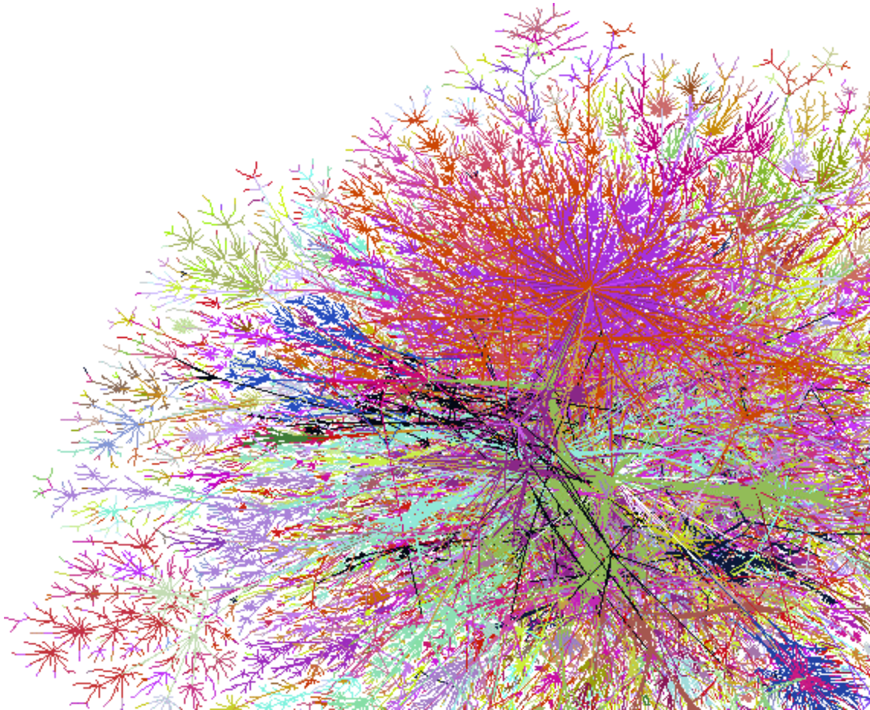
Learning on Networks

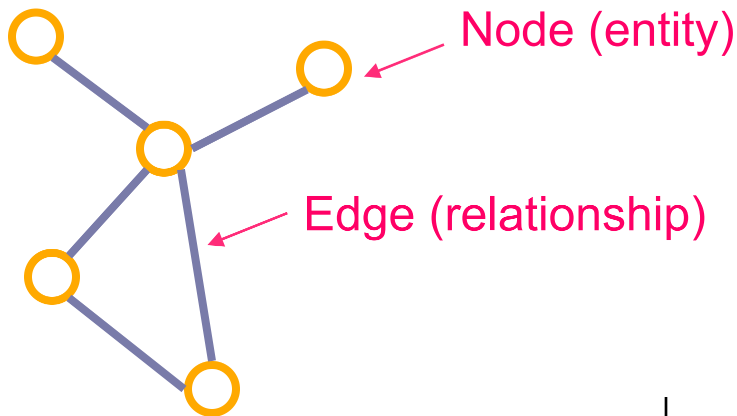


Some materials adapted from
Lada Adamic (UMichigan)

What are networks?

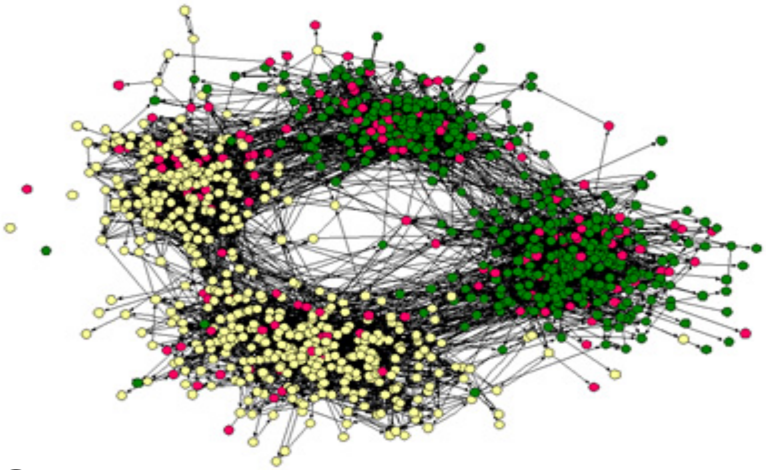
- Networks are collections of entities joined by relationships

"Network" ≡ "Graph"

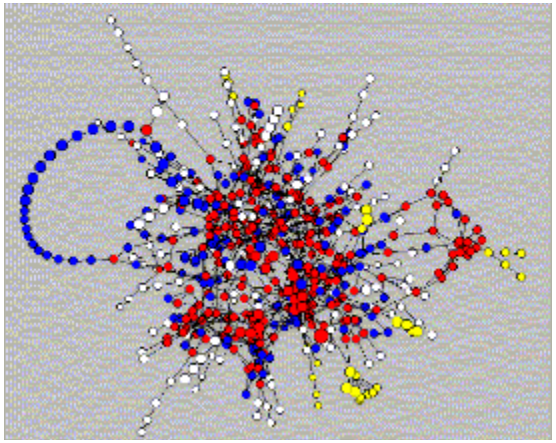


points	lines	
vertices	edges, arcs	math
nodes	links	computer science
sites	bonds	physics
actors	ties, relations	sociology

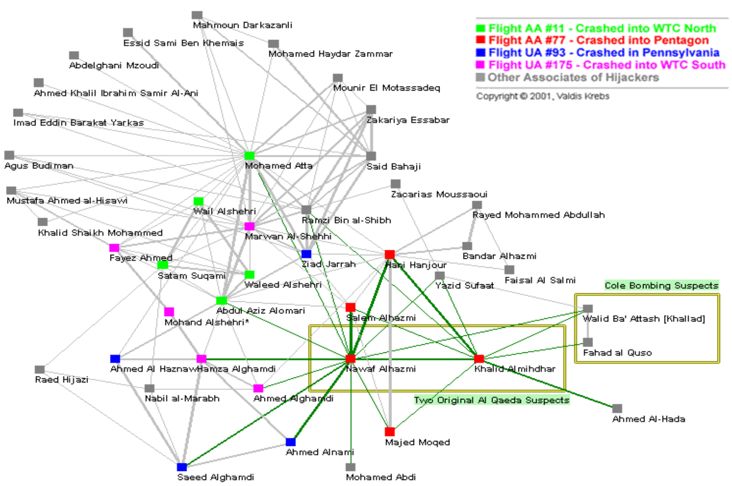
Example Relational Networks



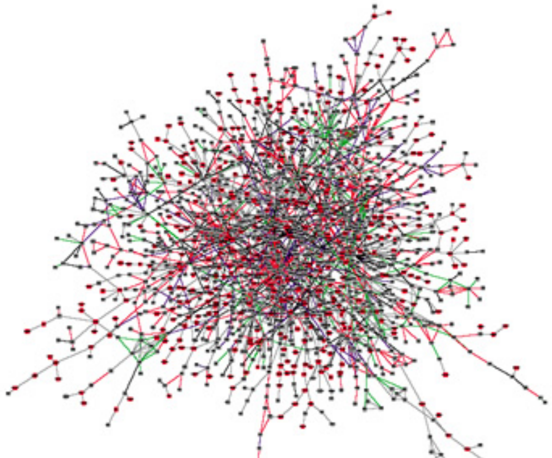
School Friendship Network
(from Moody 2001)



Yeast Metabolic Network
(from https://www.nd.edu/~networks/cell/)

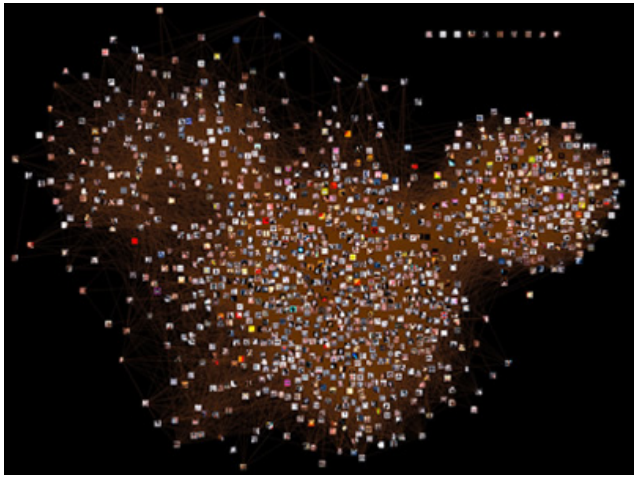


Terrorist Network
(by Valdis Krebs, Orgnet.com)

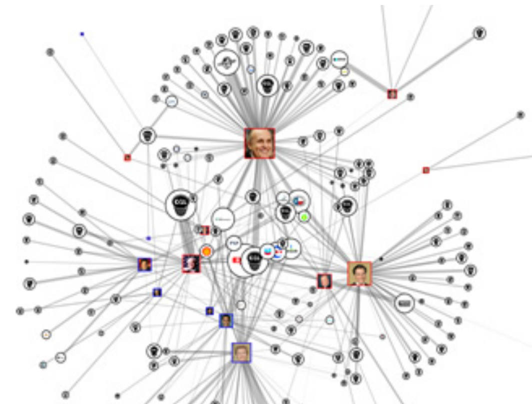


Protein-Protein Interactions
(by Peter Uetz)

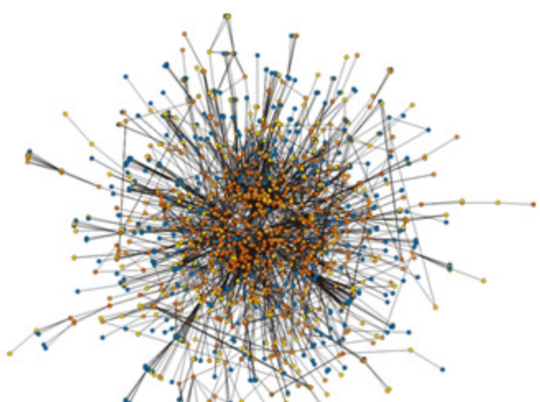
More Relational Networks



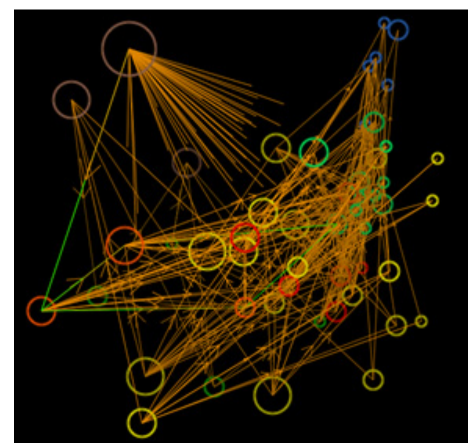
Flikr Social Network
(from http://www.flickr.com/photos/gustavog/sets/164006/)



Campaign Contributions from Oil Companies
(from http://oilmoney.priceofoil.org/)



Genomic Associations
(from Snel et al., 2002)

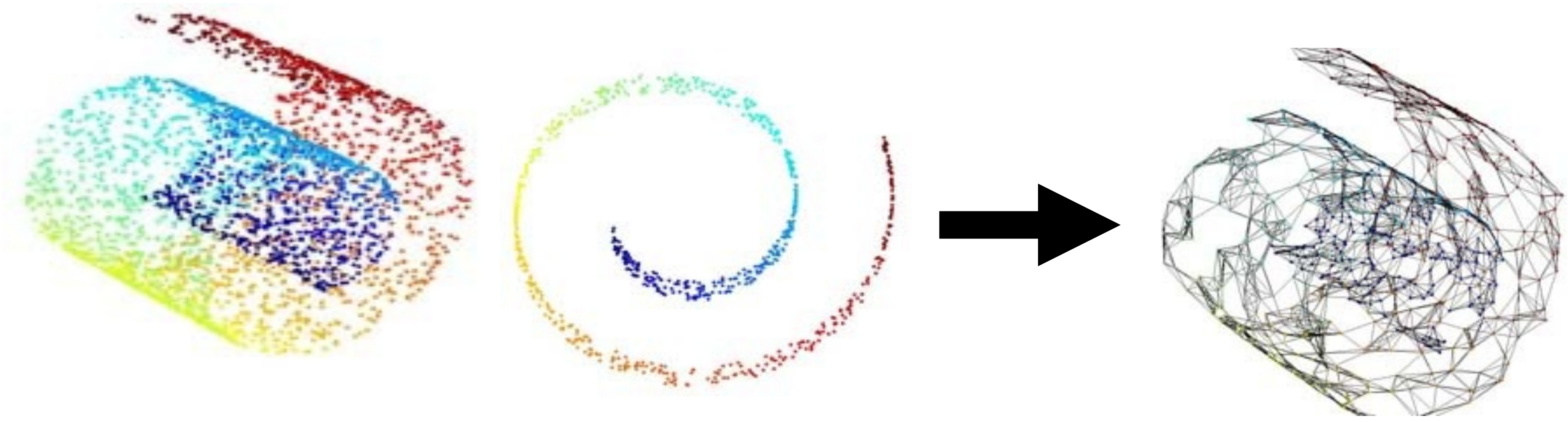


Seagrass Food Web
(generated at http://drjoe.biology.ecu.edu)

Creating a network from a surface

- Sample points from the surface
- Connect each point to the k closest points as measured by Euclidean distance

$$\mathrm{dist}(\mathbf{x}_i, \mathbf{x}_j) = \|\mathbf{x}_i - \mathbf{x}_j\|_2$$



Creating a network from data

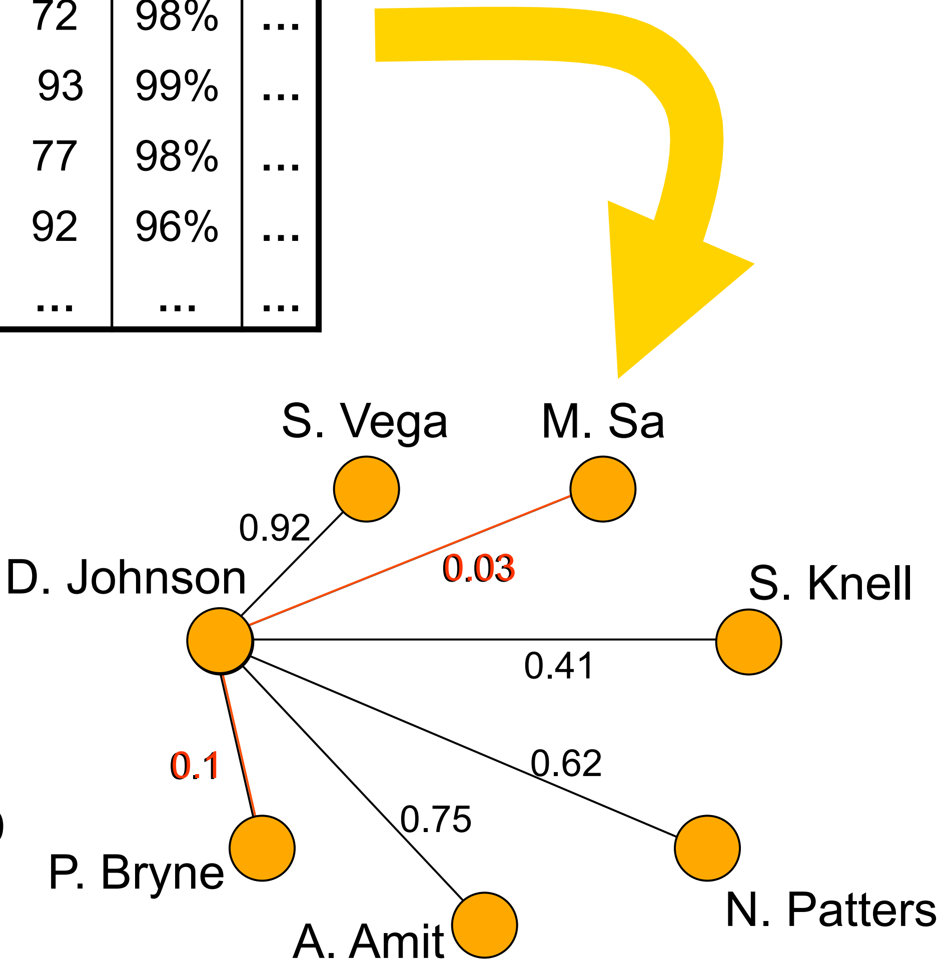
Medical Patients

Name	Age	Weight	Height	HR	SBP	DBP	SpO_2	…
D. Johnson	32	153	70	82	134	72	98%	…
S. Knell	47	169	65	130	169	93	99%	…
P. Bryne	42	128	61	102	129	77	98%	…
A. Amit	39	191	68	121	143	92	96%	…
…	…	…	…	…	…	…	…	…

1.) Measure the distance between pairs

$$\mathrm{dist}(\mathbf{x}_i, \mathbf{x}_j) = \|\mathbf{x}_i - \mathbf{x}_j\|_2$$

2.) Connect each patient to its k nearest neighbors



Creating a network from data

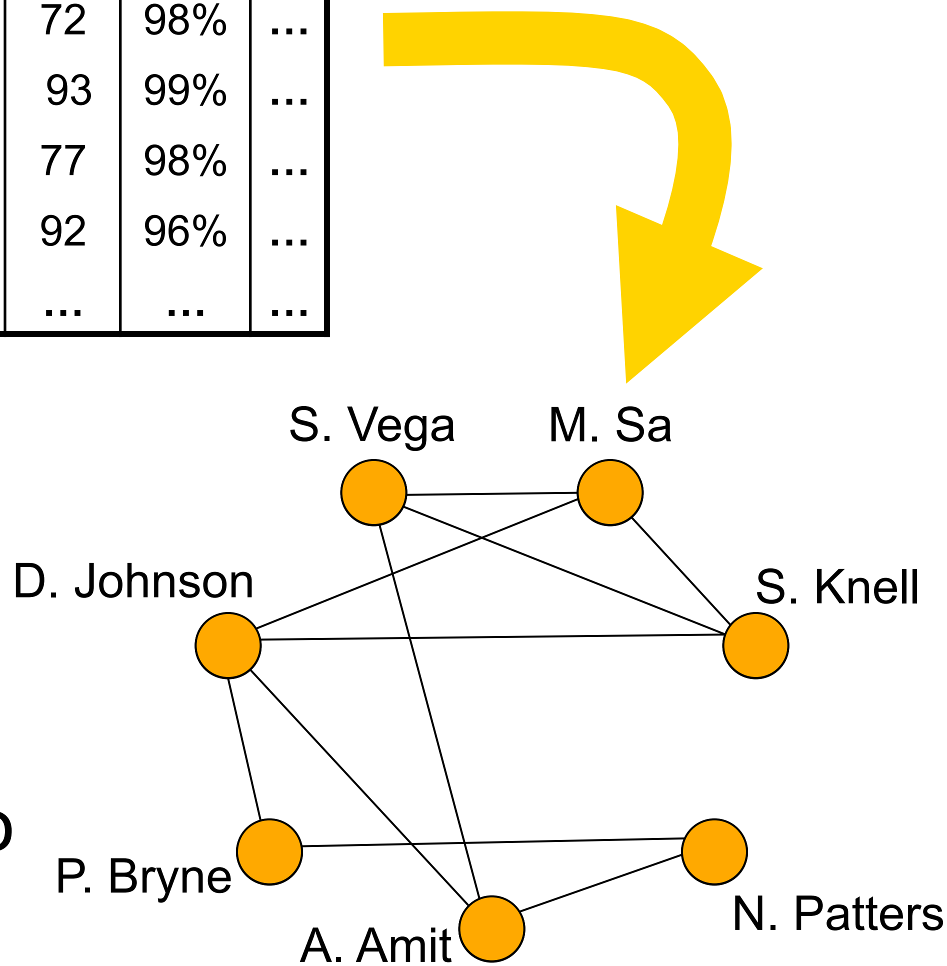
Medical Patients

Name	Age	Weight	Height	HR	SBP	DBP	SpO_2	…
D. Johnson	32	153	70	82	134	72	98%	…
S. Knell	47	169	65	130	169	93	99%	…
P. Bryne	42	128	61	102	129	77	98%	…
A. Amit	39	191	68	121	143	92	96%	…
…	…	…	…	…	…	…	…	…

1.) Measure the distance between pairs

$$\mathrm{dist}(\mathbf{x}_i, \mathbf{x}_j) = \|\mathbf{x}_i - \mathbf{x}_j\|_2$$

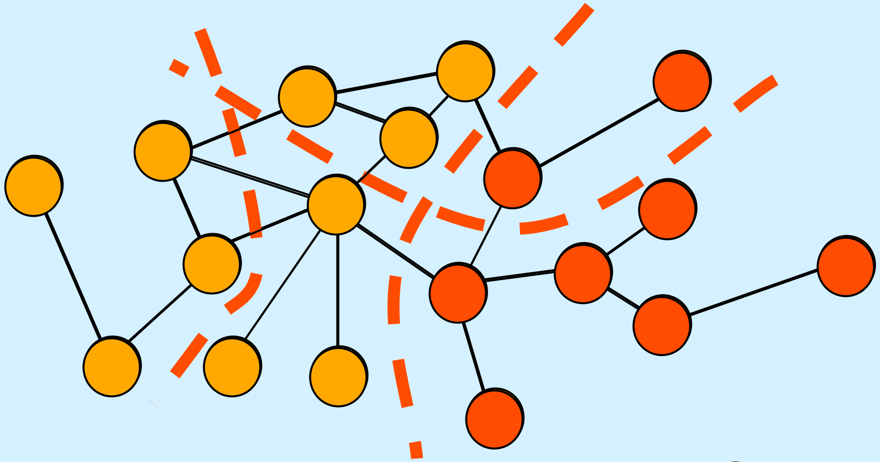
2.) Connect each patient to its k nearest neighbors



Graph partitioning

Goal: Partition the graph into multiple groups (*clusters*)

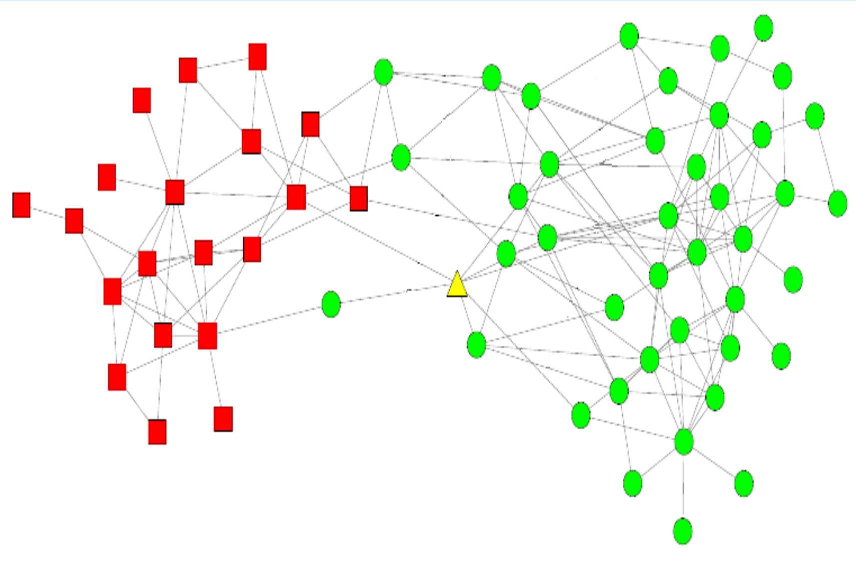
Find the two clusters



But where do we cut?

Identify the different parts of the rabbit

Social network of 62 dolphins (Lusseau et al., 2003)



Predict how the community will split when △ departs

Spectral Clustering

Based on materials by Rebecca Nugent and Larissa Stanberry
(University of Washington)

Spectral Clustering

- Algorithms that cluster points using eigenvectors of matrices derived from the data
- Obtain data representation in the low-dimensional space that can be easily clustered
- Variety of methods that use the eigenvectors differently

Spectral Clustering

- Empirically very successful
- Authors disagree:
  – Which eigenvectors to use
  – How to derive clusters from these eigenvectors

- Two general methods

Definitions

- n x n Adjacency matrix A.
  - A(i,j) = weight on edge from *i* to *j*
  - If the graph is undirected A(i,j)=A(j,i), i.e. A is symmetric

- n x n Transition matrix P.
  - P is row stochastic
  - P(i,j) = probability of stepping on node j from node i

    $= A(i,j)/\sum_i A(i,j)$

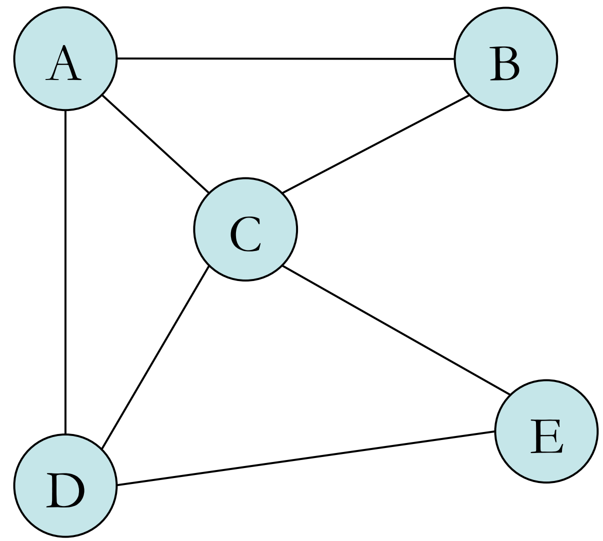
- n x n Laplacian Matrix L.
  - $L(i,j)=\sum_i A(i,j)-A(i,j)$
  - Symmetric positive semi-definite for undirected graphs
  - Singular

Definitions

Adjacency Matrix

A=

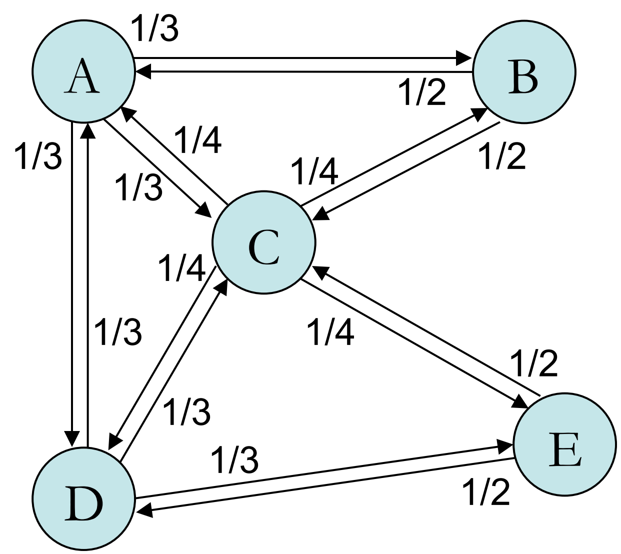
	1	1	1	
1		1		
1	1		1	1
1		1		1
		1	1	



Transition Matrix

P=

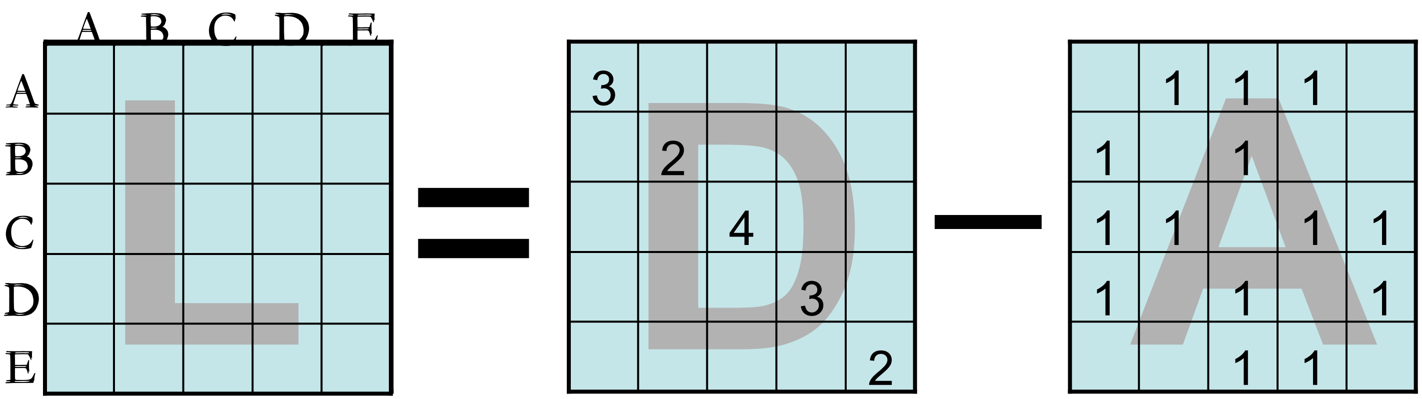
	1/3	1/3	1/3	
1/2		1/2		
1/4	1/4		1/4	1/4
1/3		1/3		1/3
		1/2	1/2	

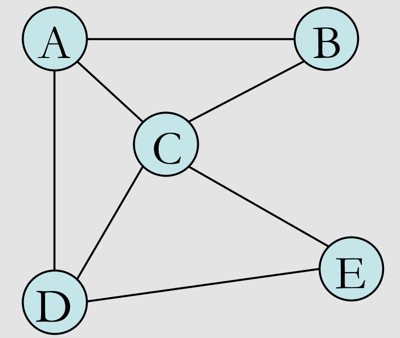


Spectral Graph Analysis

Graph Laplacian

$$L = D - A \qquad D = \mathrm{diag}(d)$$

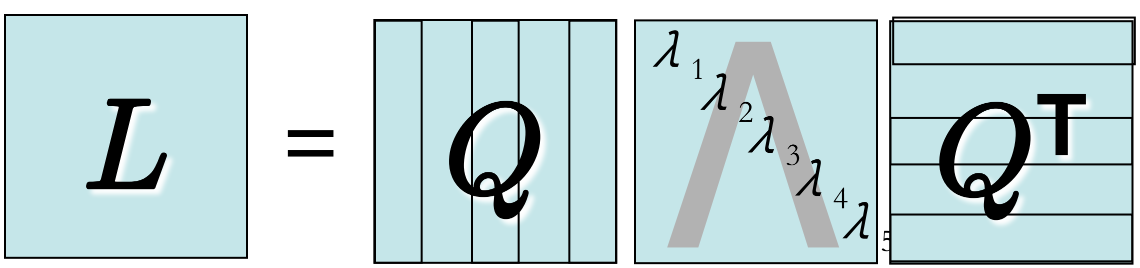




$$\mathbf{A} = \begin{bmatrix} & 1 & 1 & 1 & \\ 1 & & 1 & & \\ 1 & 1 & & 1 & 1 \\ 1 & & 1 & & 1 \\ & & 1 & 1 & \end{bmatrix}$$

$$\mathbf{d} = \begin{bmatrix} 3 & 2 & 4 & 3 & 2 \end{bmatrix}$$

Take the *eigendecomposition* of L



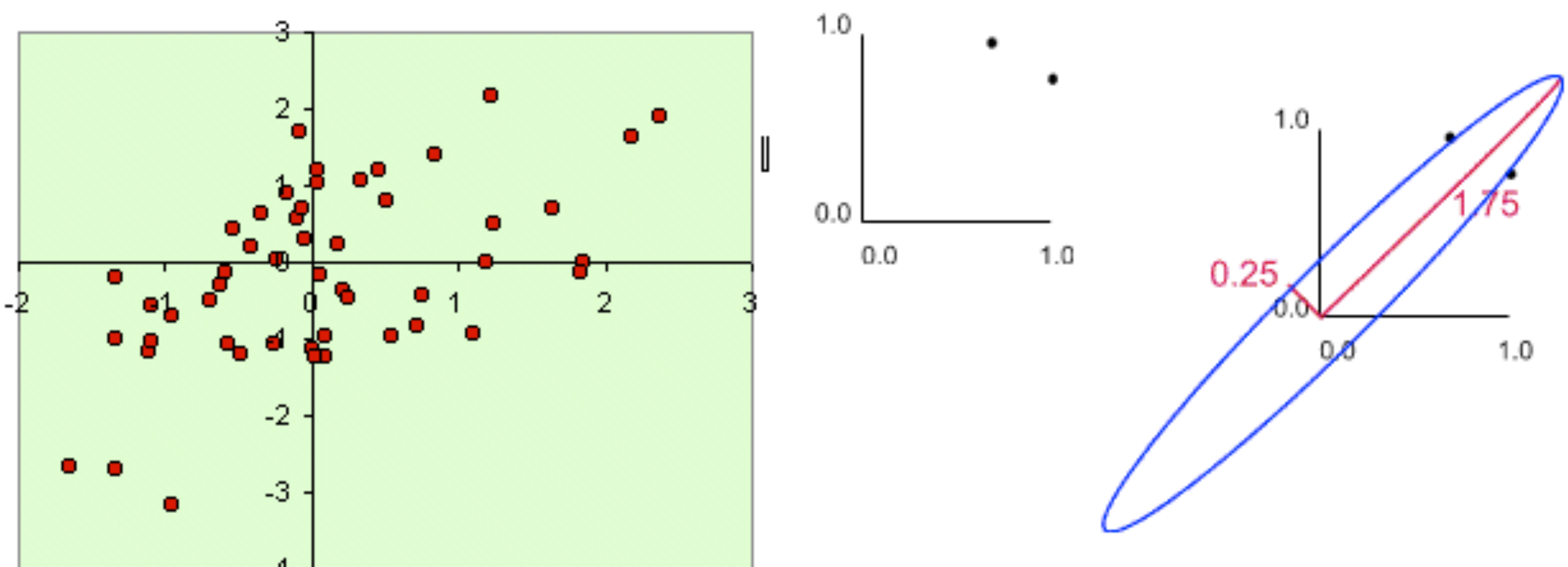
Eigenvectors

- Intuitive definition: An eigenvector is a direction for a matrix
- An eigenvector of an *n x n* matrix A is a vector such that $Av = \lambda v$, where *v* is the eigenvector and λ is the corresponding eigenvalue
  - Multiplying vector v by the scalar λ effectively stretches or shrinks the vector
- An *n x n* matrix should have *n* linearly independent eigenvectors

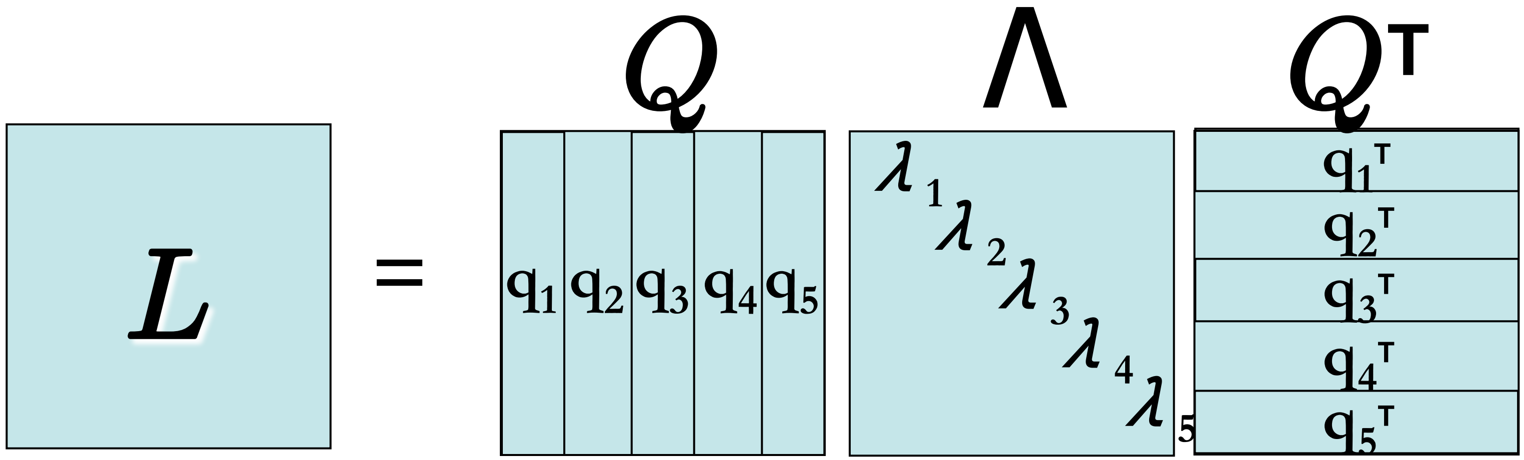
Eigenvectors Illustrated

- Consider an elliptical data cloud. The eigenvectors are then the major and minor axes of the ellipse

1.00	0.75
0.75	1.00



Spectral Graph Analysis



Eigenvector q_1 is constant

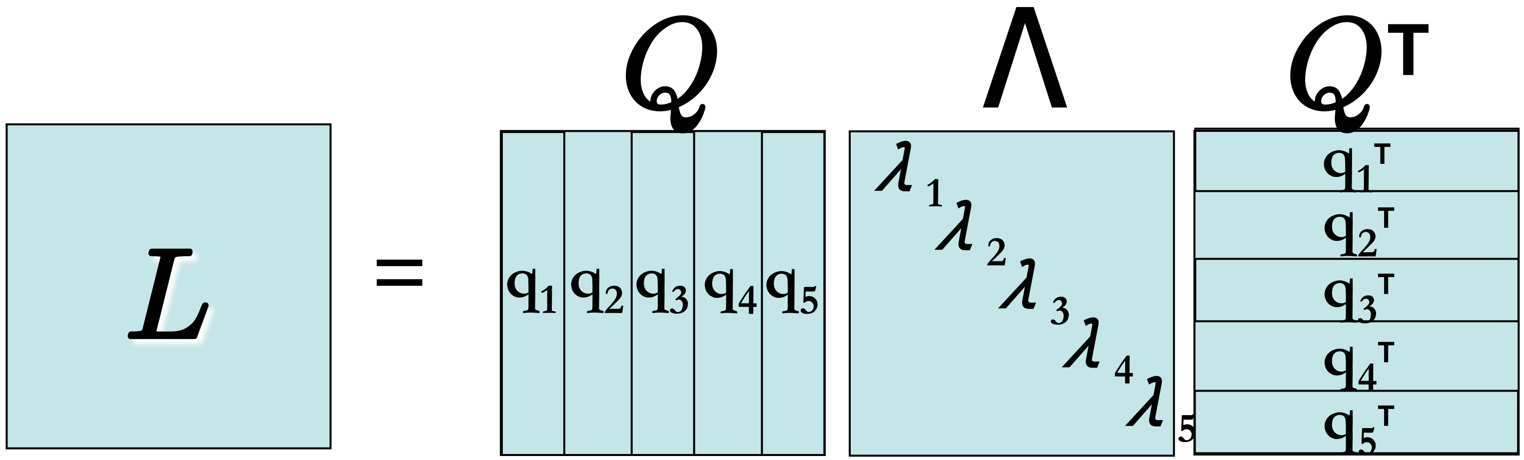
Eigenvalue $\lambda_1 = 0$

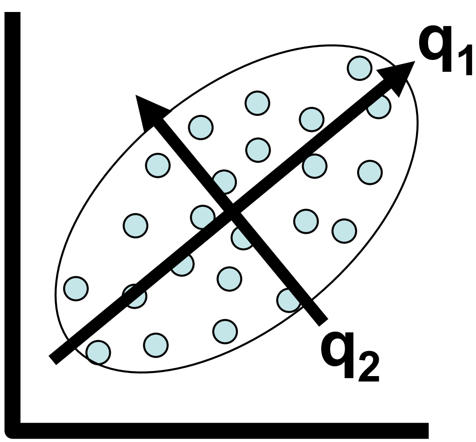
3	-1	-1	-1	
-1	**2**	-1		
-1	-1	**4**	-1	-1
-1		-1	**3**	-1
		-1	-1	**2**

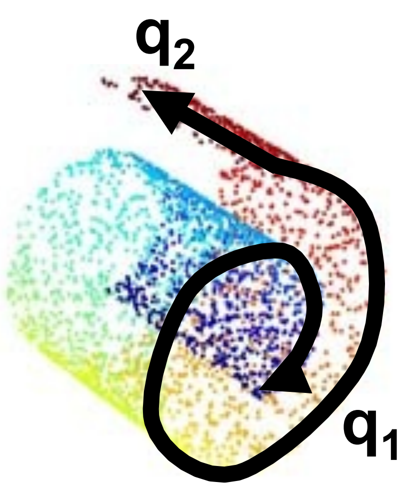
	q_1	q_2	q_3	q_4	q_5
A	0.45	-0.27	-0.5	-0.65	0.22
B	0.45	-0.65	0.5	0.27	0.22
C	0.45	-0.00	0.00	0.00	-0.89
D	0.45	0.27	-0.5	0.65	0.22
E	0.45	0.65	0.5	-0.27	0.22

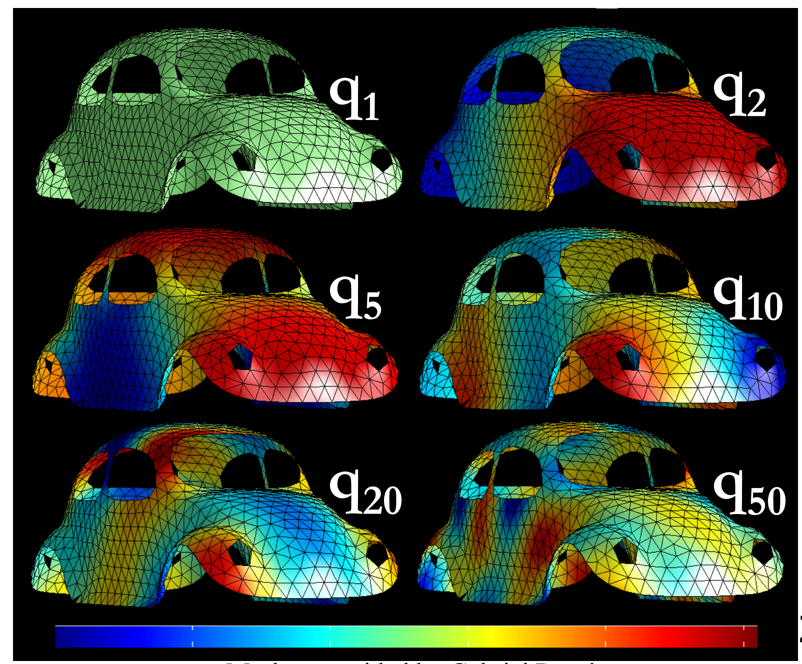
1	2	3	4	5
0.00	0	0	0	0
0	1.59	0	0	0
0	0	3.00	0	0
0	0	0	4.41	0
0	0	0	0	5.00

Spectral Graph Analysis









Meshes provided by Gabriel Peyré

Method #1

- Partition using only one eigenvector at a time
- Use procedure recursively
- Example: Image Segmentation
  - Uses 2^{nd} (smallest) eigenvector to define optimal cut
  - Recursively generates two clusters with each cut

Method #2

- Use k eigenvectors (k chosen by user)
- Directly compute k-way partitioning
- Experimentally has been seen to be “better”

Spectral Clustering Algorithm
(by Ng, Jordan, and Weiss)

- Given a set of points $\mathrm{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$
- Form the affinity matrix

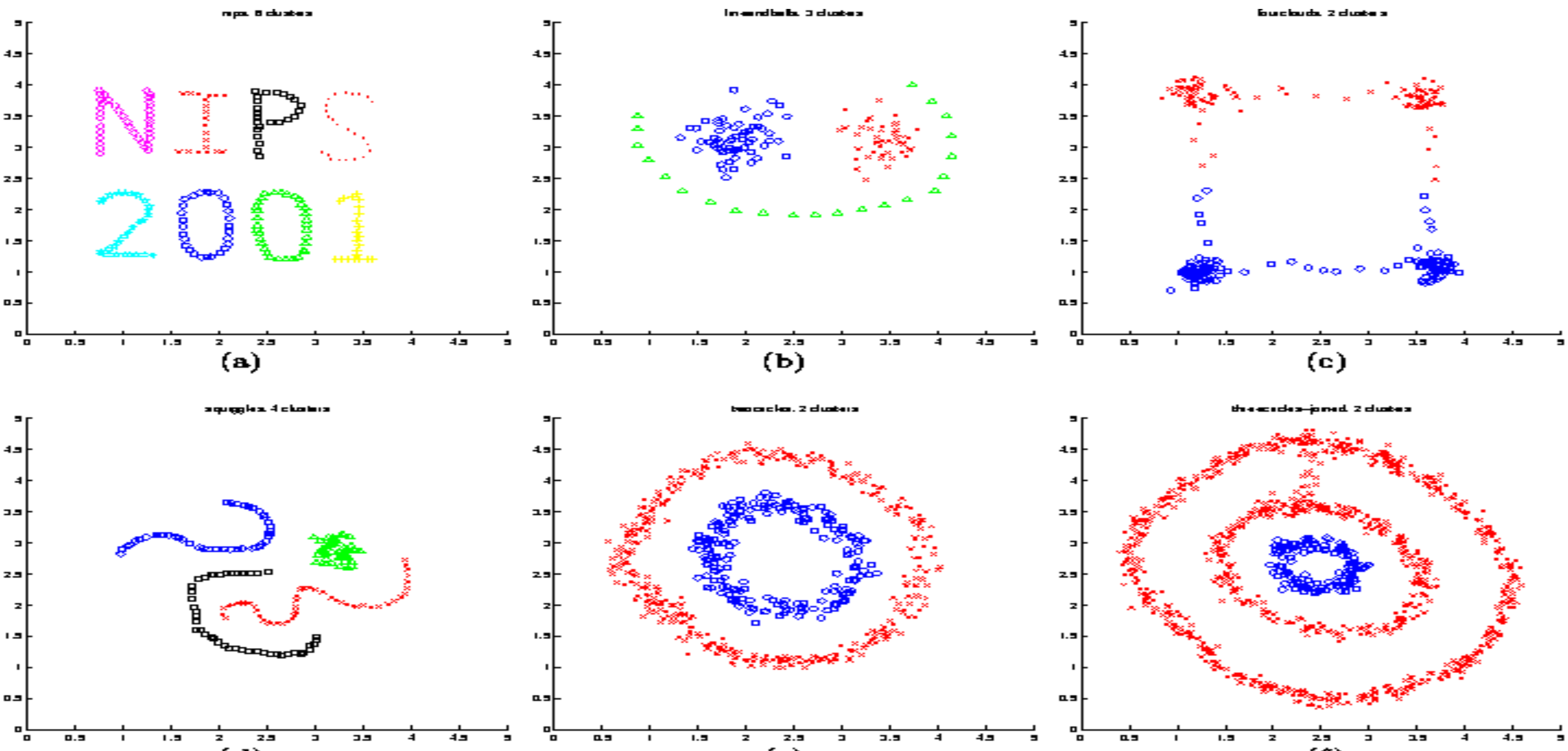
$$A_{ij} = \exp\left(-\frac{\|\mathbf{x}_i - \mathbf{x}_j\|_2^2}{2\sigma^2}\right) \quad \forall i \neq j \quad A_{ii} = 0$$

- Define diagonal matrix $D_{ii} = \sum_k A_{ik}$
- Form the matrix $\mathbf{L} = \mathbf{D}^{-1/2}\mathbf{A}\mathbf{D}^{-1/2}$
- Stack the k largest eigenvectors of $\mathbf{L}$ to form the columns of the new matrix: $\mathbf{E} = [\mathbf{e}_1 \ \mathbf{e}_2 \ \ldots \ \mathbf{e}_k]$
- Normalize each of $\mathbf{E}$'s rows to have unit length
- Cluster rows of $\mathbf{E}$ into k clusters using K-means

Why?

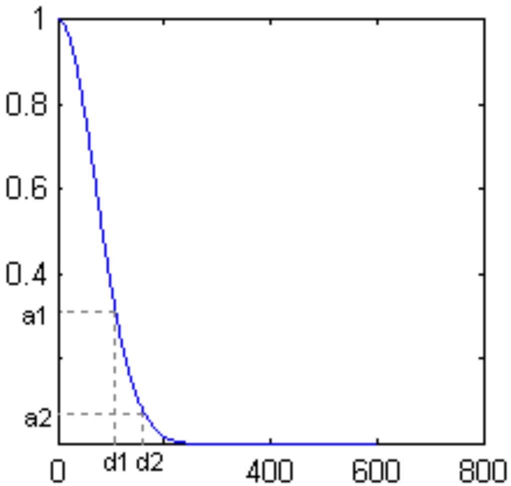
Q: If we eventually use K-means, why not just apply K-means to the original data?

A: This method allows us to cluster non-convex regions

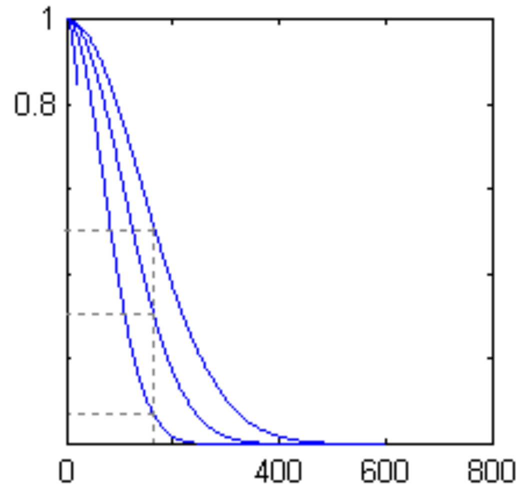


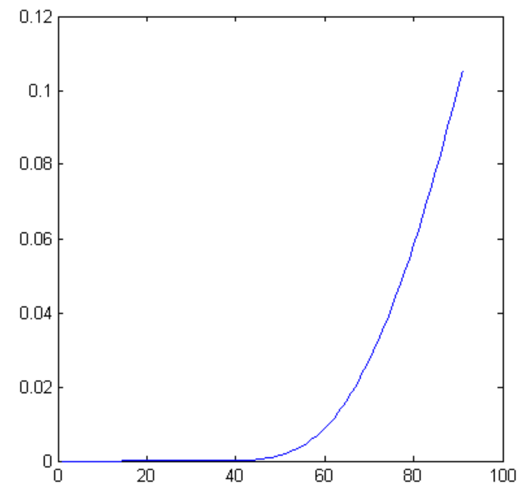
Nature of the Affinity Matrix

$$A_{ij} = e^{-(s_i - s_j)^2 / 2\sigma^2} \quad i \neq j \qquad A_{ii} = 0$$



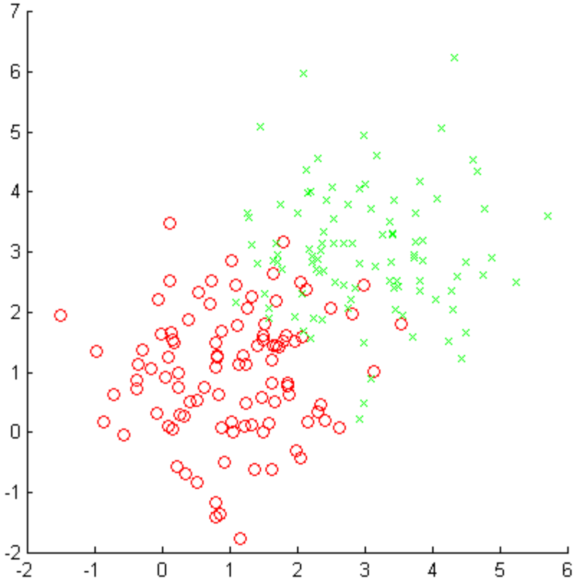
"closer" vertices will get larger weight

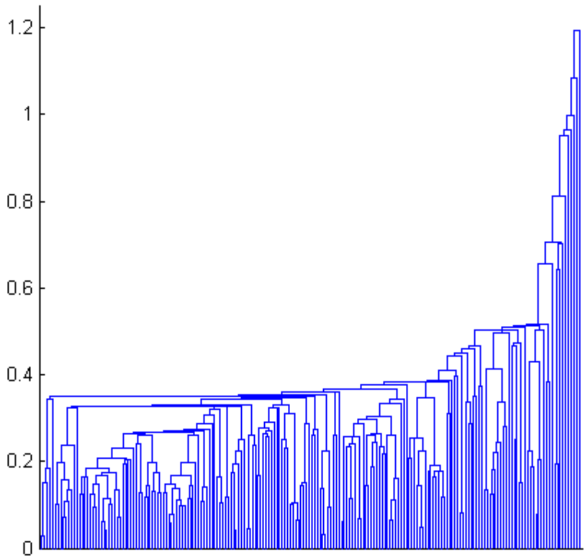




Weight as a function of σ

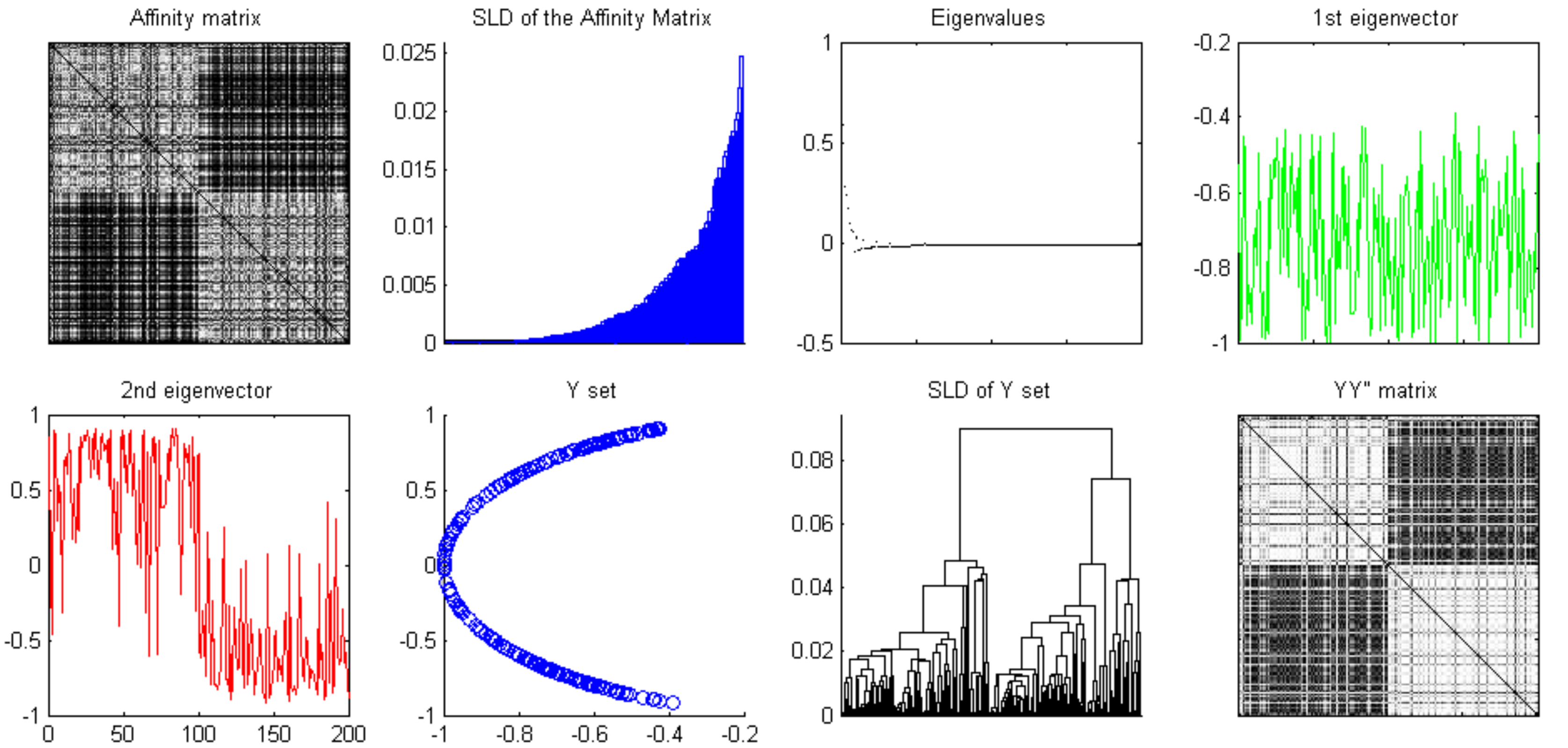
Simple Example



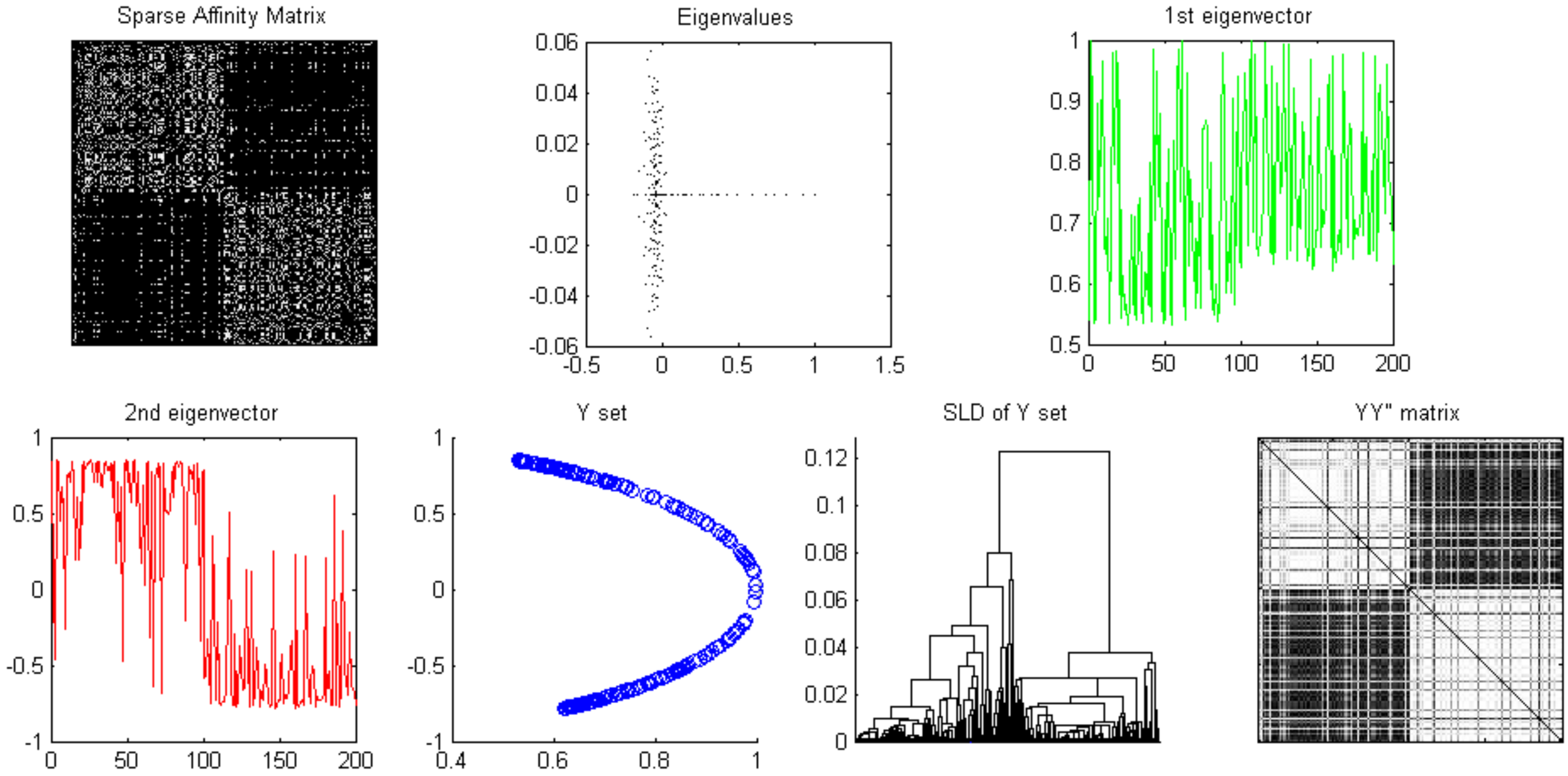


- Consider two 2-dimensional slightly overlapping Gaussian clouds each containing 100 points.

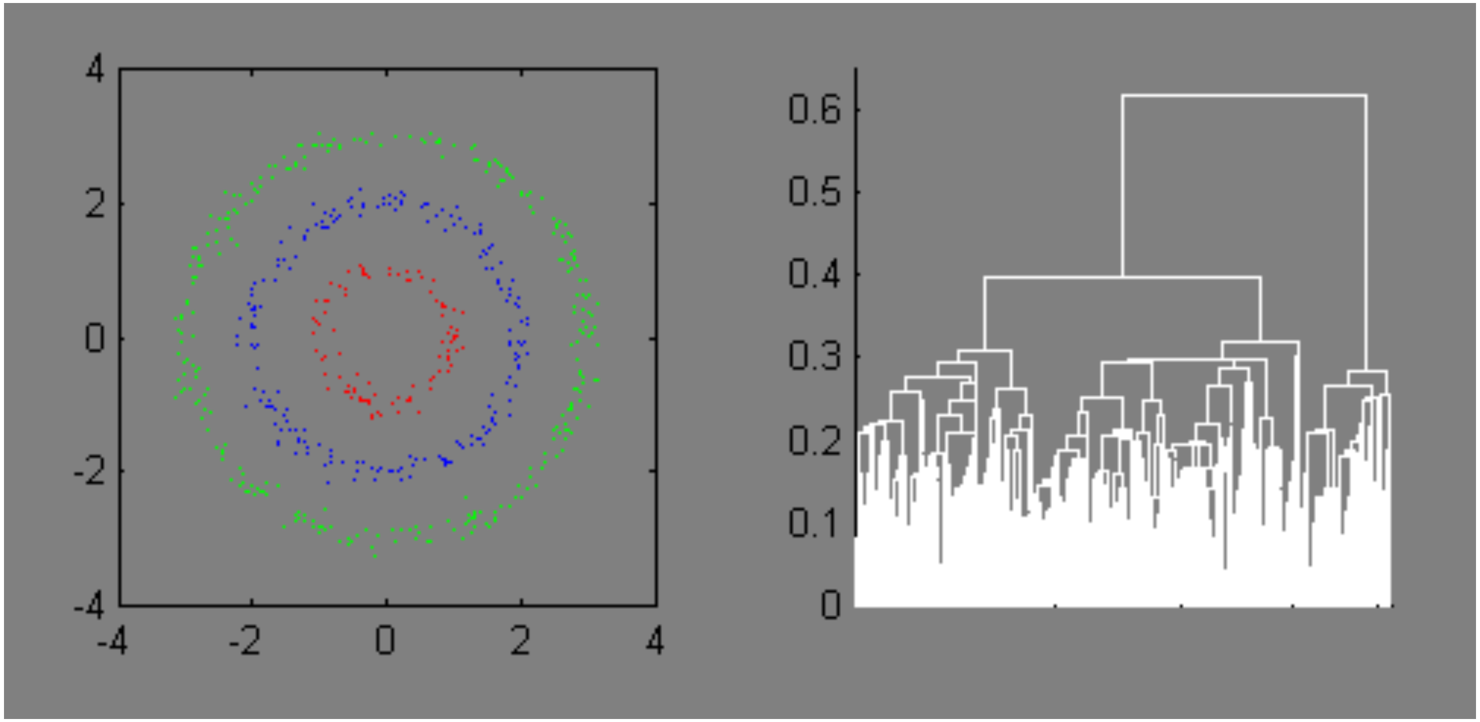
Simple Example cont-d I



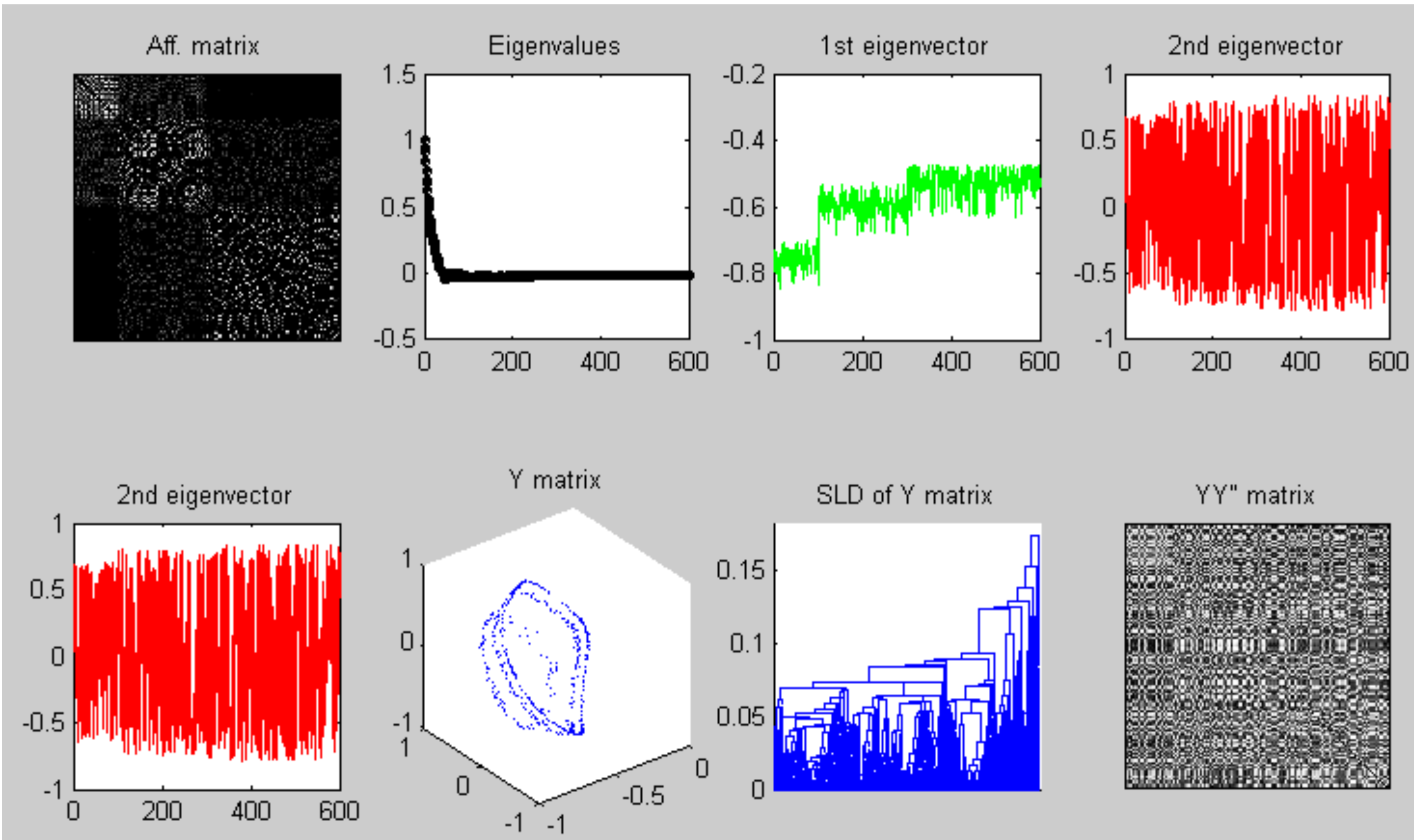
Simple Example cont-d II



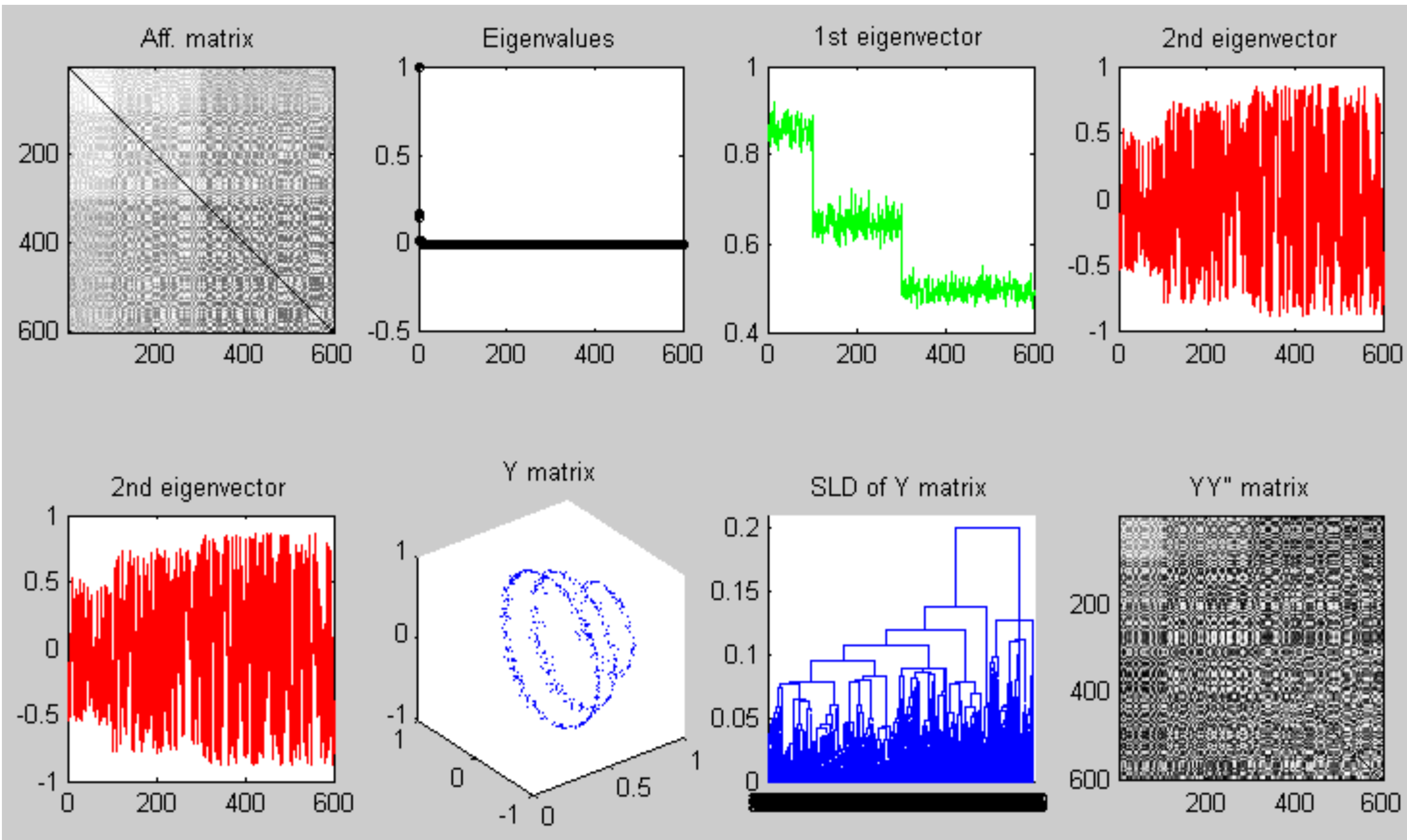
Example 2 (not so simple)



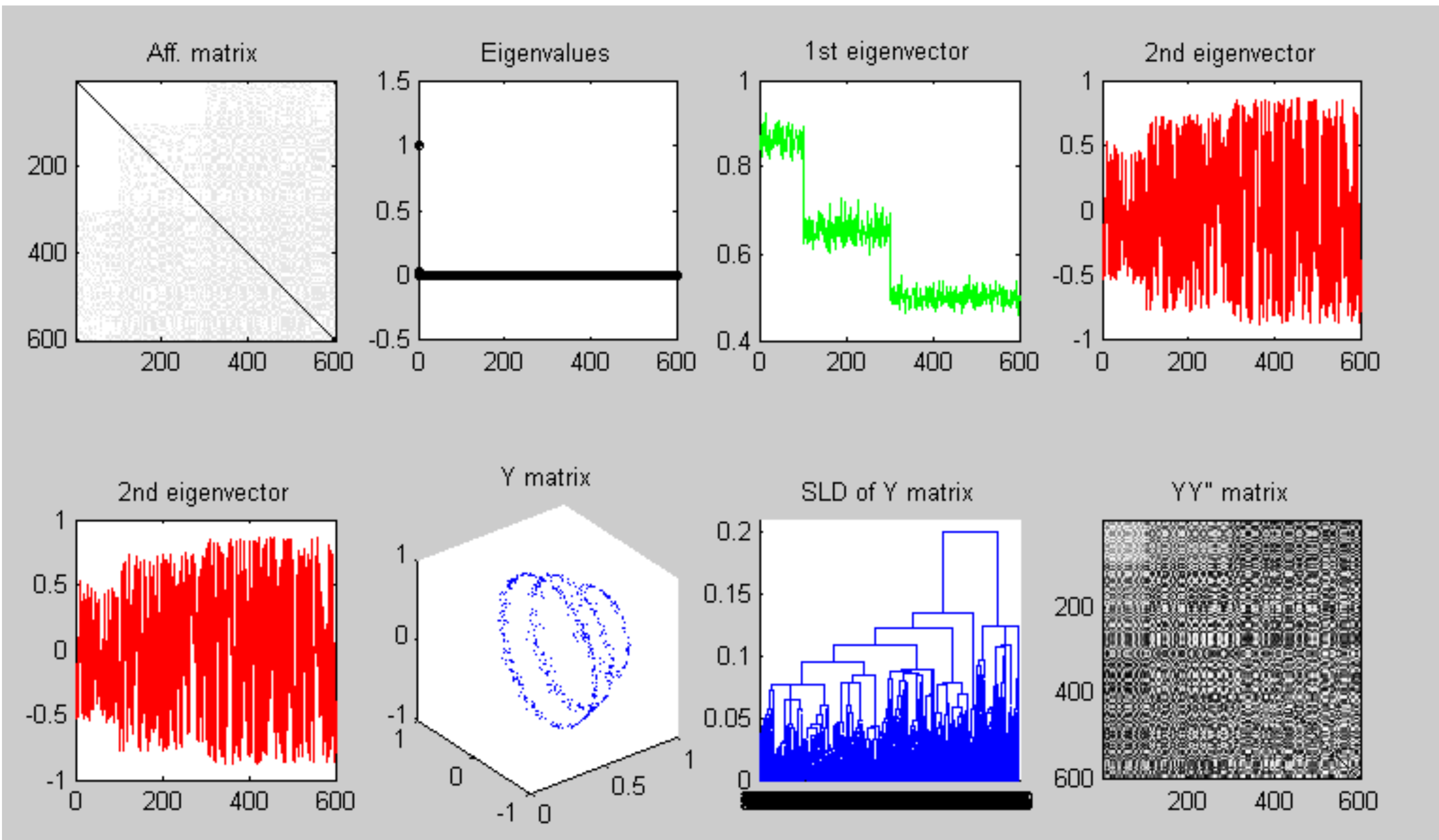
Example 2 cont-d I



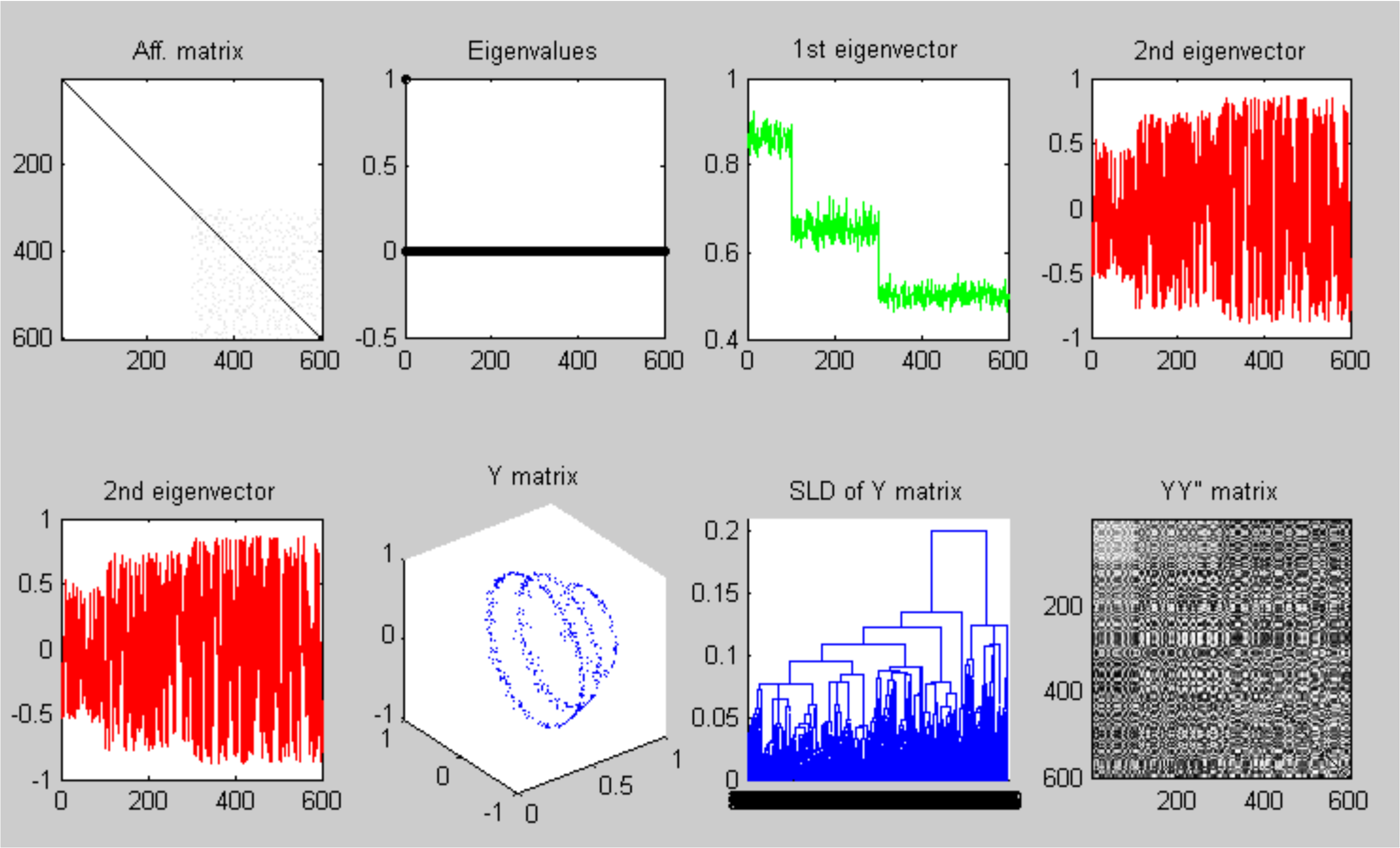
Example 2 cont-d II



Example 2 cont-d III



Example 2 cont-d IV



User's Prerogative

- Choice of k, the number of clusters
- Choice of scaling factor
  - Realistically, search over σ^2 and pick value that gives the tightest clusters
- Choice of clustering method

Comparison of Methods

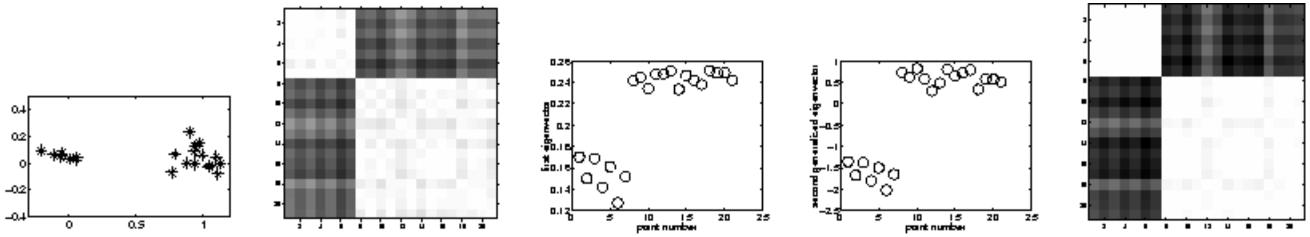
Authors	Matrix used	Procedure/Eigenvectors used
Perona/ Freeman	Affinity A	1st x: $Ax = \lambda x$ Recursive procedure
Shi/Malik	D-A with D a degree matrix $D(i,i) = \sum_j A(i,j)$	2nd smallest *generalized* eigenvector $(D-A)x = \lambda Dx$ Also recursive
Scott/ Longuet-Higgins	Affinity A, User inputs k	Finds k eigenvectors of A, forms V. Normalizes rows of V. Forms Q = VV' . Segments by Q. Q(i,j)=1 -> same cluster
Ng, Jordan, Weiss	Affinity A, User inputs k	Normalizes A. Finds k eigenvectors, forms X. Normalizes X, clusters rows

Advantages/Disadvantages

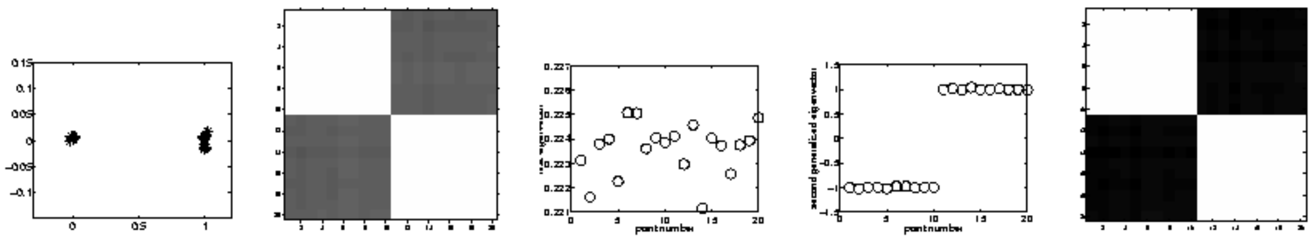
- Perona/Freeman
  - For block diagonal affinity matrices, the first eigenvector finds points in the "dominant" cluster; not very consistent
- Shi/Malik
  - 2^{nd} generalized eigenvector minimizes affinity between groups by affinity within each group; no guarantee, constraints

Advantages/Disadvantages

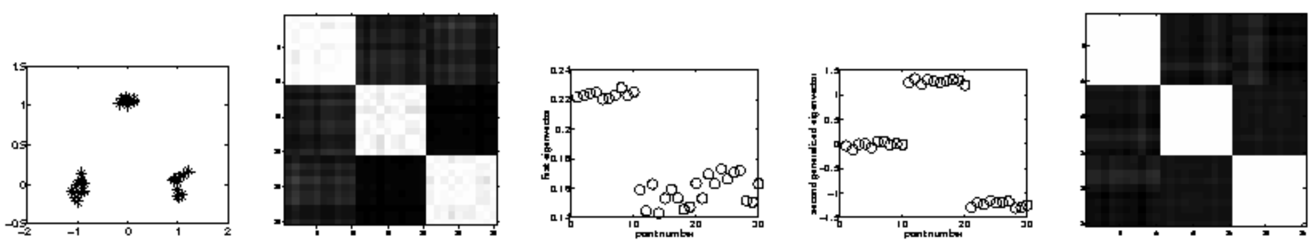
- Scott/Longuet-Higgins
  - Depends largely on choice of k
  - Good results
- Ng, Jordan, Weiss
  - Again depends on choice of k
  - Claim: effectively handles clusters whose overlap or connectedness varies across clusters



Affinity Matrix | Perona/Freeman 1st eigenv. | Shi/Malik 2nd gen. eigenv. | Scott/Lon.Higg Q matrix



Affinity Matrix | Perona/Freeman 1st eigenv. | Shi/Malik 2nd gen. eigenv. | Scott/Lon.Higg Q matrix



Affinity Matrix | Perona/Freeman 1st eigenv. | Shi/Malik 2nd gen. eigenv. | Scott/Lon.Higg Q matrix

Inherent Weakness

- At some point, a clustering method is chosen
- Each clustering method has its strengths and weaknesses
- Some methods also require a priori knowledge of k

References

- Alpert et al.   Spectral partitioning with multiple eigenvectors
- Brand&Huang.   A unifying theorem for spectral embedding and clustering
- Belkin&Niyogi.   Laplasian maps for dimensionality reduction and data representation
- Blatt et al. Data clustering using a model granular magnet
- Buhmann. Data clustering and learning
- Fowlkes et al. Spectral grouping using the Nystrom method
- Meila&Shi. A random walks view of spectral segmentation
- Ng et al. On Spectral clustering: analysis and algorithm
- Shi&Malik. Normalized cuts and image segmentation
- Weiss et al. Segmentation using eigenvectors: a unifying view

Community structures

Slides modified from Huan Liu, Lei Tang, Nitin Agarwal

Community Detection

- A community is a set of nodes between which the interactions are (relatively) frequent

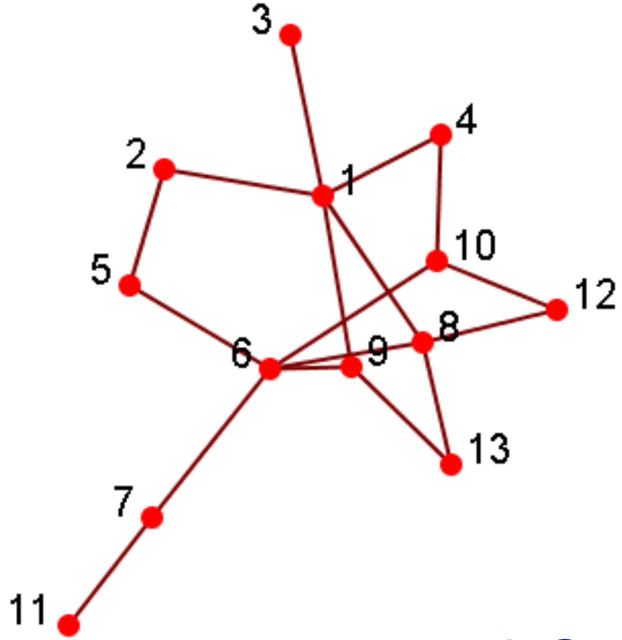
  a.k.a. group, subgroup, module, cluster

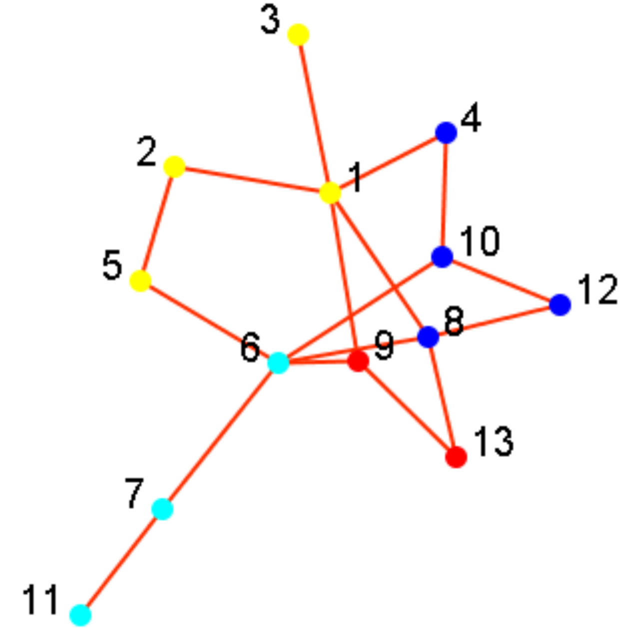
- Community detection

  a.k.a. grouping, clustering, finding cohesive subgroups
  - Given: a social network
  - Output: community membership of (some) actors

- Applications
  - Understanding the interactions between people
  - Visualizing and navigating huge networks
  - Forming the basis for other tasks such as data mining

Visualization after Grouping





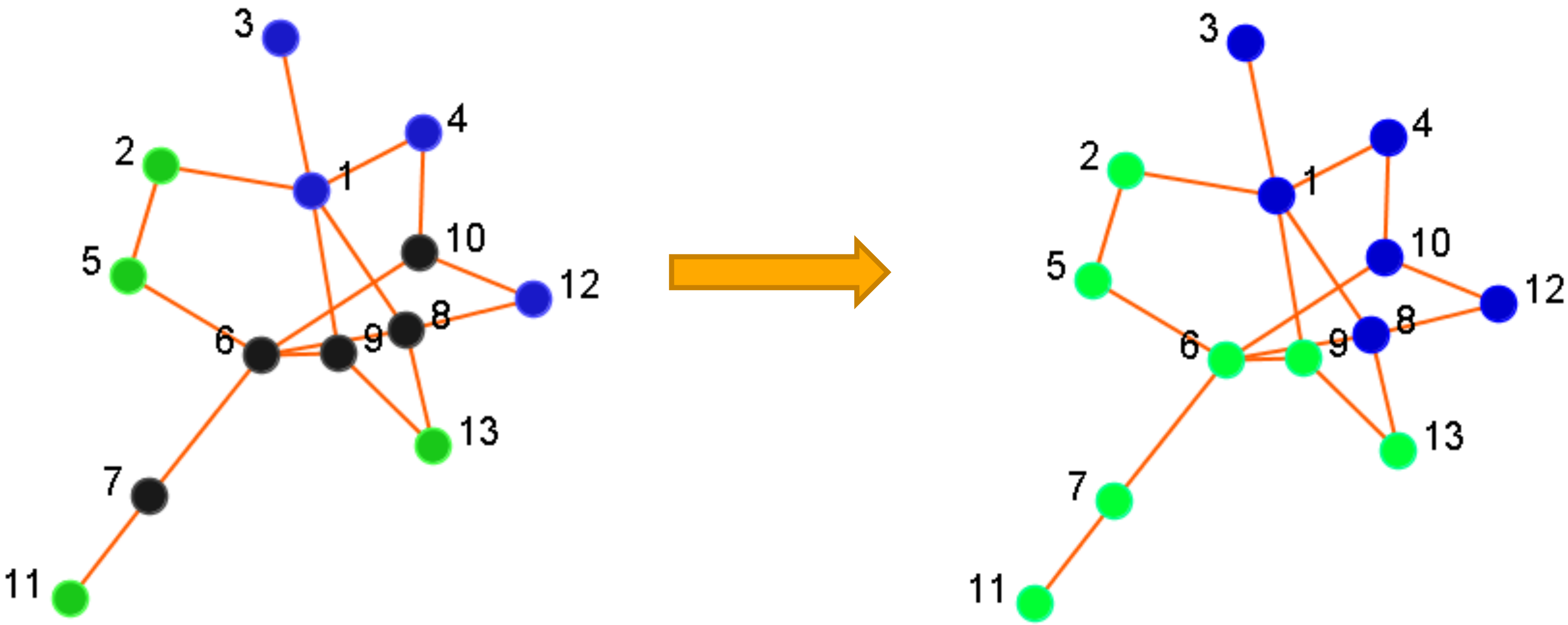
4 Groups:
{1,2,3,5}
{4,8,10,12}
{6,7,11}
{9,13}

(Nodes colored by Community Membership)

Classification

- User Preference or Behavior can be represented as class labels
  - Whether or not clicking on an ad
  - Whether or not interested in certain topics
  - Subscribed to certain political views
  - Like/Dislike a product
- Given
  - A social network
  - Labels of some actors in the network
- Output
  - Labels of remaining actors in the network

Visualization after Prediction



: Smoking

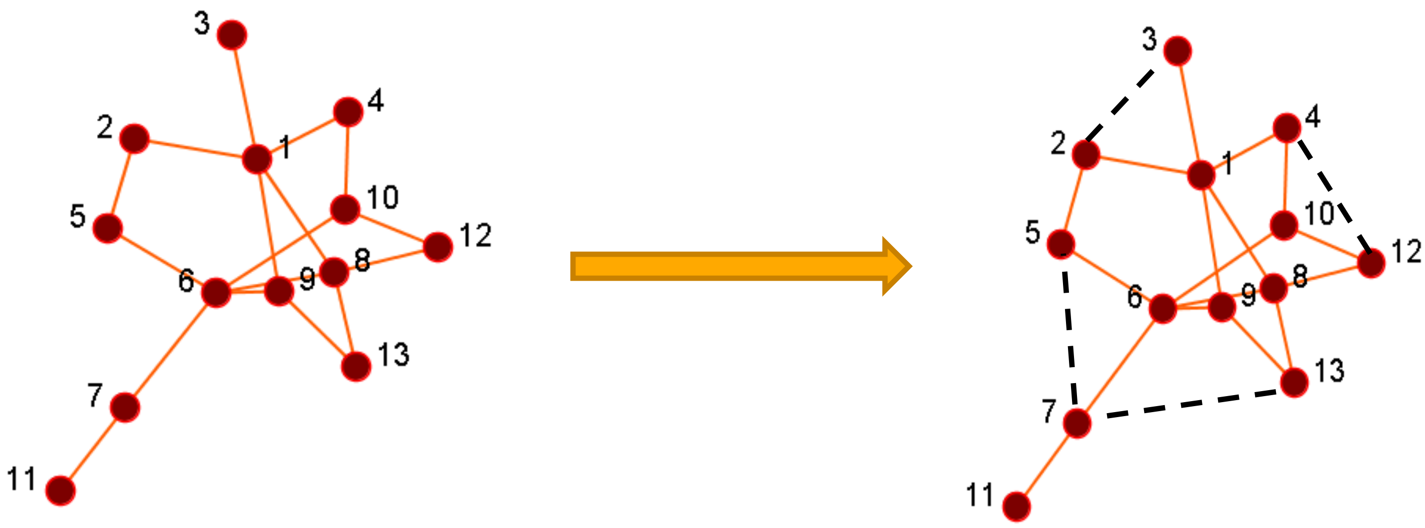
: Non-Smoking

: ? Unknown

Predictions
6: Non-Smoking
7: Non-Smoking
8: Smoking
9: Non-Smoking
10: Smoking

Link Prediction

- Given a social network, predict which nodes are likely to get connected
- Output a list of (ranked) pairs of nodes
- Example: Friend recommendation in Facebook



(2, 3)
(4, 12)
(5, 7)
(7, 13)

PRINCIPLES OF COMMUNITY DETECTION

Communities

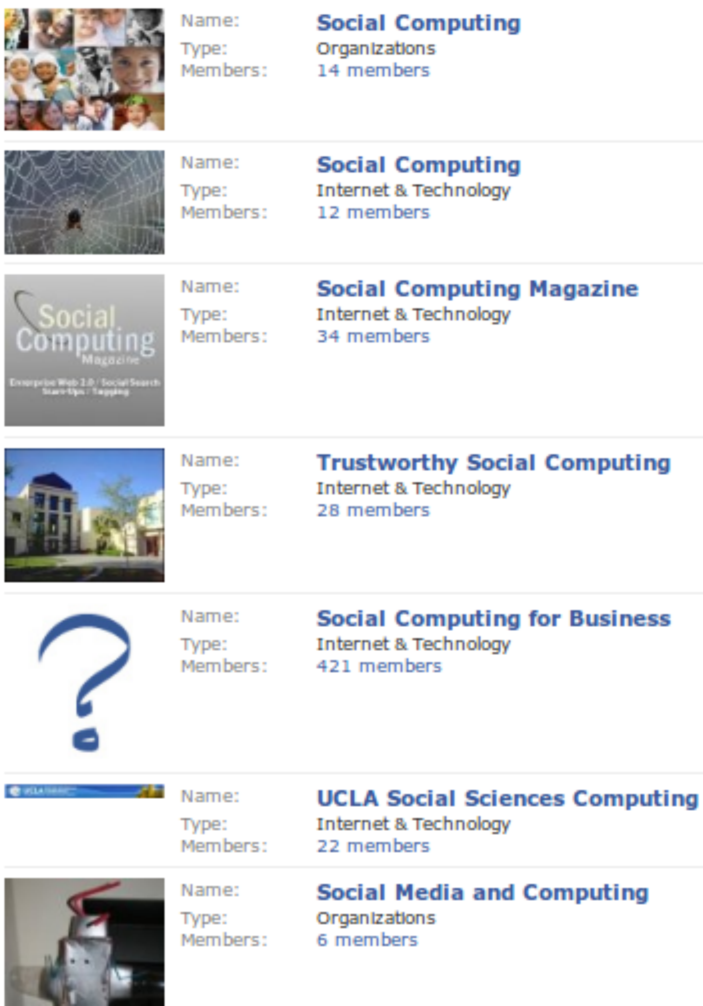
- Community: "subsets of actors among whom there are relatively strong, direct, intense, frequent or positive ties."
  -- Wasserman and Faust, *Social Network Analysis, Methods and Applications*

- Community is a set of actors interacting with each other *frequently*

- A set of people without interaction is NOT a community
  - e.g. people waiting for a bus at station but don't talk to each other

Example of Communities

Communities from Facebook


Name: Social Computing
Type: Organizations
Members: 14 members

Name: Social Computing
Type: Internet & Technology
Members: 12 members

Name: Social Computing Magazine
Type: Internet & Technology
Members: 34 members

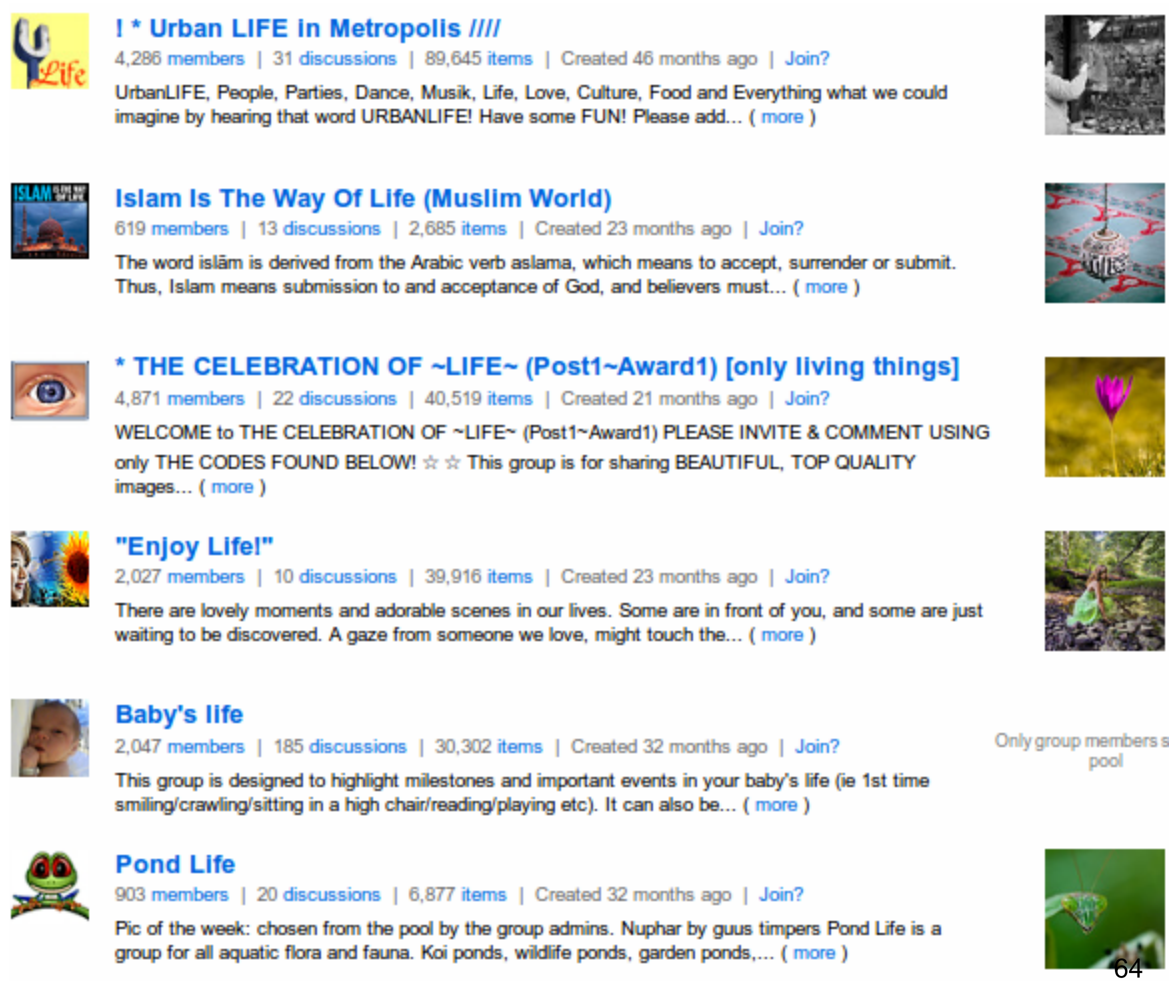
Name: Trustworthy Social Computing
Type: Internet & Technology
Members: 28 members

Name: Social Computing for Business
Type: Internet & Technology
Members: 421 members

Name: UCLA Social Sciences Computing
Type: Internet & Technology
Members: 22 members

Name: Social Media and Computing
Type: Organizations
Members: 6 members

Communities from Flickr


! * Urban LIFE in Metropolis ////
4,286 members | 31 discussions | 89,645 items | Created 46 months ago | Join?
UrbanLIFE, People, Parties, Dance, Musik, Life, Love, Culture, Food and Everything what we could imagine by hearing that word URBANLIFE! Have some FUN! Please add... (more)

Islam Is The Way Of Life (Muslim World)
619 members | 13 discussions | 2,685 items | Created 23 months ago | Join?
The word islām is derived from the Arabic verb aslama, which means to accept, surrender or submit. Thus, Islam means submission to and acceptance of God, and believers must... (more)

*** THE CELEBRATION OF ~LIFE~ (Post1~Award1) [only living things]**
4,871 members | 22 discussions | 40,519 items | Created 21 months ago | Join?
WELCOME to THE CELEBRATION OF ~LIFE~ (Post1~Award1) PLEASE INVITE & COMMENT USING only THE CODES FOUND BELOW! ☆ ☆ This group is for sharing BEAUTIFUL, TOP QUALITY images... (more)

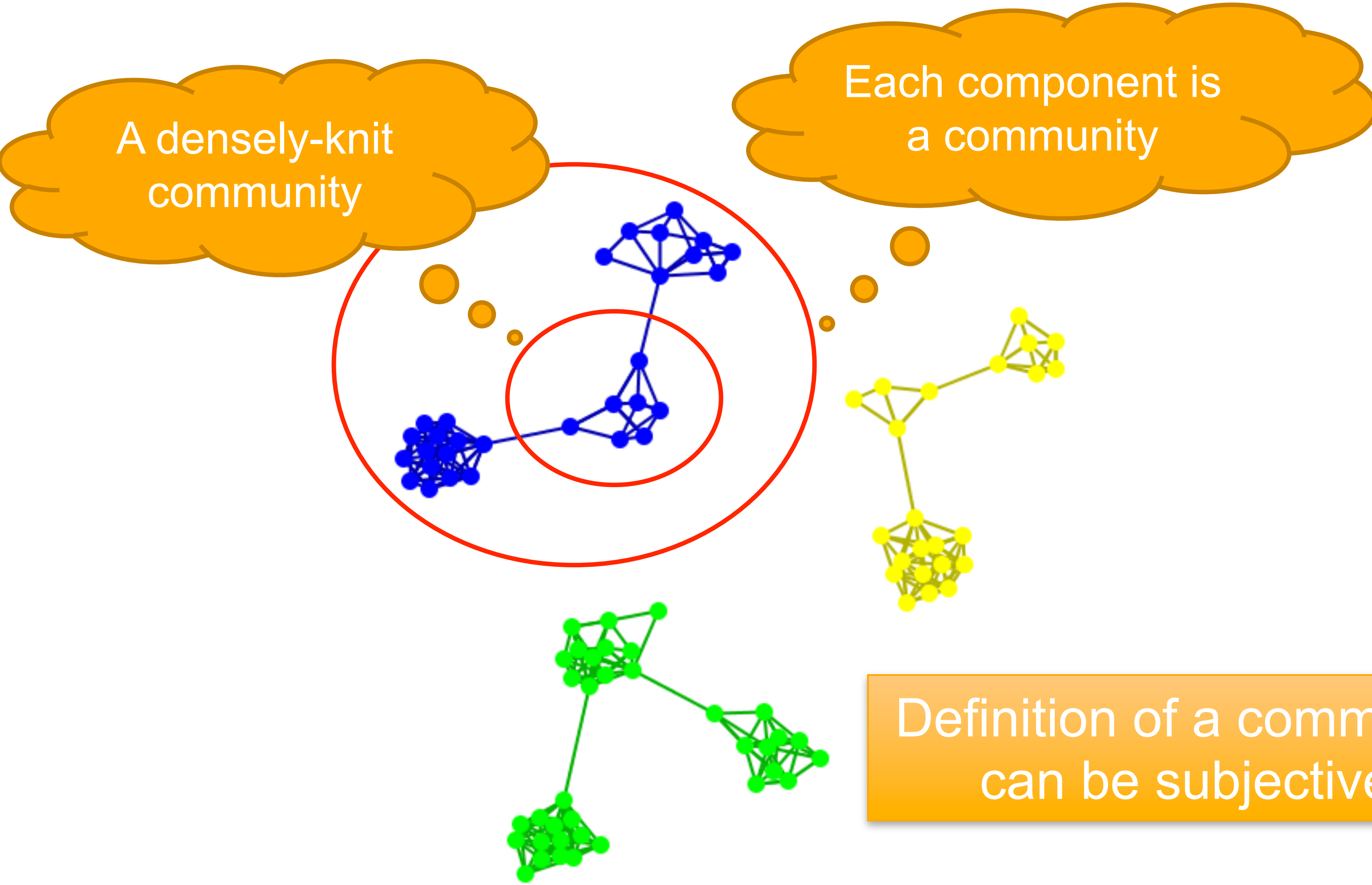
"Enjoy Life!"
2,027 members | 10 discussions | 39,916 items | Created 23 months ago | Join?
There are lovely moments and adorable scenes in our lives. Some are in front of you, and some are just waiting to be discovered. A gaze from someone we love, might touch the... (more)

Baby's life
2,047 members | 185 discussions | 30,302 items | Created 32 months ago | Join?
This group is designed to highlight milestones and important events in your baby's life (ie 1st time smiling/crawling/sitting in a high chair/reading/playing etc). It can also be... (more)

Only group members s pool

Pond Life
903 members | 20 discussions | 6,877 items | Created 32 months ago | Join?
Pic of the week: chosen from the pool by the group admins. Nuphar by guus timpers Pond Life is a group for all aquatic flora and fauna. Koi ponds, wildlife ponds, garden ponds,... (more)

Subjectivity of Community Definition



Definition of a community can be subjective.

Taxonomy of Community Criteria

- Criteria vary depending on the tasks
- Roughly, community detection methods can be divided into 4 categories (not exclusive):
- Node-Centric Community
  - Each node in a group satisfies certain properties
- Group-Centric Community
  - Consider the connections within a group as a whole. The group has to satisfy certain properties without zooming into node-level
- Network-Centric Community
  - Partition the whole network into several disjoint sets
- Hierarchy-Centric Community
  - Construct a hierarchical structure of communities

Our focus

Network-Centric Community Detection

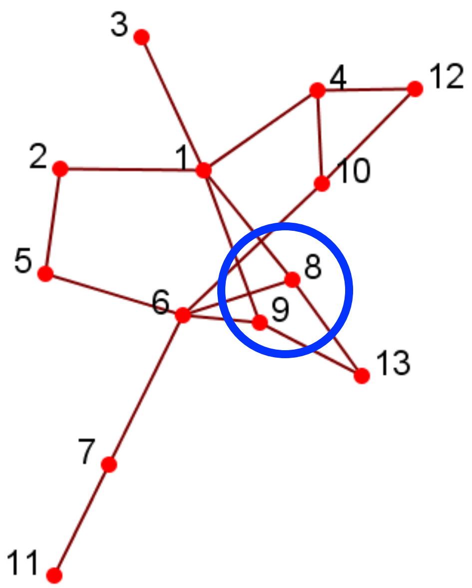
- To form a group, we need to consider the connections of the nodes globally.

- Goal: partition the network into disjoint sets

- Groups based on
  - Node Similarity
  - Latent Space Model
  - Block Model Approximation
  - Cut Minimization
  - Modularity Maximization

Node Similarity

- Node similarity is defined by how similar their interaction patterns are
- Two nodes are structurally equivalent if they connect to the same set of actors
  - e.g., nodes 8 and 9 are structurally equivalent
- Groups are defined over equivalent nodes
  - Too strict
  - Rarely occur in a large-scale
  - Relaxed equivalence class is difficult to compute
- In practice, use vector similarity
  - e.g., cosine similarity, Jaccard similarity

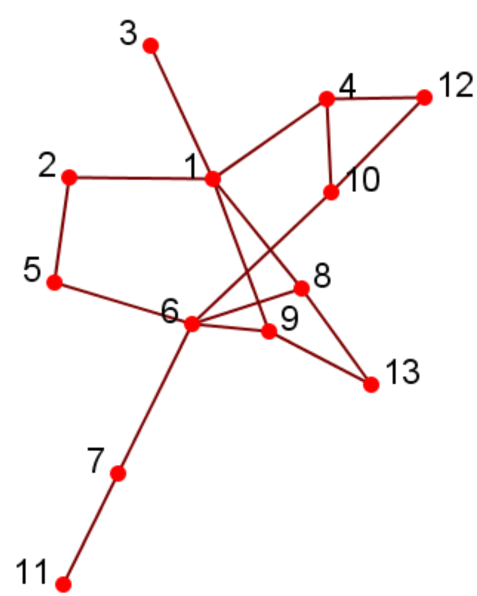


Vector Similarity Based on Adjacency Matrix

a vector → row 5

structurally equivalent: rows 8 and 9

	1	2	3	4	5	6	7	8	9	10	11	12	13
5		1				1							
8	1					1							1
9	1					1							1



Cosine Similarity:

$$\text{similarity} = \cos(\theta) = \frac{A \cdot B}{\|A\| \|B\|}.$$

$$sim(5,8) = \frac{1}{\sqrt{2} \times \sqrt{3}} = \frac{1}{\sqrt{6}}$$

Jaccard Similarity:

$$J(A,B) = \frac{|A \cap B|}{|A \cup B|}.$$

$$J(5,8) = \frac{|\{6\}|}{|\{1,2,6,13\}|} = 1/4$$

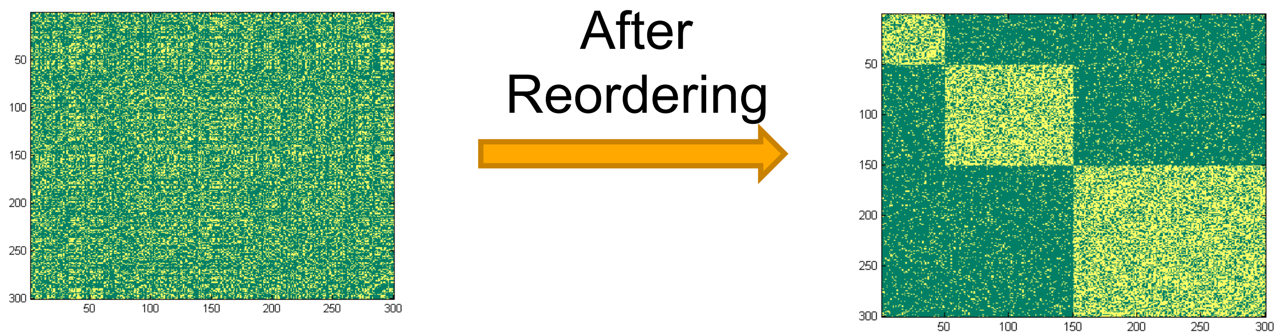
Clustering based on Node Similarity

- For practical use with huge networks:
  - Consider the connections as features
  - Use Cosine or Jaccard similarity to compute vertex similarity
  - Apply classical k-means clustering Algorithm

Algorithm 1 Basic K-means Algorithm.

```
Select K points as the initial centroids.
repeat
    Form K clusters by assigning all points to the closest centroid.
    Recompute the centroid of each cluster.
until The centroids don't change
```

Block-Model Approximation



Network Interaction Matrix Block Structure

- Objective: Minimize the difference between an interaction matrix and a block structure

S is a community indicator matrix

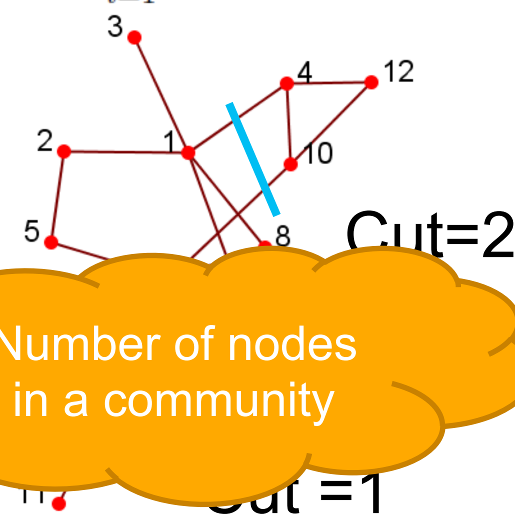
$$\min_{S,\Sigma} \|A - S\Sigma S^T\|_F$$

$$s.t. \quad S \in \{0,1\}^{n\times k}, \Sigma \in R^{k\times k} \text{ is diagonal}$$

- Challenge: S is discrete, difficult to solve
- Relaxation: Allow S to be continuous satisfying $S^T S = I_k$
- Solution: the top eigenvectors of A
- Post-Processing: Apply k-means to S to find the partition

Cut-Minimization

- Between-group interactions should be infrequent
- Cut: number of edges between two sets of nodes
- Objective: minimize the cut $cut(C_1, C_2, \cdots, C_k) = \sum_{i=1}^{k} cut(C_i, \overline{C_i})$
  - Limitations: often find communities of only one node
  - Need to consider the group size



Number of nodes in a community

- Two commonly-used variants:

$$\text{Ratio-cut}(C_1, C_2, \cdots, C_k) = \sum_{i=1}^{k} \frac{cut(C_i, \overline{C_i})}{|V_i|}$$

$$\text{Normalized-cut}(C_1, C_2, \cdots, C_k) = \sum_{i=1}^{k} \frac{cut(C_i, \overline{C_i})}{vol(V_i)}$$

Number of within-group Interactions

Graph Laplacian

- Cut-minimization can be relaxed into the following min-trace problem

$$\min_{S \in R^{n \times k}} Tr(S^T L S) \quad s.t.\ S^T S = I$$

- L is the (normalized) Graph Laplacian

$$\begin{aligned} L &= D - A \\ \text{normalized-}L &= I - D^{-1/2} A D^{-1/2} \end{aligned} \qquad D = \begin{pmatrix} d_1 & 0 & \cdots & 0 \\ 0 & d_2 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & d_n \end{pmatrix}$$

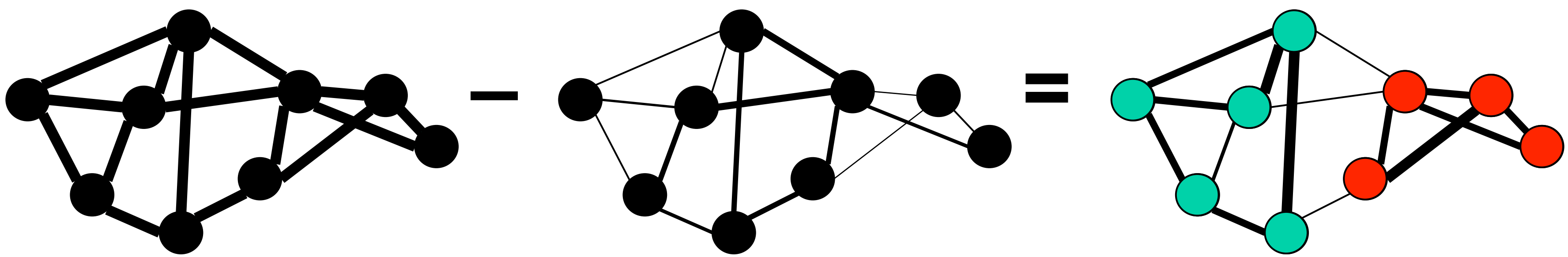
- Solution: S are the eigenvectors of L with smallest eigenvalues (except the first one)
- Post-Processing: apply k-means to S
  - a.k.a. Spectral Clustering

Graph Modularity

- Relational network given by G = (V, A)

  V : set of n vertices A : n x n adjacency matrix, m total edges

- Newman-Girvan (2006) graph modularity



Original A Null Model P Modularity (A-P)

– Measures the global community structure of G:

$$Q(C) = \frac{1}{2m} \sum_{i,j} (A_{ij} - P_{ij}) \delta(C_i, C_j) \qquad P_{ij} = \frac{d_i d_j}{2m}$$

Kronecker delta

– Foundation for a large number of methods (Fortunato, 2010)

Modularity Maximization

- Modularity measures the group interactions compared with the expected random connections in the group
- In a network with m edges, for two nodes with degree d_i and d_j, expected random connections between them are

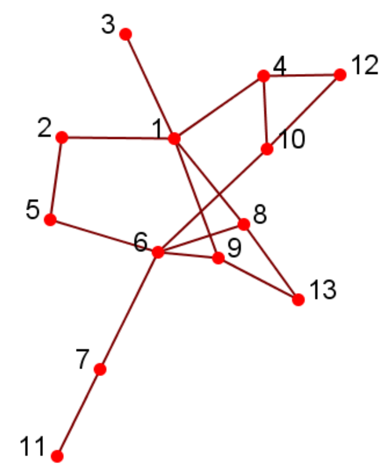
$$d_i d_j / 2m$$

- The interaction utility in a group:

$$\sum_{i \in C, j \in C} A_{ij} - d_i d_j / 2m$$

- To partition the group into multiple groups, we maximize

$$\frac{1}{2m} \sum_{C} \sum_{i \in C, j \in C} A_{ij} - d_i d_j / 2m$$



Expected Number of edges between 6 and 9 is
5*3/(2*17) = 15/34

Modularity Matrix

- The modularity maximization can also be formulated in matrix form

$$Q = \frac{1}{2m} Tr(S^T B S)$$

- B is the modularity matrix

$$B_{ij} = A_{ij} - d_i d_j / 2m$$

- Solution: top eigenvectors of the modularity matrix

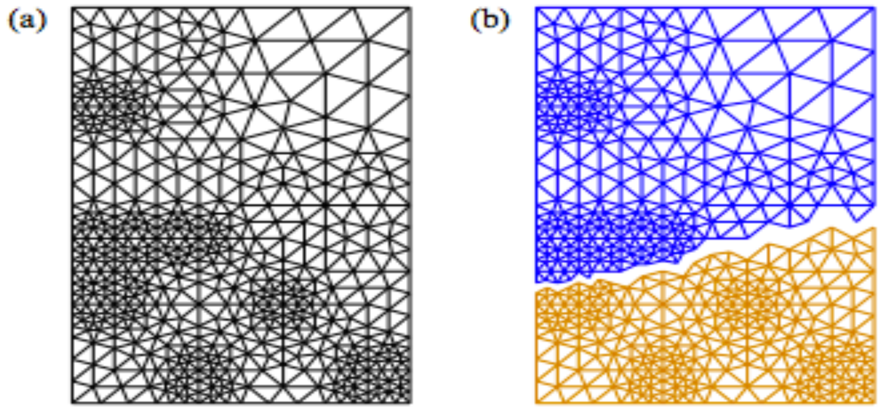
Properties of Modularity

- Properties of modularity:
  - Between (-1, 1)
  - Modularity = 0 If all nodes are clustered into one group
  - Can automatically determine optimal number of clusters

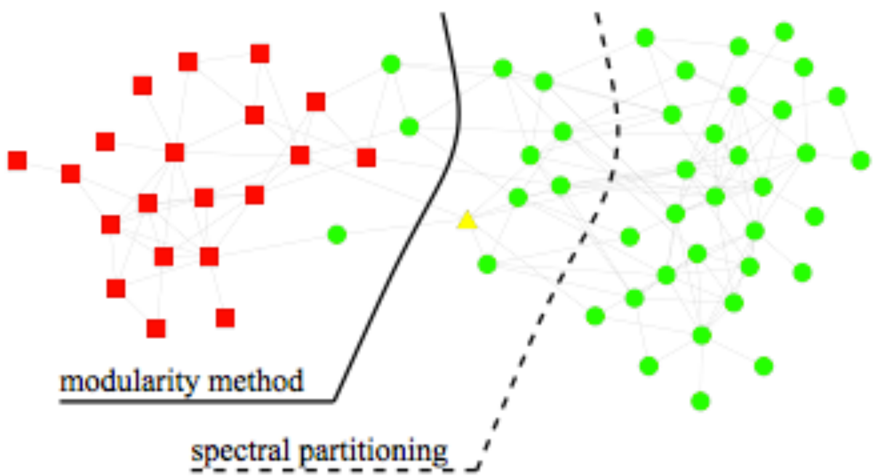
- Resolution limit of modularity
  - Modularity maximization might return a community consisting multiple small modules

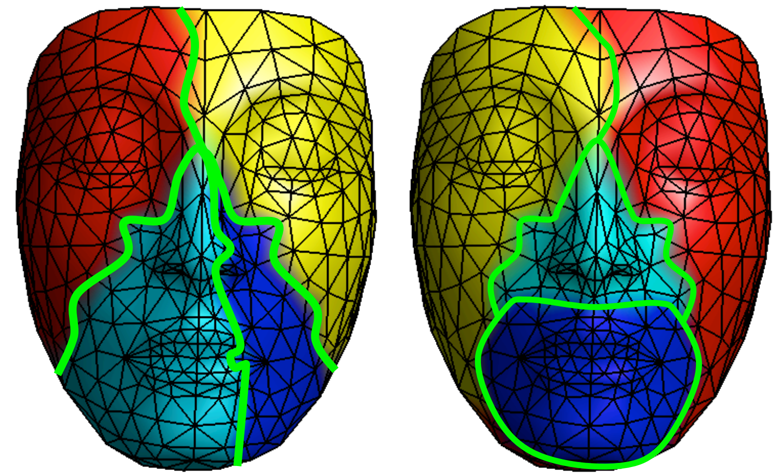
Graph Laplacian vs Graph Modularity

Mesh Network by Bern et al.
partitioned by the Laplacian

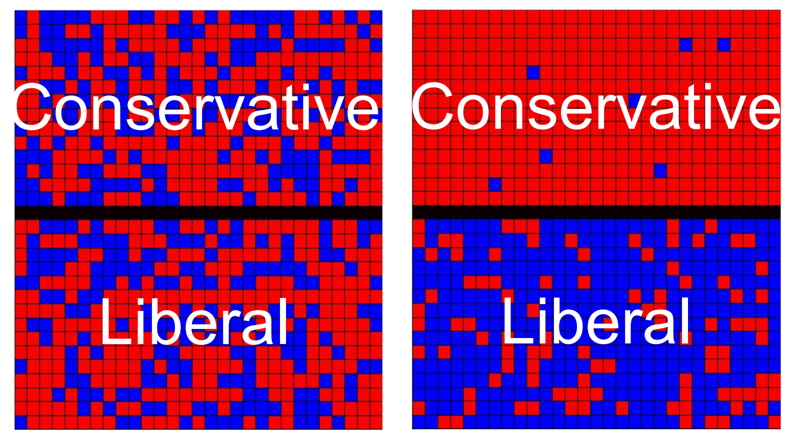


Dolphin social network





Laplacian Modularity



Political Blogs from 2004 U.S. Election,
data set from Adamic & Glance (2005)

Recap of Network-Centric Community

- Network-Centric Community Detection
  - Groups based on
    - Node Similarity
    - Latent Space Models
    - Cut Minimization
    - Block-Model Approximation
    - Modularity maximization

- Goal: Partition network nodes into several disjoint sets

- Limitation: Require the user to specify the number of communities beforehand