{links}(\mathcal{V}_i, \mathcal{V} \setminus \mathcal{V}_i)}{|\mathcal{V}_i|}$$

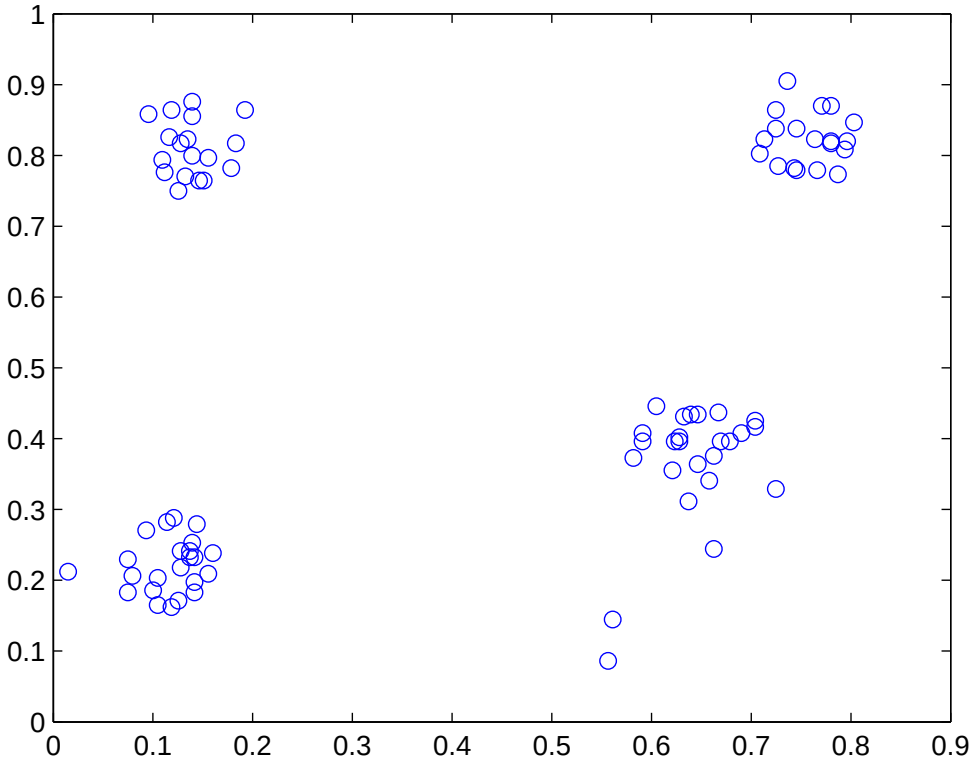
[Shi & Malik, IEEE Pattern Analysis & Machine Intelligence, 2000]

[Chan, Schlag & Zien, IEEE Integrated Circuits & Systems, 1994]

Spectral Clustering

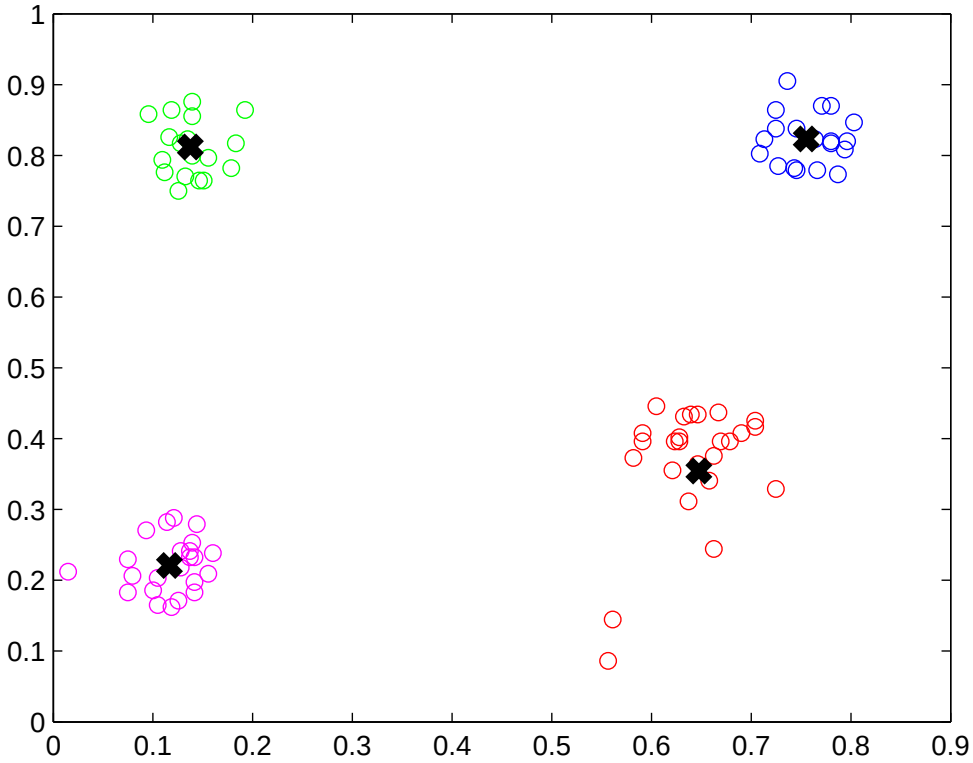
- Take a real relaxation of the clustering objective
- Globally optimal solution of the relaxed problem is given by eigenvectors
  - For ratio cut: compute smallest eigenvectors of the Laplacian $L = D - A$
  - For normalized cut: compute smallest eigenvectors of the normalized Laplacian $I - D^{-1/2}AD^{-1/2}$
  - Post-process eigenvectors to obtain a discrete clustering
- Problem: Can be expensive if many eigenvectors of a very large graph are to be computed

K-Means Clustering



Goal: partition points into k clusters

K-Means Clustering



Minimizes squared Euclidean distance from points to their cluster centroids

The k-means Algorithm

- Given a set of vectors and an initial clustering, alternate between computing cluster means and assigning points to the closest mean

1. Initialize clusters π_c and cluster means $\mathbf{m}_c$ for all clusters c.
2. For every vector $\mathbf{a}_i$ and all clusters c, compute

$$d(\mathbf{a}_i, c) = \|\mathbf{a}_i - \mathbf{m}_c\|^2$$

and

$$c^*(\mathbf{a}_i) = \text{argmin}_c \;\; d(\mathbf{a}_i, c)$$

3. Update clusters: $\pi_c = \{\mathbf{a} \;:\; c^*(\mathbf{a}_i) = c\}$.
4. Update means: $\mathbf{m}_c = \frac{1}{|\pi_c|} \sum_{\mathbf{a}_i \in \pi_c} \mathbf{a}_i$
5. If not converged, go to Step 2. Otherwise, output final clustering.

From k-means to Weighted Kernel k-means

- Introduce weights w_i for each point $\mathbf{a}_i$: use the weighted mean instead
- Expanding the distance computation yields:

$$\|\mathbf{a}_i - \mathbf{m}_c\|^2 = \mathbf{a}_i \cdot \mathbf{a}_i - \frac{2\sum_{\mathbf{a}_j \in \pi_c} w_j \mathbf{a}_i \cdot \mathbf{a}_j}{\sum_{\mathbf{a}_i \in \pi_c} w_j} + \frac{\sum_{\mathbf{a}_i, \mathbf{a}_j \in \pi_c} w_j w_l \mathbf{a}_j \cdot \mathbf{a}_l}{(\sum_{\mathbf{a}_j \in \pi_c} w_j)^2}$$

- Computation can be done only using inner products of data points
- Given a *kernel* matrix K that gives inner products in feature space, can compute distances using the above formula
- Objective function for weighted kernel k-means:

$$\text{Minimize } \mathcal{D}(\{\pi_{c=1}^k\}) = \sum_{c=1}^{k} \sum_{\mathbf{a}_i \in \pi_c} w_i \|\varphi(\mathbf{a}_i) - \mathbf{m}_c\|^2$$

$$\text{where } \mathbf{m}_c = \frac{\sum_{\mathbf{a}_i \in \pi_c} w_i \varphi(\mathbf{a}_i)}{\sum_{\mathbf{a}_i \in \pi_c} w_i}$$

The Weighted Kernel k-means Algorithm

- Given a kernel matrix (positive semi-definite similarity matrix), run k-means in the feature space

1. Initialize clusters π_c
2. For every vector $\mathbf{a}_i$ and all clusters c, compute

$$d(\mathbf{a}_i, c) = K_{ii} - \frac{2\sum_{\mathbf{a}_j \in \pi_c} w_j K_{ij}}{\sum_{\mathbf{a}_j \in \pi_c} w_j} + \frac{\sum_{\mathbf{a}_j, \mathbf{a}_l \in \pi_c} w_j w_l K_{jl}}{(\sum_{\mathbf{a}_j \in \pi_c} w_j)^2}$$

   and

$$c^*(\mathbf{a}_i) = \text{argmin}_c \; d(\mathbf{a}_i, c)$$

3. Update clusters: $\pi_c = \{\mathbf{a} \; : \; c^*(\mathbf{a}_i) = c\}$.
4. If not converged, go to Step 2. Otherwise, output final clustering.

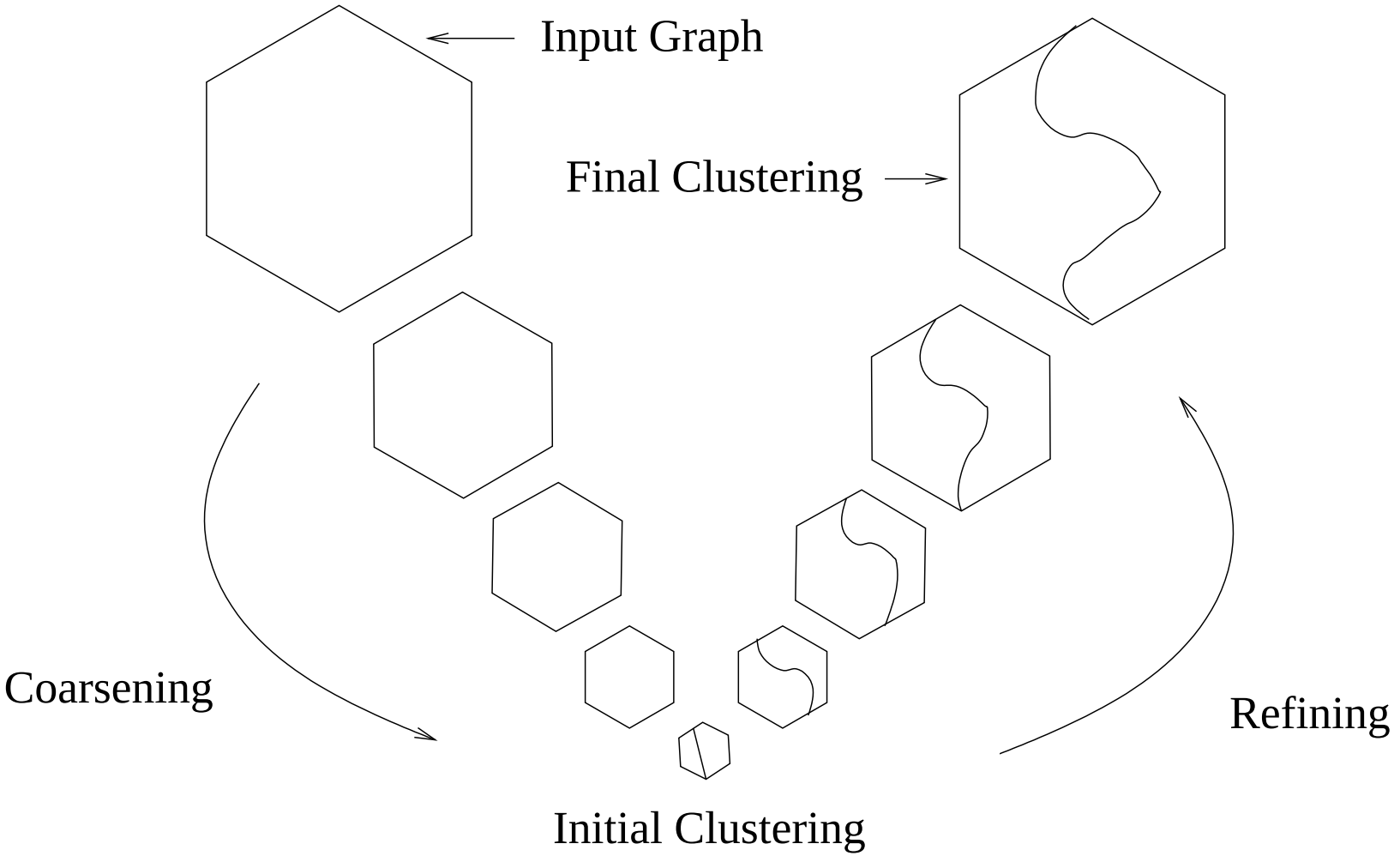
Equivalence to Graph Clustering

- "Surprising" Theoretical Equivalence:
  - Weighted graph clustering objective is *mathematically identical* to the weighted kernel k-means objective
- Follows by rewriting both objectives as trace maximization problems
- Popular graph clustering objectives and corresponding weights and kernels for weighted kernel k-means given affinity matrix A:

Objective	Node Weight	Kernel
Ratio Association	1 for each node	$K = \sigma I + A$
Ratio Cut	1 for each node	$K = \sigma I - L$
Kernighan-Lin	1 for each node	$K = \sigma I - L$
Normalized Cut	Degree of the node	$K = \sigma D^{-1} + D^{-1} A D^{-1}$

- Implication: Can minimize graph cuts such as normalized cut and ratio cut without any eigenvector computation.

The Multilevel Approach



[CHACO, Hendrickson & Leland, 1994]

[METIS, Karypis & Kumar, 1999]

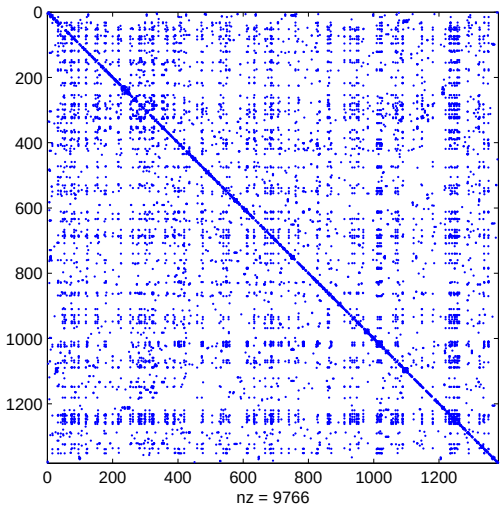
The Multilevel Approach

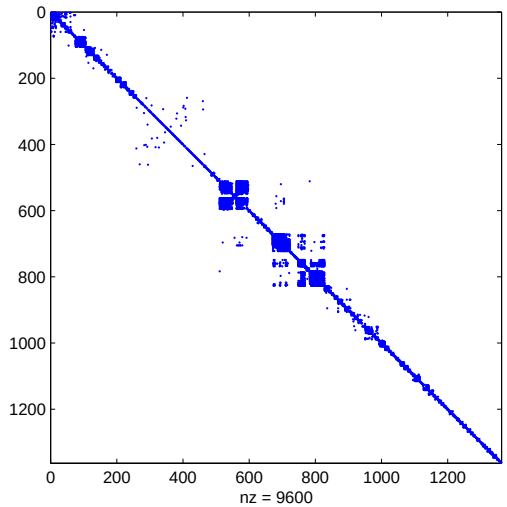
- Phase I: Coarsening
  - Coarsen the graph by merging nodes together to form smaller and smaller graphs
  - Use a simple greedy heuristic specialized to each graph cut objective function
- Phase II: Base Clustering
  - Once the graph is small enough, perform a base clustering
  - Variety of techniques possible for this step
- Phase III: Refining
  - Uncoarsen the graph, level by level
  - Use weighted kernel k-means to refine the clusterings at each level
  - Input clustering to weighted kernel k-means is the clustering from the previous level

Experiments: gene network

Mycobacterium tuberculosis gene network: 1381 genes & 9766 functional linkages.

- Spy plots of the functional linkage matrix before and after clustering (128 clusters)—each dot indicates a non-zero entry





Experiments: gene network

Mycobacterium tuberculosis gene network: 1381 genes & 9766 functional linkages.

- Normalized cut values generated by Graclus and the spectral method

# clusters	4	8	16	32	64	128
Graclus	0	.009	.018	.53824	3.1013	18.735
Spectral	0	.036556	.1259	.92395	5.3647	25.463

Experiments: IMDB movie data set

IMDB data set contains 1.4 million nodes and 4.3 million edges.

- We generate 5000 clusters using Graclus, which takes 12 minutes.
- If we use the spectral method, we would have to store 5000 eigenvectors of length 1.4M; that is 24 GB main memory.
- An example cluster: Harry Potter

Movies	Actors
Harry Potter and the Sorcerer's Stone	Daniel Radcliffe, Rupert Grint,
Harry Potter and the Chamber of Secrets	Emma Watson, Peter Best,
Harry Potter and the Prisoner of Azkaban	Joshua Herdman, Harry Melling,
Harry Potter and the Goblet of Fire	Robert Pattinson, James Phelps,
Harry Potter and the Order of the Phoenix	Tom Felton, Devon Murray,
Harry Potter: Behind the Magic	Jamie Waylett, Shefali Chowdhury,
Harry Potter und die Kammer des Schreckens:	Stanislav Ianevski, Jamie Yeates,
Das grobe RTL Special zum Film	Bonnie Wright, Alfred Enoch, Scott Fern,
J.K. Rowling: Harry Potter and Me	Chris Rankin, Matthew Lewis, Katie Leung
	Sean Biggerstaff, Oliver Phelps

Experiments: IMDB movie data set

IMDB data set contains 1.4 million nodes and 4.3 million edges.

- Normalized cut values and computation time for a varied number of clusters, using Graclus and the spectral method

Normalized cut values—lower cut values are better

# clusters	2	4	8	16	32	64	128	256
Graclus	.049	.163	.456	1.39	3.72	9.42	24.13	64.04
Spectral	.00	.016	.775	2.34	5.65	-	-	-

Computation time (in seconds)

Graclus	34.57	37.3	37.96	46.61	49.93	53.95	64.83	81.42
Spectral	261.32	521.69	597.23	1678.05	5817.96	-	-	-

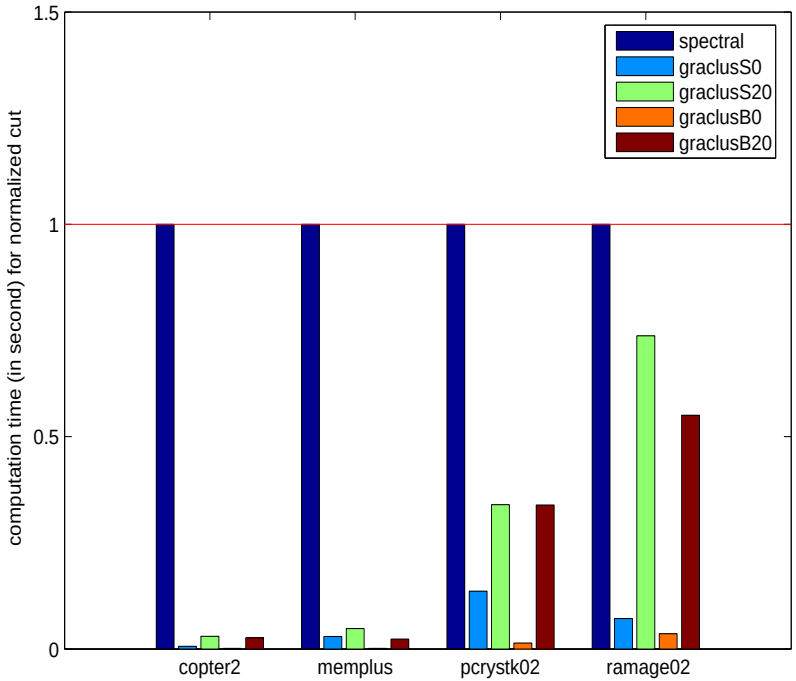
Experiments: Benchmark graph clustering

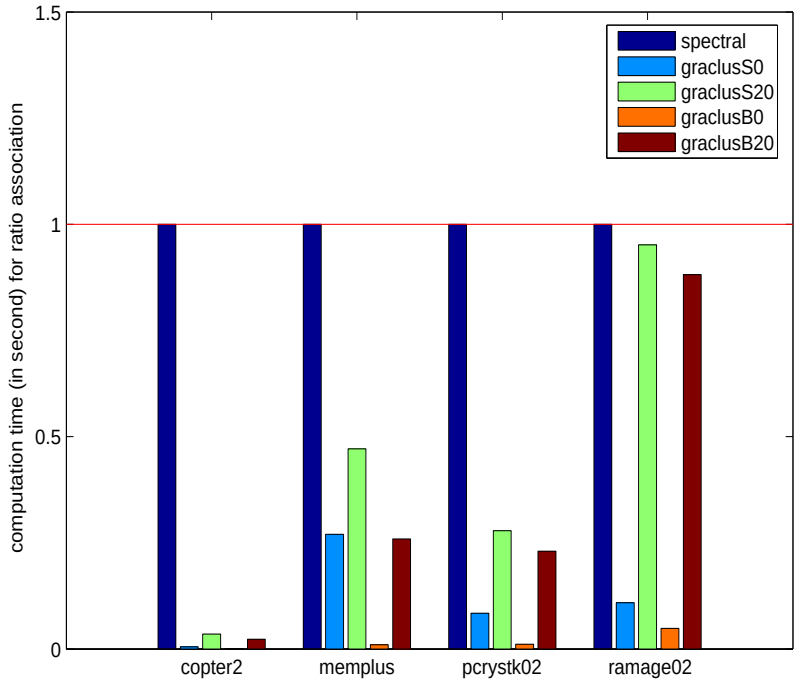
Test graphs:

Graph name	No. of nodes	No. of edges	Application
copter2	55476	352238	Helicopter mesh
memplus	17758	54196	Memory circuit
pcrystk02	13965	477309	Structural engineering
ramage02	16830	1424761	Navier Stokes equations

Experiments: Benchmark graph clustering

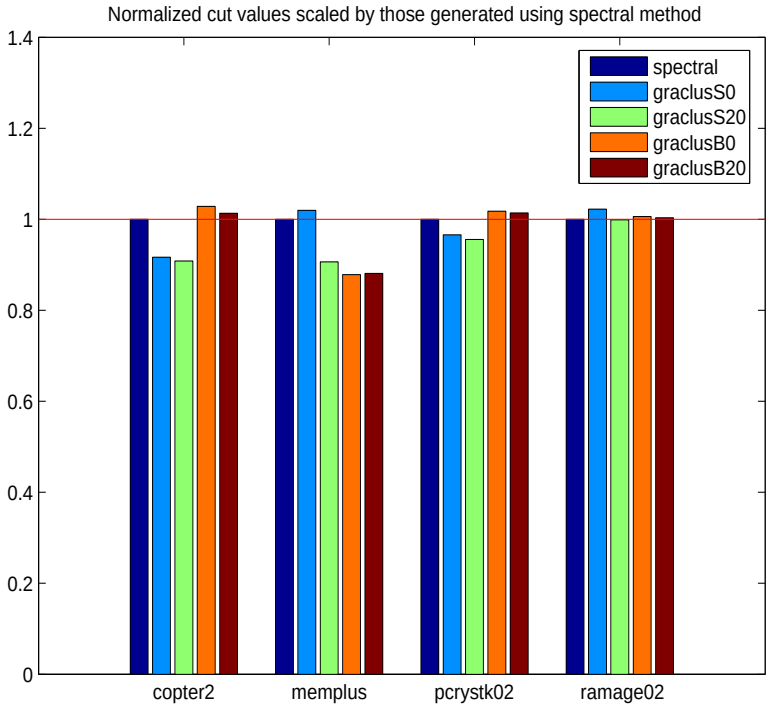
- Computation time:

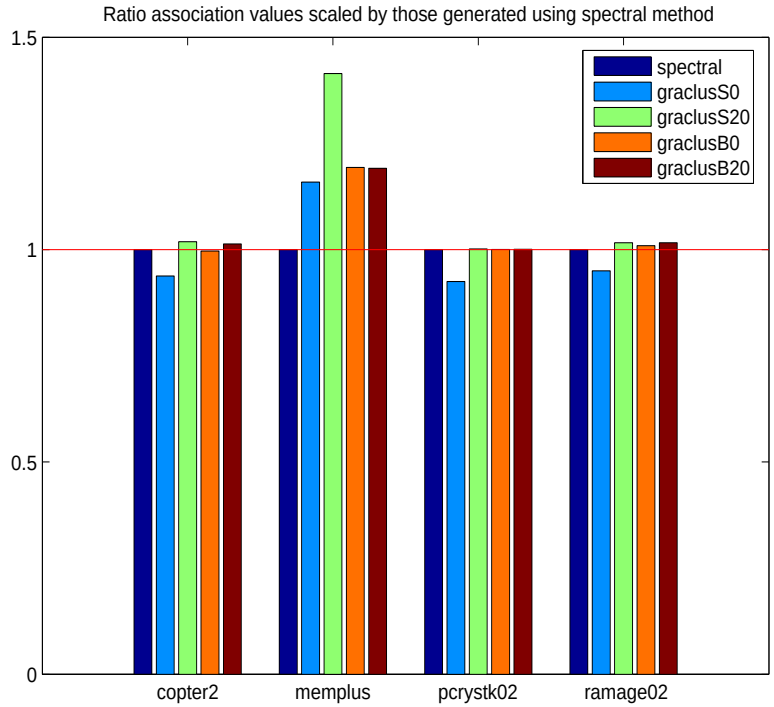




Experiments: Benchmark graph clustering

- Quality (normalized cut and ratio association):





Co-clustering

- Given a data matrix, partition the rows as well as columns

Original Matrix

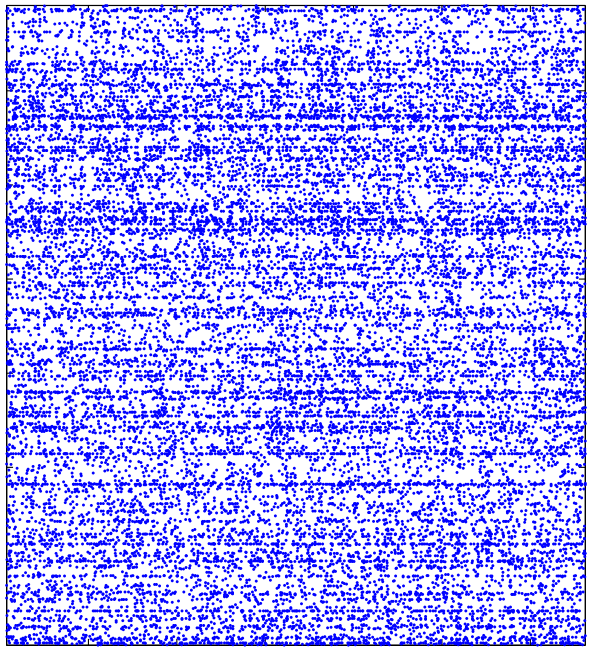
z	x	z	–	–	x
+	o	+	*	*	o
z	x	z	–	–	x
+	o	+	*	*	o
+	o	+	*	*	o

After co-clustering and permutation

x	x	–	–	z	z
x	x	–	–	z	z
o	o	*	*	+	+
o	o	*	*	+	+
o	o	*	*	+	+

Text Mining

- Sample term-document matrix:

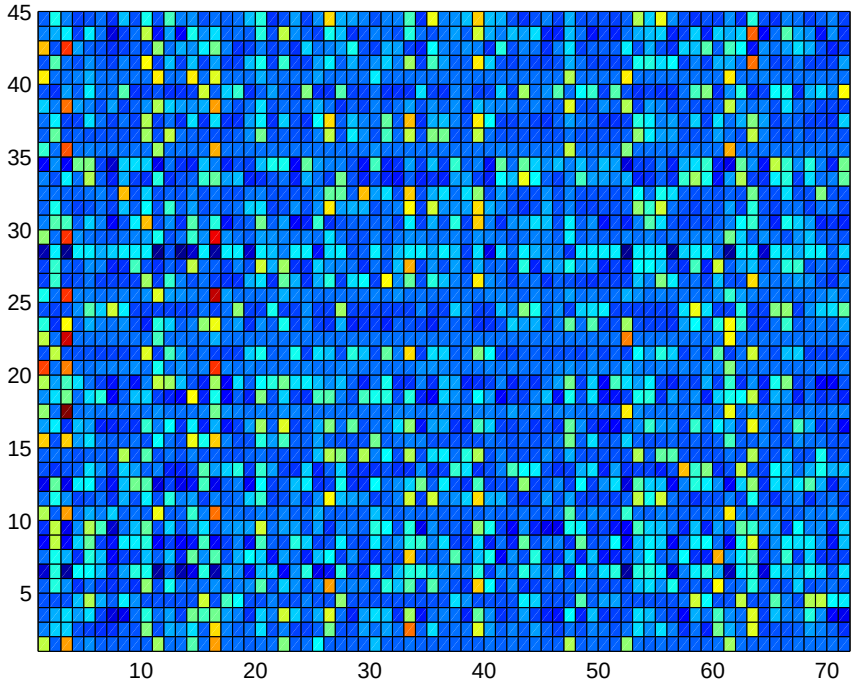


nz = 176347

- Matrix Characteristics:
  - Large
  - Sparse
  - Nonnegative

Bioinformatics

- Sample gene-condition matrix:



- Matrix Characteristics:
  - Smaller
  - Dense
  - Negative as well as positive entries

Co-clustering Applications

- Bioinformatics: co-cluster genes and conditions

  [Kluger, Basri, Chang & Gerstein, Genome Biology, 2003], [Cho, Dhillon, Guan & Sra, SIAM DM 2004]

- Text Mining: co-cluster terms and documents (and categories)

  [Dhillon, Mallela & Modha, KDD 2003], [Gao, Liu, Zheng, Cheng & Ma, KDD 2005], [Takamura & Matsumoto, Information Processing Society of Japan (IPSJ) Journal, 2003]

- Natural Language Processing: co-cluster terms & their contexts for Named Entity Recognition

  [Rohwer & Freitag, HLT-NAACL 2004], [Freitag, ACL 2004]

- Image Analysis: co-cluster images and features

  [Qiu, ICPR 2004], [Guan, Qiu & Xue, IEEE Multimedia Signal Processing, 2005]

- Video Content Analysis: co-cluster video segments & prototype images, co-cluster auditory scenes & key audio effects for scene categorization

  [Zhong, Shi & Visontai, IEEE CVPR 2004], [Cai, Lu, & Cai, IEEE Acoustics, Speech and Signal Processing (ICASSP05)]

- Miscellaneous: co-cluster advertisers and keywords

  [Carrasco, Fain, Lang & Zhukov, ICDM, 2003]

Co-Clustering & Matrix Approximation

- Let $A = [\mathbf{a}_1, \ldots, \mathbf{a}_n]$ be an $m \times n$ data matrix
- Goal: partition A into k row clusters and ℓ column clusters
- How do we judge the quality of co-clustering?

Co-Clustering & Matrix Approximation

- Let $A = [\mathbf{a}_1, \ldots, \mathbf{a}_n]$ be an $m \times n$ data matrix
- Goal: partition A into k row clusters and ℓ column clusters
- How do we judge the quality of co-clustering?
- Use quality of "associated" matrix approximation
  - Associate matrix approximation using the Minimum Bregman Information (MBI) principle
- Objective: Find optimal co-clustering $\leftrightarrow$ optimal MBI approximation

Minimum Bregman Information

- Matrix Approximation from a co-clustering:

Minimum Bregman Information

- Matrix Approximation from a co-clustering:

  Alice

Minimum Bregman Information

- Matrix Approximation from a co-clustering:

  Alice

  Knows input matrix **A**

Minimum Bregman Information

- Matrix Approximation from a co-clustering:

Alice

Bob

Knows input matrix **A**

Minimum Bregman Information

- Matrix Approximation from a co-clustering:

Alice	Bob
Knows input matrix **A**	Does not know **A**

Minimum Bregman Information

- Matrix Approximation from a co-clustering:

Alice	Bob
Knows input matrix **A**	Does not know **A**
Determines a co-clustering	

Minimum Bregman Information

- Matrix Approximation from a co-clustering:

Alice $\xrightarrow[\text{\& summary statistics}]{\text{Transmits co-clustering}}$ Bob

Knows input matrix **A**

Does not know **A**

Determines a co-clustering

Minimum Bregman Information

- Matrix Approximation from a co-clustering:

Alice $\xrightarrow[\text{\& summary statistics}]{\text{Transmits co-clustering}}$ Bob

Knows input matrix $\mathbf{A}$

Does not know $\mathbf{A}$

Determines a co-clustering

Reconstructs an approximation $\hat{\mathbf{A}}$ given co-clustering & summary statistics

Minimum Bregman Information

- Matrix Approximation from a co-clustering:

Alice $\xrightarrow[\text{\& summary statistics}]{\text{Transmits co-clustering}}$ Bob

Knows input matrix $\mathbf{A}$

Does not know $\mathbf{A}$

Determines a co-clustering

Reconstructs an approximation $\hat{\mathbf{A}}$ given co-clustering & summary statistics

- **Key Idea:** Bob will reconstruct $\hat{\mathbf{A}}$ using the Minimum Bregman Information principle:

$$\hat{\mathbf{A}} = \underset{\substack{\mathbf{X} \text{ satisfies} \\ \text{summary statistics}}}{\operatorname{argmin}} \sum_{i=1}^{m}\sum_{j=1}^{n} D_\varphi(X_{ij}, \mu_{\mathbf{A}})$$

  - generalizes the maximum entropy approach

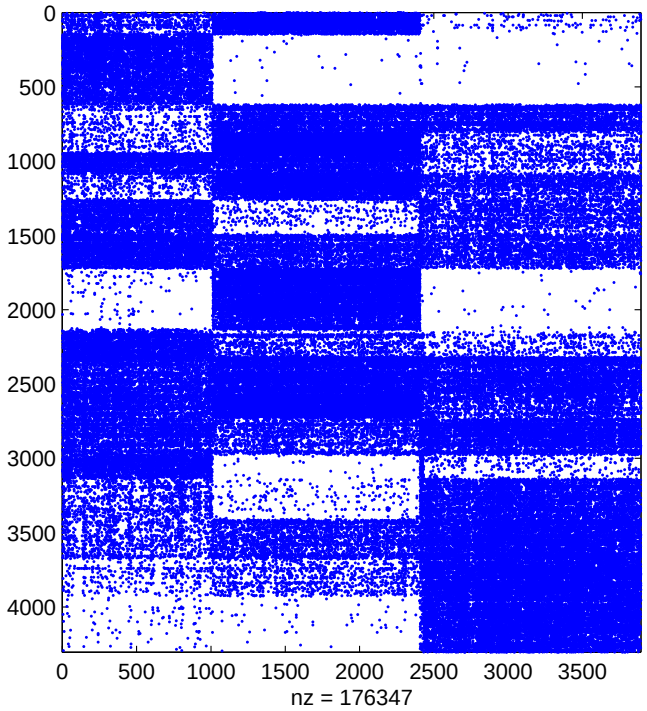
Results — Document Clustering

- Document data set with 3 known clusters
- Co-clustering with Relative Entropy
  - superior performance as compared to just column clustering
  - performs implicit dimensionality reduction at each iteration

(3 doc;20 word)			(3 doc;500 word)			(3 doc;2500 word)		
1389	1	2	1364	3	18	920	49	292
9	1455	33	5	1446	21	31	1239	404
0	4	998	29	11	994	447	172	337

Confusion matrices for a document data set with different number of word clusters

Co-clustered term-document matrix

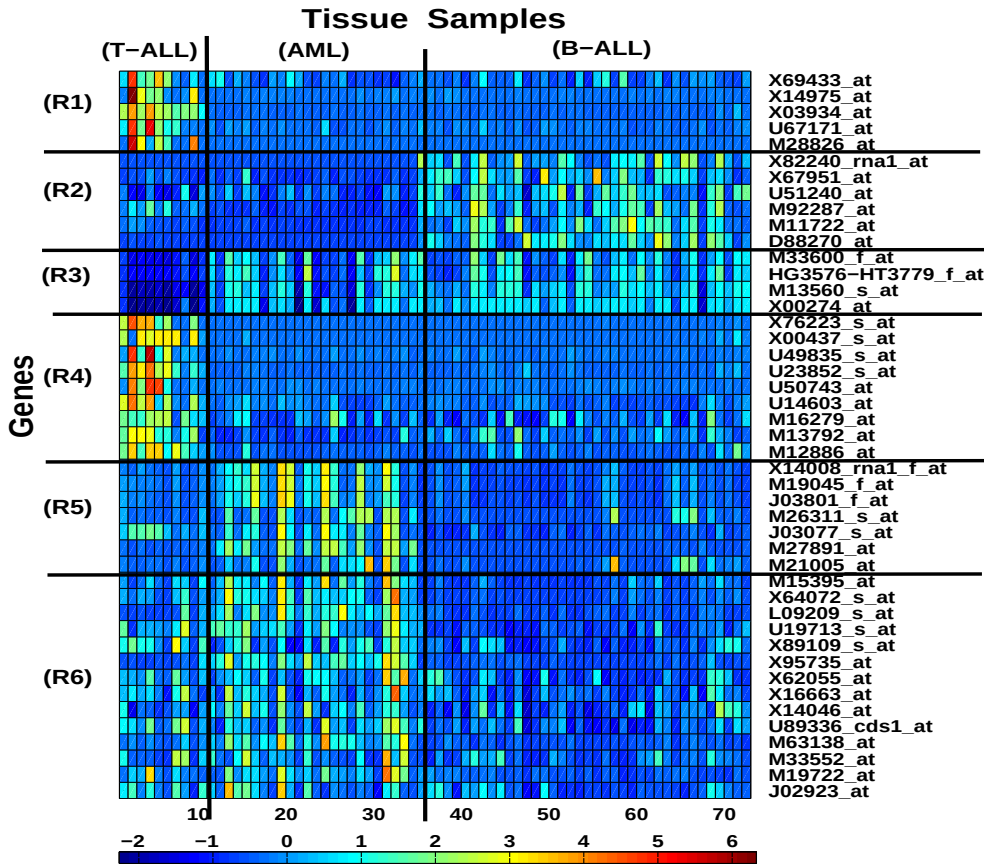


Results — Bioinformatics

- Gene Expression Leukemia data
- Matrix contains expression levels of genes in different tissue samples

Results — Bioinformatics

- Gene Expression Leukemia data
- Matrix contains expression levels of genes in different tissue samples
- Co-clustering recovers cancer samples & functionally related genes



Conclusions

- A mathematical equivalence between spectral graph clustering objectives and weighted kernel k-means objectives
- Superfast multilevel algorithm uses kernel k-means in its refinement phase
- Rich co-clustering framework/formulation/algorithm
- Co-clustering is becoming a technology

References

- **Graph Clustering Software:** "Graclus" available at http://www.cs.utexas.edu/users/dml/Software/graclus.html
- **Graph Clustering Paper:** I. S. Dhillon, Y. Guan, and B. Kulis, "Weighted Graph Cuts without Eigenvectors: A Multilevel Approach", *IEEE Transactions on Pattern Analysis and Machine Intelligence(PAMI)*, vol. 29:11, pages 1944–1957, November 2007.
- **Co-clustering Paper:** A. Banerjee, I. S. Dhillon, J. Ghosh, S. Merugu and D. S. Modha, "A Generalized Maximum Entropy Approach to Bregman Co-Clustering and Matrix Approximations", *Journal of Machine Learning Research(JMLR)*, vol. 8, pages 1919–1986, August 2007.