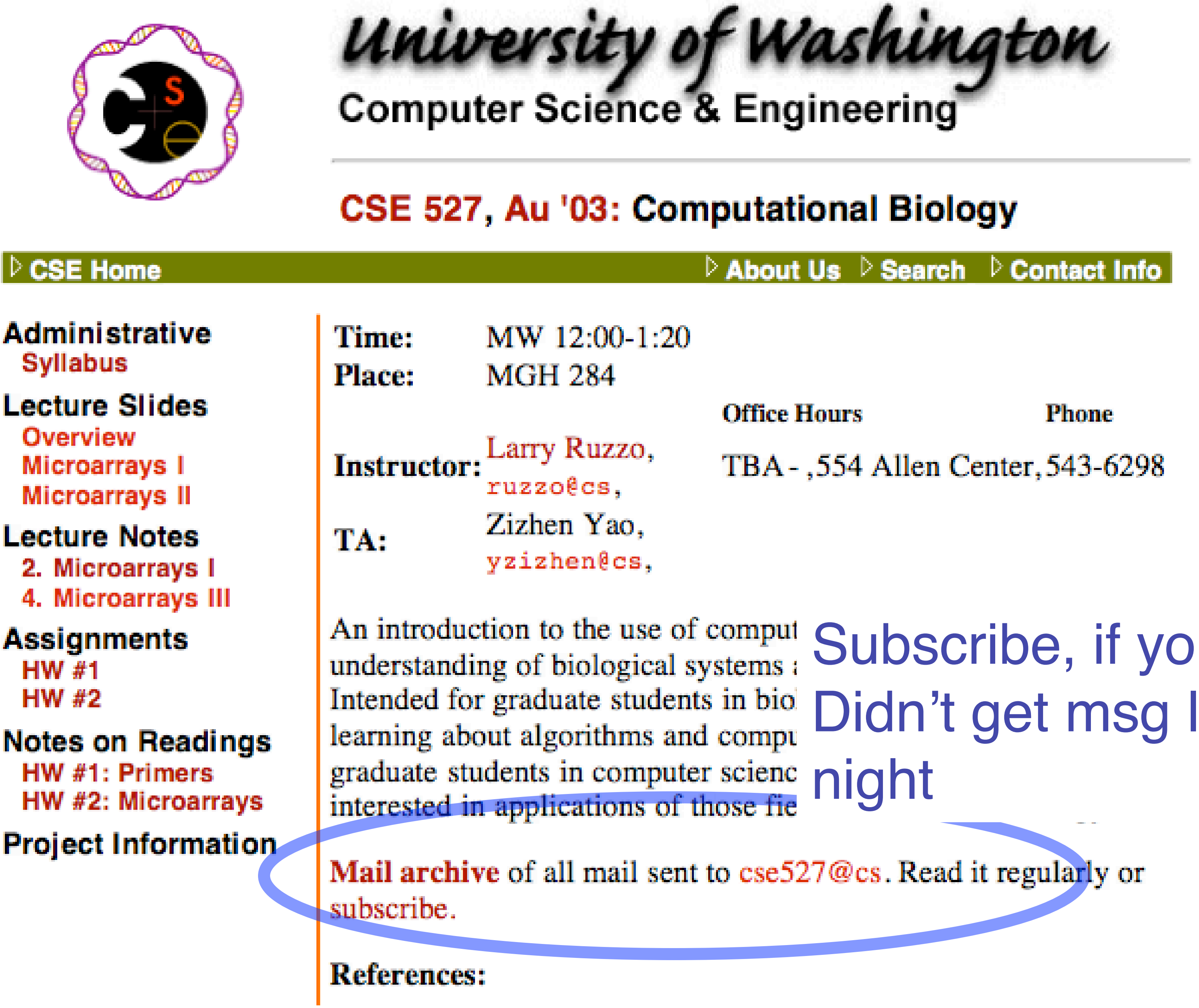
www.cs.washington.edu/527



University of Washington
Computer Science & Engineering

CSE 527, Au '03: Computational Biology

▷ CSE Home ▷ About Us ▷ Search ▷ Contact Info

Administrative
Syllabus

Lecture Slides
Overview
Microarrays I
Microarrays II

Lecture Notes
2. Microarrays I
4. Microarrays III

Assignments
HW #1
HW #2

Notes on Readings
HW #1: Primers
HW #2: Microarrays

Project Information

Time: MW 12:00-1:20
Place: MGH 284

		Office Hours	Phone
Instructor:	Larry Ruzzo, ruzzo@cs,	TBA - ,554 Allen Center,	543-6298
TA:	Zizhen Yao, yzizhen@cs,		

An introduction to the use of comput
understanding of biological systems
Intended for graduate students in bio
learning about algorithms and compu
graduate students in computer scienc
interested in applications of those fie

Mail archive of all mail sent to cse527@cs. Read it regularly or subscribe.

References:

Subscribe, if you Didn't get msg last night

Clustering Expression Data

- Why cluster gene expression data?
  - Tissue classification
  - Find biologically related genes
  - First step in inferring regulatory networks
  - Look for common promoter elements
  - Hypothesis generation
  - One of the tools of choice for expression analysis

Clustering Expression Data

- What has been done?
  - Hierarchical average-link [Eisen et al. 98]
  - Self Organizing Maps (SOM) [Tamayo et al. 99]
  - CAST [Ben-Dor et al. 99]
  - Support Vector Machines (SVM) [Grundy et al. 00]
  - etc., etc., etc.
- Why so many methods?
  - Clustering is NP-hard, even with simple objectives, data
  - Hard problem: high dimensionality, noise, …
  - □ many heuristic, local search, & approximation algorithms
  - No clear winner

Clustering Algorithms

- **Partitional**
  - CAST (Ben-Dor et al. 1999)
  - k-means, variously initialized (Hartigan 1975)
- **Hierarchical**
  - single-link
  - average-link
  - complete-link
- **Random** (as a control)
  - Randomly assign genes to clusters
- Others

The following slides largely from
http://staff.washington.edu/kayee/research.html
Errors are mine.

Clustering 101

Ka Yee Yeung
Center for Expression Arrays
University of Washington

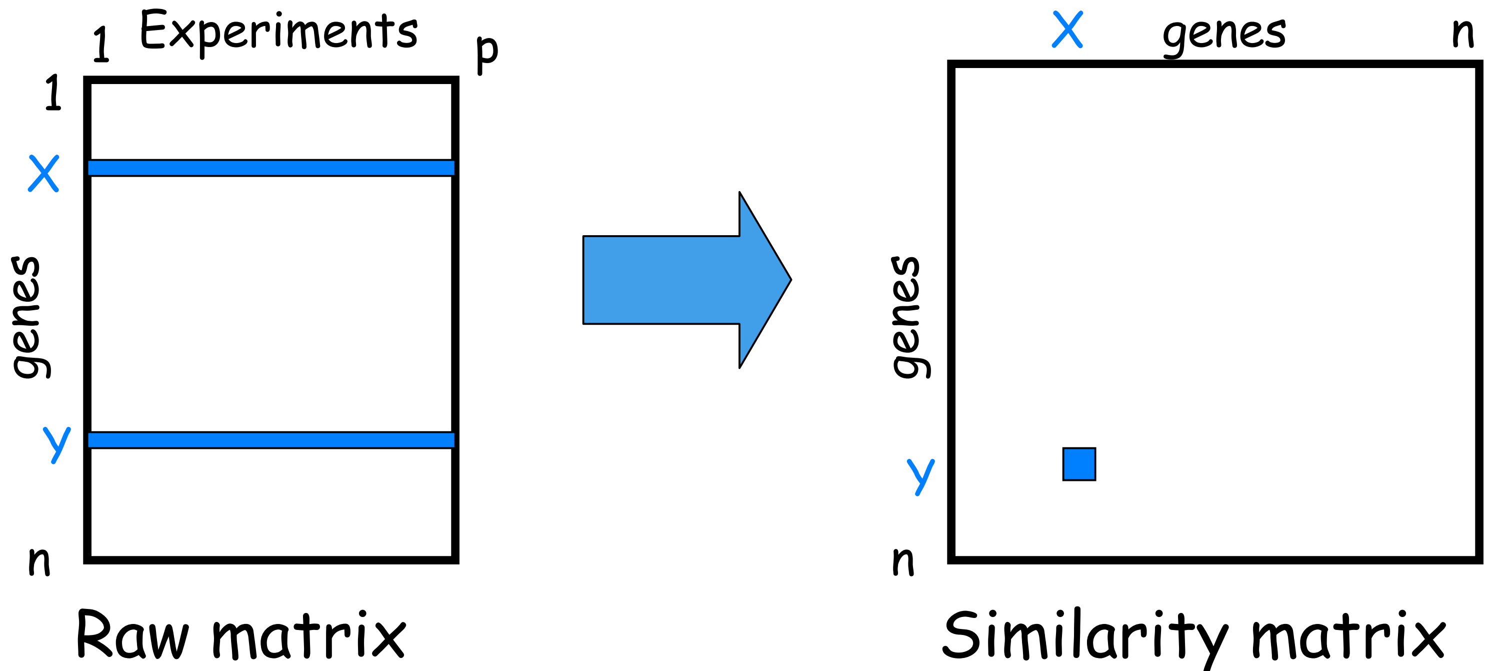
Overview

- What is clustering?
- Similarity/distance metrics
- Hierarchical clustering algorithms
    - Made popular by Stanford, ie. [Eisen *et al*. 1998]
- K-means
    - Made popular by many groups, eg. [Tavazoie *et al*. 1999]
- Self-organizing map (SOM)
    - Made popular by Whitehead, ie. [Tamayo *et al*. 1999]

What is clustering?

- Group *similar* objects together
- Objects in the same cluster (group) are more similar to each other than objects in different clusters
- Data exploratory tool

How to define similarity?



- Similarity metric:
  - A measure of *pairwise* similarity or dissimilarity
  - Examples:
    - Correlation coefficient
    - Euclidean distance

Similarity metrics

- Euclidean distance

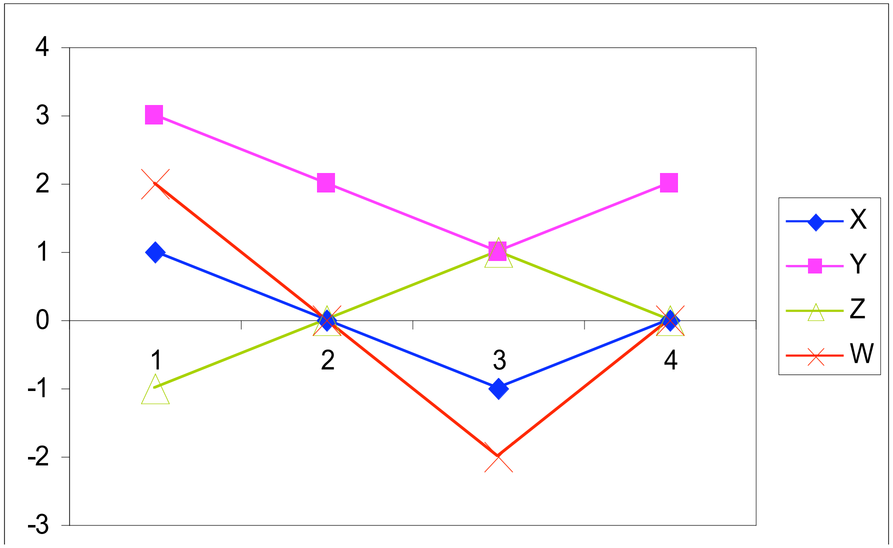
$$\sqrt{\sum_{j=1}^{p}(X[j]-Y[j])^2}$$

- Correlation coefficient

$$\frac{\sum_{j=1}^{p}(X[j]-\overline{X})(Y[j]-\overline{Y})}{\sqrt{\sum_{j=1}^{p}(X[j]-\overline{X})^2\sum_{j=1}^{p}(Y[j]-\overline{Y})^2}},\quad where\ \overline{X}=\frac{\sum_{j=1}^{p}X[j]}{p}$$

Example

X	1	0	-1	0
Y	3	2	1	2
Z	-1	0	1	0
W	2	0	-2	0



Correlation (X,Y) = 1 Distance (X,Y) = 4

Correlation (X,Z) = -1 Distance (X,Z) = 2.83

Correlation (X,W) = 1 Distance (X,W) = 1.41

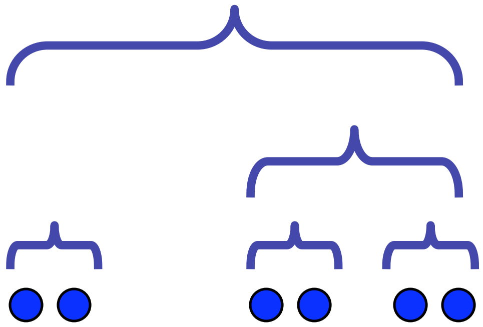
Lessons from the example

- Correlation – direction only
- Euclidean distance – magnitude & direction
- Min # attributes (experiments) to compute pairwise similarity
  - >= 2 attributes for Euclidean distance
  - >= 3 attributes for correlation
- Array data is noisy ➔ need many experiments to robustly estimate pairwise similarity

Clustering algorithms

- Inputs:
  - Raw data matrix or similarity matrix
  - Number of clusters or some other parameters
- Many different classifications of clustering algorithms:
  - Hierarchical vs partitional
  - Heuristic-based vs model-based
  - Soft vs hard

Hierarchical Clustering [Hartigan 1975]

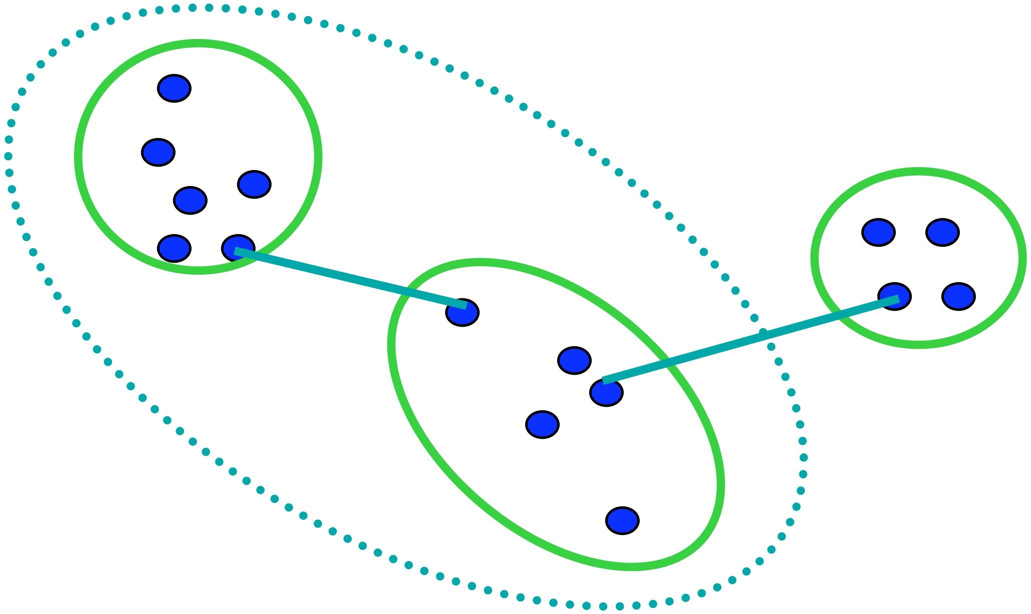


dendrogram

- Agglomerative (bottom-up)
- Algorithm:
  - – Initialize: each item a cluster
  - – Iterate:
    - select two most similar clusters
    - merge them
  - – Halt: when required number of clusters is reached

Hierarchical: Single Link

- cluster similarity = similarity of two most similar members



- Potentially long and skinny clusters

+ Fast

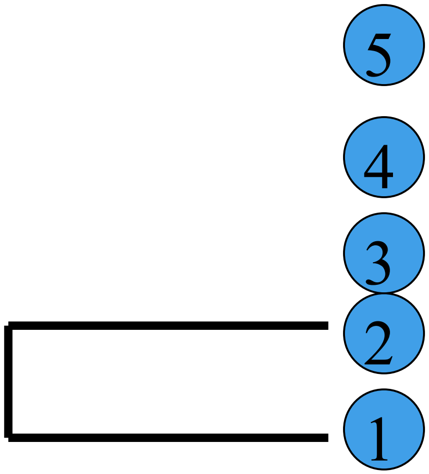
Example: single link

$$\begin{array}{cc} & \begin{array}{ccccc} 1 & 2 & 3 & 4 & 5 \end{array} \\ \begin{array}{c} 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{array} & \left[\begin{array}{ccccc} 0 & & & & \\ 2 & 0 & & & \\ 6 & 3 & 0 & & \\ 10 & 9 & 7 & 0 & \\ 9 & 8 & 5 & 4 & 0 \end{array}\right] \end{array} \Rightarrow \begin{array}{cc} & \begin{array}{cccc} (1,2) & 3 & 4 & 5 \end{array} \\ \begin{array}{c} (1,2) \\ 3 \\ 4 \\ 5 \end{array} & \left[\begin{array}{cccc} 0 & & & \\ 3 & 0 & & \\ 9 & 7 & 0 & \\ 8 & 5 & 4 & 0 \end{array}\right] \end{array}$$

$$d_{(1,2),3} = \min\{ d_{1,3}, d_{2,3} \} = \min\{ 6,3 \} = 3$$

$$d_{(1,2),4} = \min\{ d_{1,4}, d_{2,4} \} = \min\{ 10,9 \} = 9$$

$$d_{(1,2),5} = \min\{ d_{1,5}, d_{2,5} \} = \min\{ 9,8 \} = 8$$

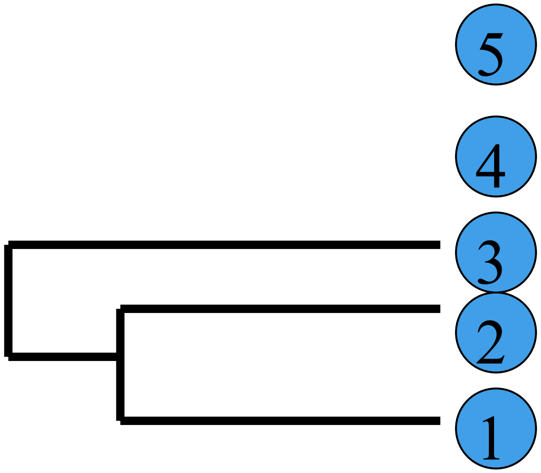


Example: single link

$$
\begin{array}{cccccc}
 & 1 & 2 & 3 & 4 & 5 \\
1 & 0 & & & & \\
2 & 2 & 0 & & & \\
3 & 6 & 3 & 0 & & \\
4 & 10 & 9 & 7 & 0 & \\
5 & 9 & 8 & 5 & 4 & 0
\end{array}
\Rightarrow
\begin{array}{ccccc}
 & (1,2) & 3 & 4 & 5 \\
(1,2) & 0 & & & \\
3 & 3 & 0 & & \\
4 & 9 & 7 & 0 & \\
5 & 8 & 5 & 4 & 0
\end{array}
\Rightarrow
\begin{array}{cccc}
 & (1,2,3) & 4 & 5 \\
(1,2,3) & 0 & & \\
4 & 7 & 0 & \\
5 & 5 & 4 & 0
\end{array}
$$

$$d_{(1,2,3),4} = \min\{ d_{(1,2),4}, d_{3,4}\} = \min\{ 9,7\} = 7$$

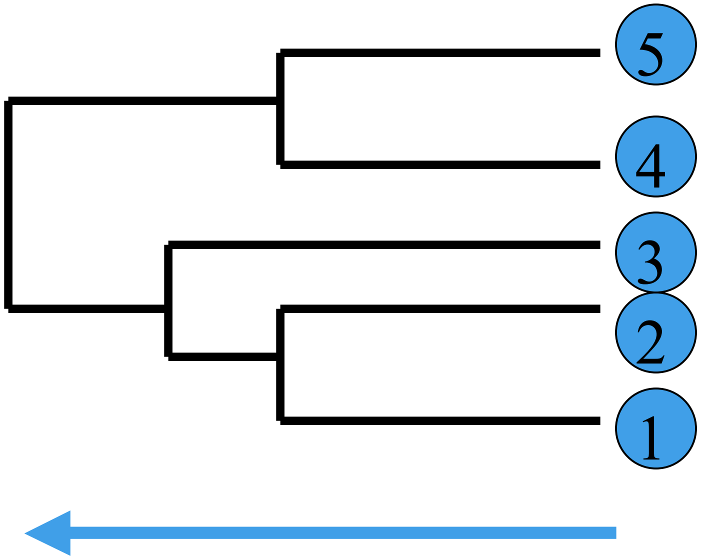
$$d_{(1,2,3),5} = \min\{ d_{(1,2),5}, d_{3,5}\} = \min\{ 8,5\} = 5$$



Example: single link

$$
\begin{array}{c|ccccc}
 & 1 & 2 & 3 & 4 & 5 \\
\hline
1 & 0 & & & & \\
2 & 2 & 0 & & & \\
3 & 6 & 3 & 0 & & \\
4 & 10 & 9 & 7 & 0 & \\
5 & 9 & 8 & 5 & 4 & 0
\end{array}
\Rightarrow
\begin{array}{c|cccc}
 & (1,2) & 3 & 4 & 5 \\
\hline
(1,2) & 0 & & & \\
3 & 3 & 0 & & \\
4 & 9 & 7 & 0 & \\
5 & 8 & 5 & 4 & 0
\end{array}
\Rightarrow
\begin{array}{c|ccc}
 & (1,2,3) & 4 & 5 \\
\hline
(1,2,3) & 0 & & \\
4 & 7 & 0 & \\
5 & 5 & 4 & 0
\end{array}
$$

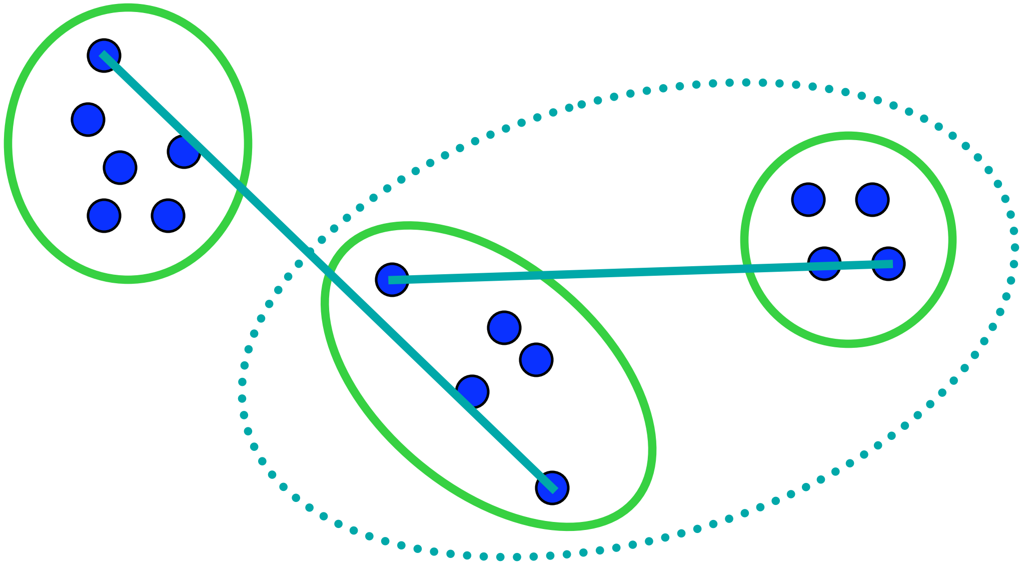
$$d_{(1,2,3),(4,5)} = \min\{ d_{(1,2,3),4}, d_{(1,2,3),5} \} = 5$$



Sometimes drawn to a scale

Hierarchical: Complete Link

- cluster similarity = similarity of two least similar members



+ tight clusters

- slow

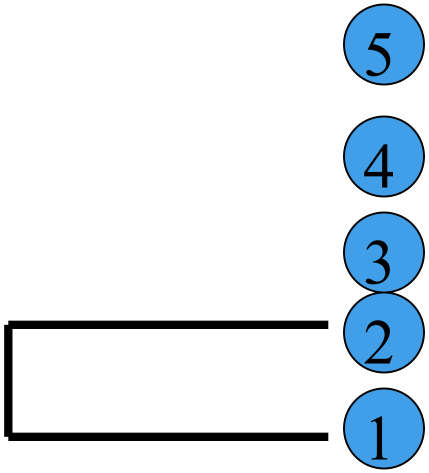
Example: complete link

$$
\begin{array}{c|ccccc}
 & 1 & 2 & 3 & 4 & 5 \\
\hline
1 & 0 & & & & \\
2 & 2 & 0 & & & \\
3 & 6 & 3 & 0 & & \\
4 & 10 & 9 & 7 & 0 & \\
5 & 9 & 8 & 5 & 4 & 0
\end{array}
\Rightarrow
\begin{array}{c|cccc}
 & (1,2) & 3 & 4 & 5 \\
\hline
(1,2) & 0 & & & \\
3 & 6 & 0 & & \\
4 & 10 & 7 & 0 & \\
5 & 9 & 5 & 4 & 0
\end{array}
$$

$$d_{(1,2),3} = \max\{d_{1,3}, d_{2,3}\} = \max\{6,3\} = 6$$

$$d_{(1,2),4} = \max\{d_{1,4}, d_{2,4}\} = \max\{10,9\} = 10$$

$$d_{(1,2),5} = \max\{d_{1,5}, d_{2,5}\} = \max\{9,8\} = 9$$

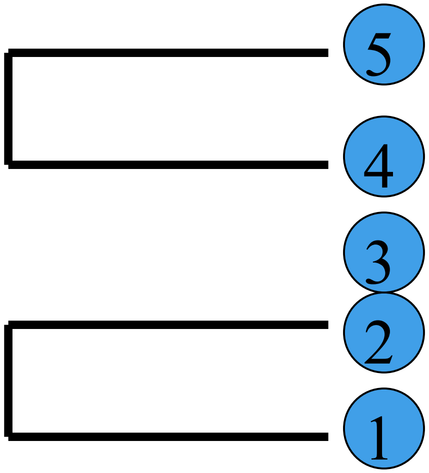


Example: complete link

$$\begin{array}{c|ccccc} & 1 & 2 & 3 & 4 & 5 \\ \hline 1 & 0 & & & & \\ 2 & 2 & 0 & & & \\ 3 & 6 & 3 & 0 & & \\ 4 & 10 & 9 & 7 & 0 & \\ 5 & 9 & 8 & 5 & 4 & 0 \end{array} \Rightarrow \begin{array}{c|cccc} & (1,2) & 3 & 4 & 5 \\ \hline (1,2) & 0 & & & \\ 3 & 6 & 0 & & \\ 4 & 10 & 7 & 0 & \\ 5 & 9 & 5 & 4 & 0 \end{array} \Rightarrow \begin{array}{c|ccc} & (1,2) & 3 & (4,5) \\ \hline (1,2) & 0 & & \\ 3 & 6 & 0 & \\ (4,5) & 10 & 7 & 0 \end{array}$$

$$d_{(1,2),(4,5)} = \max\{d_{(1,2),4}, d_{(1,2),5}\} = \max\{10,9\} = 10$$

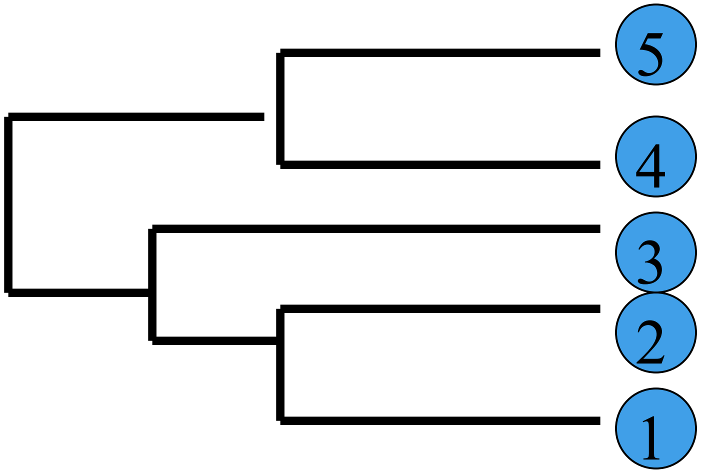
$$d_{3,(4,5)} = \max\{d_{3,4}, d_{3,5}\} = \max\{7,5\} = 7$$



Example: complete link

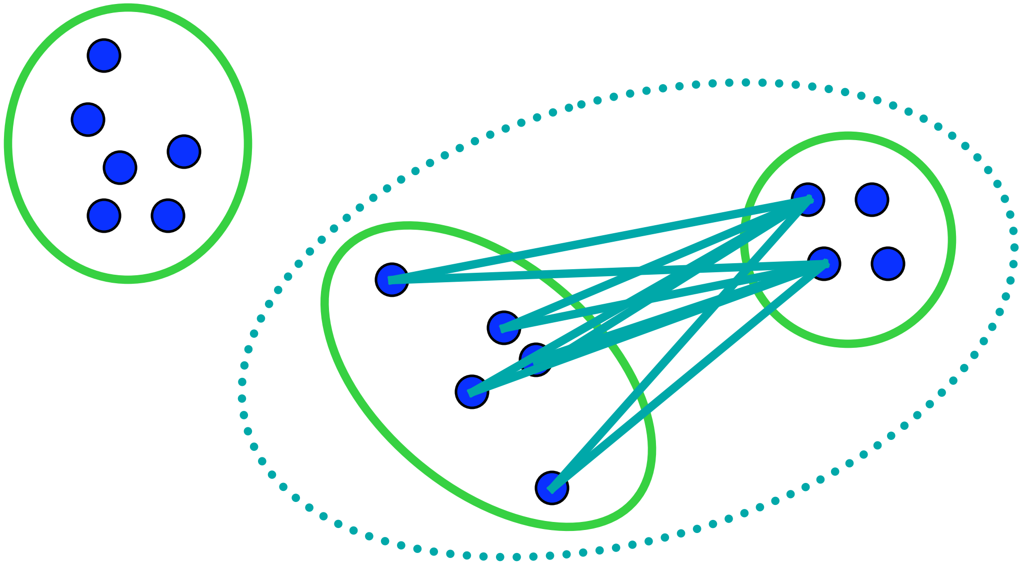
$$\begin{array}{c|ccccc} & 1 & 2 & 3 & 4 & 5 \\ \hline 1 & 0 & & & & \\ 2 & 2 & 0 & & & \\ 3 & 6 & 3 & 0 & & \\ 4 & 10 & 9 & 7 & 0 & \\ 5 & 9 & 8 & 5 & 4 & 0 \end{array} \Rightarrow \begin{array}{c|cccc} & (1,2) & 3 & 4 & 5 \\ \hline (1,2) & 0 & & & \\ 3 & 6 & 0 & & \\ 4 & 10 & 7 & 0 & \\ 5 & 9 & 5 & 4 & 0 \end{array} \Rightarrow \begin{array}{c|ccc} & (1,2) & 3 & (4,5) \\ \hline (1,2) & 0 & & \\ 3 & 6 & 0 & \\ (4,5) & 10 & 7 & 0 \end{array}$$

$$d_{(1,2,3),(4,5)} = \max\{ d_{(1,2),(4,5)}, d_{3,(4,5)} \} = 10$$



Hierarchical: Average Link

- cluster similarity = average similarity of all pairs



+ tight clusters

- slow

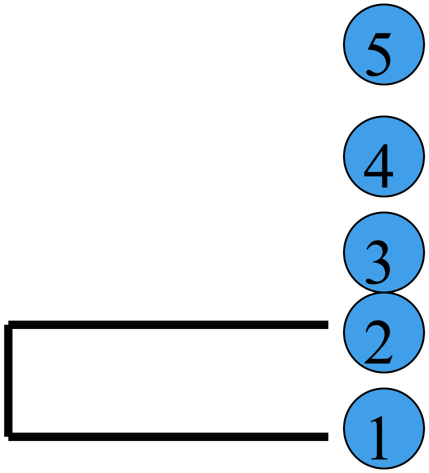
Example: average link

$$
\begin{array}{c|ccccc}
 & 1 & 2 & 3 & 4 & 5 \\
\hline
1 & 0 & & & & \\
2 & 2 & 0 & & & \\
3 & 6 & 3 & 0 & & \\
4 & 10 & 9 & 7 & 0 & \\
5 & 9 & 8 & 5 & 4 & 0
\end{array}
\quad \Rightarrow \quad
\begin{array}{c|cccc}
 & (1,2) & 3 & 4 & 5 \\
\hline
(1,2) & 0 & & & \\
3 & 4.5 & 0 & & \\
4 & 9.5 & 7 & 0 & \\
5 & 8.5 & 5 & 4 & 0
\end{array}
$$

$$d_{(1,2),3} = \frac{1}{2}(d_{1,3} + d_{2,3}) = \frac{6+3}{2} = 4.5$$

$$d_{(1,2),4} = \frac{1}{2}(d_{1,4} + d_{2,4}) = \frac{10+9}{2} = 9.5$$

$$d_{(1,2),5} = \frac{1}{2}(d_{1,5} + d_{2,5}) = \frac{9+8}{2} = 8.5$$

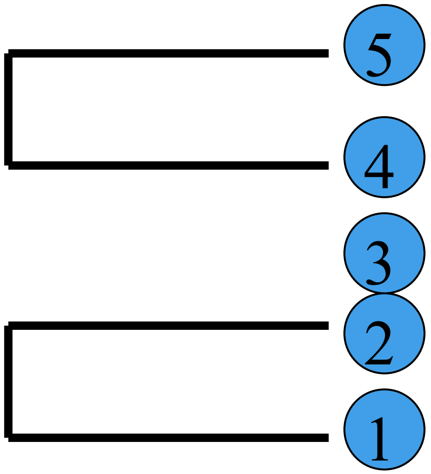


Example: average link

$$\begin{array}{cccccc} & 1 & 2 & 3 & 4 & 5 \\ 1 & 0 & & & & \\ 2 & 2 & 0 & & & \\ 3 & 6 & 3 & 0 & & \\ 4 & 10 & 9 & 7 & 0 & \\ 5 & 9 & 8 & 5 & 4 & 0 \end{array} \Rightarrow \begin{array}{ccccc} & (1,2) & 3 & 4 & 5 \\ (1,2) & 0 & & & \\ 3 & 4.5 & 0 & & \\ 4 & 9.5 & 7 & 0 & \\ 5 & 8.5 & 5 & 4 & 0 \end{array} \Rightarrow \begin{array}{cccc} & (1,2) & 3 & (4,5) \\ (1,2) & 0 & & \\ 3 & 4.5 & 0 & \\ (4,5) & 9 & 6 & 0 \end{array}$$

$$d_{(1,2),(4,5)} = \frac{1}{4}(d_{1,4} + d_{1,5} + d_{2,4} + d_{2,5}) = 9$$

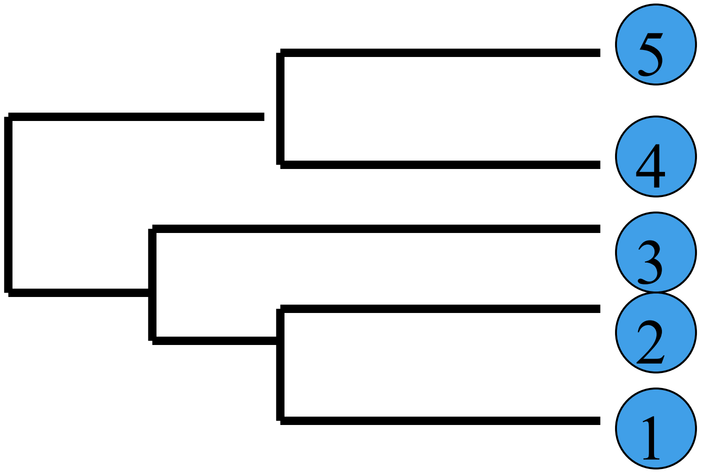
$$d_{3,(4,5)} = \frac{1}{2}(d_{3,4} + d_{3,5}) = 6$$



Example: average link

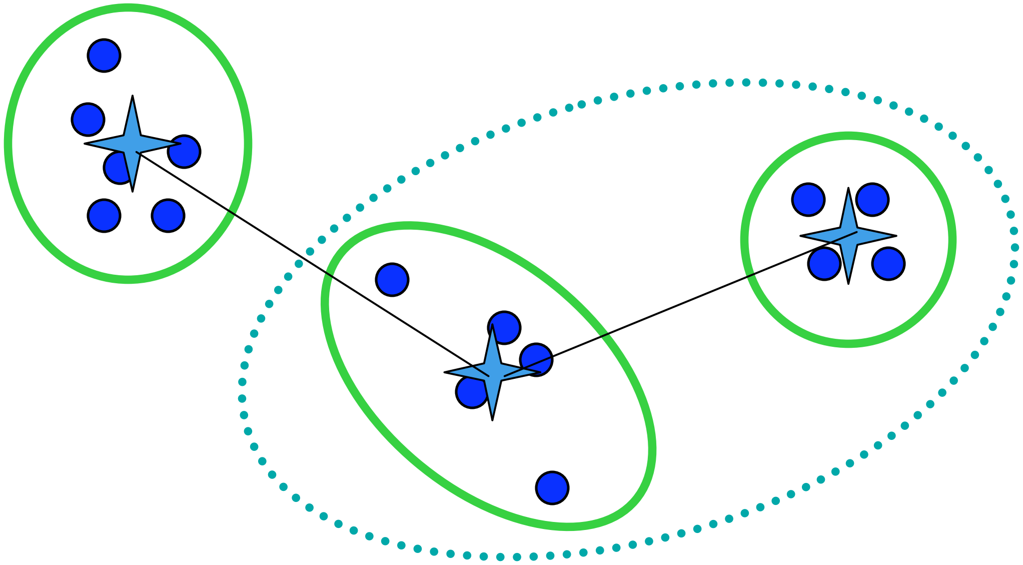
$$
\begin{array}{c|ccccc}
 & 1 & 2 & 3 & 4 & 5 \\
\hline
1 & 0 & & & & \\
2 & 2 & 0 & & & \\
3 & 6 & 3 & 0 & & \\
4 & 10 & 9 & 7 & 0 & \\
5 & 9 & 8 & 5 & 4 & 0
\end{array}
\Rightarrow
\begin{array}{c|cccc}
 & (1,2) & 3 & 4 & 5 \\
\hline
(1,2) & 0 & & & \\
3 & 4.5 & 0 & & \\
4 & 9.5 & 7 & 0 & \\
5 & 8.5 & 5 & 4 & 0
\end{array}
\Rightarrow
\begin{array}{c|ccc}
 & (1,2) & 3 & (4,5) \\
\hline
(1,2) & 0 & & \\
3 & 4.5 & 0 & \\
(4,5) & 9 & 6 & 0
\end{array}
$$

$$d_{(1,2,3),(4,5)} = \frac{1}{6}(d_{1,4} + d_{1,5} + d_{2,4} + d_{2,5} + d_{3,4} + d_{3,5}) = 8$$



Hierarchical: Centroid Link

- cluster centroid = average of all points
- cluster similarity = distance between centroids

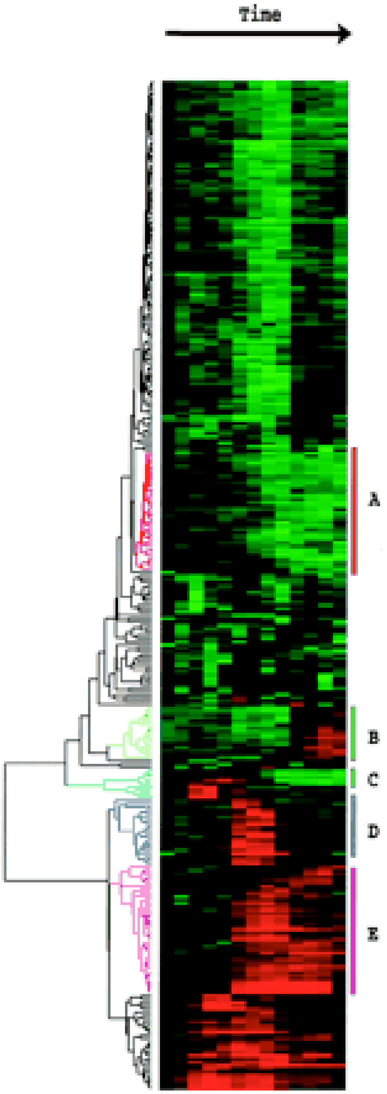


In Expression literature, often called “Average link”

+ faster

- discards shape

Software: TreeView [Eisen et al. 1998]



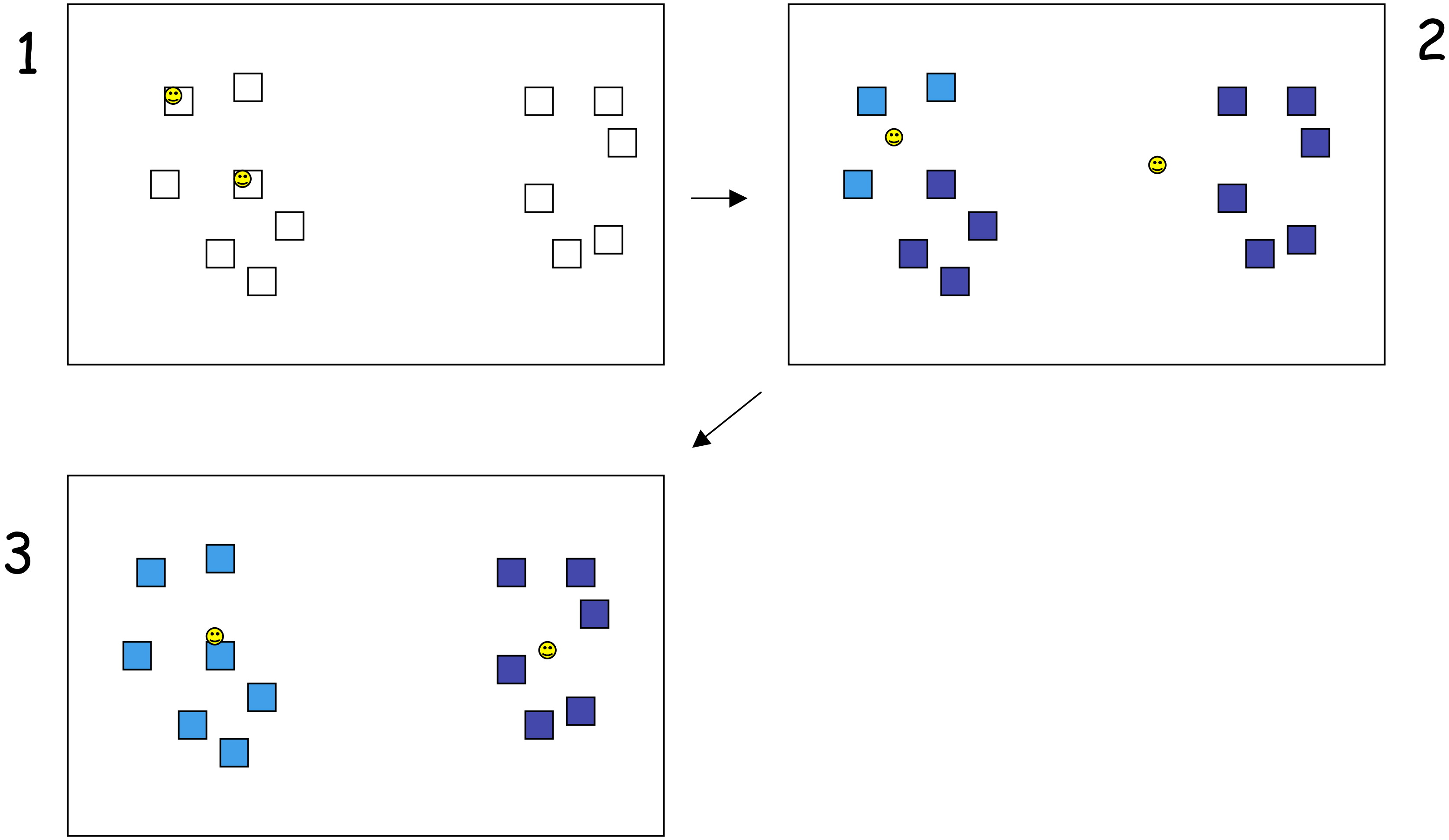
- Fig 1 in Eisen's PNAS 99 paper
- Time course of serum stinulation of primary human fibrolasts
- cDNA arrays with approx 8600 spots
- Similar to average-link
- Free download at: http://rana.lbl.gov/EisenSoftware.htm

Hierarchical divisive clustering algorithms

- Top down
  - Start with all the objects in one cluster
  - Successively split into smaller clusters
- Tend to be less efficient than agglomerative
- Resolver implemented a deterministic annealing approach from [Alon et al. 1999]

Partitional: K-Means

[MacQueen 1965]



Details of k-means

- Iterate until converge:
  - Assign each data point to the closest centroid
  - Compute new centroid

Objective function:

$$\sum_{i=1}^{k}\sum_{x\in C_i}\left(x - Centroid(C_i)\right)^2$$

Minimize

Properties of k-means

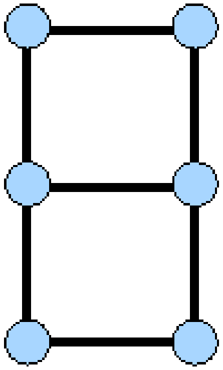
- Fast
- Proved to converge to *local* optimum
- In practice, converge quickly
- Tend to produce spherical, equal-sized clusters
- Related to the model-based approach

Self-organizing maps (SOM)

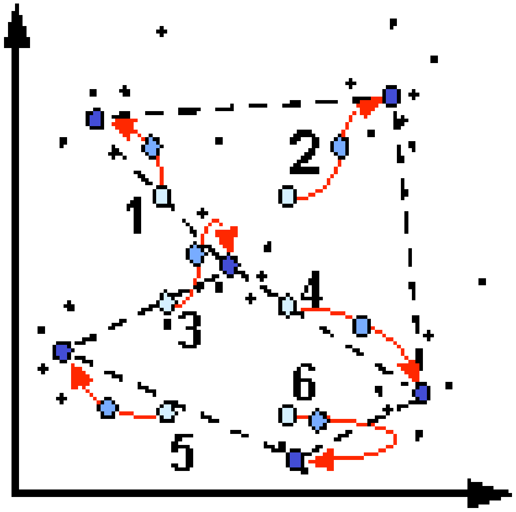
[Kohonen 1995]

- Basic idea:
  - map high dimensional data onto a 2D grid of nodes
  - Neighboring nodes are more similar than points far away

SOM



- Grid (geometry of nodes)
- Input vectors that are close to each other mapped to the same or neighboring nodes



Properties of SOM

- Partial structure
- Easy visualization
- Tons of parameters to tune
- Sensitive to parameters

Summary

- Definition of clustering
- Pairwise similarity:
  - Correlation
  - Euclidean distance
- Clustering algorithms:
  - Hierarchical (single-link, complete-link, average-link)
  - K-means
  - SOM
- Different clustering algorithms → different clusters

Which clustering algorithm should I use?

- Good question ☺
- No definite answer: on-going research
- Feel free to read my thesis:

  http://staff.washington.edu/kayee/research

General Suggestions

- Avoid single-link
- Try:
  - K-means
  - Average-link/ complete-link
- If you are interested in capturing “patterns” of expression, use correlation instead of Euclidean distance
- Visualization of data
  - Eisen-gram
  - Dendrogram
  - PCA, MDS etc

Misc Notes

- Greedy algorithms. Can get trapped in local minima. Can be sensitive to addition of new points, order of points,…

+ simple, intuitive algorithms, reasonably fast, ok on simple data, no obvious preconception about structure

- no model of structure; biases unclear