CPSC 340: Machine Learning and Data Mining

Spectral Clustering

Fall 2015

Admin

- Assignment 5 due Friday:
  - For ISOMAP, graph should be undirected/symmetric.
  - Include i-j if ‘i’ is a neighbour of ‘j’ or ‘j’ is a neighbour of ‘i’.
- Fill out course evaluations online.
- Assignment 6 out soon:
  - 2 questions: discrete loss functions and graph-based SSL.
  - Due Friday of next week.
- Practice final coming next week.

Last Time: Ranking

- In ranking, input is a set of objects (and possibly a query).
- We discussed supervised ranking:
  - Given item relevance, formulate as regression or ordinal regression.

$$\underset{w \in \mathbb{R}^d}{\operatorname{argmin}} \sum_{(i,j) \in R} -\log(p(y_{ij} \mid w, x_{ij}))$$

ratings we have

If prob is Gaussian: $\sum_{(i,j) \in \text{ratings}} \frac{1}{2}(y_{ij} - w^T x_{ij})^2$

x_{ij}: features for object 'i' with query 'j'

  - Given pairwise preferences, define loss by probability ratios.

$$\underset{w \in \mathbb{R}^d}{\operatorname{argmin}} \sum_{(i,j,k) \in R} \max\{0, 1 - \log(p(y_{ik} \mid w, x_{ik})) + \log(p(y_{jk} \mid w, x_{jk}))\}$$

$$-\log\left(\frac{p(y_{ik} \mid w, x_{ik})}{p(y_{jk} \mid w, x_{jk})}\right)$$

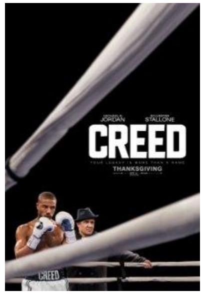
preference for 'i' over 'j' when query is 'k'.

If prob is softmax: $\underset{w \in \mathbb{R}^d}{\operatorname{argmin}} \sum_{(i,j,k) \in R} \max\{0, 1 - w^T x_{ik} + w^T x_{jk}\}$

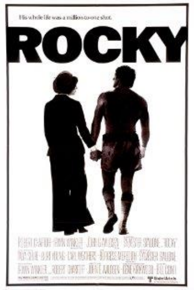
you could smooth max, then apply gradient descent. (convex)

Ranking: Beyond Pairwise Preferences

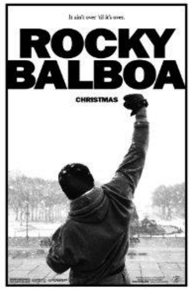
IMDB ranking



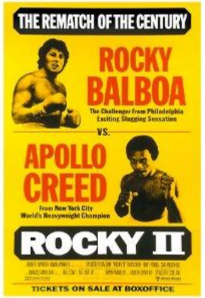
>



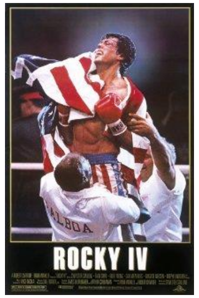
>



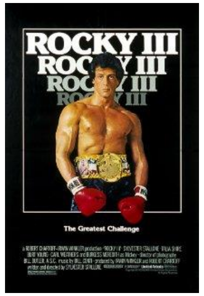
>



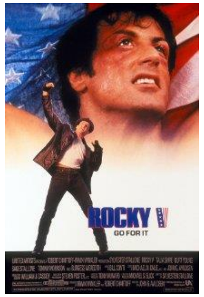
>



≥



>



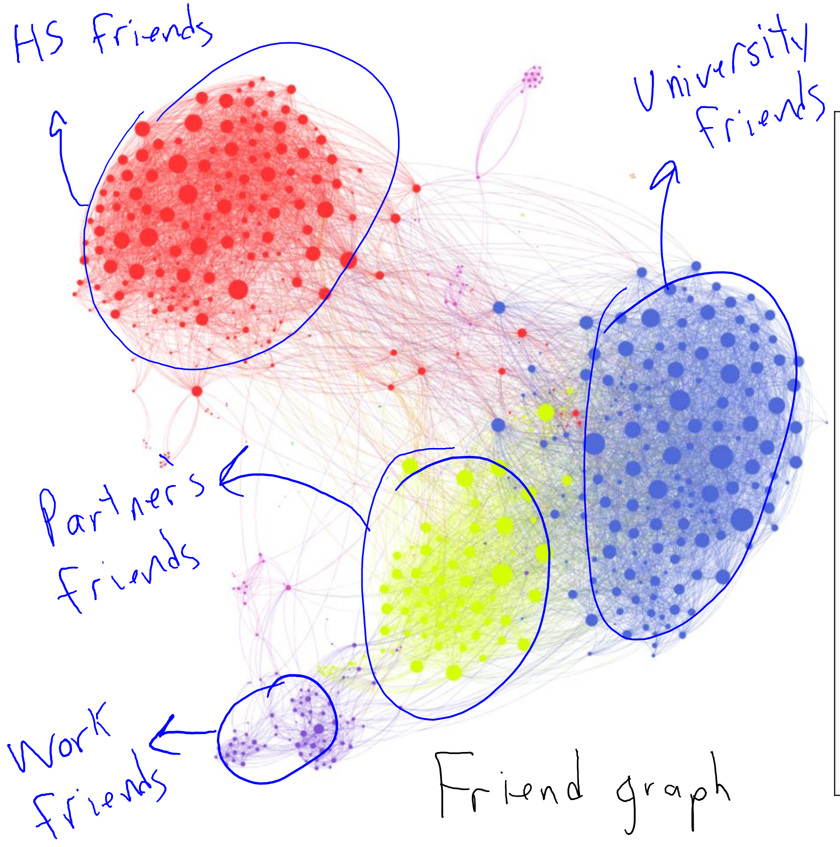
- Modern ranking methods are more advanced:
  - Take into account that you often only care about top rankings.
  - Define losses that are not additive across ratings.
    - "Precision at k": if we return k documents, how many are relevant?
    - "Average precision": precision at k averaged across values of 'k'.
  - You can still define losses based on probability ratios:
    - But you get exponential number of terms, need more advanced optimization tricks.
  - Also work on diversity of rankings:
    - E.g., divide objects into sub-topics and do weighted 'covering' of topics.

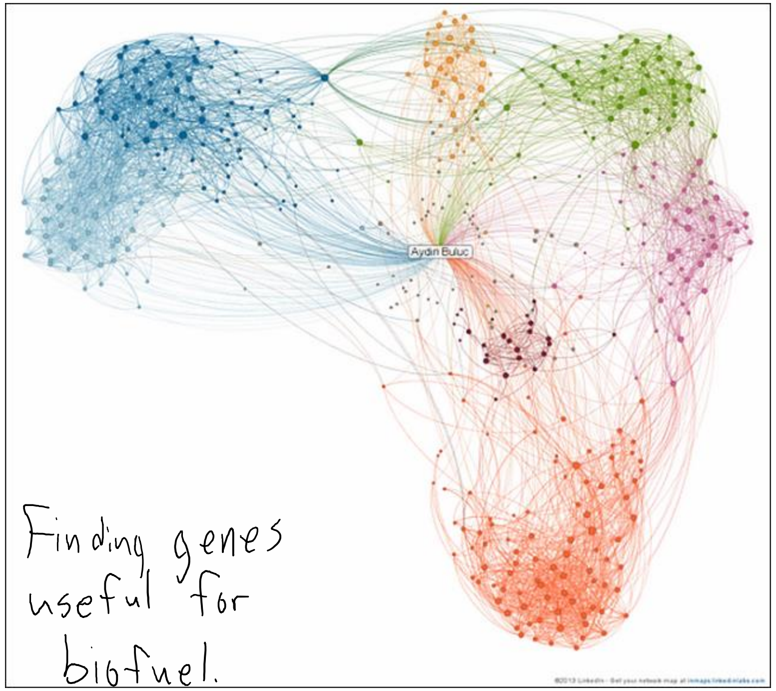
Unsupervised Graph-Based Ranking

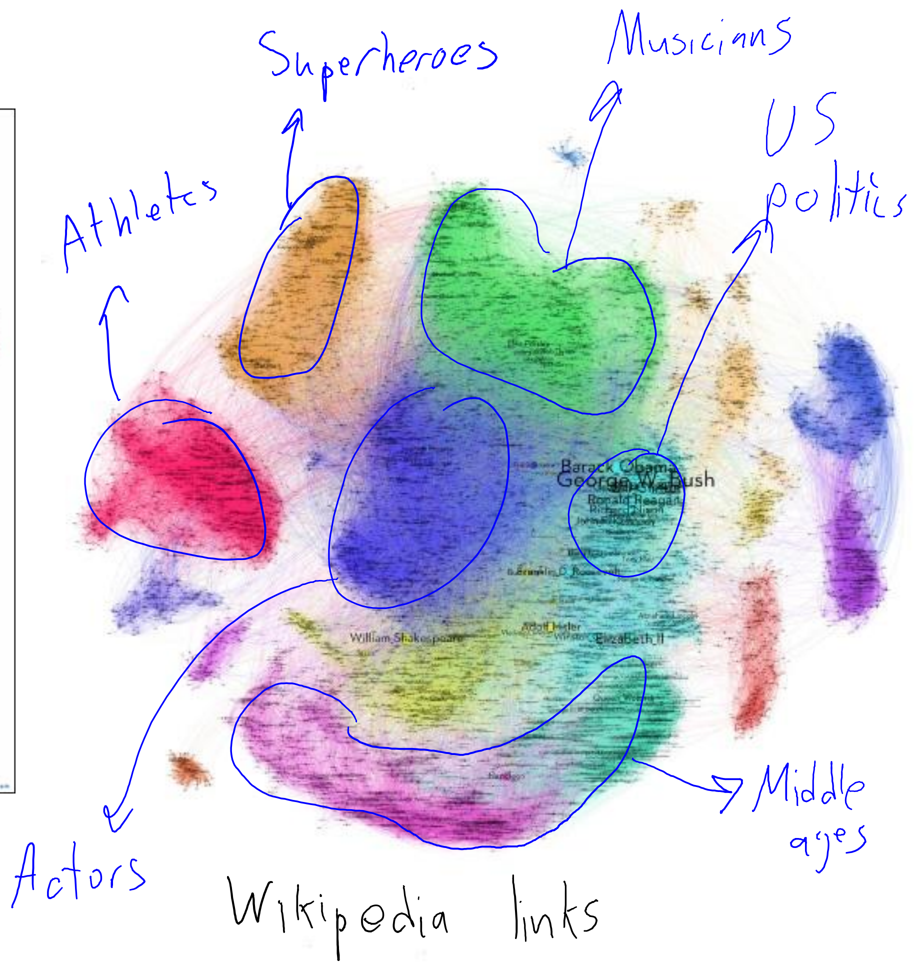
- PageRank algorithm is graph-based unsupervised ranking.
  - Important pages are linked to by many pages.
  - Link is more meaningful if a page has few links.
- 'Random surfer' view of PageRank algorithm:
  - At time 0, start out at a random webpage.
  - At time $t > 0$:
    - With probability 'p', follow a random link from page at time (t-1).
    - With probability (1-p), go to a random webpage ('damping').
- PageRank: probability of random surfer landing on page at $t = \infty$.
  - Interpretation as 'Markov chain' (links on webpage, discussed next week).
  - Can be solved via SVD, or at large scale using 'power method'.

Today: Clustering on Graphs

- Consider the problem of clustering data represented as a graph.

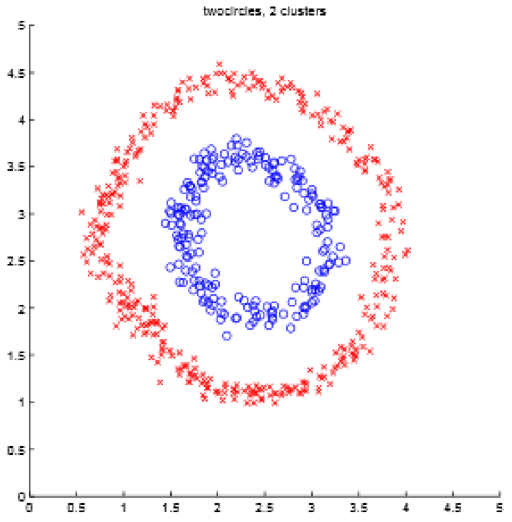






https://griffsgraphs.wordpress.com/tag/clustering/
http://ascr-discovery.science.doe.gov/2013/09/sifting-genomes/
https://www.hackdiary.com/2012/04/05/extracting-a-social-graph-from-wikipedia-people-pages/

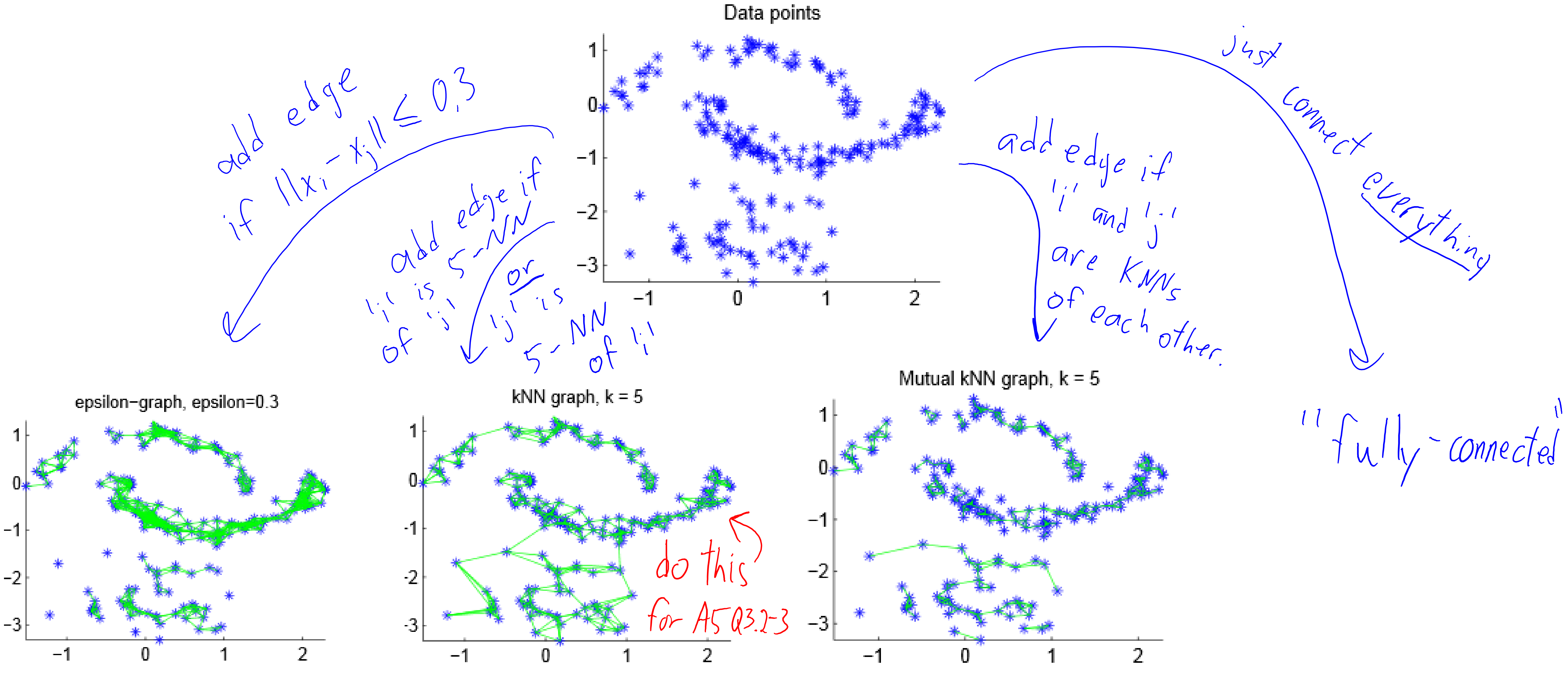
Today: Clustering on Graphs



- Setting 1:
  - We have explicit features ‘x_i’ for each object ‘i’.
  - Want to detect non-convex clusters.
- Setting 2:
  - We don’t have explicit features.
  - We have graph of links between objects (links, friends, hybridization, etc.)
  - Graph is undirected and could be weighted.
- We can convert from Setting 1 to Setting 2:
  - Taking all points within radius, KNN graph, etc.

http://ai.stanford.edu/~ang/papers/nips01-spectral.pdf

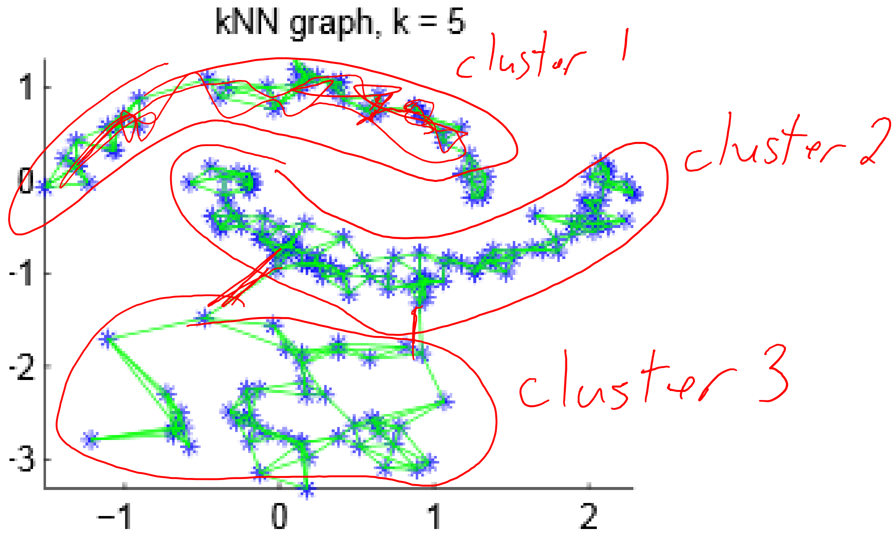
Converting from Features to Graph



http://www.kyb.mpg.de/fileadmin/user_upload/files/publications/attachments/Luxburg07_tutorial_4488%5B0%5D.pdf

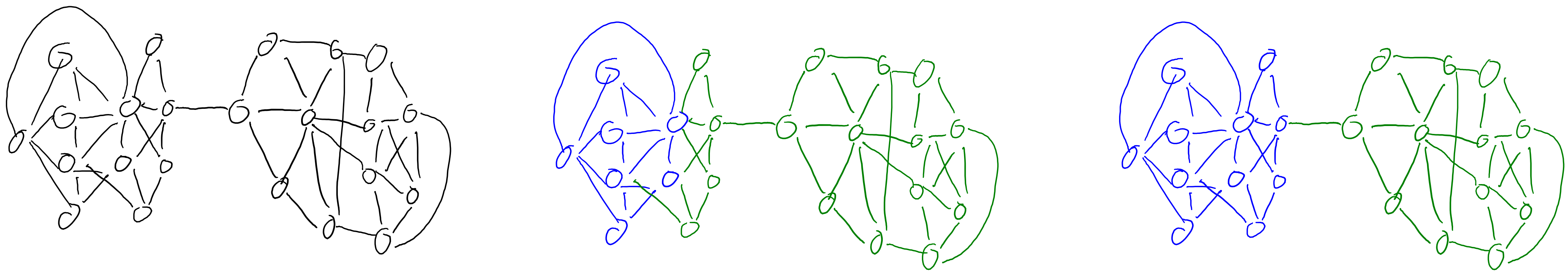
Spectral Clustering

- Most common method for data on graphs: spectral clustering.
- As with ranking, we focus on a variation based on random walks.



- Key property we want a 'good' clustering to satisfy:
  - If we start in cluster 'c', random walk should stay in cluster 'c'.

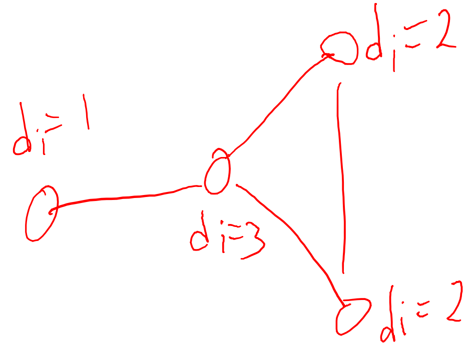
Random Walks on Graphs



Original graph　　Bad clustering　　Good clustering

- Graph-based algorithms in terms of random walks:
  - PageRank: how often does a long random walk land no node?
  - Spectral clustering: which groups of nodes does random walk stay in?
  - Graph-based SSL: which label is mostly likely to be visited first?

Biased Random Walks



- Unbiased random walk:
  - Move to a random neighbor, with each one getting equal probability.

$$p(j \leftarrow i) = \frac{1}{d_i}$$

number of neighbours of 'degree' of node 'i'

- What if we have edge weights 'w_{ij}'?
  - In spectral clustering, we want edge weights to be measure of similarity.
    - Edge weights must be non-negative.
    - High edge weight means we prefer nodes to be in the same cluster.
- With edge weights, use biased random walk:

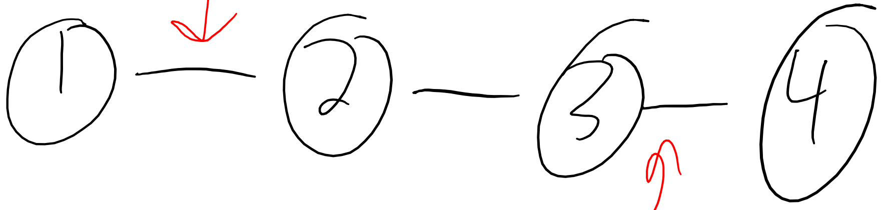
$$p(j \leftarrow i) = \frac{w_{ij}}{\sum_{(i,k) \in E} w_{ik}}$$

"weighted" degree
(normal degree is special case where all $w_{ij}=1$)

Adjacency Matrix

- Define matrix ‘A’, where ‘Aij’ when there is edge between ‘i’ and ‘j’.

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



- Powers of adjacency give number of paths of degree length:

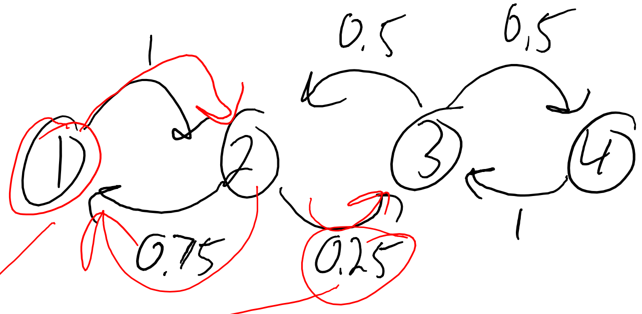
$$A^2 = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 2 & 0 & 1 \\ 1 & 0 & 2 & 0 \\ 0 & 1 & 0 & 1 \end{bmatrix}$$

$$A^3 = \begin{bmatrix} 0 & 2 & 0 & 1 \\ 2 & 0 & 3 & 0 \\ 0 & 3 & 0 & 2 \\ 1 & 0 & 2 & 0 \end{bmatrix}$$

Transition Matrix

- Define a matrix ‘P’, where ‘P_{ij}’ is probability of going from ‘i’ to ‘j’.

$$P = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0.75 & 0 & 0.25 & 0 \\ 0 & 0.5 & 0 & 0.5 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$



- Now powers give probability of landing in ‘j’ if you start at ‘i’.

$$P^2 = \begin{bmatrix} 0.75 & 0 & 0.25 & 0 \\ 0 & 0.875 & 0 & 0.125 \\ 0.375 & 0 & 0.625 & 0 \\ 0 & 0.5 & 0 & 0.5 \end{bmatrix}$$

Multiplication by Transition Matrix

- PageRank finds finds ‘left’ eigenvector:

$$\pi^T P = \pi^T \quad \text{(largest eigenvalue guaranteed to be 1)}$$

  – It’s about finding a ‘pi’ where transition maintains distribution.
- Spectral clustering finds the usual ‘right’ eigenvector:

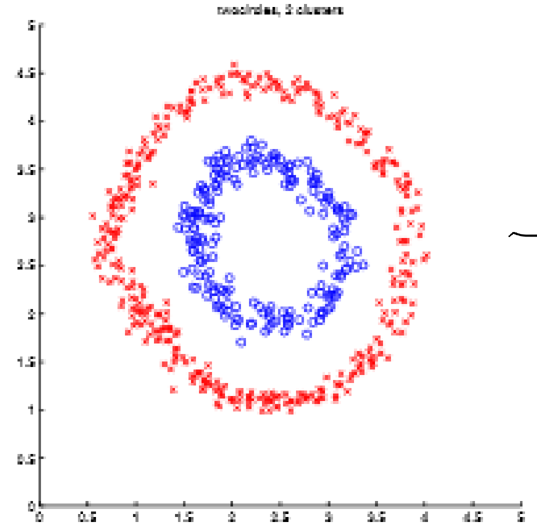
$$P\pi = \pi$$

- Largest eigenvalue again guaranteed to be 1, but it’s degenerate:
  – If graph is connected, pi is a vector of ones.
  – Multiple connected components: multiple eigenvalues of 1.
    - These eigenvectors give the connected components.
  – Further largest eigenvalues:
    - Eigenvectors will tend to cluster based on connectivity.
    - Area of ‘spectral graph theory’ explores this.

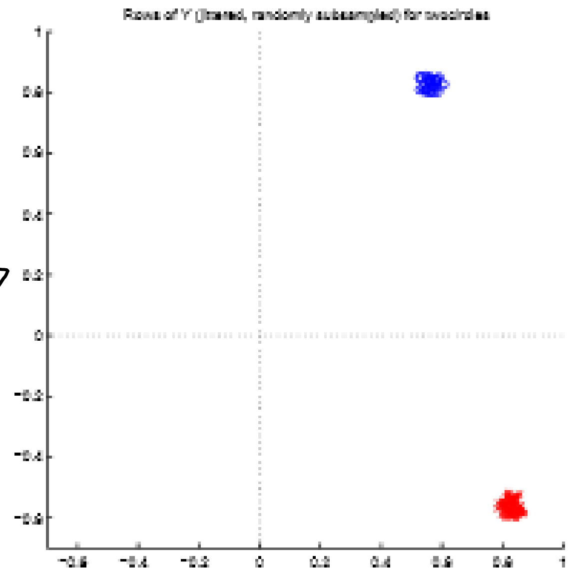
Spectral Clustering

- Spectral clustering method:
  - Compute eigenvectors of largest eigenvalues of P.
  - Run a clustering algorithm using the eigenvectors.

Original data:



→ Use RBF kernel for similarity → Find top eigenvectors →



Now run k-means.

http://ai.stanford.edu/~ang/papers/nips01-spectral.pdf

Connections to Other Algorithms

with centered X.

linear kernel: XX^T

- Kernel PCA: Eigenvectors of 'K', where is 'centered' kernel matrix.
- Spectral clustering:

  Eigenvectors of $P = D^{-1}K$, where D^{-1} normalizes rows.
- Connection to graph Laplacian and spectral graph theory:

  You can equivalently use smallest eigenvectors of $I - D^{-1}K$.

  Alternative methods consider 'Laplacian' $L = D - W$

  and 'normalized Laplacian' $L = I - D^{-1/2} W D^{-1/2}$

Application: Image Segmentation

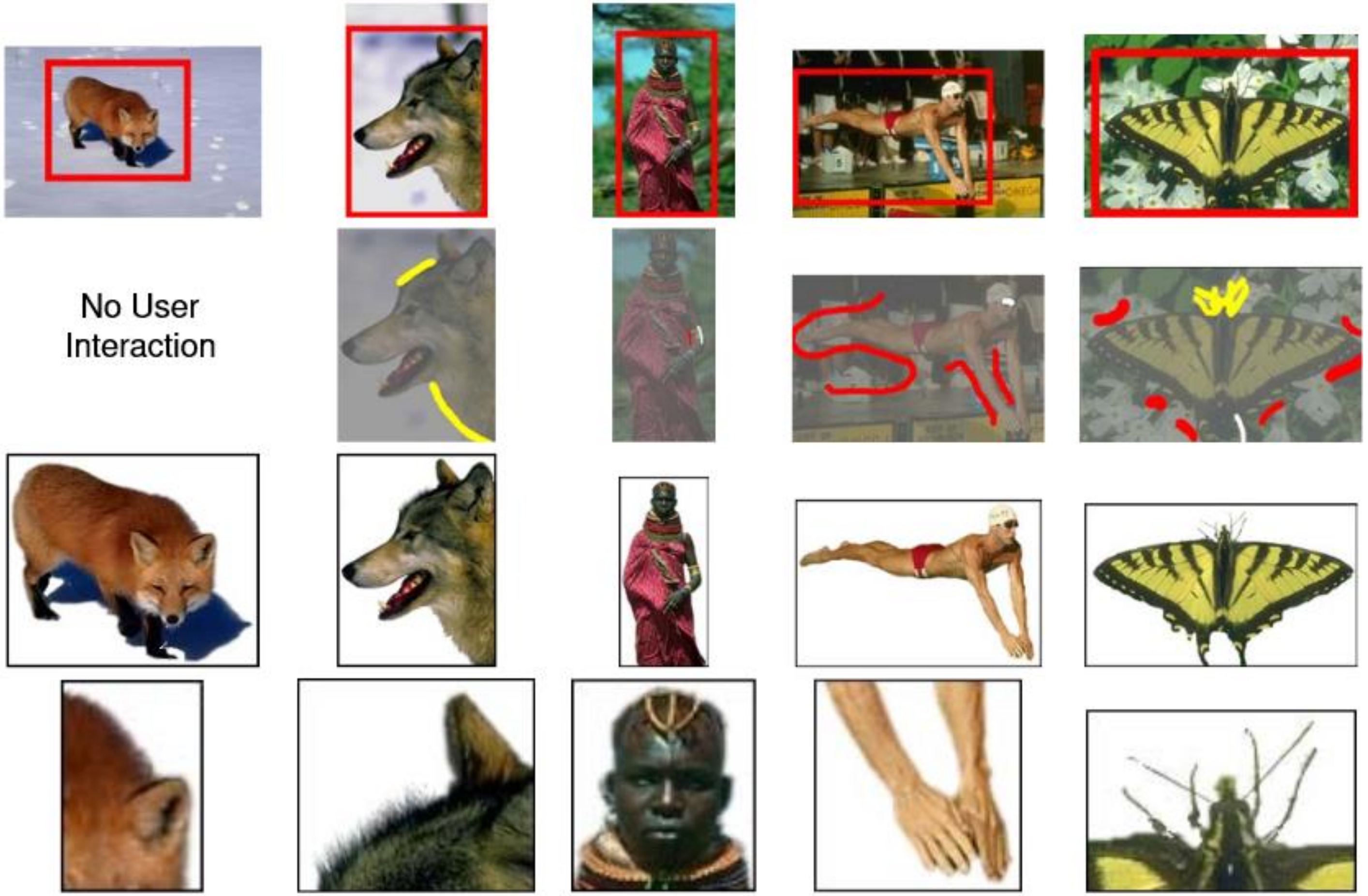
(a) (b)

Graph-Based SSL for Image Segmentation

http://cns-web.bu.edu/~lgrady/grady2006random.pdf

More Advanced Graph Cut Methods

- Combining classification model with graph-based SSL:



http://cvg.ethz.ch/teaching/cvl/2012/grabcut-siggraph04.pdf

Summary

- Spectral clustering considers clustering on graphs.
- Non-convex clusters can be found on data represented as features.
- Biased random walks lead to one variant of spectral clustering.
- Top eigenvectors give spectral clustering solution.
- Graph Laplacian studied in field of 'spectral graph theory'.

- Next time:
  - Finding in patterns in your genes, and most cited science paper of 1990s.
    (and three of the top 15 all-time).