Clustering
Lecture 8

David Sontag

New York University

Slides adapted from Luke Zettlemoyer, Vibhav Gogate, Carlos Guestrin, Andrew Moore, Dan Klein

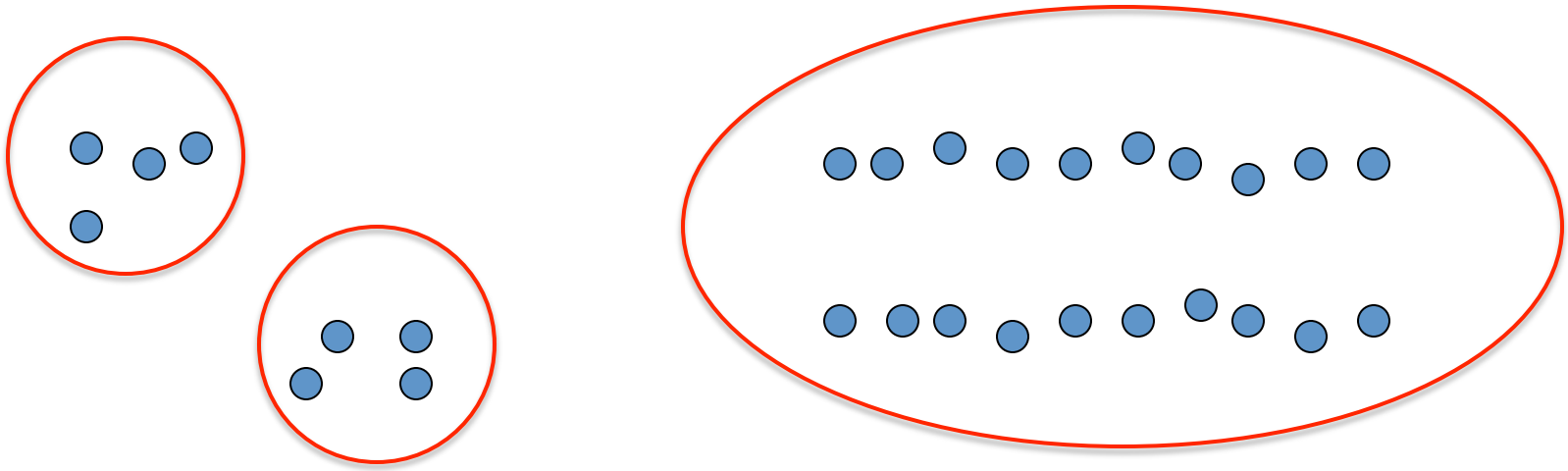
Clustering

Clustering:

- Unsupervised learning
- Requires data, but no labels
- Detect patterns e.g. in
  - Group emails or search results
  - Customer shopping patterns
  - Regions of images
- Useful when don't know what you're looking for
- But: can get gibberish

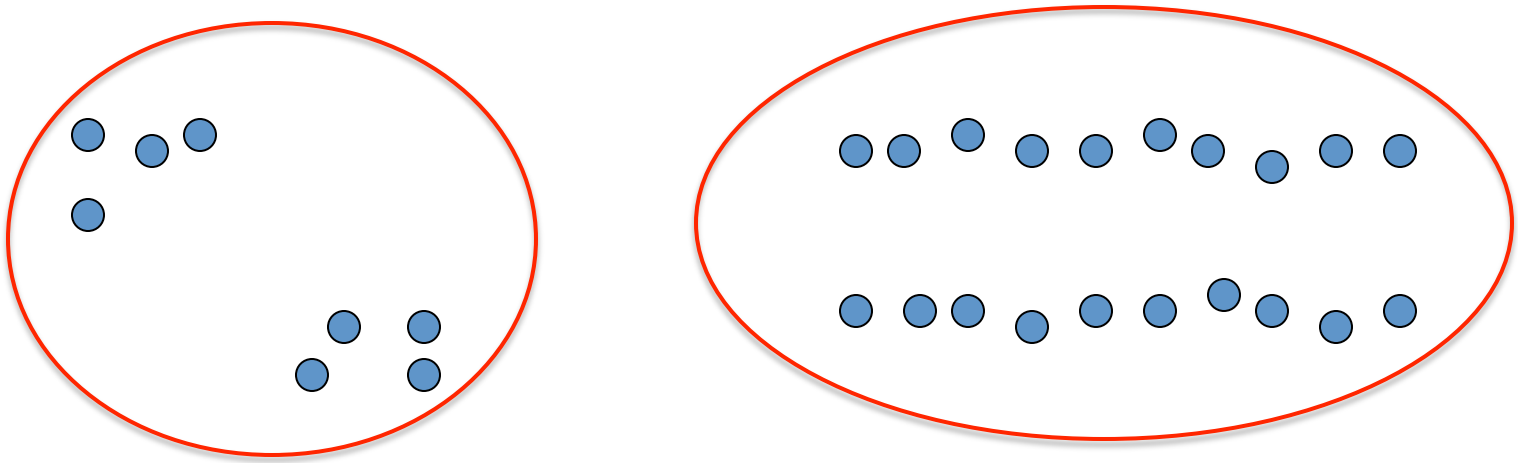
Clustering

- Basic idea: group together similar instances
- Example: 2D point patterns



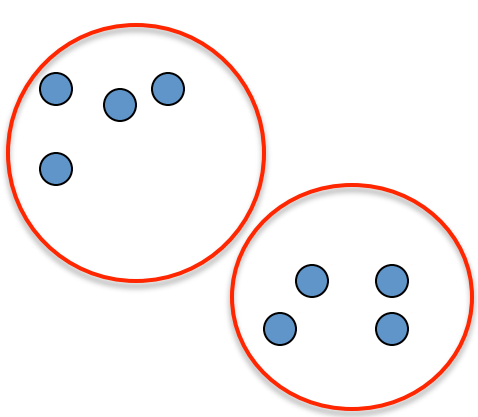
Clustering

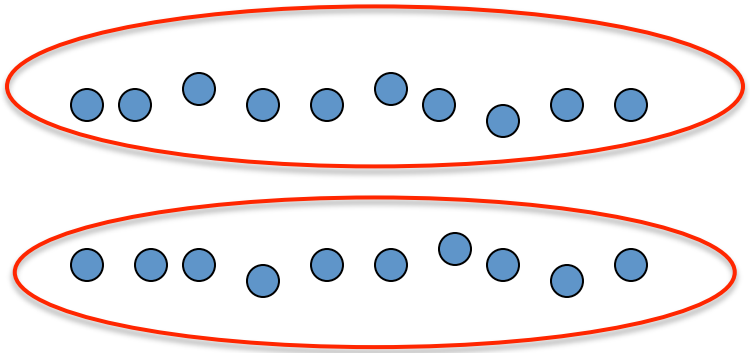
- Basic idea: group together similar instances
- Example: 2D point patterns



Clustering

- Basic idea: group together similar instances
- Example: 2D point patterns





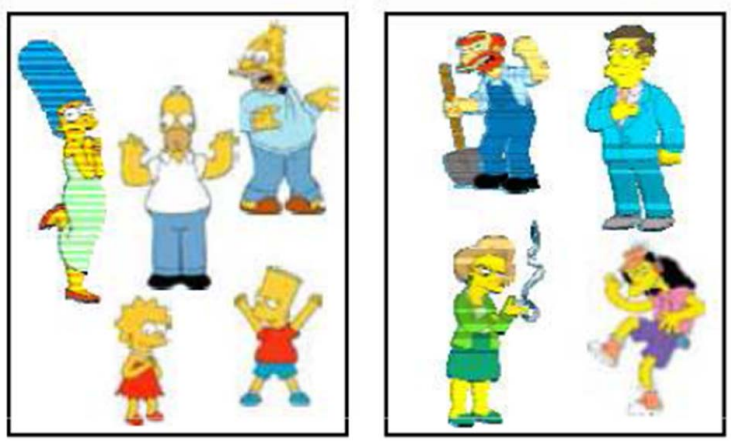
- What could "similar" mean?
  - One option: small Euclidean distance (squared)

$$\mathrm{dist}(\vec{x}, \vec{y}) = ||\vec{x} - \vec{y}||_2^2$$

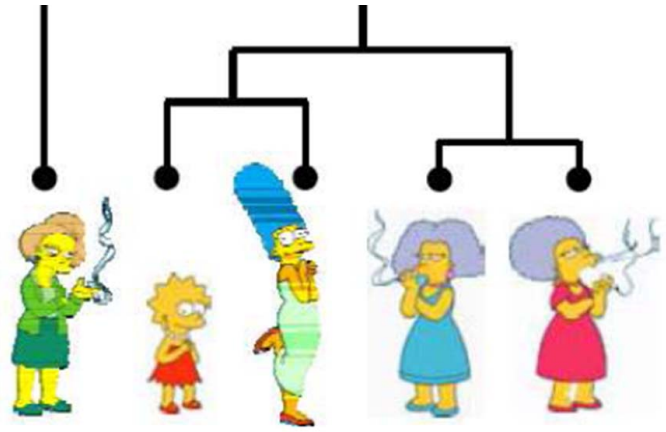
  - Clustering results are crucially dependent on the measure of similarity (or distance) between "points" to be clustered

Clustering algorithms

- Partition algorithms (Flat)
  - K-means
  - Mixture of Gaussian
  - Spectral Clustering



- Hierarchical algorithms
  - Bottom up – agglomerative
  - Top down – divisive



Clustering examples

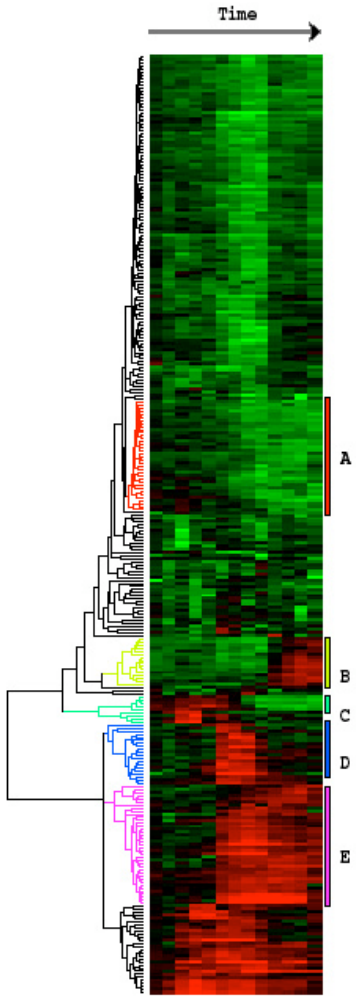
Image segmentation

Goal: Break up the image into meaningful or perceptually similar regions

[Slide from James Hayes]

Clustering examples

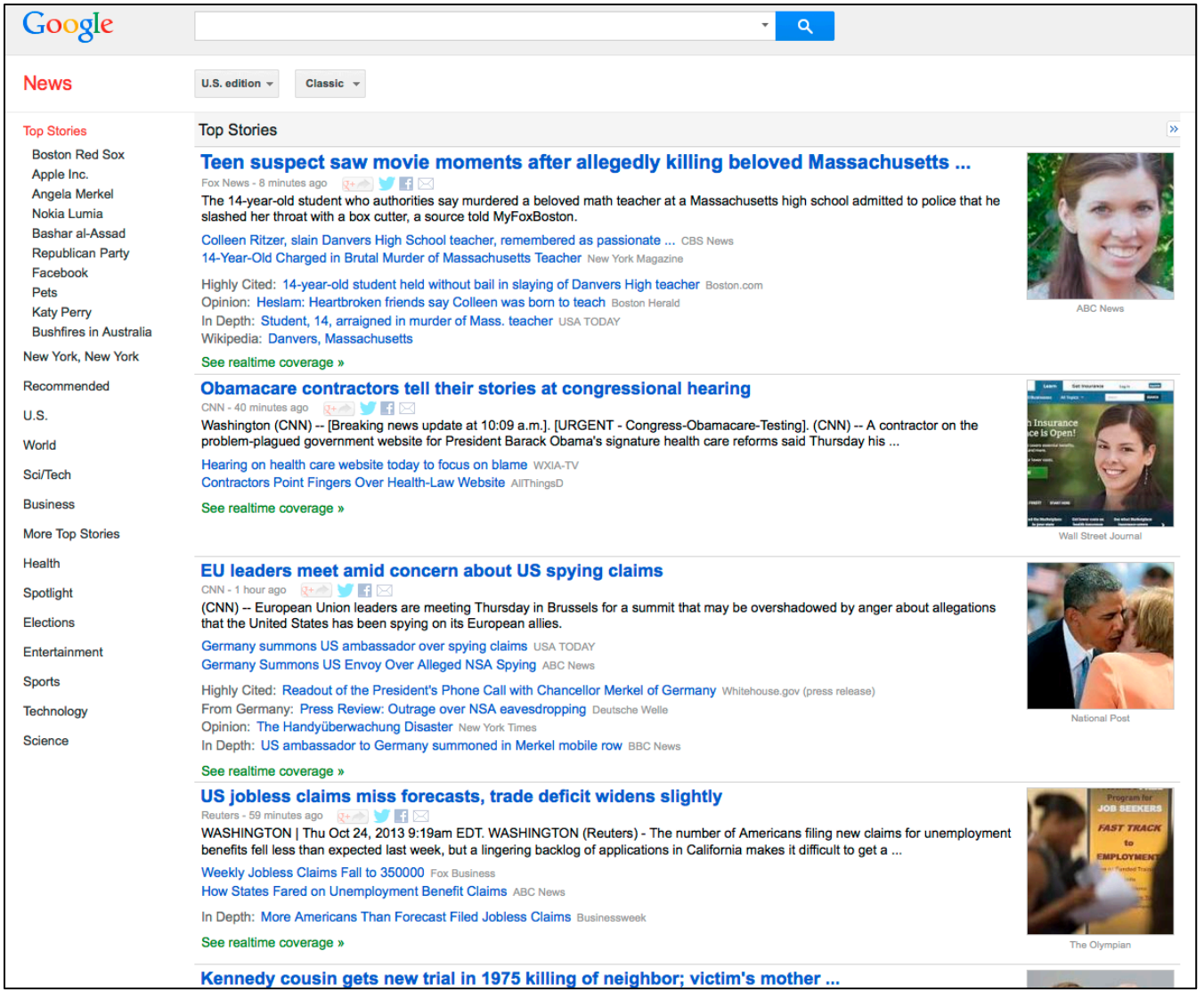
Clustering gene expression data



Eisen et al, PNAS 1998

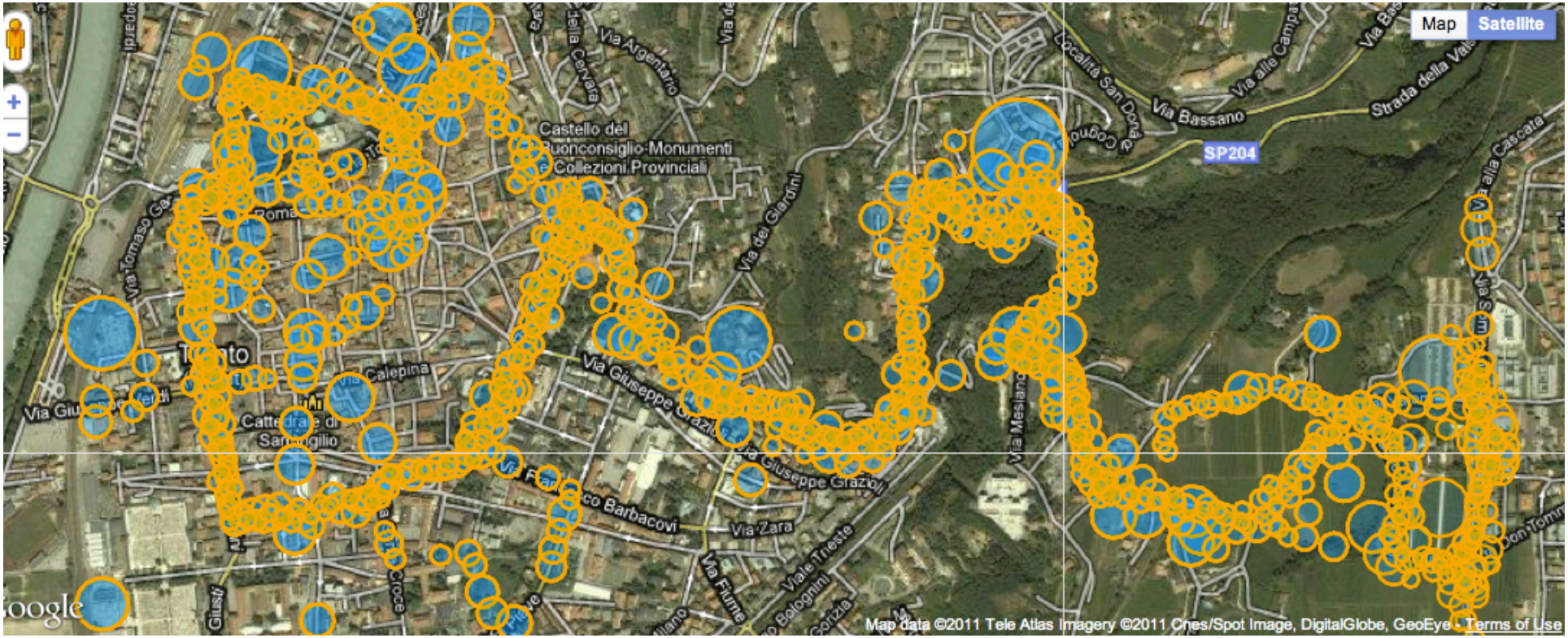
Clustering examples

Cluster news articles



Clustering examples

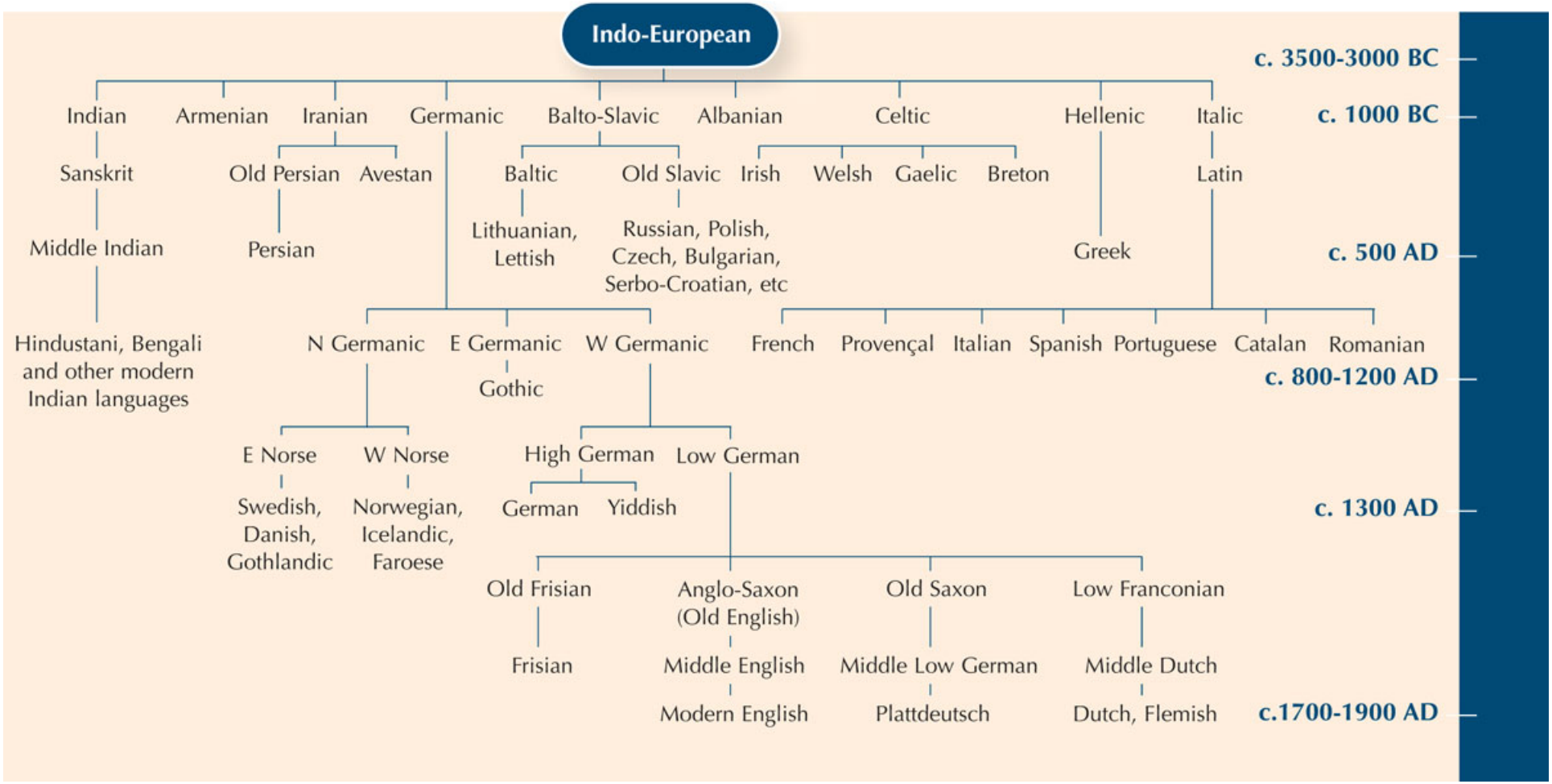
Cluster people by space and time



[Image from Pilho Kim]

Clustering examples

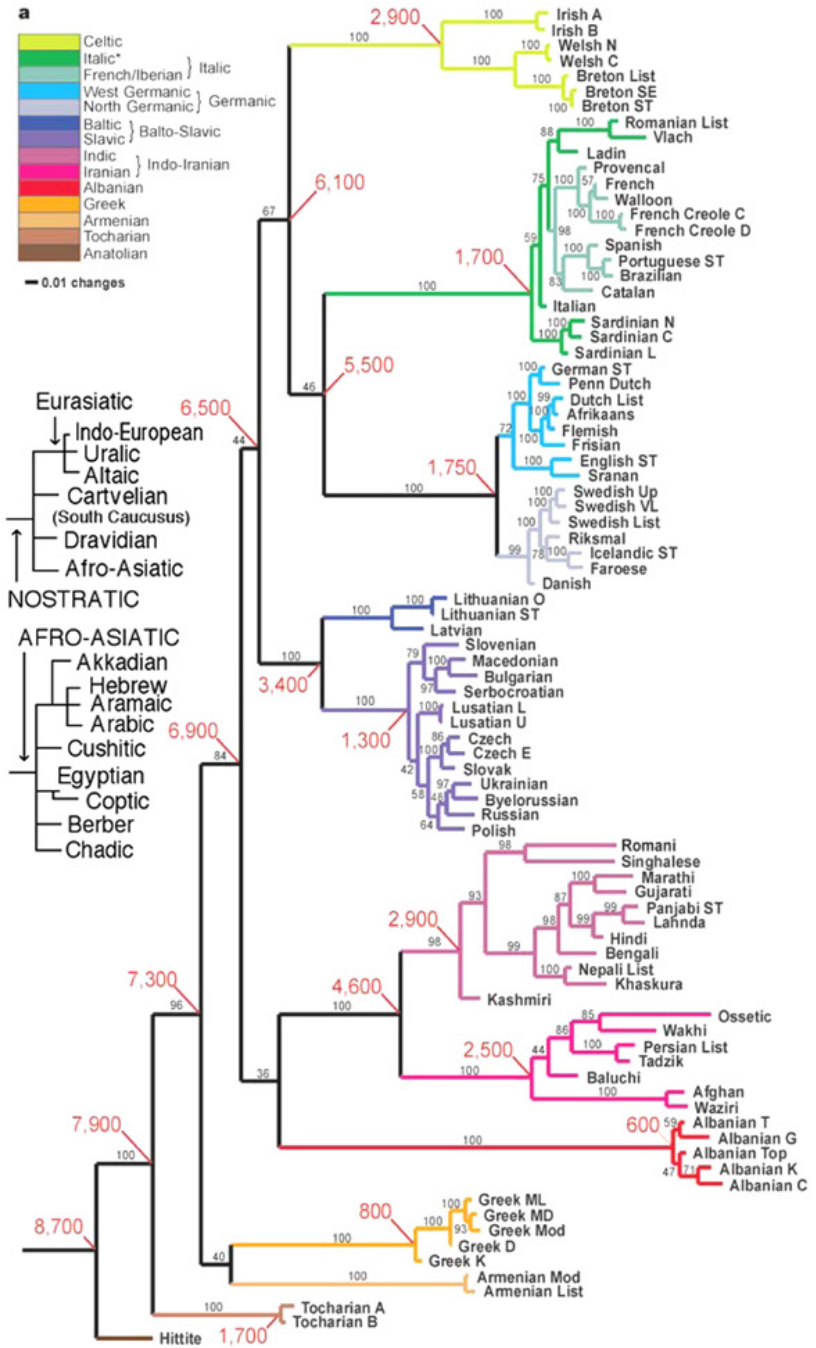
Clustering languages



[Image from scienceinschool.org]

Clustering examples

Clustering languages

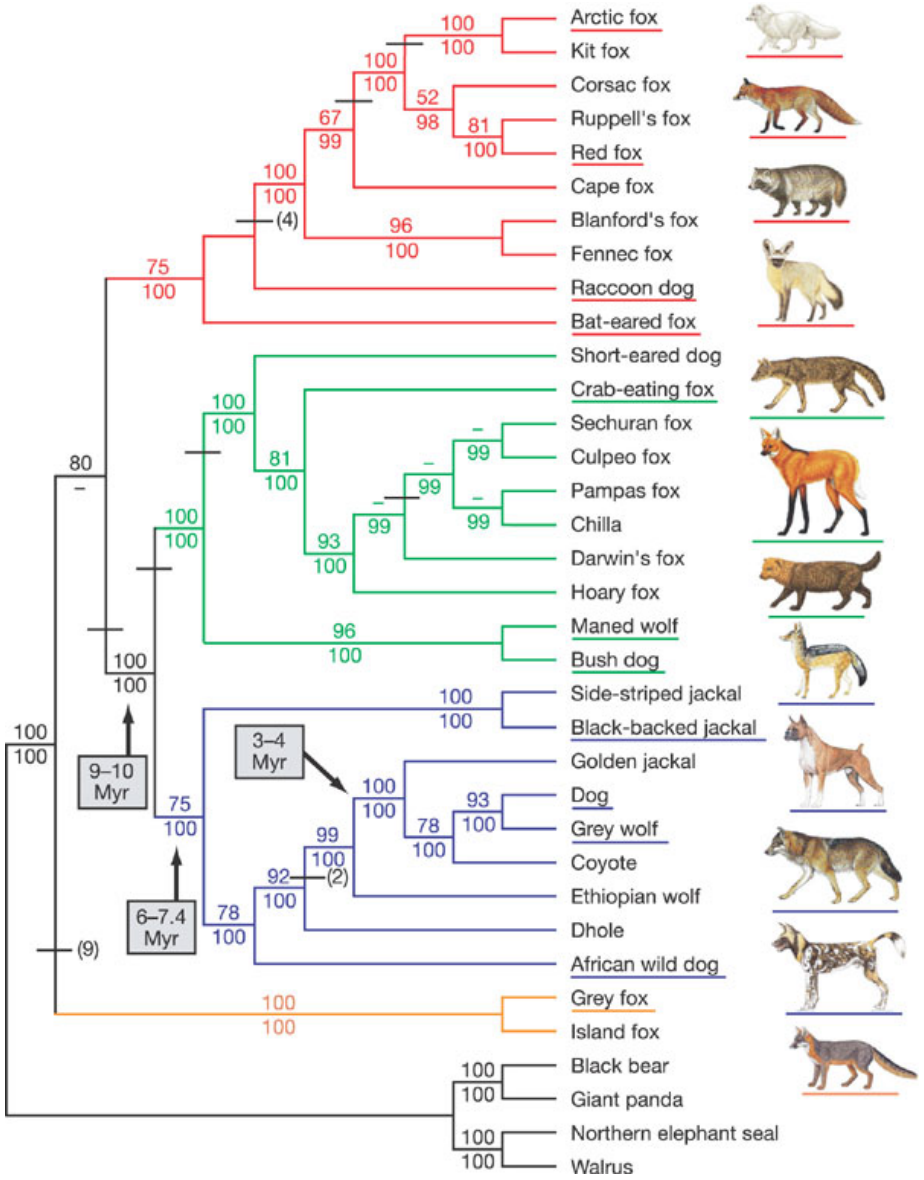


[Image from dhushara.com]

Clustering examples

Clustering species

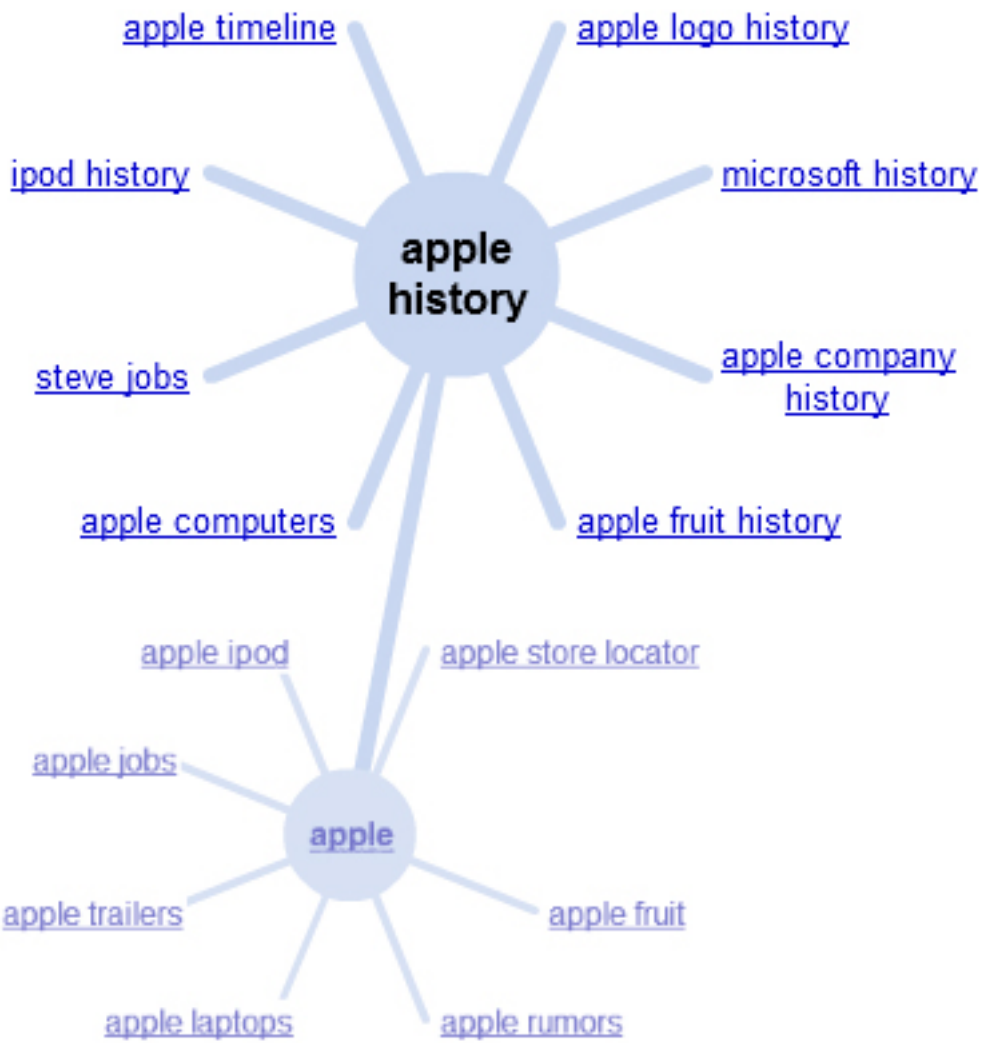
(“phylogeny”)



[Lindblad-Toh et al., Nature 2005]

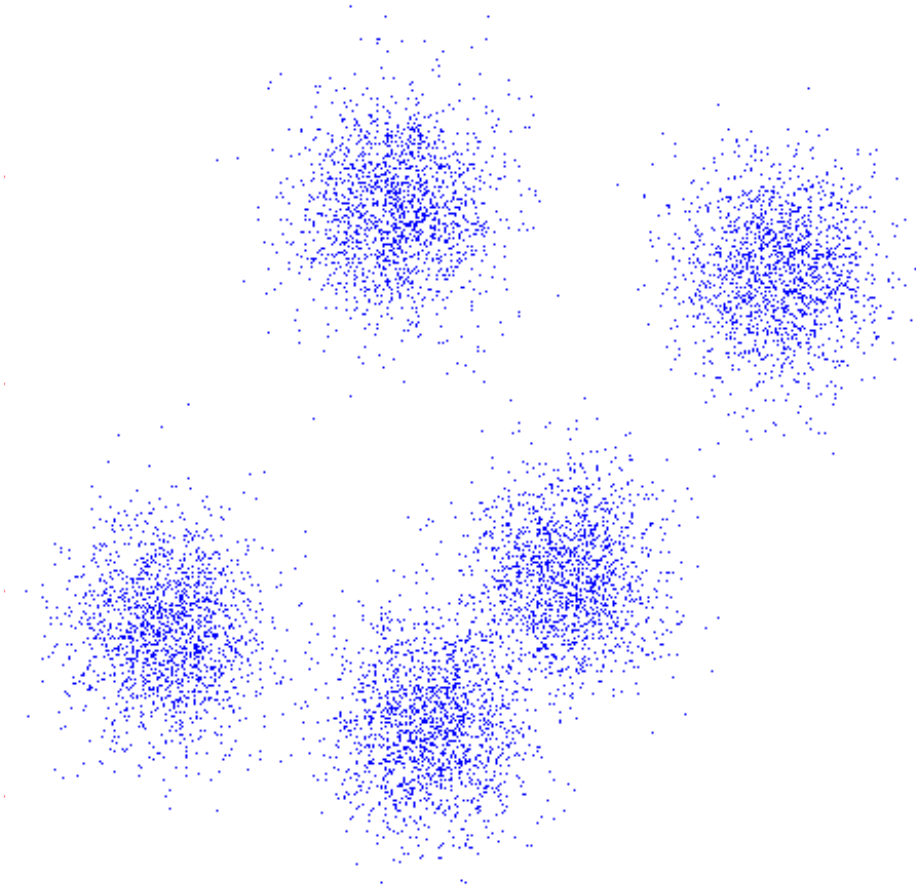
Clustering examples

Clustering search queries



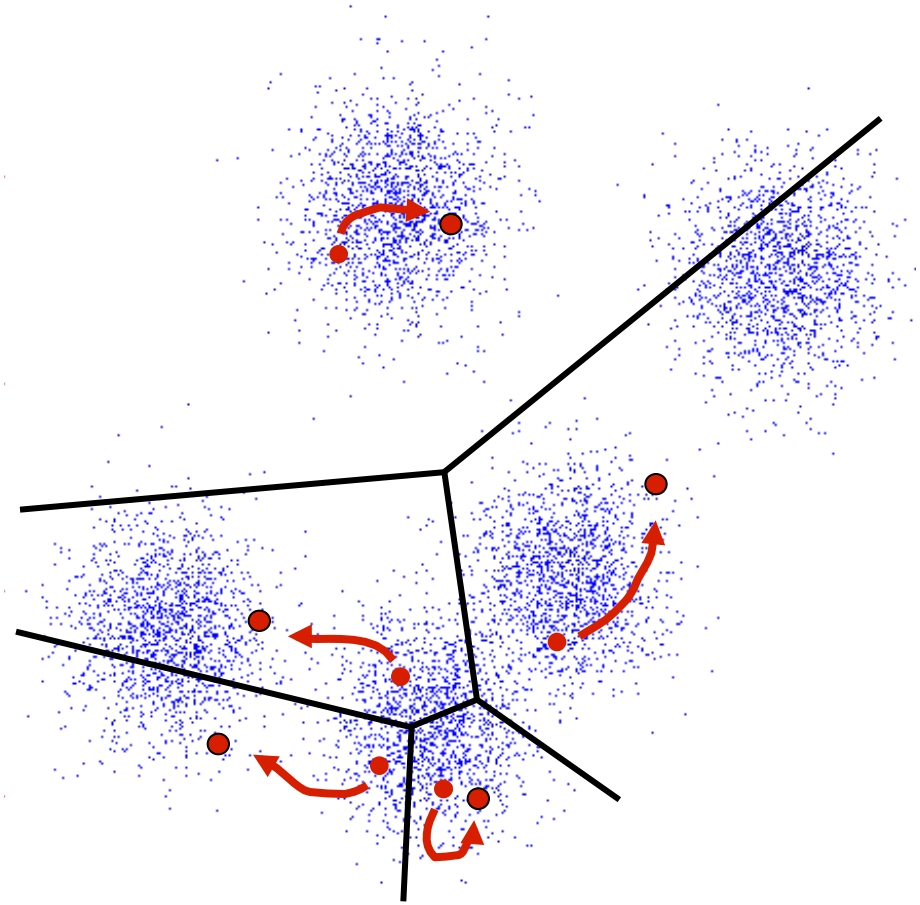
K-Means

- An iterative clustering algorithm
  - Initialize: Pick *K* random points as cluster centers
  - Alternate:
    1. Assign data points to closest cluster center
    2. Change the cluster center to the average of its assigned points
  - Stop when no points' assignments change

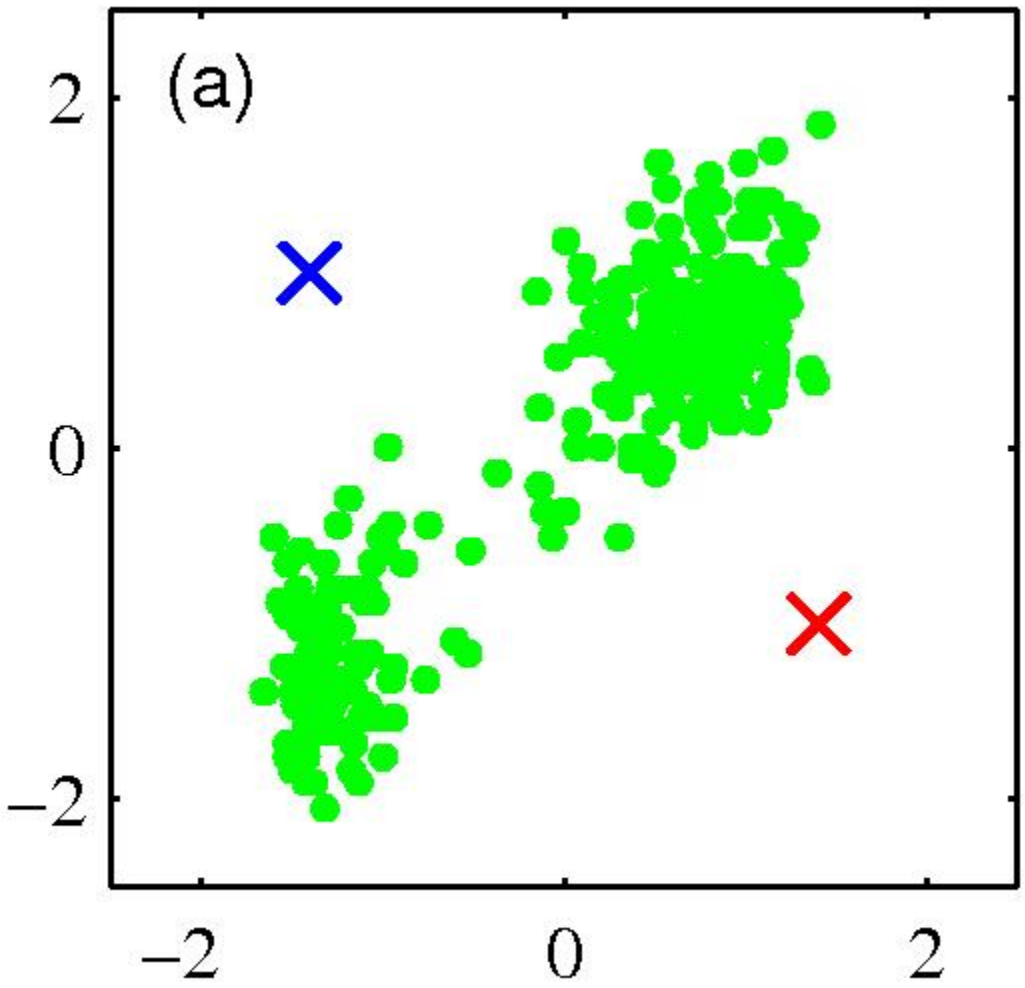


K-Means

- An iterative clustering algorithm
  - Initialize: Pick *K* random points as cluster centers
  - Alternate:
    1. Assign data points to closest cluster center
    2. Change the cluster center to the average of its assigned points
  - Stop when no points' assignments change



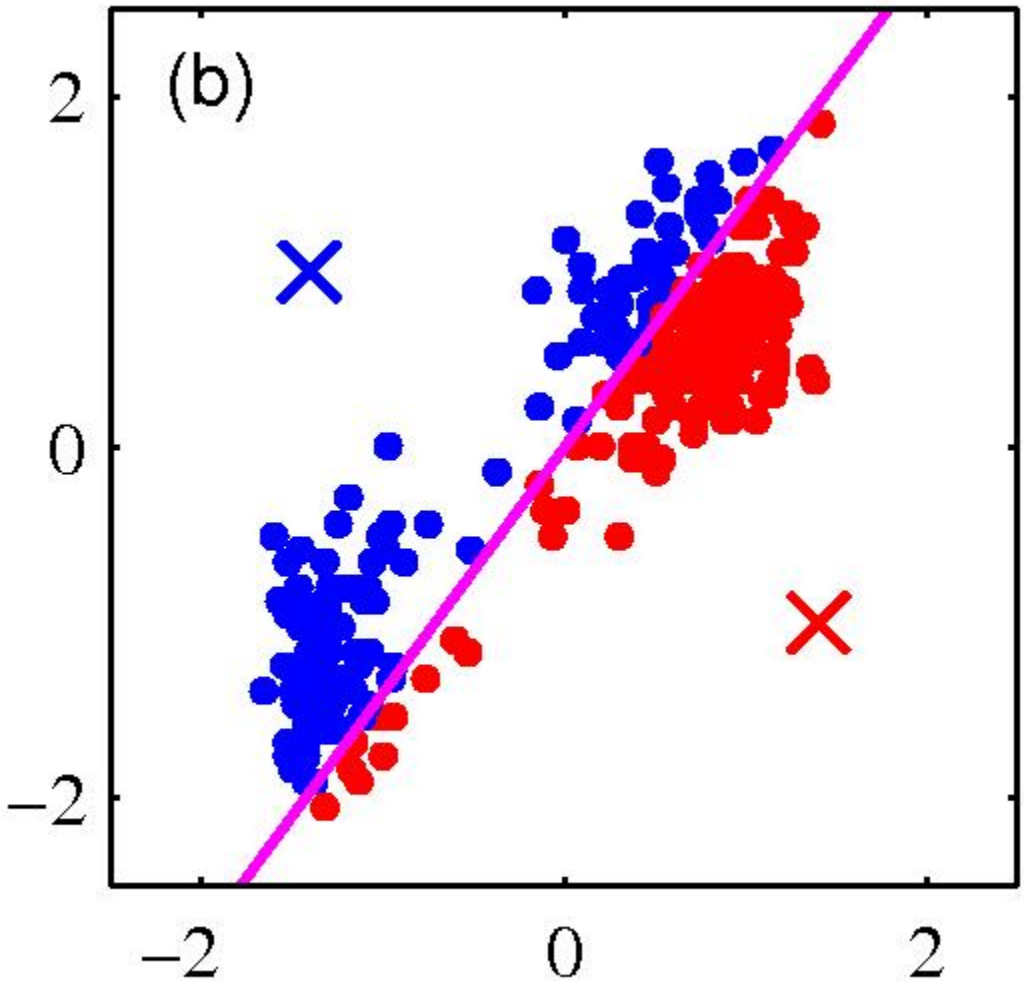
K-means clustering: Example



- Pick *K* random points as cluster centers (means)

Shown here for *K*=2

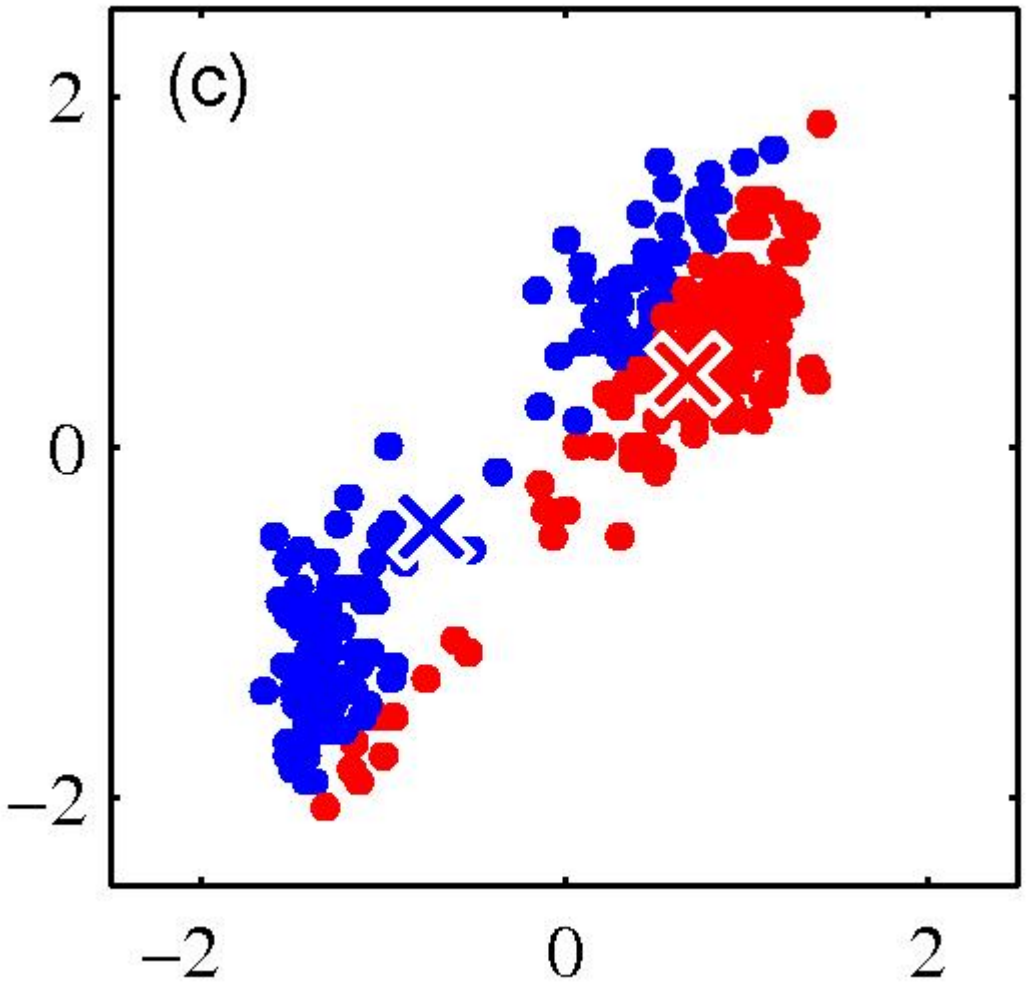
K-means clustering: Example



Iterative Step 1

- Assign data points to closest cluster center

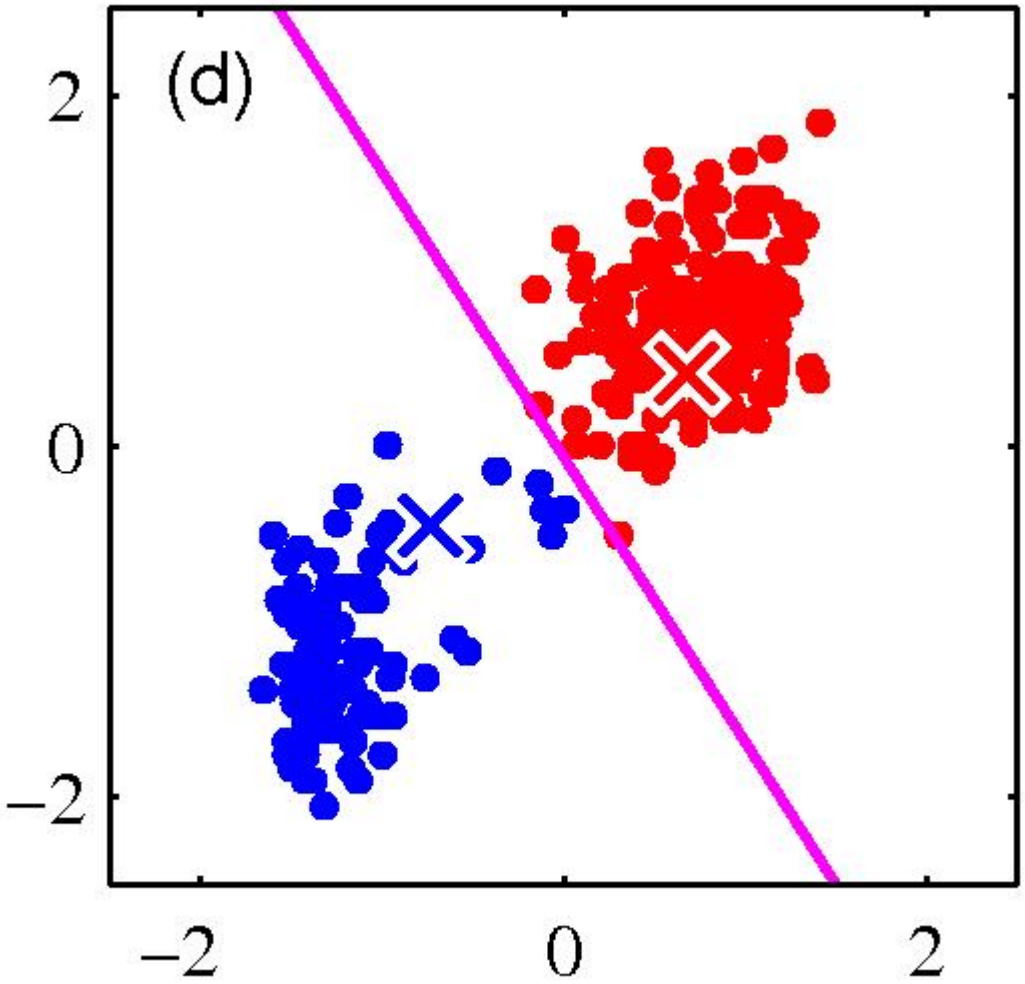
K-means clustering: Example



Iterative Step 2

- Change the cluster center to the average of the assigned points

K-means clustering: Example



- Repeat until convergence

Properties of K-means **algorithm**

- Guaranteed to converge in a finite number of iterations

- Running time per iteration:
    1. Assign data points to closest cluster center

       O(KN) time

    2. Change the cluster center to the average of its assigned points

       O(N)

Kmeans Convergence

Objective

$$\min_{\mu}\min_{C}\sum_{i=1}^{k}\sum_{x\in C_i}|x-\mu_i|^2$$

1. Fix μ, optimize C:

$$\min_{C}\sum_{i=1}^{k}\sum_{x\in C_i}|x-\mu_i|^2=\min_{c}\sum_{i}^{n}\left|x_i-\mu_{x_i}\right|^2$$

Step 1 of kmeans

2. Fix C, optimize μ:

$$\min_{\mu}\sum_{i=1}^{k}\sum_{x\in C_i}|x-\mu_i|^2$$

   – Take partial derivative ~~of~~ with respect to μ_i and set to zero, we have

$$\mu_i=\frac{1}{|C_i|}\sum_{x\in C_i}x$$

Step 2 of kmeans

Kmeans takes an alternating optimization approach, each step is guaranteed to decrease the objective – thus guaranteed to converge

[Slide from Alan Fern]

Example: K-Means for Segmentation

K=2

Goal of Segmentation is to partition an image into regions each of which has reasonably homogenous visual appearance.

Original

Example: K-Means for Segmentation

K=2

K=3

Original

Example: K-Means for Segmentation

K=2

K=3

K=10

Original

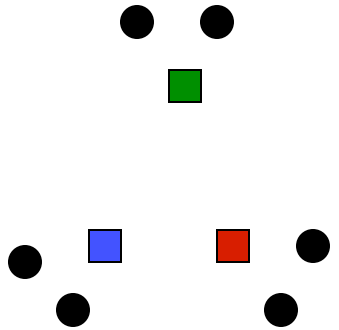
Example: Vector quantization

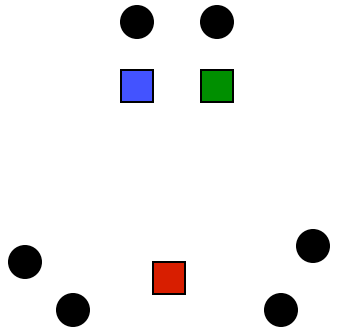
FIGURE 14.9. *Sir Ronald A. Fisher* $(1890 - 1962)$ *was one of the founders of modern day statistics, to whom we owe maximum-likelihood, sufficiency, and many other fundamental concepts. The image on the left is a* 1024×1024 *grayscale image at* 8 *bits per pixel. The center image is the result of* 2×2 *block VQ, using* 200 *code vectors, with a compression rate of* 1.9 *bits/pixel. The right image uses only four code vectors, with a compression rate of* 0.50 *bits/pixel*

[Figure from Hastie *et al.* book]

Initialization

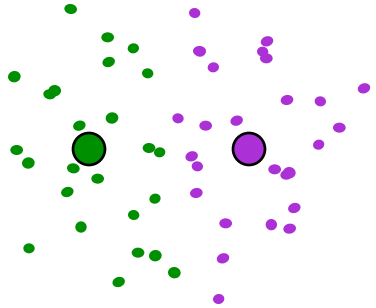
- K-means **algorithm** is a heuristic
  - Requires initial means
  - It does matter what you pick!
  - What can go wrong?
  - Various schemes for preventing this kind of thing: variance-based split / merge, initialization heuristics



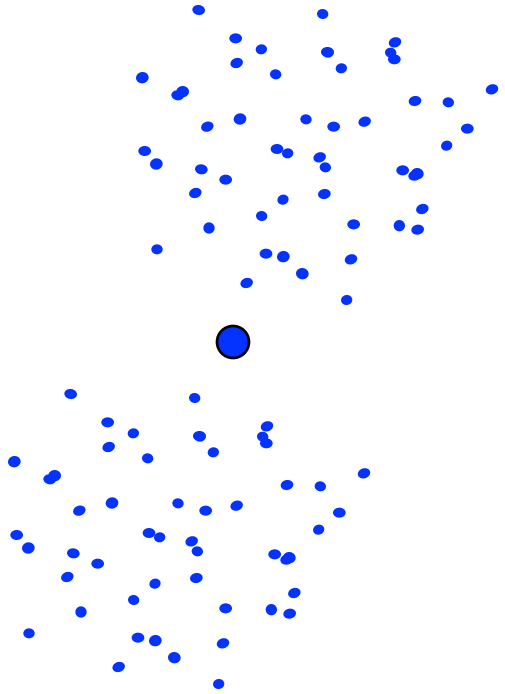


K-Means Getting Stuck

A local optimum:

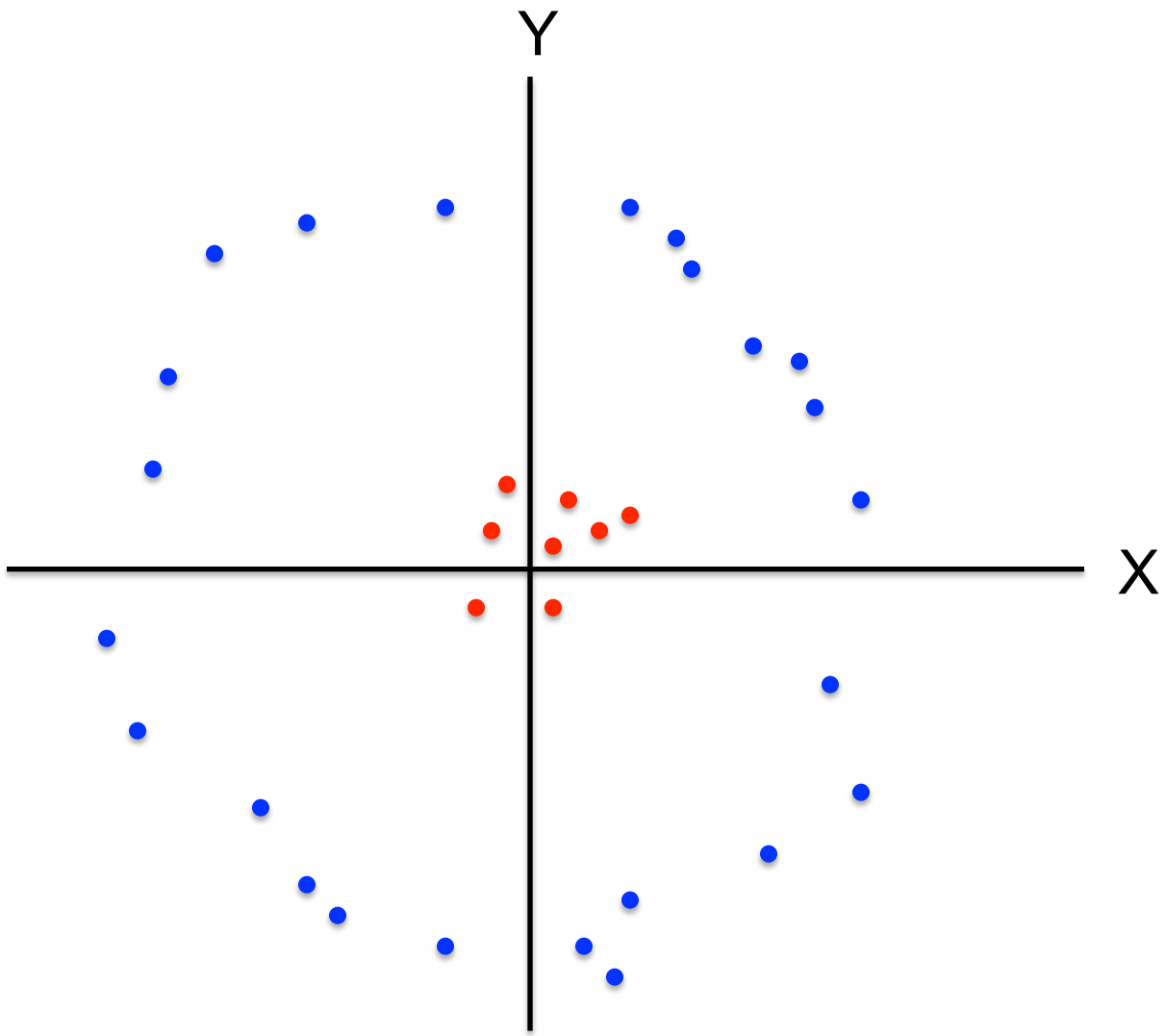


Would be better to have
one cluster here

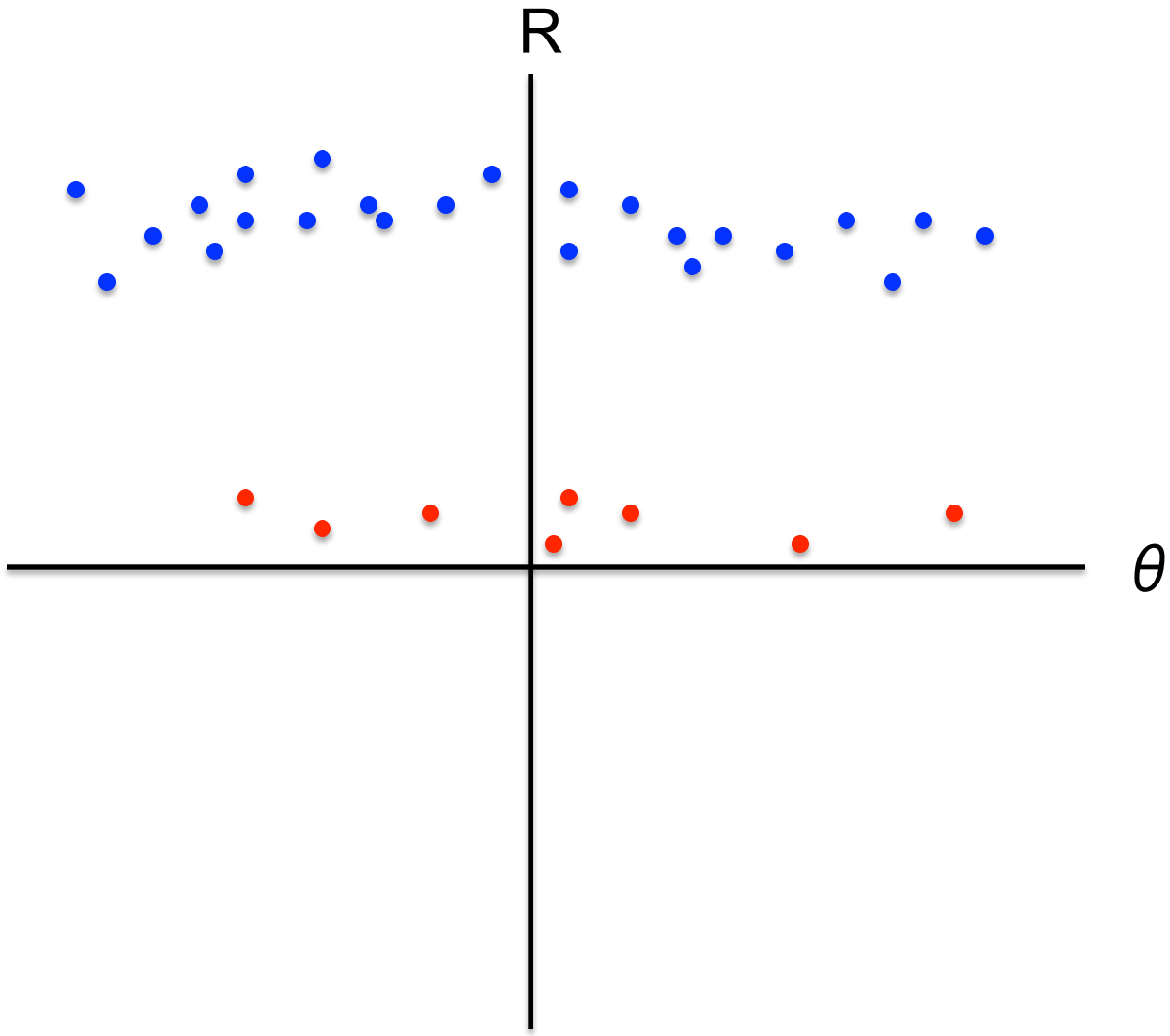


… and two clusters here

K-means not able to properly cluster



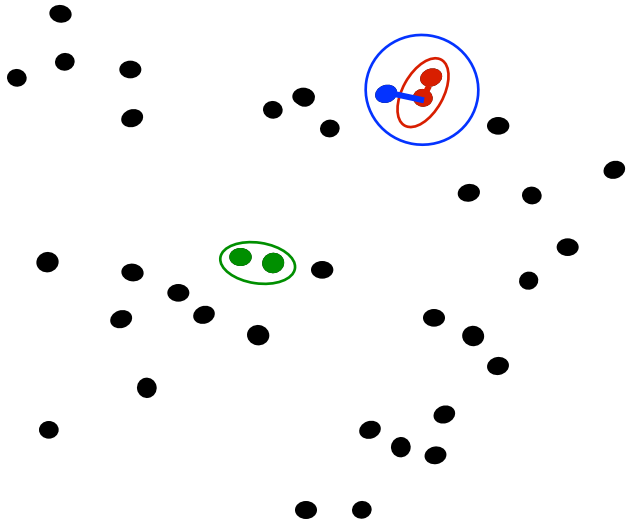
Changing the features (distance function) can help



Hierarchical Clustering

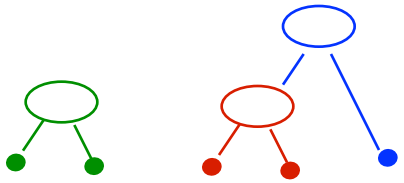
Agglomerative Clustering

- Agglomerative clustering:
  - First merge very similar instances
  - Incrementally build larger clusters out of smaller clusters



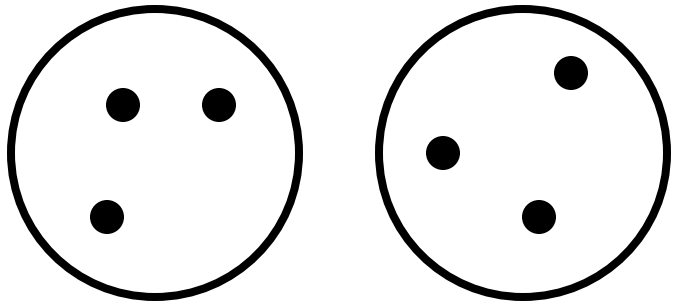
- Algorithm:
  - Maintain a set of clusters
  - Initially, each instance in its own cluster
  - Repeat:
    - Pick the two closest clusters
    - Merge them into a new cluster
    - Stop when there's only one cluster left

- Produces not one clustering, but a family of clusterings represented by a dendrogram



Agglomerative Clustering

- How should we define “closest” for clusters with multiple elements?

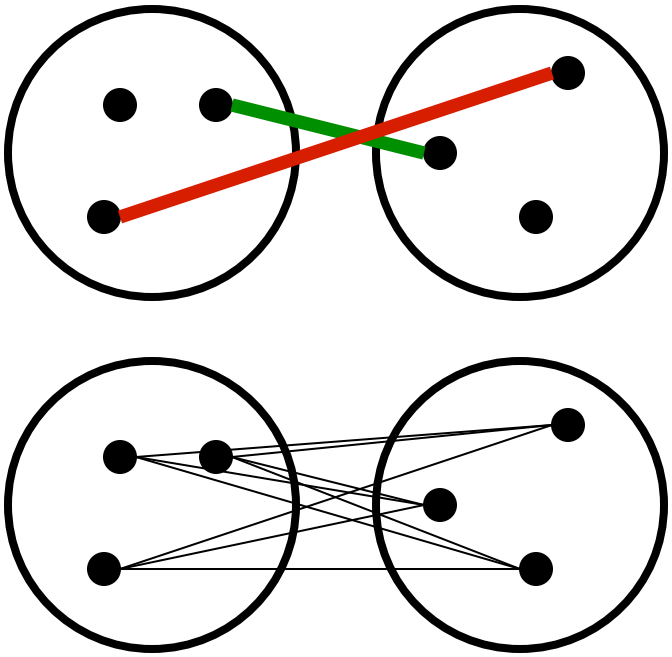


Agglomerative Clustering

- How should we define "closest" for clusters with multiple elements?

- Many options:
  - Closest pair
    (single-link clustering)
  - Farthest pair
    (complete-link clustering)
  - Average of all pairs

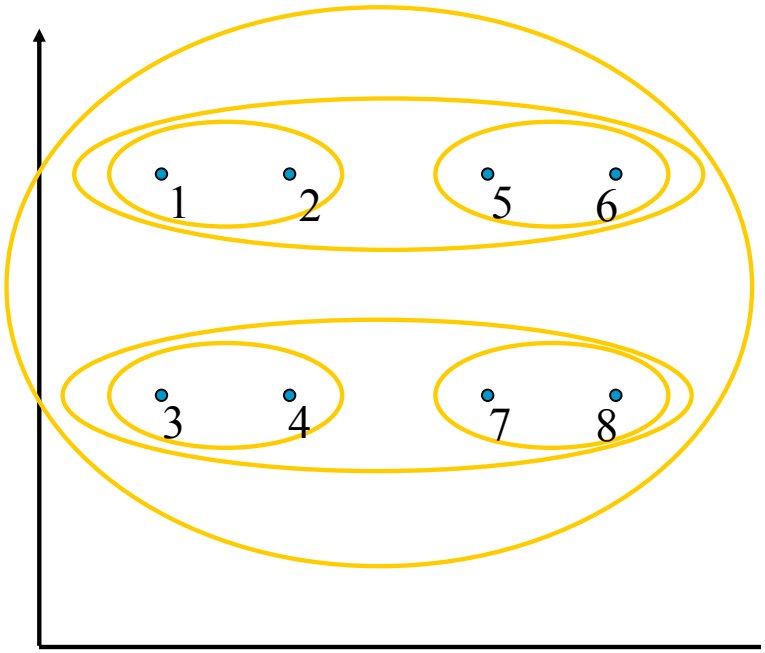
- Different choices create different clustering behaviors



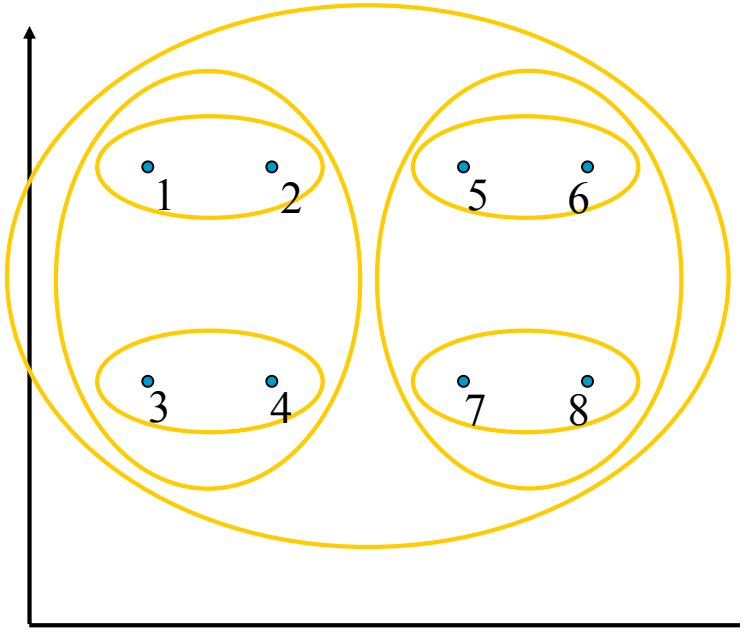
Agglomerative Clustering

- How should we define "closest" for clusters with multiple elements?

Closest pair
(single-link clustering)

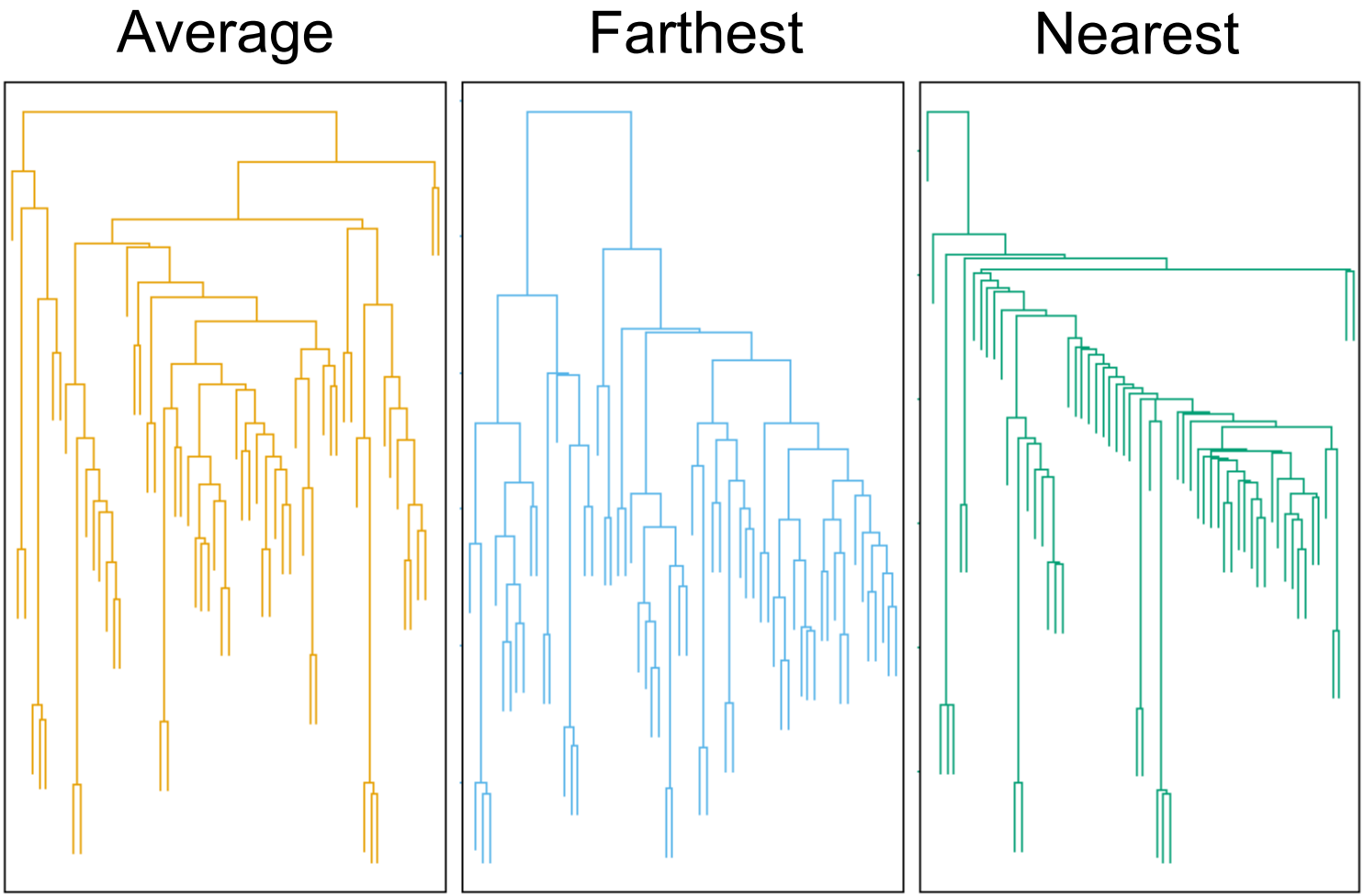


Farthest pair
(complete-link clustering)



[Pictures from Thorsten Joachims]

Clustering Behavior

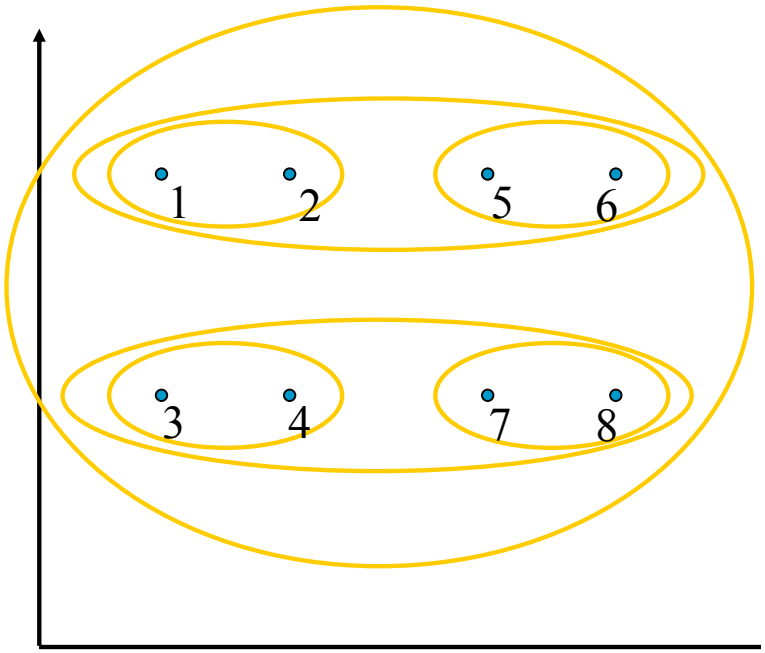


Mouse tumor data from [Hastie *et al.*]

Agglomerative Clustering

When can this be expected to work?

Closest pair
(single-link clustering)



Strong separation property:
All points are more similar to points in their own cluster than to any points in any other cluster

Then, the true clustering corresponds to some **pruning** of the tree obtained by single-link clustering!

Slightly weaker (stability) conditions are solved by average-link clustering

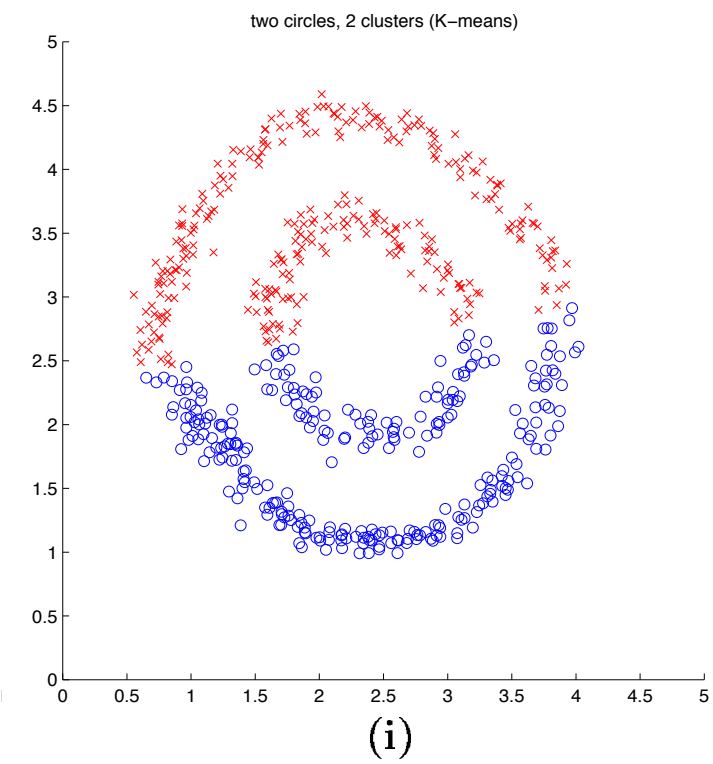
(Balcan et al., 2008)

Spectral Clustering

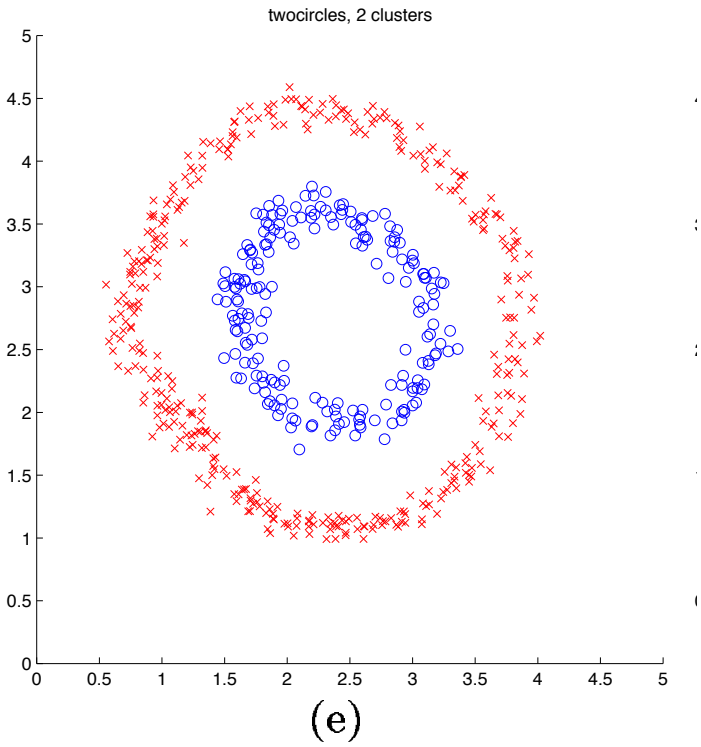
Slides adapted from James Hays, Alan Fern, and Tommi Jaakkola

Spectral clustering

K-means

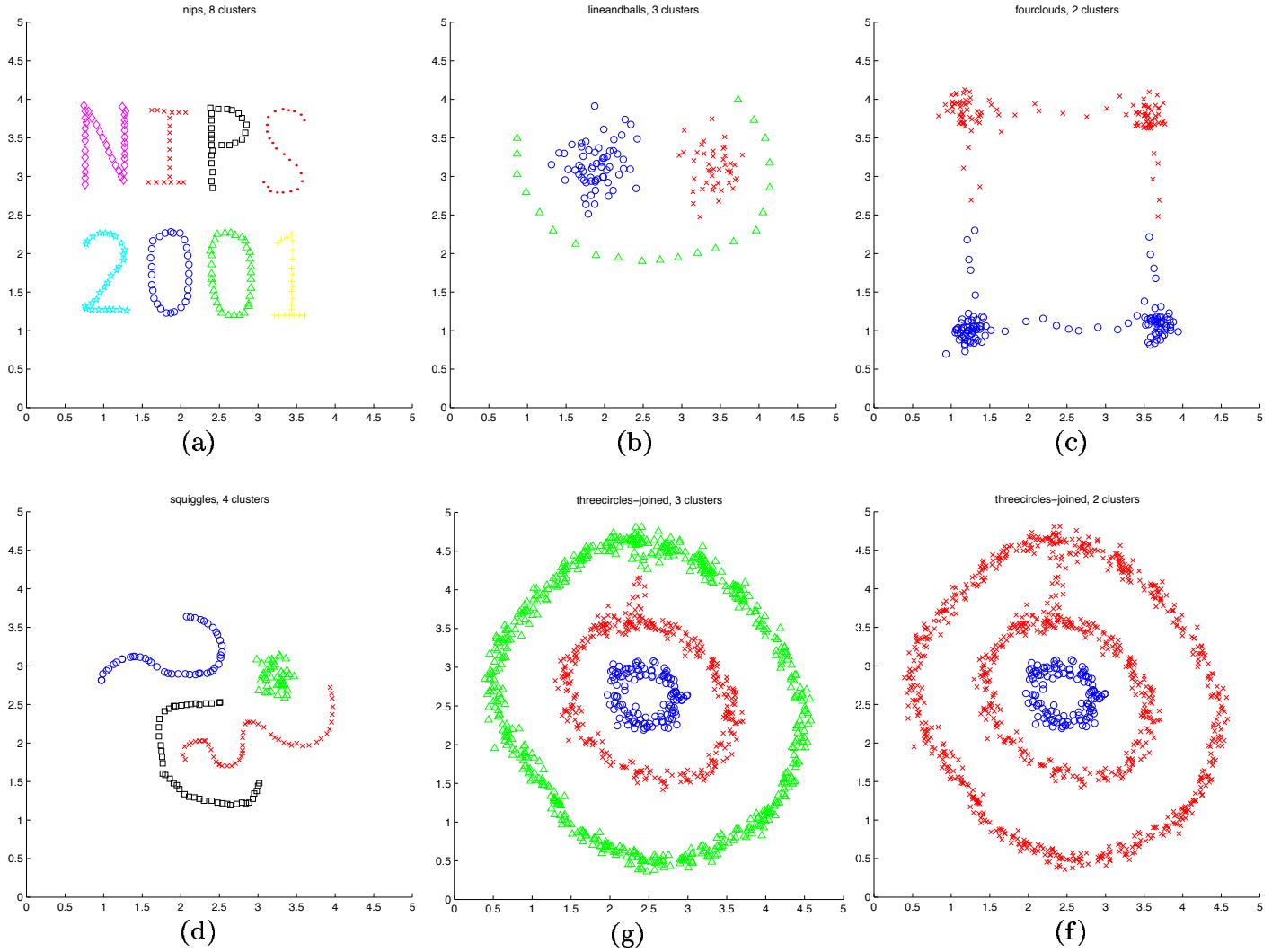


Spectral clustering



[Shi & Malik ‘00; Ng, Jordan, Weiss NIPS ‘01]

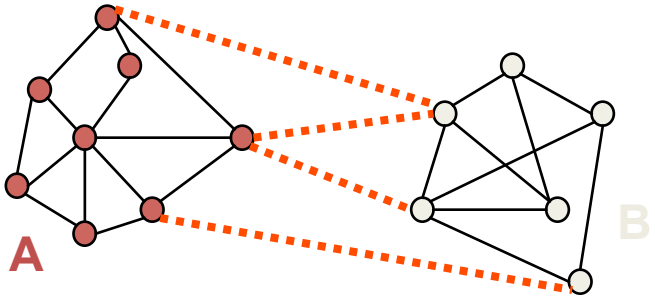
Spectral clustering

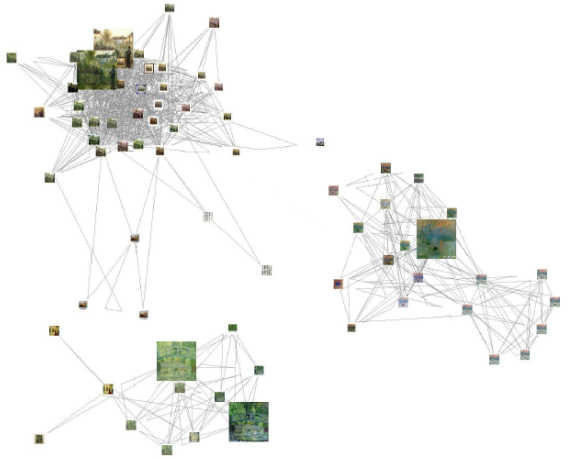


[Figures from Ng, Jordan, Weiss NIPS ‘01]

Spectral clustering

Group points based on links in a graph





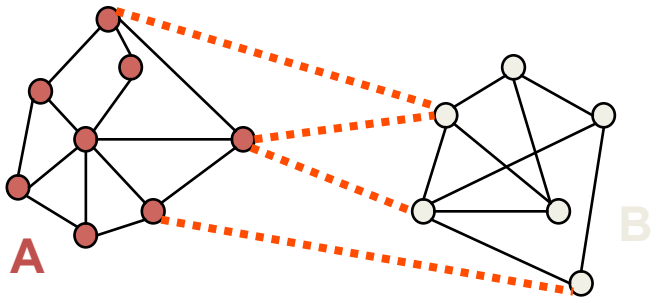
[Slide from James Hays]

How to Create the Graph ?

- It is common to use a Gaussian Kernel to compute similarity between objects

$$W(i,j) = \exp \frac{-\left|x_i - x_j\right|^2}{\sigma^2}$$

- One could create
  - A fully connected graph
  - K-nearest neighbor graph (each node is only connected to its K-nearest neighbors)



[Slide from Alan Fern]

Spectral clustering for segmentation

[Slide from James Hays]

Can we use minimum cut for clustering?

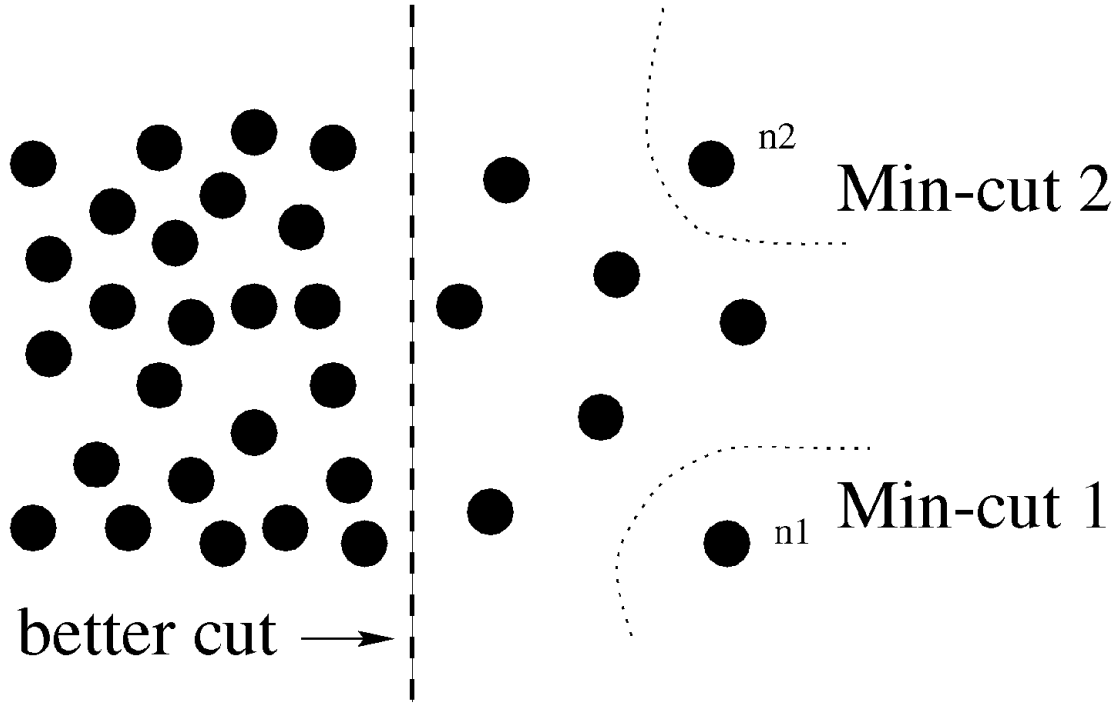
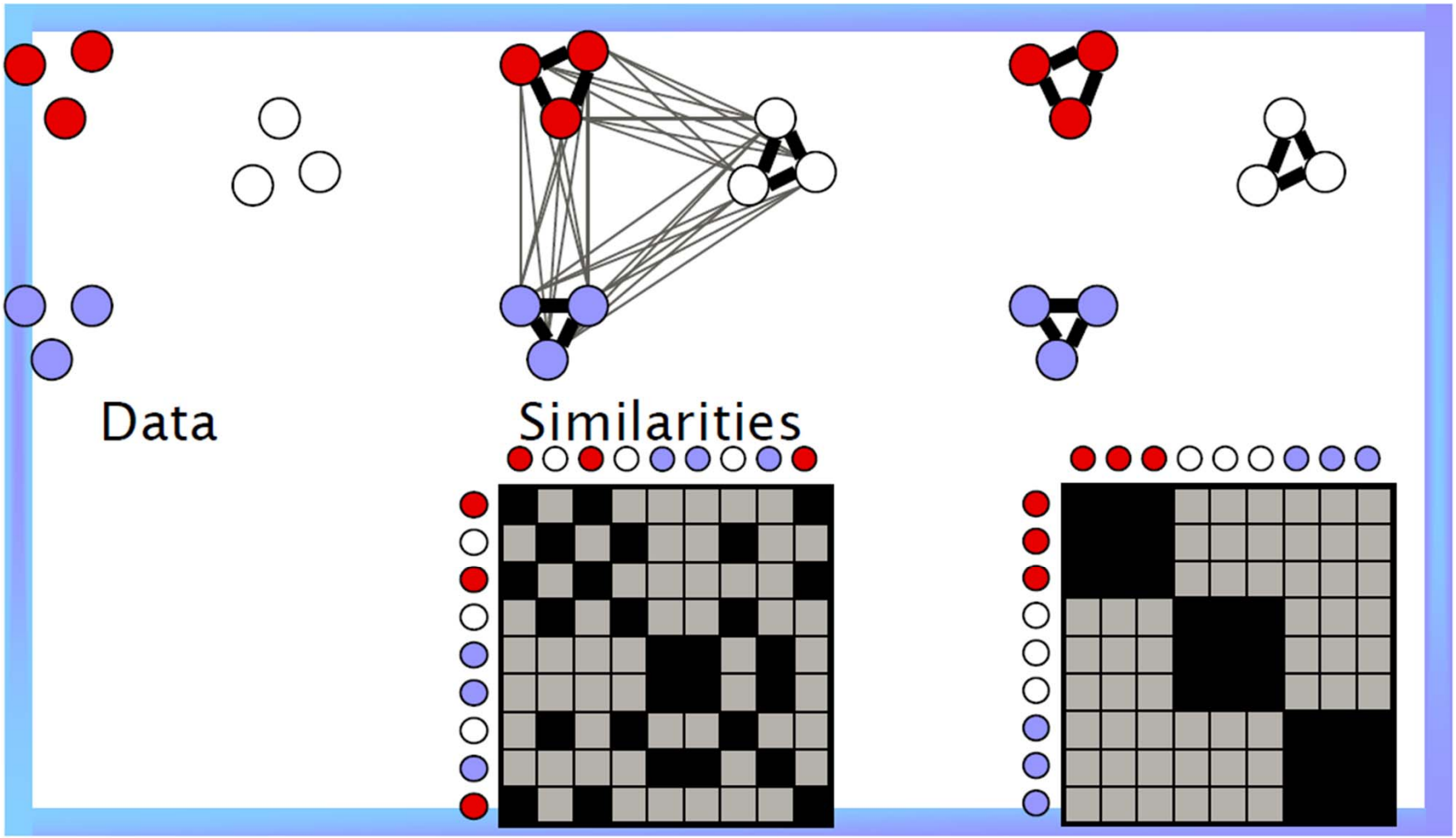


Fig. 1. A case where minimum cut gives a bad partition.

[Shi & Malik ‘00]

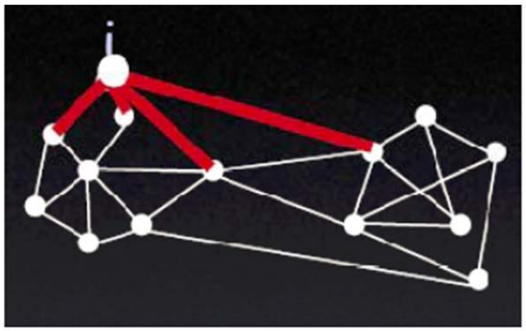
Graph partitioning

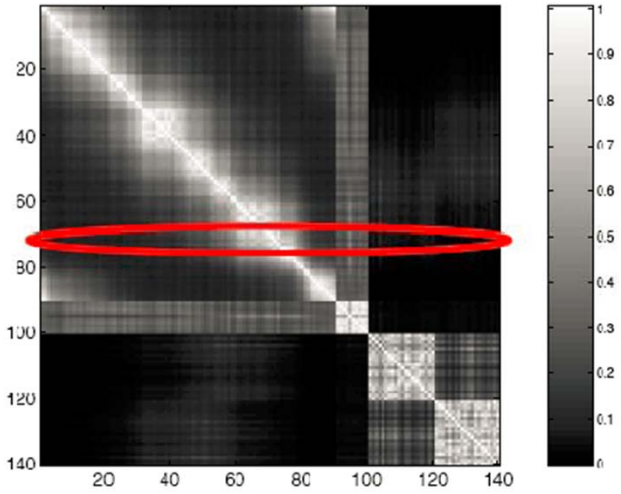


Graph Terminologies

- Degree of nodes

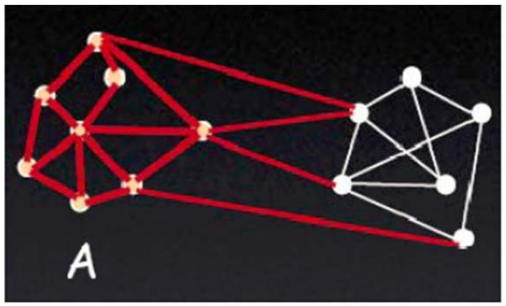
$$d_i = \sum_j w_{i,j}$$

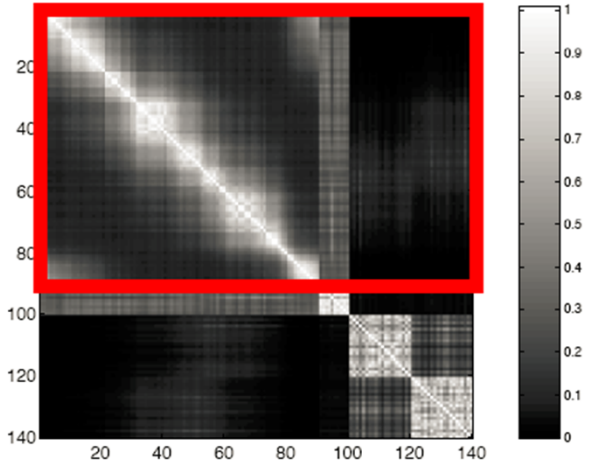




- Volume of a set

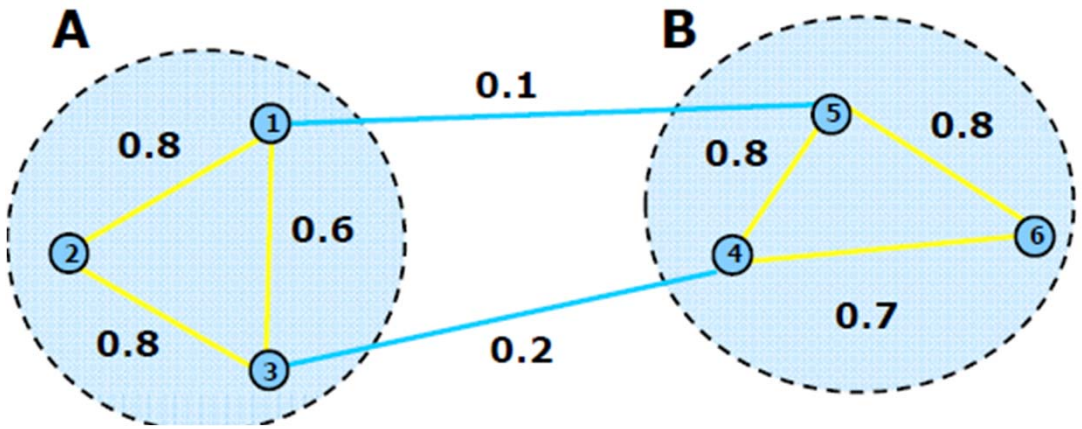
$$vol(A) = \sum_{i \in A} d_i, A \subseteq V$$





Graph Cut

- Consider a partition of the graph into two parts A and B



- ***Cut(A, B)***: sum of the weights of the set of edges that connect the two groups

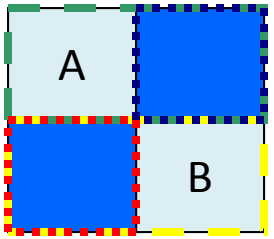
$$cut(A,B) = \sum_{i \in A, j \in B} w_{ij} = \mathbf{0.3}$$

- An intuitive goal is find the partition that minimizes the cut

Normalized Cut

- Consider the connectivity between groups relative to the volume of each group

$$Ncut(A,B) = \frac{cut(A,B)}{Vol(A)} + \frac{cut(A,B)}{Vol(B)}$$



$$Ncut(A,B) = cut(A,B)\frac{Vol(A)+Vol(B)}{Vol(A)Vol(B)}$$

Minimized when Vol(A) and Vol(B) are equal.
Thus encourage balanced cut

Solving NCut

- How to minimize $Ncut$?

  Let W be the similarity matrix, $W(i,j) = W_{i,j}$;
  Let D be the diag. matrix, $D(i,i) = \sum_j W(i,j)$;
  Let x be a vector in $\{1,-1\}^N$, $x(i) = 1 \Leftrightarrow i \in A$.

- With some simplifications, we can show:

$$\min_x Ncut(x) = \min_y \frac{y^T(D-W)y}{y^T Dy}$$

Rayleigh quotient

Subject to: $y^T D\mathbf{1} = 0$ (***y*** *takes discrete values*)

NP-Hard!

Solving NCut

- Relax the optimization problem into the continuous domain by solving generalized eigenvalue system:

$$\min_y y^T(D-W)y \text{ subject to } y^T D y = 1$$

- Which gives: $(D-W)y = \lambda D y$
- Note that $(D-W)1 = 0$, so the first eigenvector is $y_0 = 1$ with eigenvalue 0.
- The second smallest eigenvector is the real valued solution to this problem!!

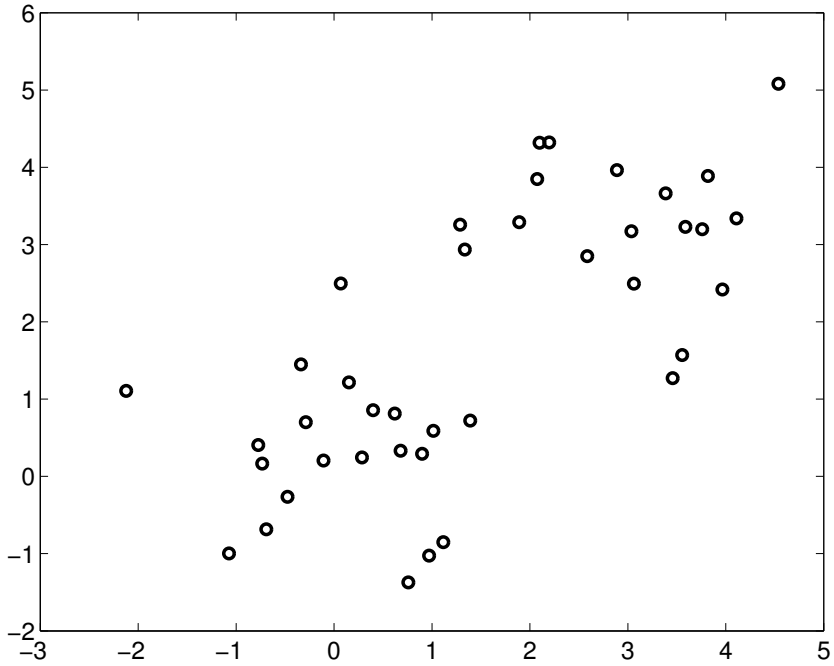
2-way Normalized Cuts

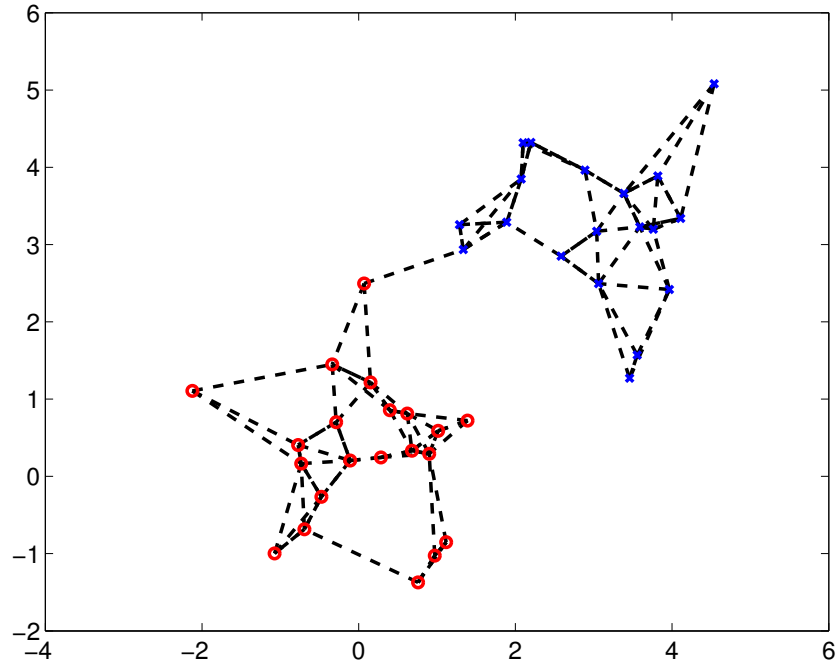
1. Compute the affinity matrix W, compute the degree matrix (D), D is diagonal and $D(i,i) = \sum_{j \in V} W(i,j)$
2. Solve $(D - W)y = \lambda D y$, where $D - W$ is called the Laplacian matrix
3. Use the eigenvector with the second smallest eigen-value to bipartition the graph into two parts.

Creating Bi-partition Using 2nd Eigenvector

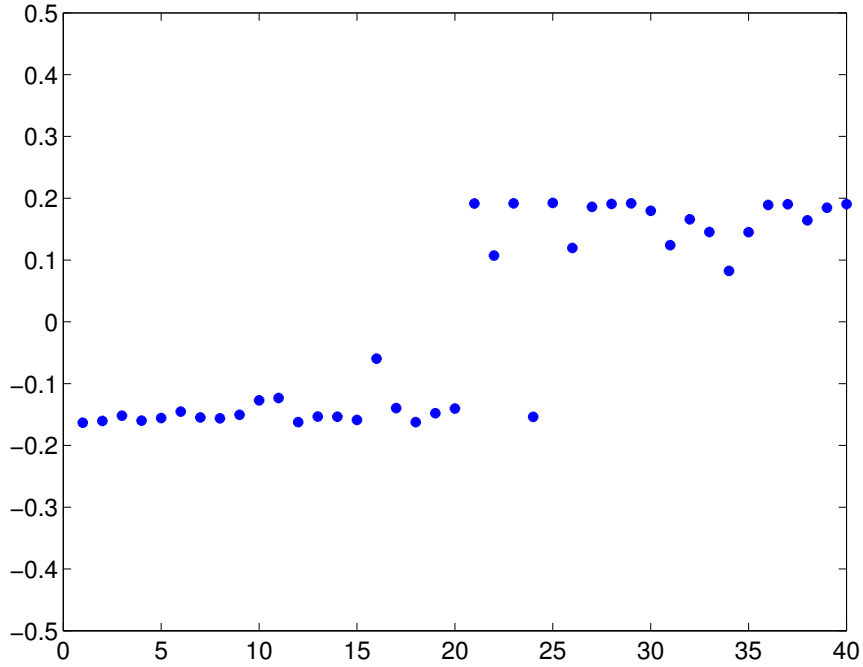
- Sometimes there is not a clear threshold to split based on the second vector since it takes continuous values
- How to choose the splitting point?
  a) Pick a constant value (0, or 0.5).
  b) Pick the median value as splitting point.
  c) Look for the splitting point that has the minimum $Ncut$ value:
     1. Choose n possible splitting points.
     2. Compute $Ncut$ value.
     3. Pick minimum.

Spectral clustering: example





Spectral clustering: example cont'd



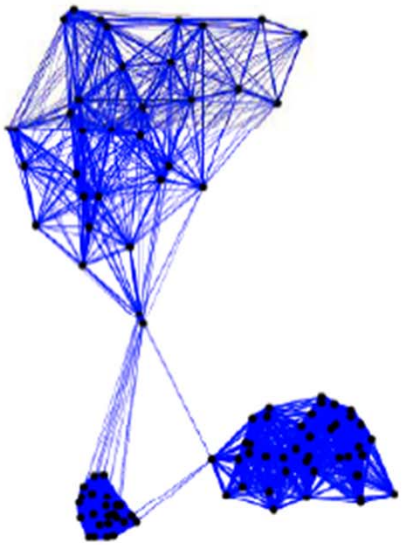
Components of the eigenvector corresponding to the second largest eigenvalue

K-way Partition?

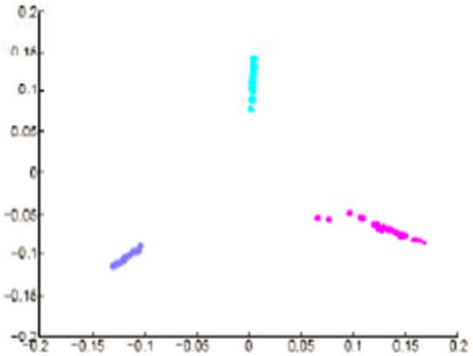
- Recursive bi-partitioning (Hagen et al.,'91)
  - Recursively apply bi-partitioning algorithm in a hierarchical divisive manner.
  - Disadvantages: Inefficient, unstable
- Cluster multiple eigenvectors
  - Build a reduced space from multiple eigenvectors.
  - Commonly used in recent papers
  - A preferable approach… its like doing dimension reduction then k-means

Beyond bi-partition

Graph, 20-NN



$\mathbf{Z}$



Clustering