Examples of Graph Data

Zhen Zhang

Southern University of Science and Technology

Outlines

Some Graph Data

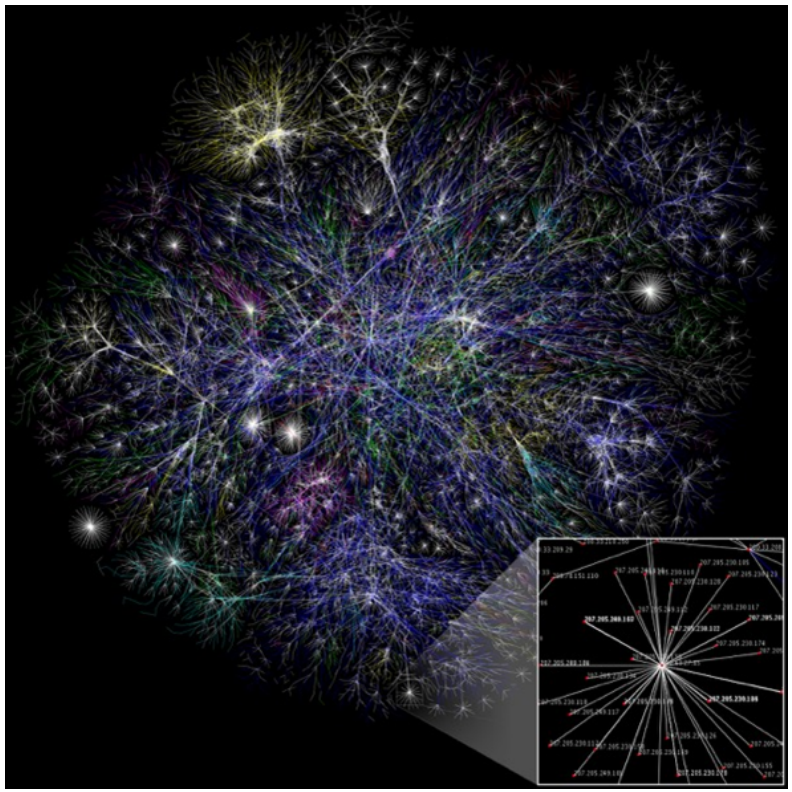


FIGURE: Graphical representation of webpage linkage

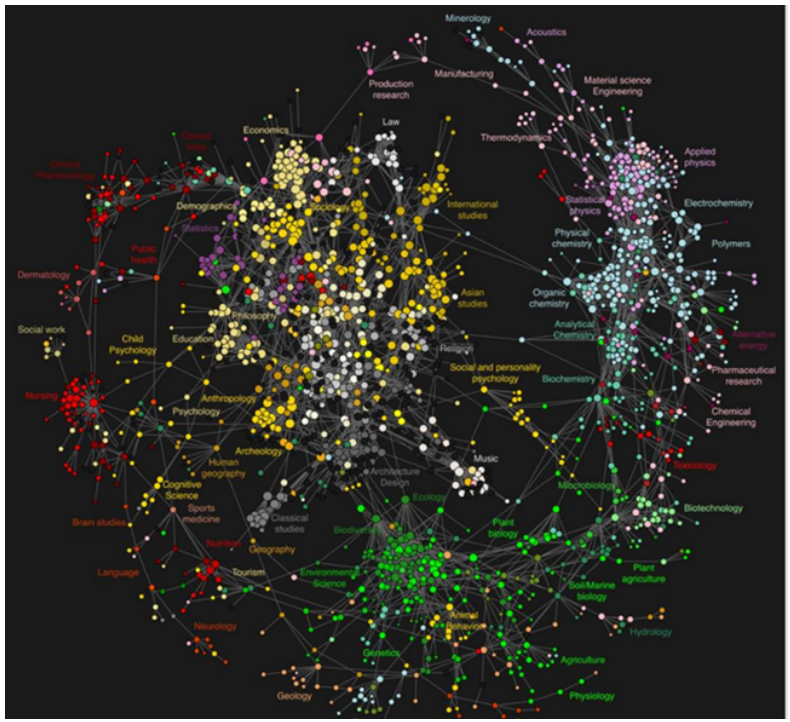


FIGURE: Graphical representation of relationships of scientific journals

Some Graph Data

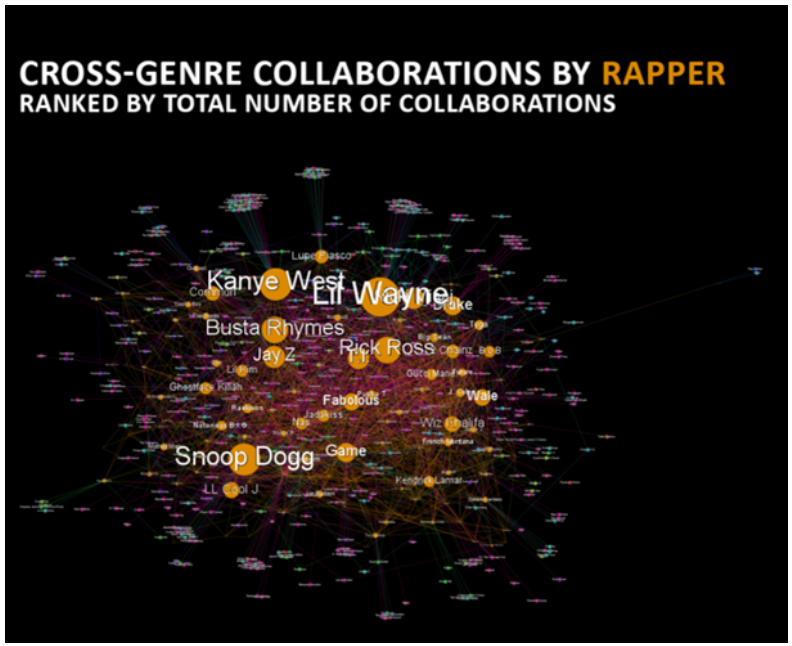


FIGURE: Network of collaborations among rappers

FIGURE: Network of US airlines

Biological Graph Data

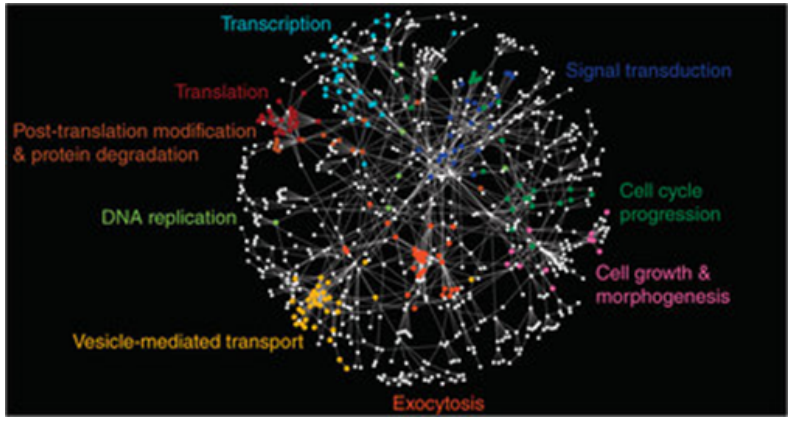


FIGURE: Gene Regulatory Network (GRN). The mRNA concentration follows a dynamic process (e.g. ODE) controlled by other related genes.

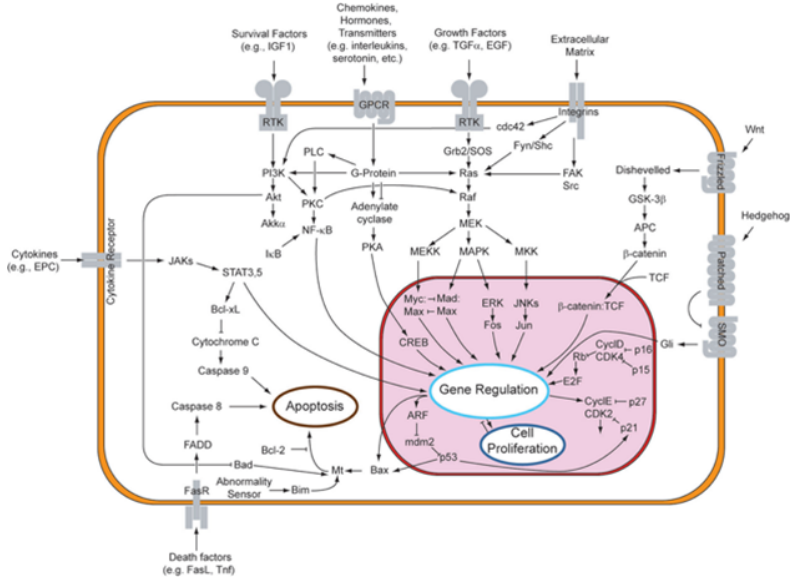


FIGURE: Cell Signal Transcriptional Network

Biological Graph Data

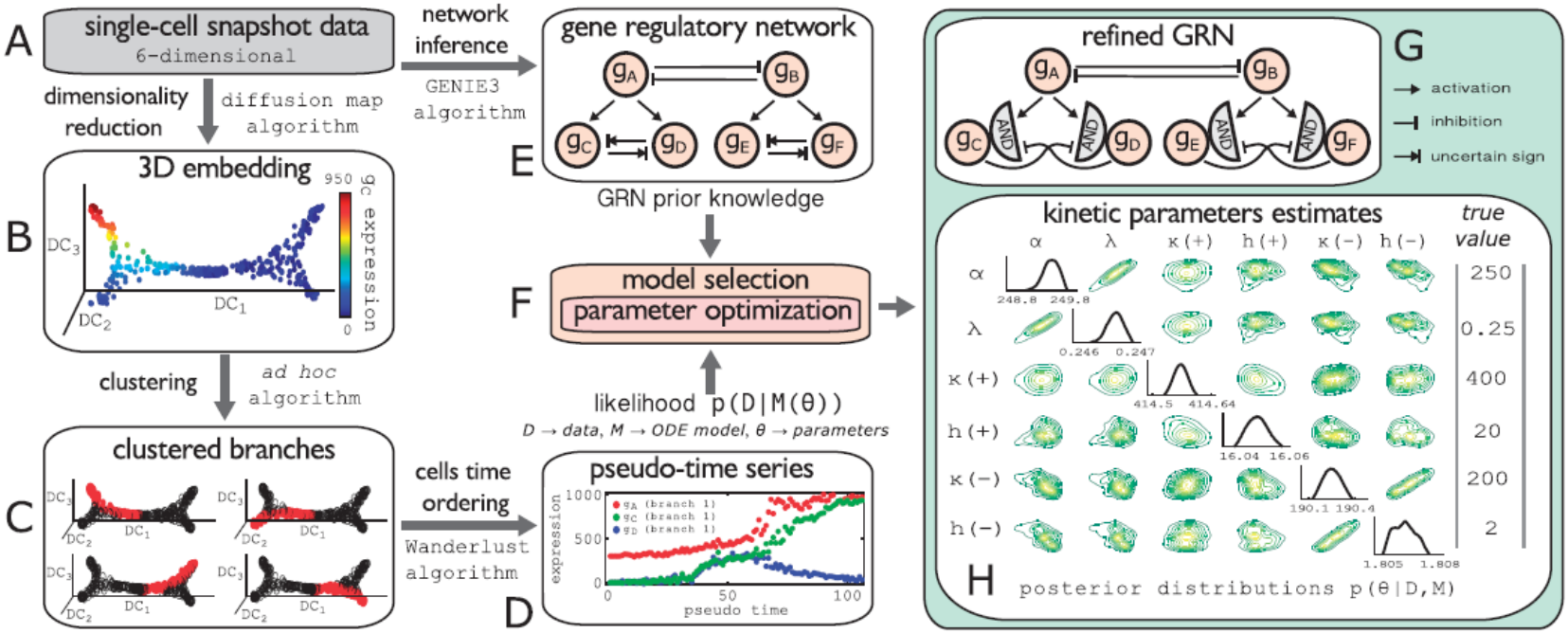
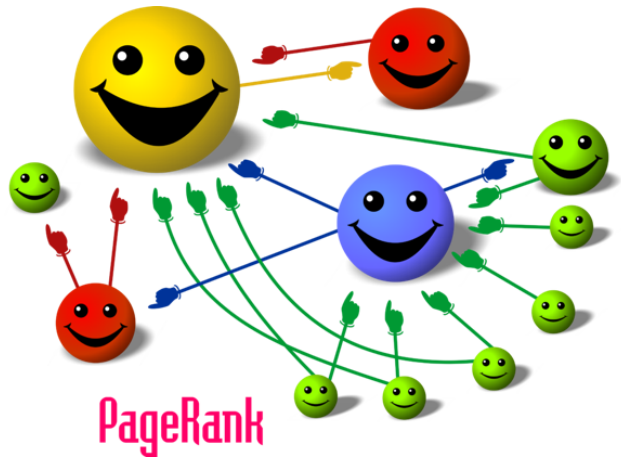


FIGURE: Framework of GRN produced from single-cell data.

Outlines

Linking Websites

- Scoring websites by counting number of links
- Rescoring (reweighting) by considering the importance of the websites



What sites link to pku.edu.cn?

Total Sites Linking In 6,649

Site	Page
1. baidu.com	bdl.baidu.com/publication.html
2. msn.com	msn.com/de-at/nachrichten/wissenundtec...
3. qq.com	edu.qq.com/bschool
4. hupu.com	bbs.hupu.com/14788328.html
5. 163.com	biz.163.com

What sites link to tsinghua.edu.cn?

Total Sites Linking In 8,579

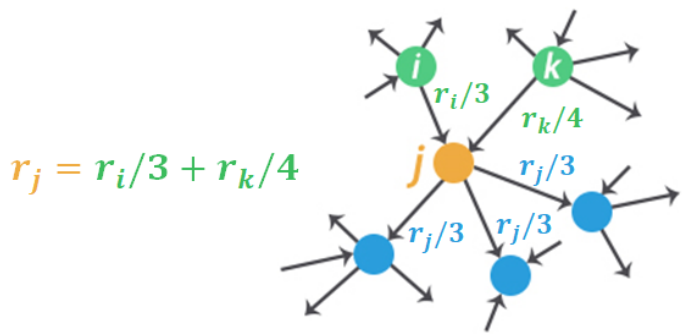
Site	Page
1. yahoo.com	travelinspirations.yahoo.com/post/:id/...
2. baidu.com	tieba.baidu.com/f?ie=utf-8&kw=清华大学
3. msn.com	msn.com/en-us/travel/tripideas/the-bes...
4. yandex.ru	ftp.yandex.ru/debian/README.mirrors.ht...
5. qq.com	city.qq.com

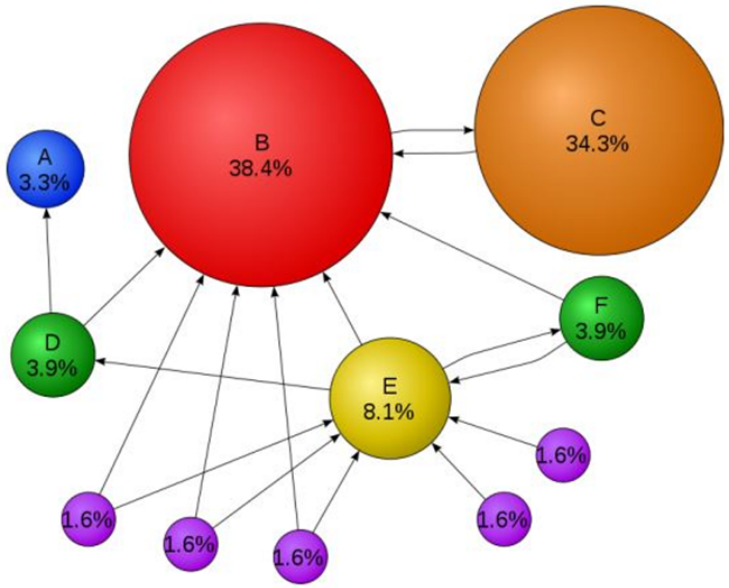
Scoring the Pages

Ranking the webpage j by computing r_j :

$$r_j = \sum_{i \to j} \frac{r_i}{d_i}, \text{(stream equation)}$$

where d_i is the out-degree of freedom of node i





Eigenvalue Problem

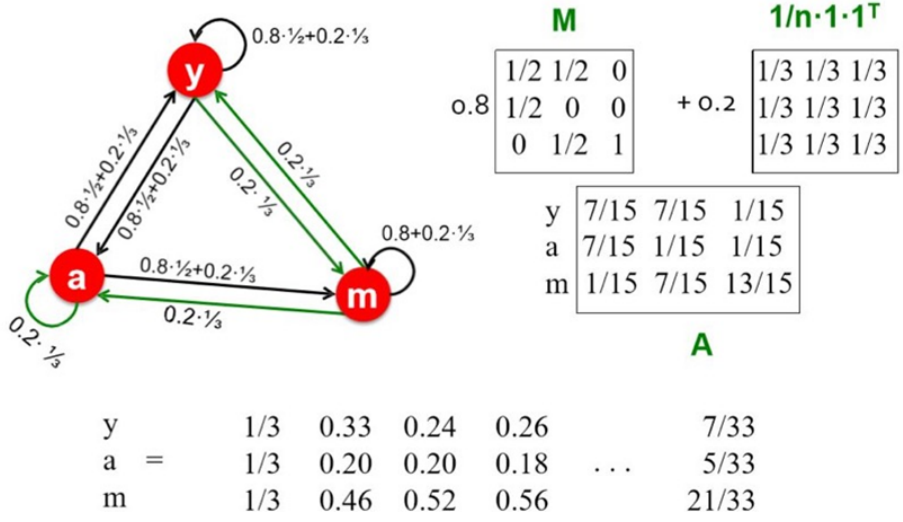
- $\mathbf{Mr} = \mathbf{r}$
- Use power method to solve for $\mathbf{r} = \lim_{t \to \infty} \mathbf{r}^{(t)}$:
  1. Initialization : $\mathbf{r}^{(0)} = (\frac{1}{N}, \ldots, \frac{1}{N})^T$
  2. Iteration : $\mathbf{r}^{(t+1)} = \mathbf{M}\mathbf{r}^{(t)}$
  3. Stopping rule : $\|\mathbf{r}^{(t+1)} - \mathbf{r}^{(t)}\| \leqslant \epsilon$.
- Random walk interpretation : $\mathbf{r}^{(t)} = (r_i^{(t)})_i$ is a probability distribution, where $r_i^{(t)}$ represents the probability that the explorer stays in the webpage i at time t ; he randomly choose the next webpage according to the probability indicated by the matrix $\mathbf{M}$.
- This produces a Markov chain. And $\mathbf{r}$ is its stationary distribution if $\mathbf{M}$ is irreducible and non-periodic by Perron-Frobenius theory.

Google PageRank

- To avoid spider traps (out-link absorbed by a small subset) and dead ends (no out-link), Google introduced the random page transition (Brin-Page,98) :

$$r_j = \sum_{i \to j} \beta \frac{r_i}{d_i} + (1-\beta)\frac{1}{N}$$

- $\mathbf{A} = \beta \mathbf{M} + (1-\beta)\frac{1}{N}\mathbf{1}\mathbf{1}^T$ is irreducible and non-periodic

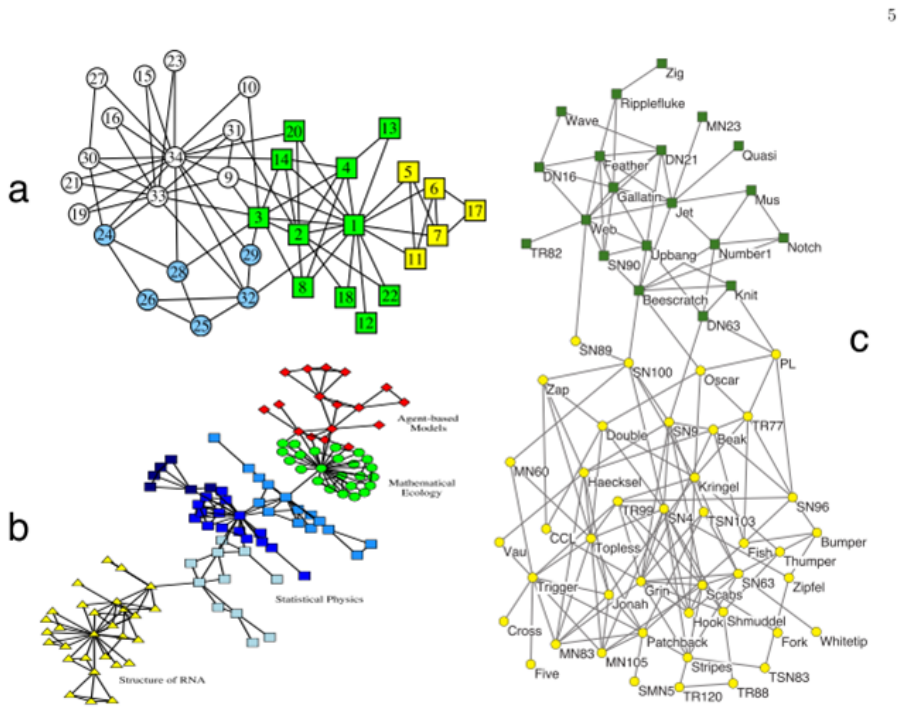


y		1/3	0.33	0.24	0.26	7/33
a	=	1/3	0.20	0.20	0.18	... 5/33
m		1/3	0.46	0.52	0.56	21/33

Outlines

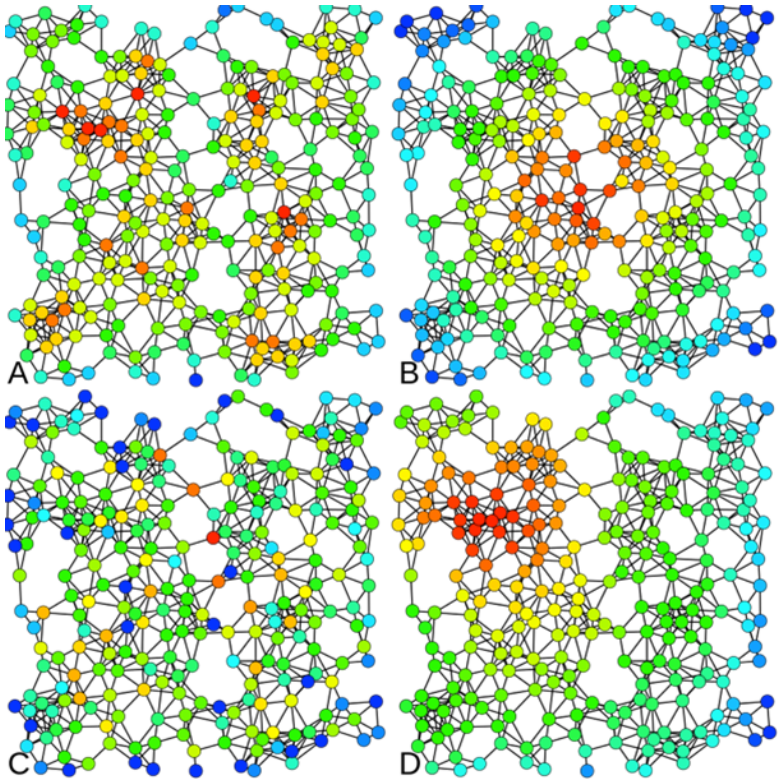
Community Detection

- Club organization from individuals (karate club)
- Collaboration network
- Social behavior of zebra



Centrality (Geometry of the Graph)

- Degree (or normalized by the total number of vertices) centrality : the number of edges linking the node
- Farness and closeness centrality, harmonic Centrality
- Betweenness centrality : the number of shortest paths passing through the current node
- Eigenvector centrality : **r** in PageRank



Community Detection Algorithms

- Hierarchical clustering based algorithms :
  - Girvan-Newman Algorithm
  - Improved by Newman's fast algorithm : A concept of "modularity" Q is introduced, agglomerate the subsets by maximizing ΔQ
- Fast Unfolding by V. D. Blondel, implemented in Gephi

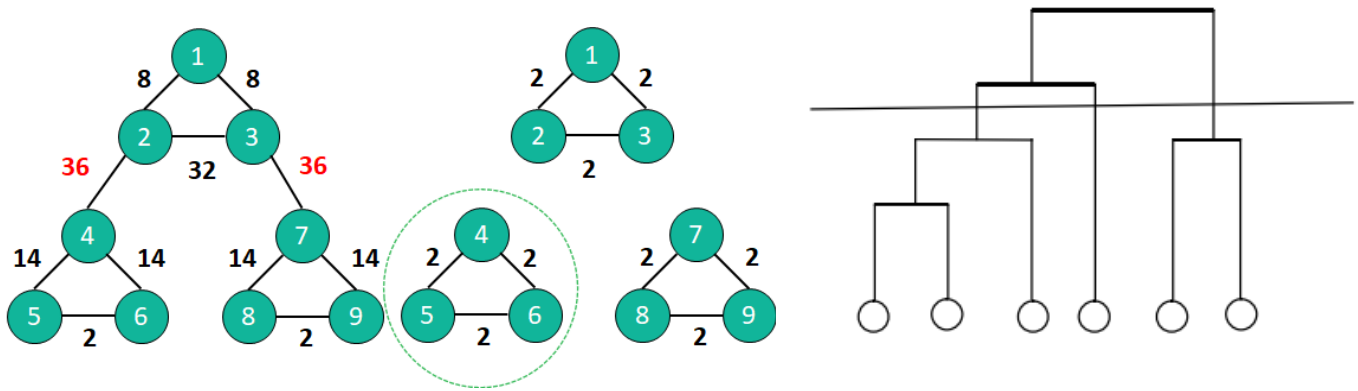
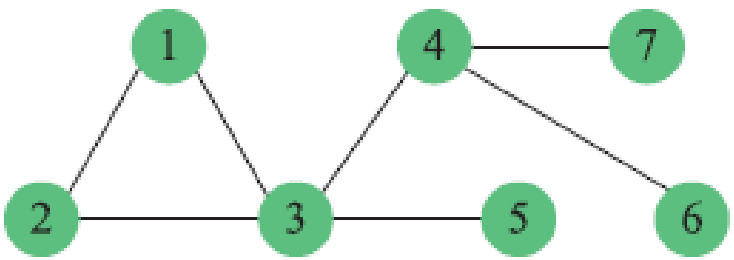


FIGURE: Left : Girvan-Newman ; Right : Newman's fast algorithm

Outlines

Graphs

- A set of data points $\{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$, similarity s_{ij} or distance d_{ij}
- Graph $G = (V, E)$, where $V = \{v_i\}_{i=1}^n$ with each v_i representing a sample $\mathbf{x}_i$
- v_i and v_j are connected ($w_{ij} > 0$) if $s_{ij} > \epsilon$ where $\epsilon \geqslant 0$ is a threshold ; then the edge is weighted by $w_{ij} = s_{ij}$
- Undirected graph $w_{ij} = w_{ji}$, adjacency matrix $W = \{w_{ij}\}$
- Degree of v_i : $d_i = \sum_{j=1}^n w_{ij}$; $D = \mathrm{diag}(d_1, \ldots, d_n)$

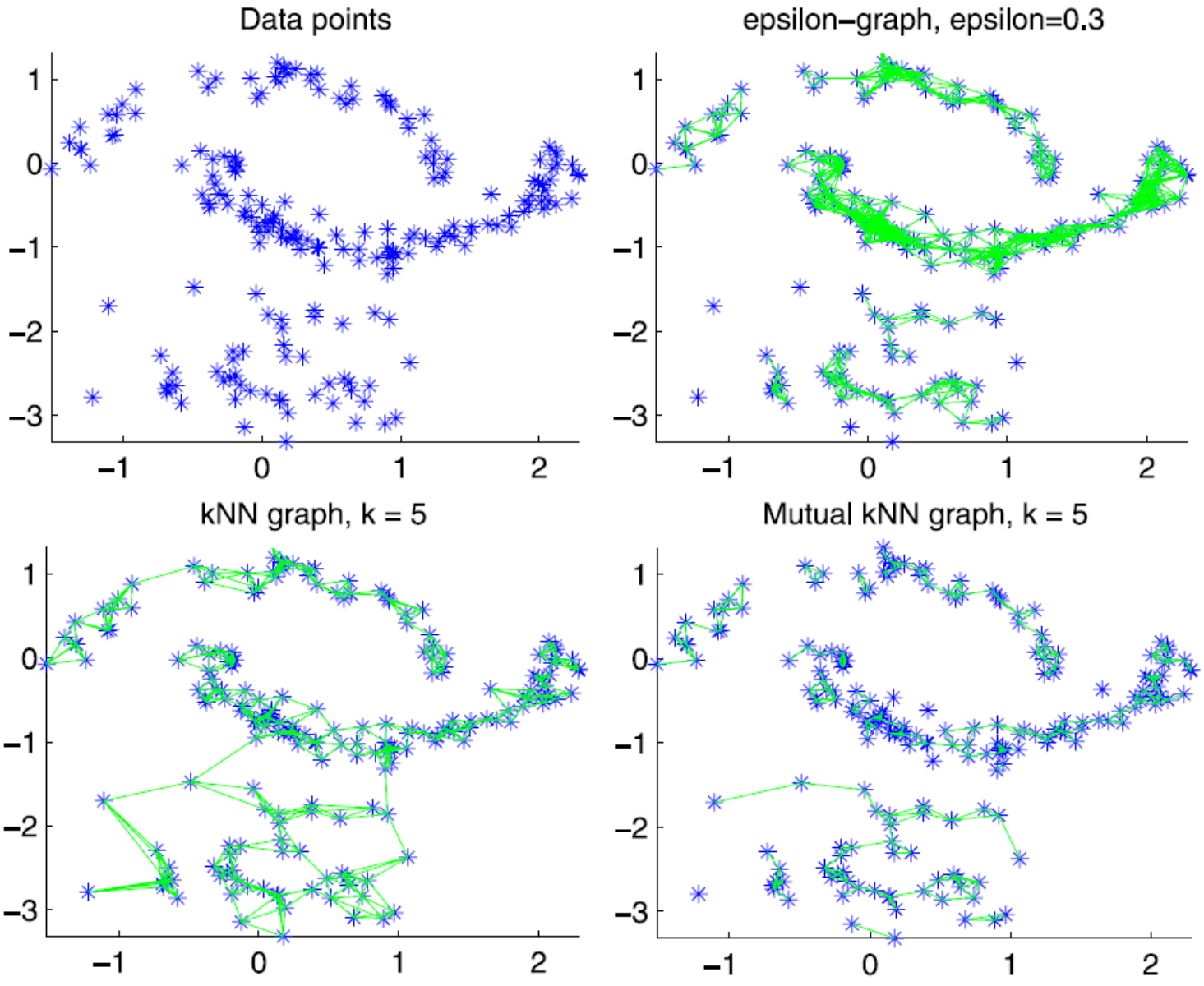


$$W = \begin{bmatrix} 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}$$

Similarity Graphs

- ϵ-neighborhood graph : v_i and v_j are connected if $d(x_i, x_j) < \epsilon$; unweighted graph ; $\epsilon \sim (\log n/n)^p$; difficult to choose ϵ for data on different scales
- k-nearest neighbor graph : connect v_i to v_j if v_j is among the k-nearest neighbors of v_i, directed graph ; connect v_i and v_j if v_i and v_j are among the k-nearest neighbors of each other, mutual k-nearest neighbor graph, undirected ; $k \sim \log n$
- Fully connected graph : connect all points with positive similarity with each other ; model local neighborhood relationships ; Gaussian similarity function $s(x_i, x_j) = \exp(-\|x_i - x_j\|^2/(2\sigma^2))$, where σ controls the width of neighborhoods ; adjacency matrix is not sparse ; $\sigma \sim \epsilon$

Similarity Graphs



Graph Laplacian

- Unnormalized graph Laplacian : $L = D - W$
  - Has $\mathbf{1}$ as an eigenvector corresponding to the eigenvalue 0
  - Symmetric and positive definite : $\mathbf{f}^T \mathbf{L}\mathbf{f} = \frac{1}{2} \sum_{i,j} w_{ij}(f_i - f_j)^2$
  - Non-negative, real-valued eigenvalues $0 = \lambda_1 \leqslant \lambda_2 \leqslant \cdots \leqslant \lambda_n$
  - The eigenspace of eigenvalue 0 is spanned by the indicator vectors $\mathbf{1}_{A_1}, \ldots, \mathbf{1}_{A_k}$, where $A_1, \ldots, A_k$ are k connected components in the graph
- Normalized graph Laplacians :
  - Symmetric Laplacian : $L_{sym} = D^{-1/2} L D^{-1/2}$
  - Random walk Laplacian : $L_{rw} = D^{-1} L$
  - Both have similar properties as L

Spectral Clustering

- Graph cut : segment G into K clusters $A_1, \ldots, A_K$, where $A_i \subset V$, this is equivalent to minimize the graph cut function

$$cut(A_1, \ldots, A_K) = \frac{1}{2} \sum_{k=1}^{K} W(A_k, \bar{A}_k)$$

  where $W(A, B) = \sum_{i \in A, j \in B} w_{ij}$. Trivial solution consists of a singleton and its complement

- RatioCut : $RatioCut(A_1, \ldots, A_K) = \frac{1}{2} \sum_{k=1}^{K} \frac{W(A_k, \bar{A}_k)}{|A_k|}$, where $|A|$ is the number of vertices in A
- Normalized cut : $Ncut(A_1, \ldots, A_K) = \frac{1}{2} \sum_{k=1}^{K} \frac{W(A_k, \bar{A}_k)}{vol(A_k)}$, where $vol(A) = \sum_{i \in A} d_i$; it is NP-hard

Relaxation of RatioCut to Eigenvalue Problems with $K = 2$

- $\min_{A \subset V} RatioCut(A, \bar{A})$
- Binary vector $f = (f_1, \ldots, f_n)^T$ as indicator function :
  $$f_i = \begin{cases} \sqrt{|\bar{A}|/|A|}, & \text{if } v_i \in A \\ -\sqrt{|A|/\bar{A}|}, & \text{if } v_i \in \bar{A} \end{cases}$$
- $f^T L f = |V| \cdot RatioCut(A, \bar{A})$, $\sum_{i=1}^{n} f_i = 0$, and $\|f\|_2^2 = n$
- Relax f to be real-valued : $\min_{f \in \mathbb{R}^n} f^T L f$, subject to $f \perp \mathbf{1}$ and $\|f\|_2 = \sqrt{n}$
- By Rayleigh-Ritz theorem, the solution f is the eigenvector corresponding to the second smallest eigenvalue of L
- Cluster $\{f_i\}_{i=1}^n$ to two groups C and $\bar{C}$: $v_i \in A$ if $f_i \in C$, and else $v_i \in \bar{A}$

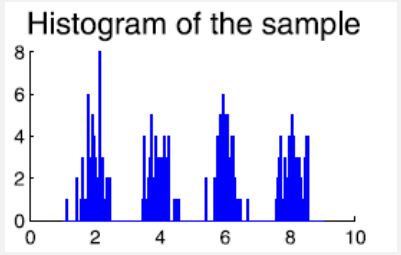
Relaxation of RatioCut and Ncut with general K

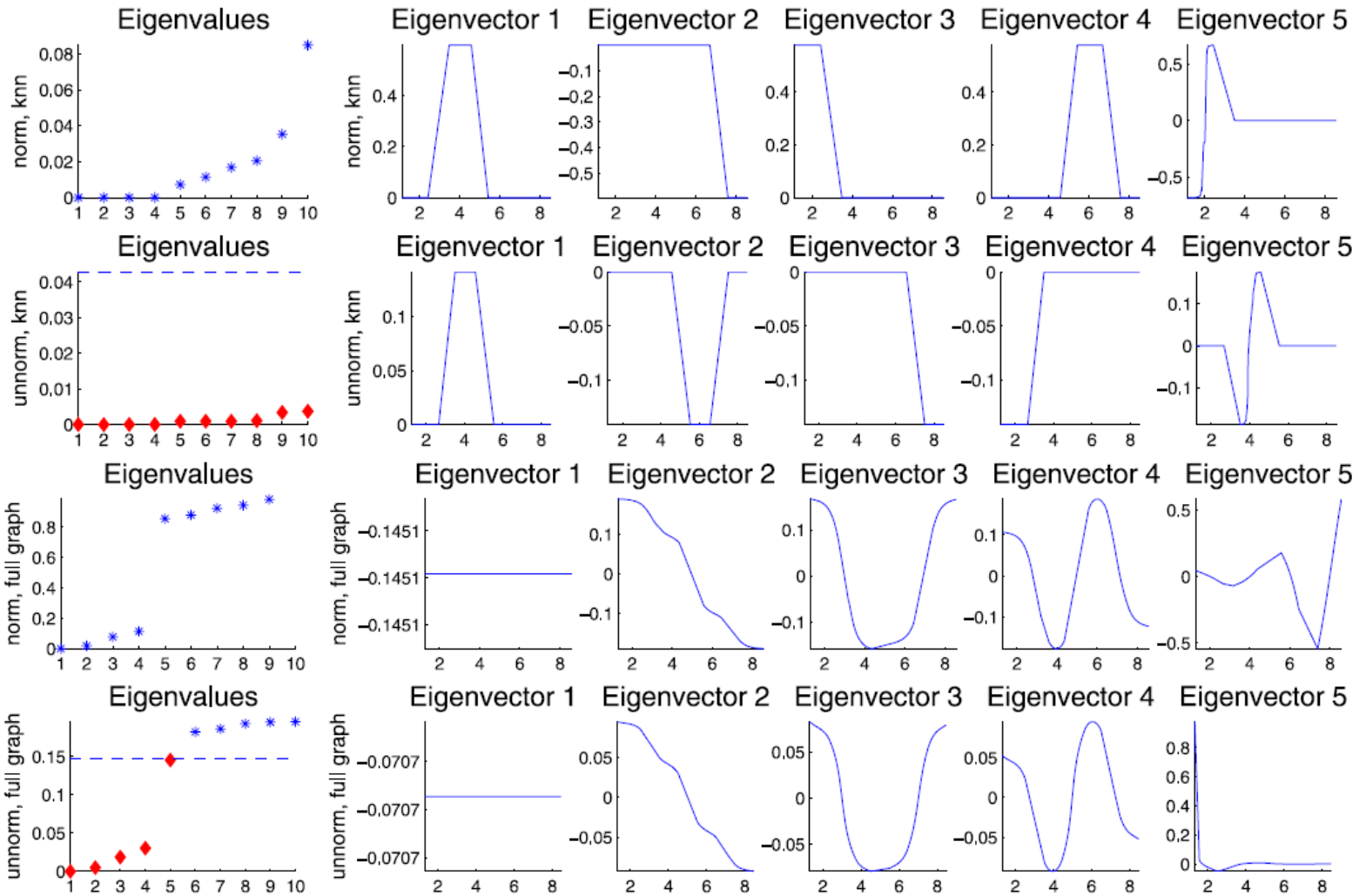
- RatioCut
  - Binary vector $h_j = (h_{1j}, \ldots, h_{nj})^T$, $j = 1, \ldots, K$, as indicator function : $h_{ij} = \begin{cases} 1/\sqrt{|A_j|}, & \text{if } v_i \in A_j \\ 0, & \text{otherwise} \end{cases}$
  - $h_j^T L h_j = Cut(A_j, \bar{A}_j)/|A_j|$, $H = (h_1, \ldots, h_K) \in \mathbb{R}^{n \times K}$, $RatioCut(A_1, \ldots, A_K) = \mathrm{Tr}(H^T L H)$, $H^T H = I$
  - Relax H : $\min_{H \in \mathbb{R}^{n \times K}} \mathrm{Tr}(H^T L H)$, subject to $H^T H = I$
  - Solution : the first K eigenvectors of L as columns
  - Cluster the rows of H to K groups
- Ncut
  - Replacing $|A_j|$ by $vol(A_j)$, the same argument for the relaxation of Ncut : $\min_{H \in \mathbb{R}^{n \times K}} \mathrm{Tr}(H^T L H)$, subject to $H^T D H = I$
  - Solution : the first K eigenvectors of L_{rw} as columns

Spectral Clustering Algorithm

- Input : Similarity matrix $S \in \mathbb{R}^{n \times n}$, number k of clusters
- Output : Clusters $A_1, \ldots, A_K$ of indices of vertices
- Algorithm :
  1. Construct a similarity graph $G = (V, E)$ with weighted adjacency matrix W
  2. Compute the unnormalized graph Laplacian L or normalized graph Laplacian L_{sym} or L_{rw}
  3. Compute the first K eigenvectors $\mathbf{U} = [\mathbf{u}_1, \ldots, \mathbf{u}_K] \in \mathbb{R}^{n \times K}$
  4. In the case of L_{sym}, normalize the rows of U to norm 1 ; for the other two cases, skip this step
  5. Let $\mathbf{y}_i \in \mathbb{R}^K$ be the i-th row of $\mathbf{U}$, use K-means to cluster the point set $\{\mathbf{y}_i\}_{i=1}^n$ into clusters $C_1, \ldots, C_K$
  6. $A_k = \{i | y_i \in C_k\}$

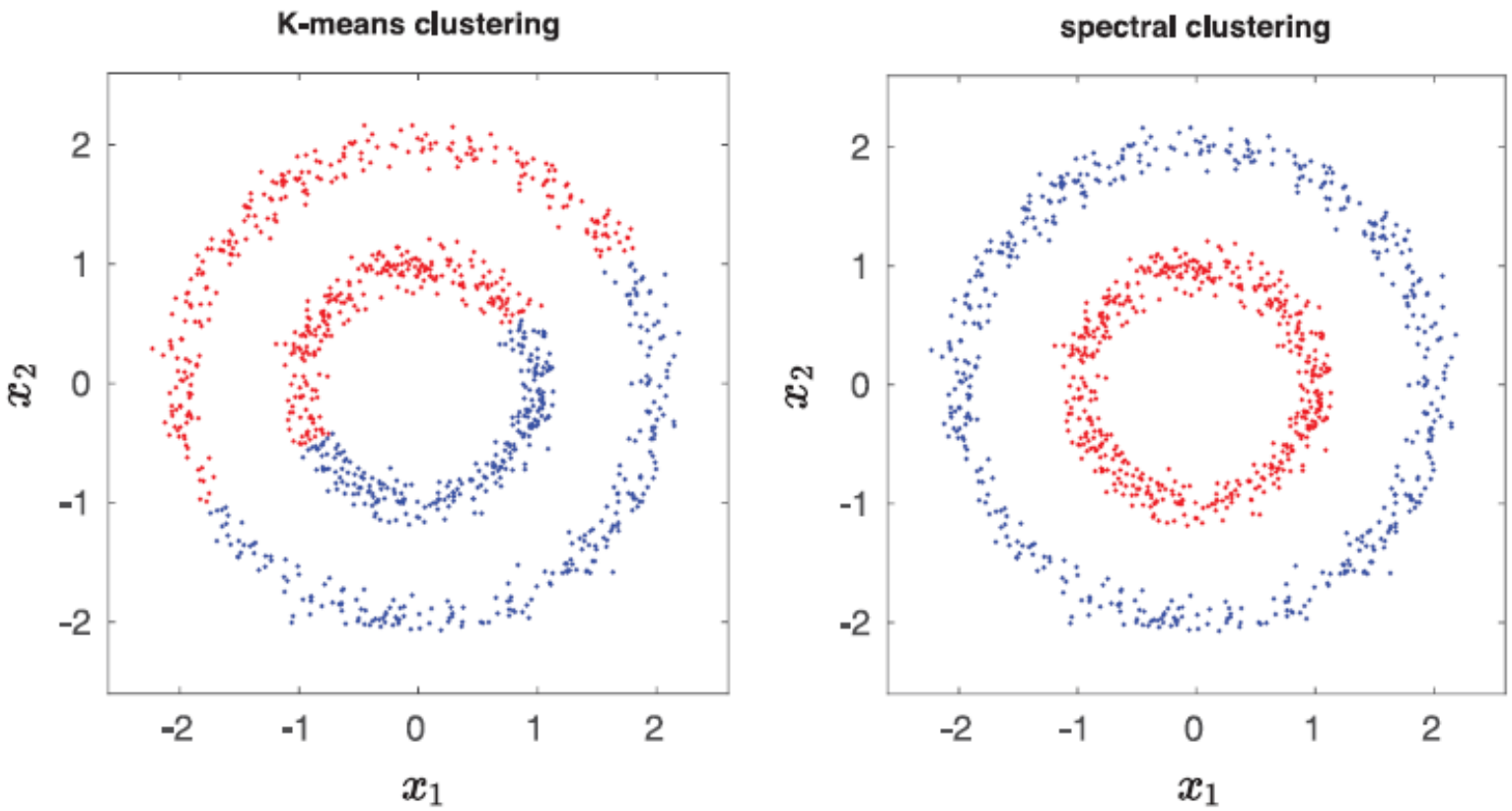
Mixture of 4 Gaussians on $\mathbb{R}$:





Interpretations

- Usually better than K-means



Random Walks Point of View

- $P = D^{-1}W$ can be interpreted as transition matrix of a Markovian random walk, which possesses a unique stationary distribution if the graph is connected and non-bipartite.
- $L_{rw} = I - P \Rightarrow \lambda(L_{rw}) = 1 - \lambda(P)$
- A probability viewpoint of Ncut : for a random walk $(X_t)_t$ starting with X_0 in the stationary distribution,

$$Ncut(A, \bar{A}) = P(X_1 \in \bar{A} | X_0 \in A) + P(X_1 \in A | X_0 \in \bar{A}).$$

  Minimizer of Ncut gives a segmentation of the graph such that a random walk seldom transitions between A and $\bar{A}$
- Commute distance : c_{ij} measures the expected time it takes the random walk to travel from vertex i to vertex j and back. Some better properties than shortest path (geodesics). A nice formula :

$$c_{ij} = vol(V)(e_i - e_j)^T L^{\dagger}(e_i - e_j)$$

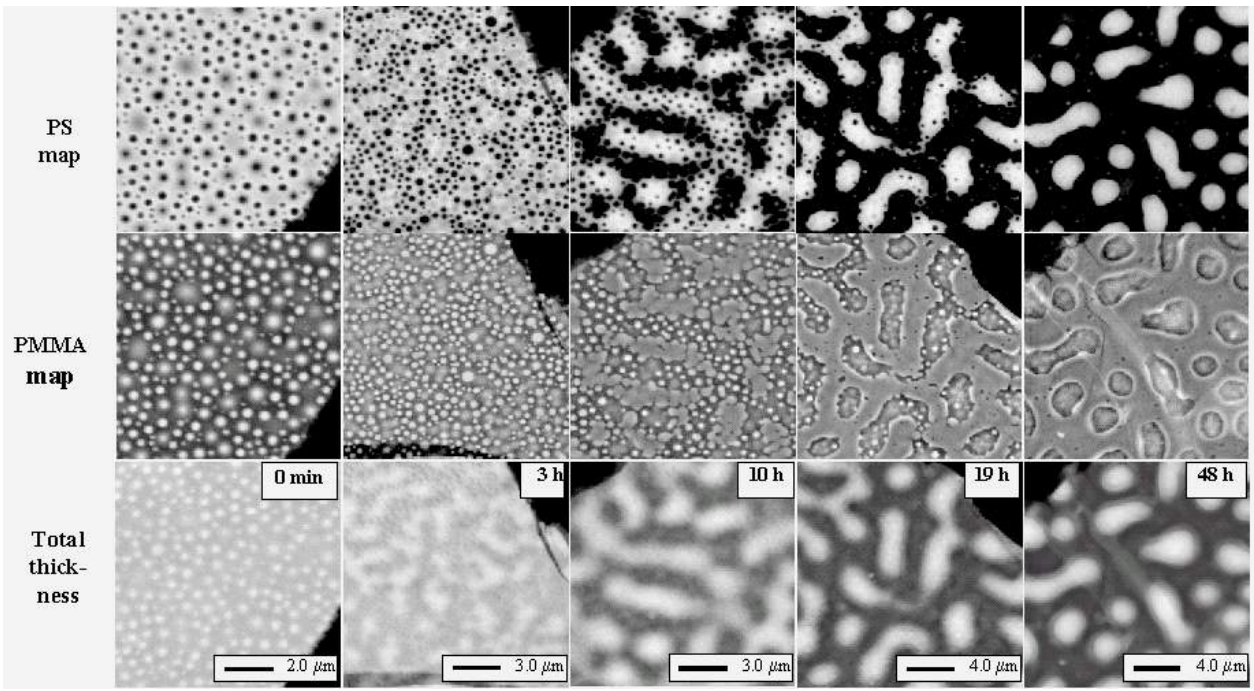
  where $L^{\dagger}$ is psudo-inverse of L.

Ginzburg-Landau Segmentation

- Ginzburg-Landau functional :

$$GL(u) = \frac{\epsilon}{2}\int |\nabla u|^2 \mathrm{d}x + \frac{1}{\epsilon}\int W(u)\mathrm{d}x,$$

where $W(u) = \frac{1}{4}(u^2-1)^2$ is a double well potential. This is used to model superconductivity, two-phase flows, etc. The minimizer naturally separates the "+1" phase from the "-1" phase.



Ginzburg-Landau Gradient Flow

- Gamma convergence : $GL(u) \to_\Gamma C|u|_{TV}$, widely used in image segmentation.
- Minimizing $E(u) = GL(u) + \lambda F(u, u_0)$ (F is data fidelity) is usually driven by a gradient flow :

$$u_t = \epsilon \Delta u - \frac{1}{\epsilon} W'(u) - \lambda \frac{\delta F}{\delta u}.$$

- Numerical PDE solver by convex splitting of $E(u) = E_{convex} - E_{concave}$:

$$\frac{u^{n+1} - u^n}{\Delta t} = -\frac{\delta E_{convex}}{\delta u}(u^{n+1}) + \frac{\delta E_{concave}}{\delta u}(u^n)$$

- Due to the Laplace operator (diagonalizable by Fourier transform), this can be solved very efficiently using FFT and iterated in spectral space

Ginzburg-Landau Segmentation on Graphs

- Bertozzi and Flenner introduced modified GL functional on graph $G = (V, E)$:

$$E(u) = \frac{\epsilon}{2} < u, Lu > + \frac{1}{4\epsilon} \sum_{z \in V} (u^2(z) - 1)^2 + \sum_{z \in V} \frac{\lambda(z)}{2} (u(z) - u_0(z))^2,$$

where u is the labeling function

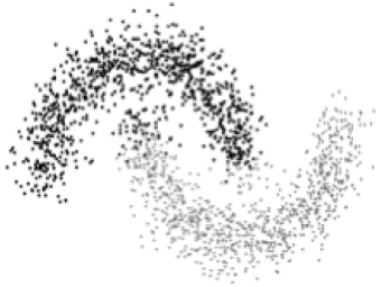
Convex Splitting for the Graph Laplacian

1. Input $\leftarrow$ an initial function u_0 and the eigenvalue-eigenvector pairs $(\tilde{\lambda}_k, \phi_k(x))$ for the graph Laplacian L_s from (2.6).
2. Set convexity parameter c and interface scale ϵ from (3.2).
3. Set the time step dt.
4. Initialize $a_k^{(0)} = \int u(x)\phi_k(x)\, dx$.
5. Initialize $b_k^{(0)} = \int [u_0(x)]^3 \phi_k(x)\, dx$.
6. Initialize $d_k^{(0)} = 0$.
7. Calculate $\mathcal{D}_k = 1 + dt\ (\epsilon\ \tilde{\lambda}_k + c)$.
8. For n less than a set number of iterations M
   (a) $a_k^{(n+1)} = \mathcal{D}_k^{-1} \left[\left(1 + \frac{dt}{\epsilon} + c\, dt\right)\ a_k^{(n)} - \frac{dt}{\epsilon} b_k^{(n)} - dt d_k^{(n)} \right]$,
   (b) $u^{(n+1)}(x) = \sum_k a_k^{(n+1)} \phi_k(x)$,
   (c) $b_k^{(n+1)} = \int [u^{(n+1)}(x)]^3 \phi_k(x)\, dx$,
   (d) $d_k^{(n+1)} = \int \lambda(x)\, (u^{(n+1)}(x) - u_0(x))\, \phi_k(x)\, dx$.
9. end for
10. Output $\leftarrow$ the function $u^{(M)}(x)$.

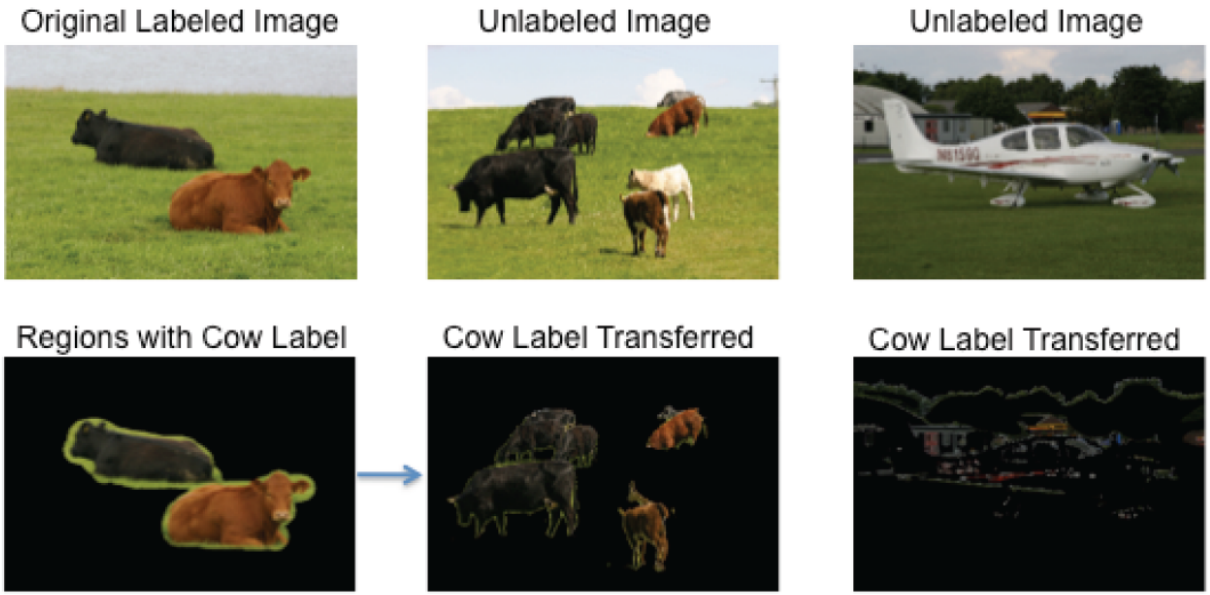
Ginzburg-Landau Segmentation on Two-Moon Data

$\epsilon = 10$

$\epsilon = 2.6$



$\epsilon = 2$



References

- A. Ocone, L. Haghverdi, N. S. Mueller, and F. J. Theis, "Reconstructing gene regulatory dynamics from high-dimensional single-cell snapshot data," Bioinformatics 31 : 89-96, 2015.
- U. von Luxburg, "A tutorial on spectral clustering," Stat. Comput. 17 :395-416, 2007.
- A. L. Bertozzi and A. Flenner, "Diffuse Interface Models on Graphs for Classification of High Dimensional Data," SIAM Review 58(2) :293-328, 2016.
- M. Schaub, A. R. Benson, P. Horn, G. Lippner, and A. Jadbabaie, "Random Walks on Simplicial Complexes and the Normalized Hodge I-Laplacian," SIAM Review 62(2) :353 – 391, 2020.