6.047 / 6.878 Computational Biology: Genomes, Networks, Evolution
Fall 2008

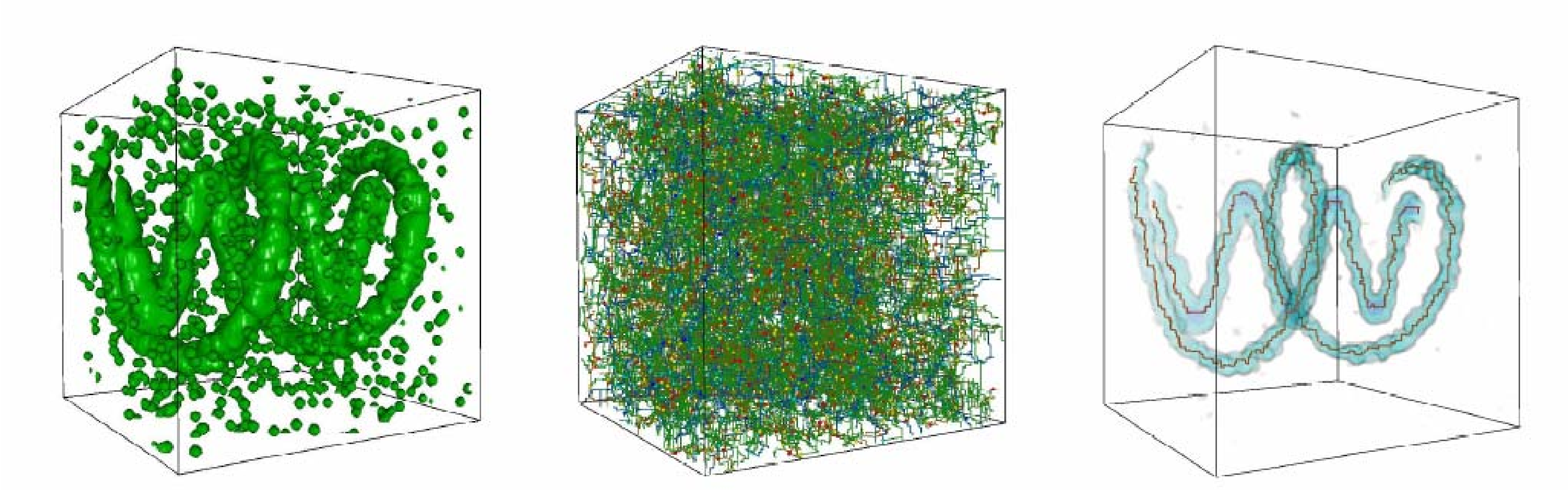
Computational Biology: Genomes, Networks, Evolution

Clustering

Lecture 3

September 16, 2008

Structure in High-Dimensional Data



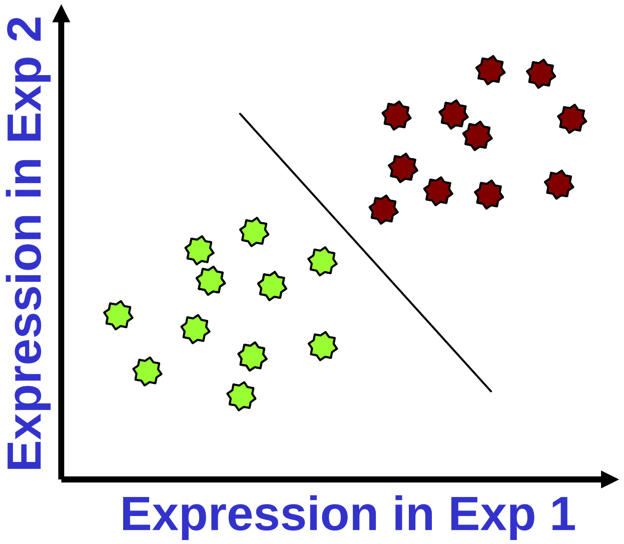
Gyulassy, Atilla, et al. "Topologically Clean Distance Fields." *IEEE Transactions on Visualization and Computer Graphics* 13, no. 6 (2007): 1432-1439.

- Structure can be used to reduce dimensionality of data
- Structure can tell us something useful about the underlying phenomena
- Structure can be used to make inferences about new data

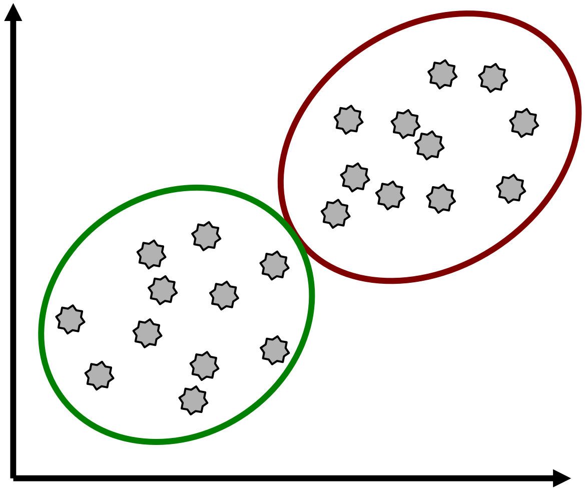
Clustering vs Classification

- Objects characterized by one or more features

- Classification
  - Have labels for some points
  - Want a “rule” that will accurately assign labels to new points
  - Supervised learning



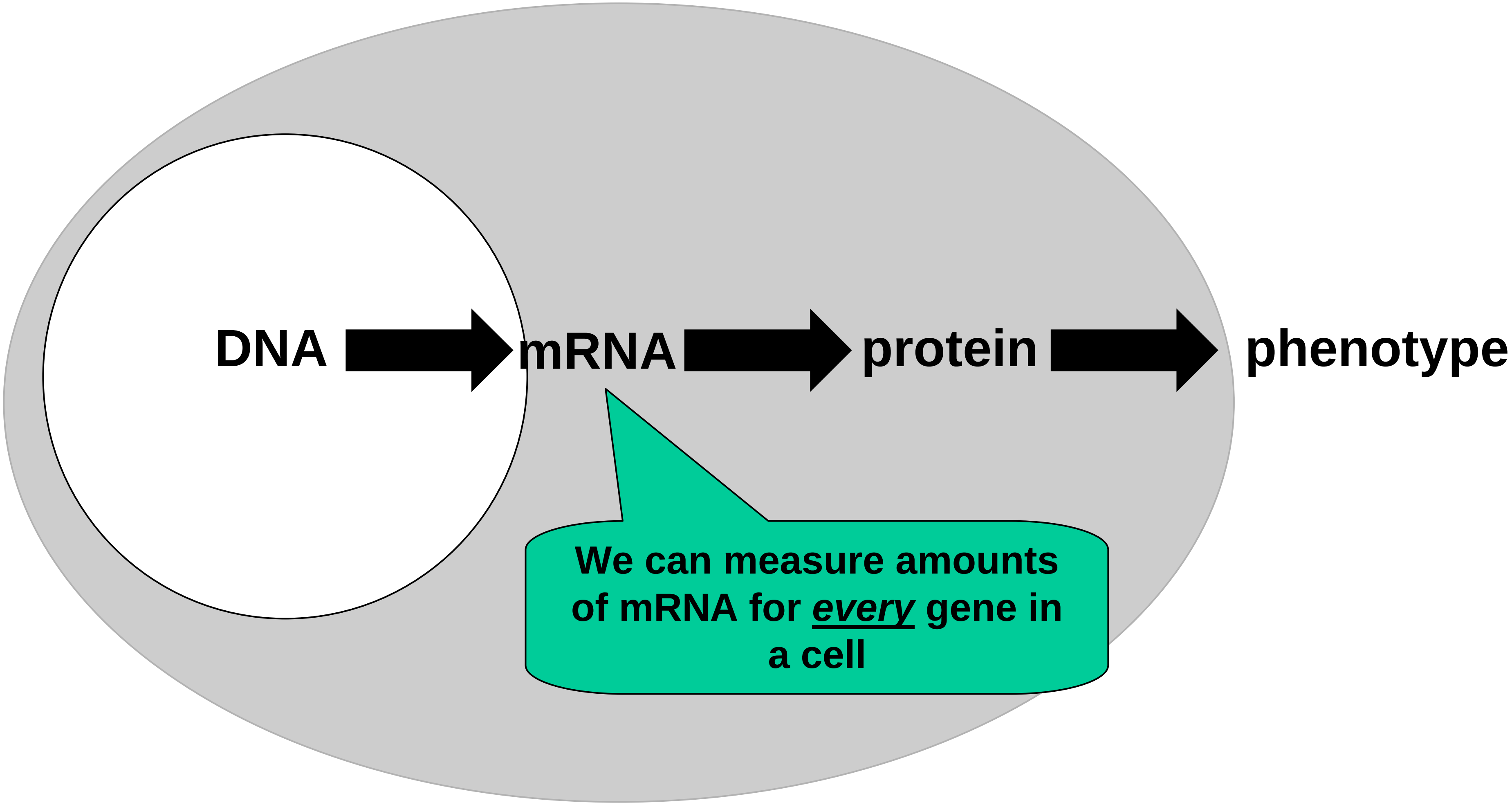
- Clustering
  - No labels
  - Group points into clusters based on how “near” they are to one another
  - Identify structure in data
  - Unsupervised learning



Today

- Microarray Data
- K-means clustering
- Expectation Maximization
- Hierarchical Clustering

Central Dogma



DNA → mRNA → protein → phenotype

We can measure amounts of mRNA for ***every*** gene in a cell

Expression Microarrays

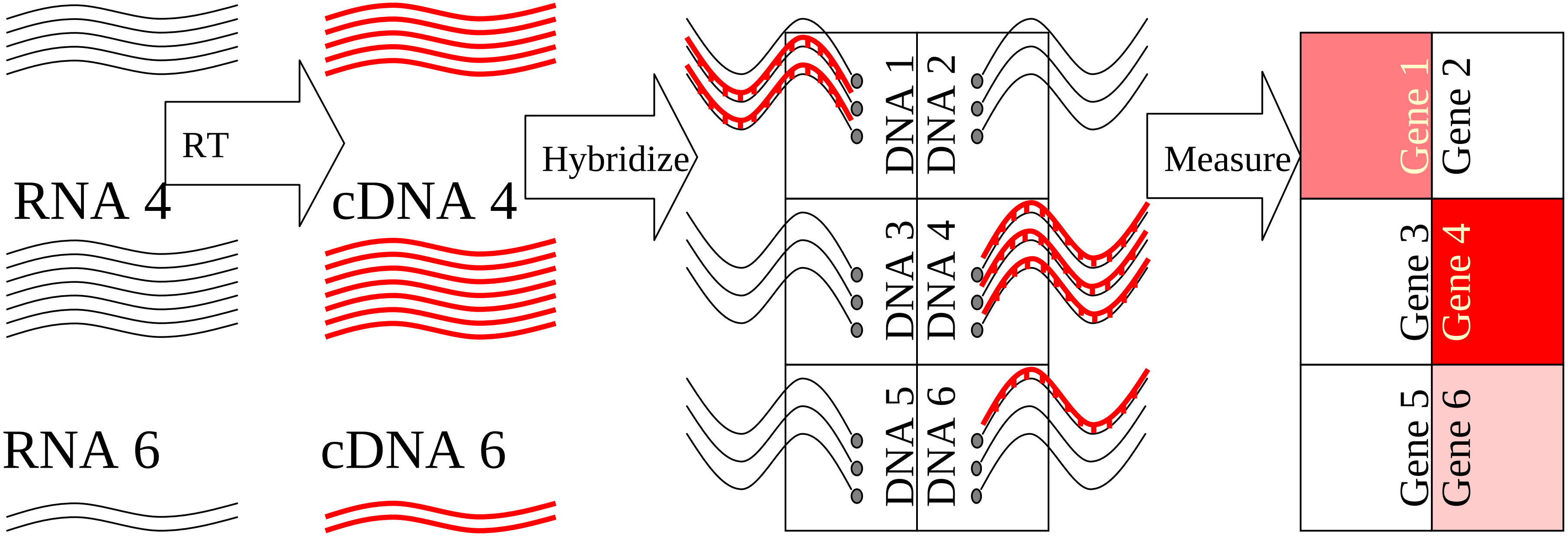
- A way to measure the levels of mRNA in every gene

- Two basic types
  - Affymetrix gene chips
  - Spotted oligonucleotides

- Both work on same principle
  - Put DNA probe on slide
  - Complementary hybridization

Expression Microarrays

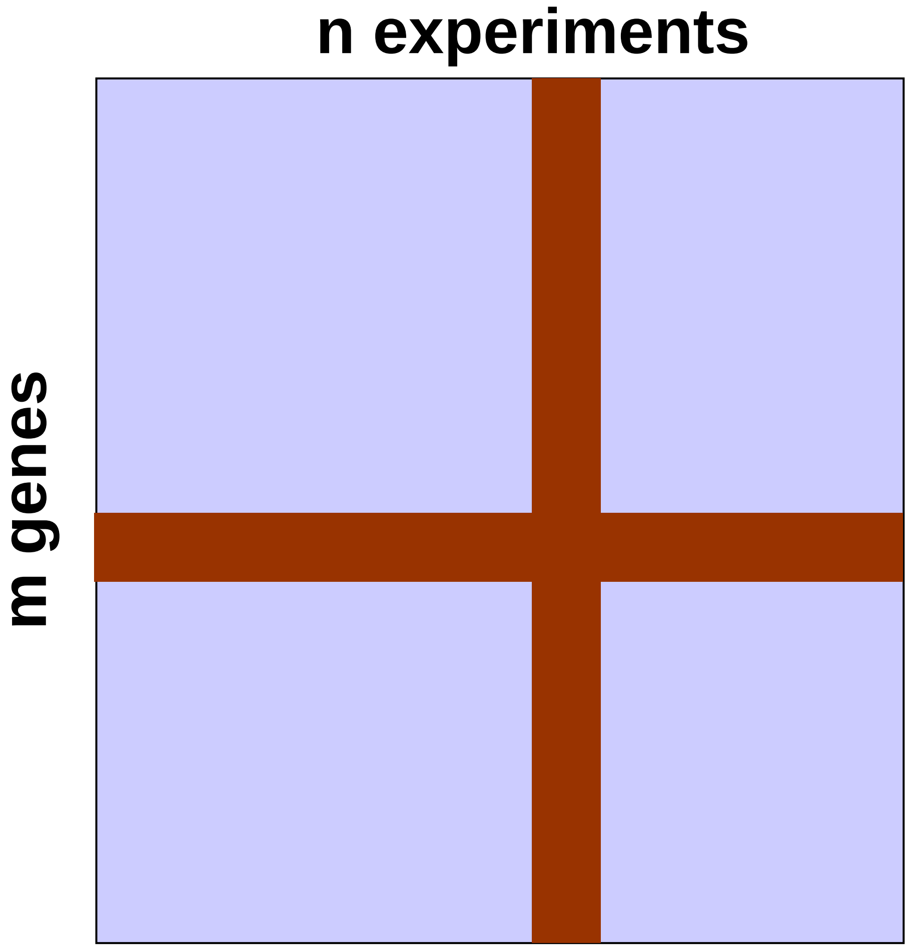
- Measure the level of mRNA messages in a cell



Expression Microarray Data Matrix

- Genes are typically given as rows

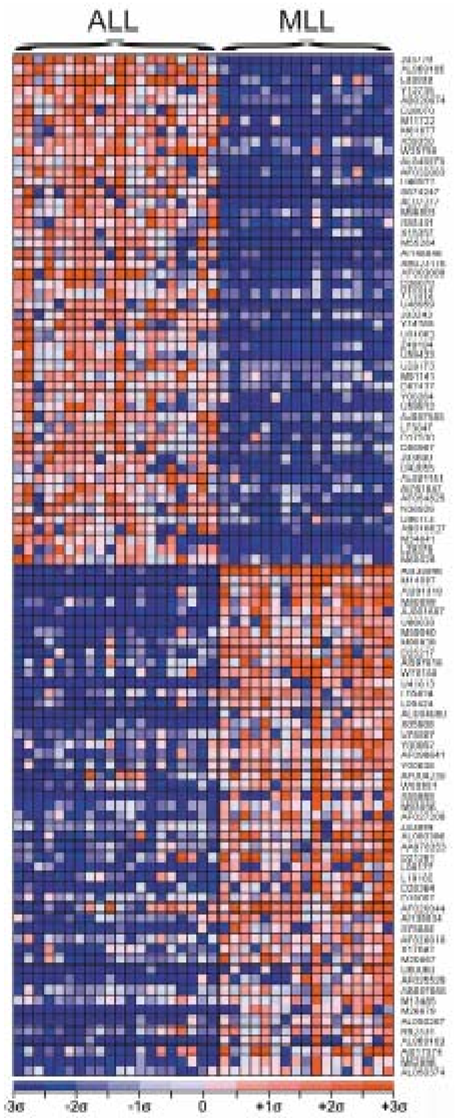
- Experiment are given by columns

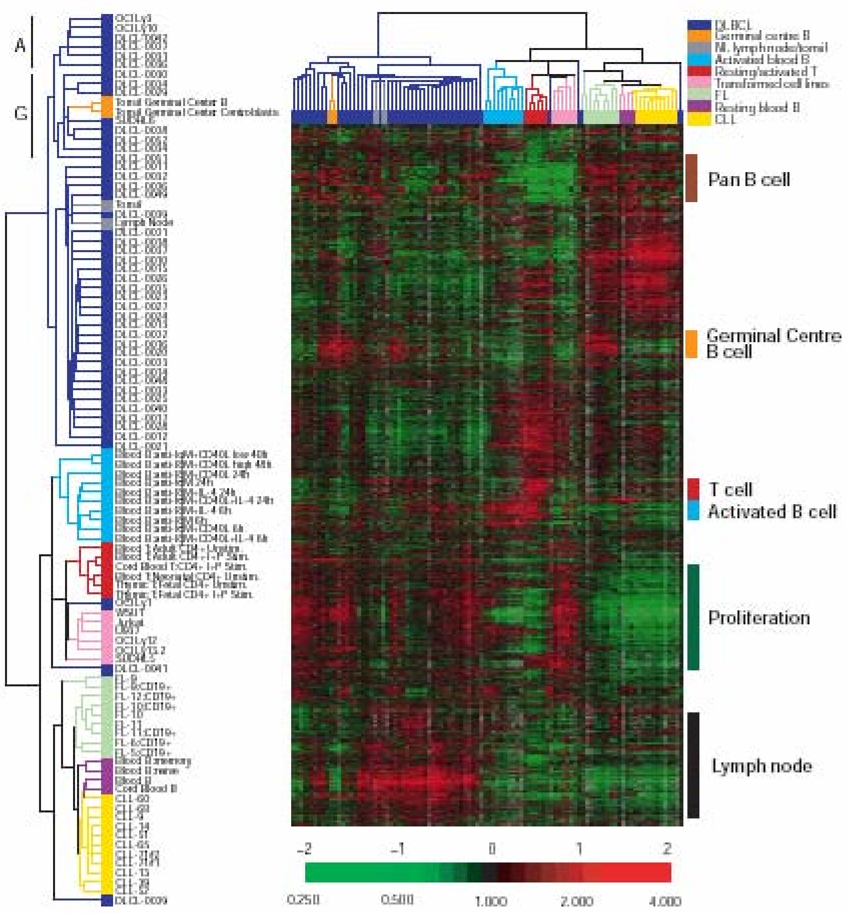


Clustering and Classification in Genomics

- Classification
  - Microarray data: classify cell state (i.e. AML vs ALL) using expression data
  - Protein/gene sequences: predict function, localization, etc.

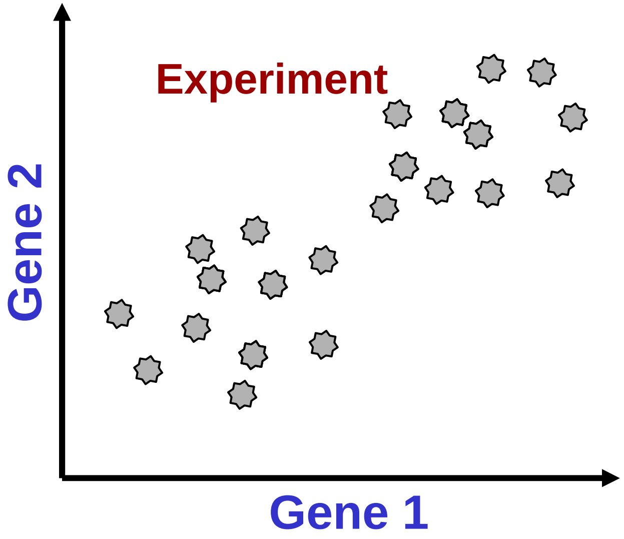
- Clustering
  - Microarray data: groups of genes that share similar function have similar expression patterns – identify regulons
  - Protein sequence: group related proteins to infer function
  - EST data: collapse redundant sequences


ALL
MLL
-3σ
-2σ
-1σ
0
+1σ
+2σ
+3σ

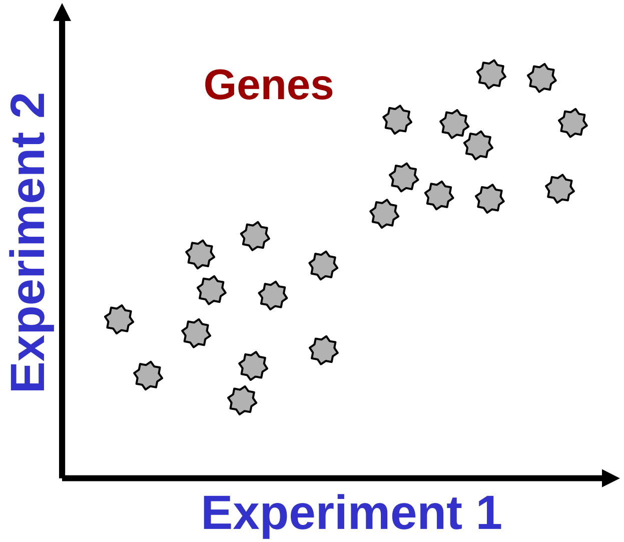

A
G
DLBCL
Germinal centre B
Nl. lymph node/tonsil
Activated blood B
Resting/activated T
Transformed cell lines
FL
Resting blood B
CLL
Pan B cell
Germinal Centre B cell
T cell
Activated B cell
Proliferation
Lymph node
-2
-1
0
1
2
0.250
0.500
1.000
2.000
4.000

Clustering Expression Data

- Cluster Experiments
  - Group by similar expression profiles

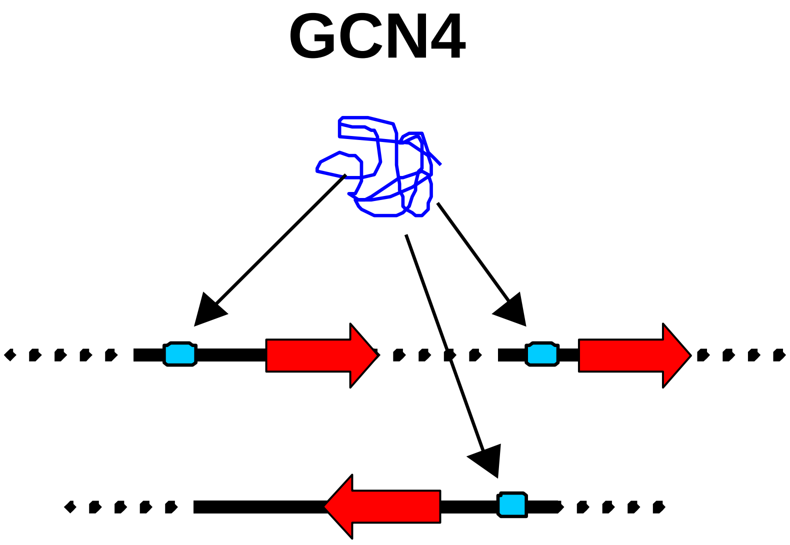


- Cluster Genes
  - Group by similar expression in different conditions



Why Cluster Genes by Expression?

- Data Exploration
  - Summarize data
  - Explore without getting lost in each data point
  - Enhance visualization
- Co-regulated Genes
  - Common expression may imply common regulation
  - Predict *cis*-regulatory promoter sequences
- Functional Annotation
  - Similar function from similar expression



His2	Amino Acids
His3	Amino Acids
Unknown	

Clustering Algorithms

- Partitioning
  - Divides objects into non-overlapping clusters such that each data object is in exactly one subset
- Agglomerative
  - A set of nested clusters organized as a hierarchy

K-Means Clustering

The Basic Idea

- Assume a fixed number of clusters, K
- Goal: create “compact” clusters

More Formally

1. Initialize K centers $\mathbf{u}_k$

For each iteration n until convergence

2. Assign each $\mathbf{x}_i$ the label of the nearest center, where the distance between $\mathbf{x}_i$ and $\mathbf{u}_k$ is

$$d_{i,k} = \left(\mathbf{x}_i - \boldsymbol{\mu}_k\right)^2$$

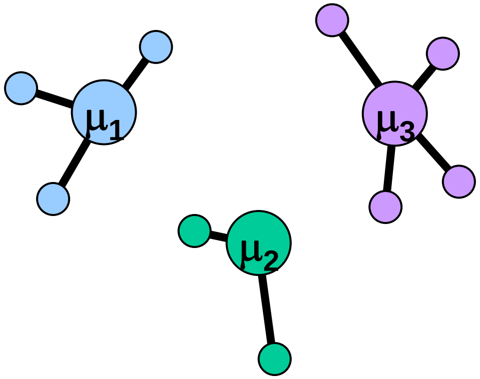
3. Move the position of each $\mathbf{u}_k$ to the centroid of the points with that label

$$\boldsymbol{\mu}_k(n+1) = \sum_{\mathrm{x_i \ with\ label\ j}} \frac{\mathbf{x}_\mathrm{i}}{\left|\mathrm{X}^\mathrm{k}\right|} \ , \ \left|\mathrm{X}^\mathrm{k}\right| = \ \#\mathbf{x}_\mathrm{i} \ \text{ with label k}$$

Cost Criterion

We can think of K-means as trying to create clusters that minimize a cost criterion associated with the size of the cluster

$$\mathrm{COST}(\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3, \ldots, \mathbf{x}_\mathrm{n}) = \sum_{\boldsymbol{\mu}_k} \sum_{\mathbf{x}_i \text{ with label k}} (\mathbf{x}_\mathrm{i} - \boldsymbol{\mu}_k)^2$$



Minimizing this means minimizing each cluster term separately:

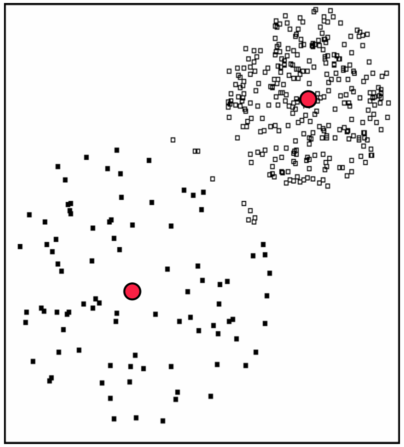
$$\sum_{\mathbf{x}_i \text{ with label k}} (\mathbf{x}_\mathrm{i} - \boldsymbol{\mu}_k)^2$$

Fuzzy K-Means

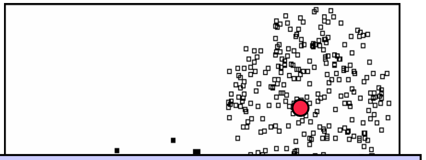
- Initialize K centers $\mathbf{u}_k$
- For each point calculate the probability of membership for each category

$$\mathrm{P}(\text{label K} \mid \mathbf{x}_\mathrm{i}, \boldsymbol{\mu}_k)$$

- Move the position of each $\mathbf{u}_k$ to the weighted centroid :



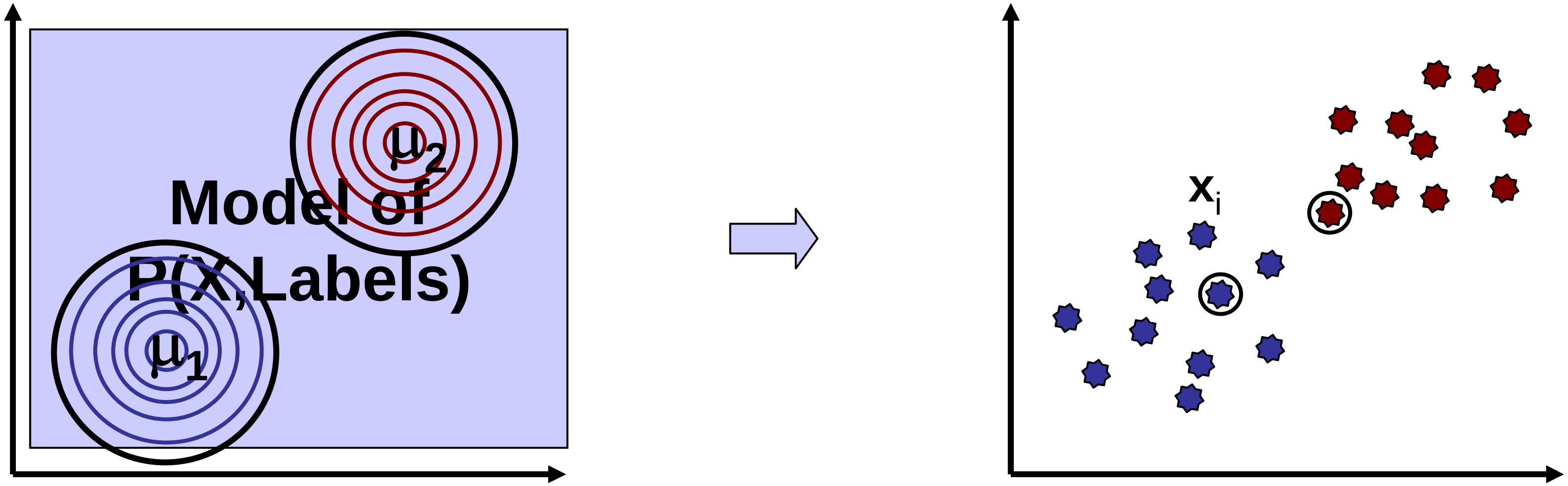
K-means



Of course, K-Means just special case where

$$\mathrm{P}(\text{label K} \mid \mathbf{x}_\mathrm{i}, \boldsymbol{\mu}_k) = \begin{cases} 1 & \text{if } \mathbf{x}_\mathrm{i} \text{ is closest to } \boldsymbol{\mu}_k \\ 0 & \textbf{otherwise} \end{cases}$$

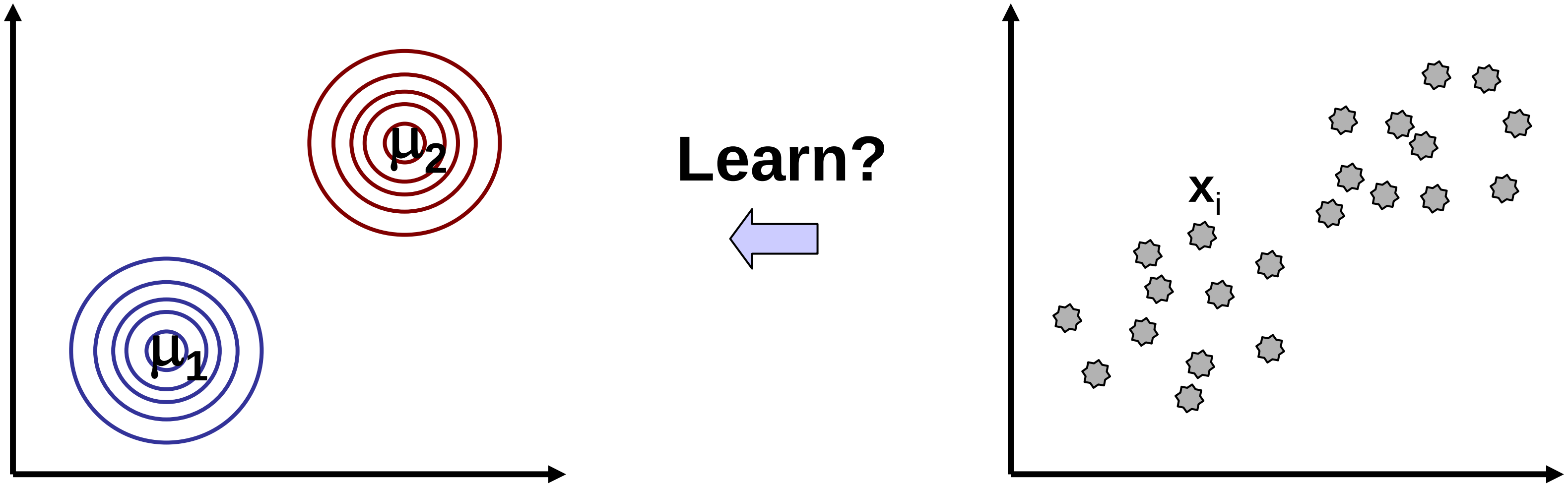
K-Means as a Generative Model



Samples drawn from two equally normal distributions with unit variance - a *Gaussian Mixture Model*

$$P\left(\mathbf{x}_i \mid \mathbf{u}_j\right) = \frac{1}{\sqrt{2\pi}} \exp\left\{ -\frac{\left(\mathbf{x}_i - \mathbf{u}_j\right)^2}{2} \right\}$$

Unsupervised Learning



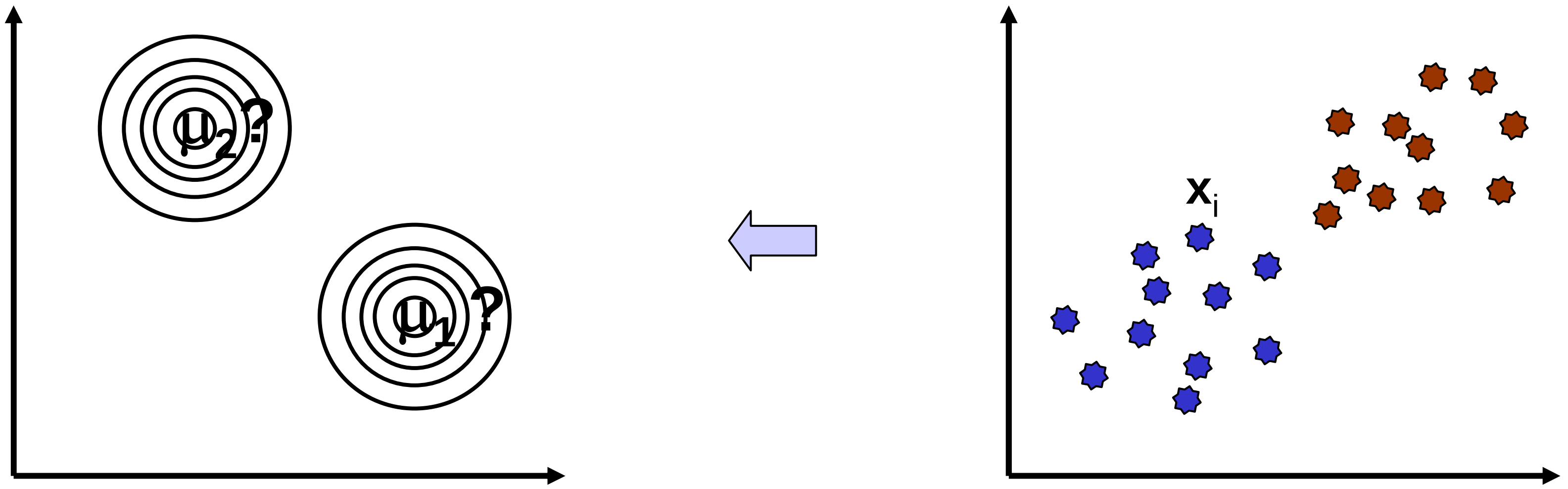
Samples drawn from two equally normal distributions with unit variance - a *Gaussian Mixture Model*

$$P\left(\mathbf{x}_i \mid \mathbf{u}_j\right) = \frac{1}{\sqrt{2\pi}} \exp\left\{ -\frac{\left(\mathbf{x}_i - \mathbf{u}_j\right)^2}{2} \right\}$$

If We Have Labeled Points

Need to estimate unknown gaussian centers from data

In general, how could we do this?
How could we “estimate” the “best” $u_{k?}$



Choose u_k to maximize probability of model

If We Have Labeled Points

Need to estimate unknown gaussian centers from data

In general, how could we do this?
How could we “estimate” the “best” $u_{k?}$

Given a set of $\mathbf{x}_i$, all with label k, we can find the maximum likelihood μ_k from

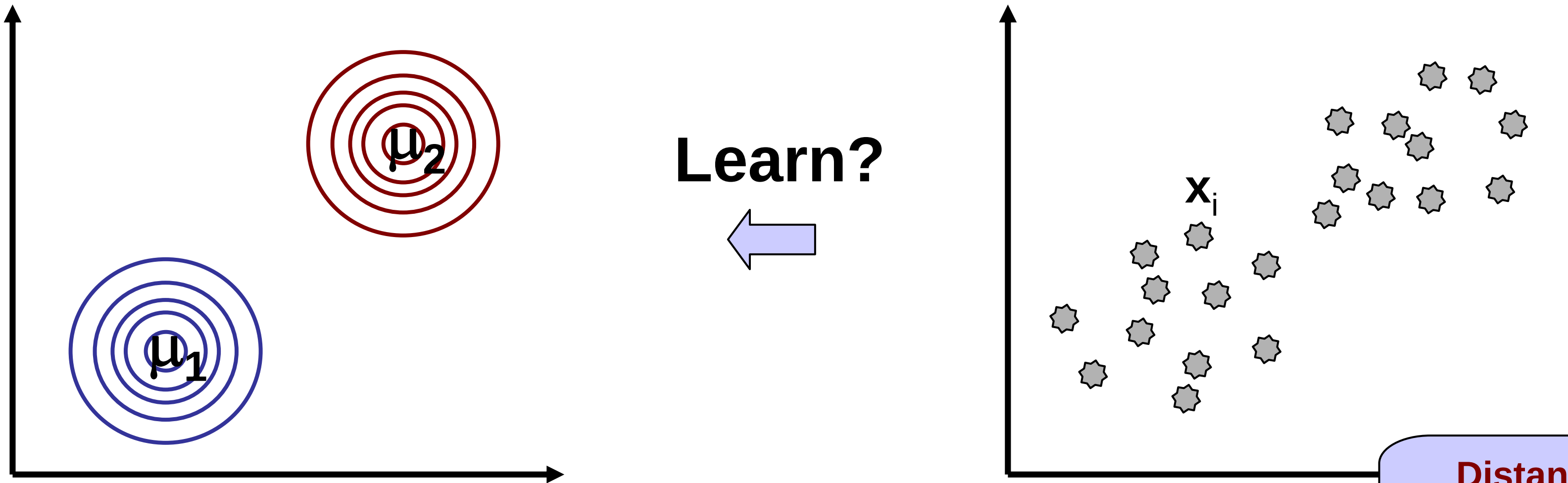
$$\arg\max_{\boldsymbol{\mu}} \left\{ \log \prod_{\mathrm{i}} \mathrm{P}\left(\mathbf{x}_i \mid \boldsymbol{\mu}\right) \right\} = \arg\max_{\boldsymbol{\mu}} \sum_i \left\{ -\frac{1}{2}\left(\mathbf{x}_i - \mathbf{u}\right)^2 + \log\left(\frac{1}{\sqrt{2\pi}} \right) \right\}$$

$$= \arg\min_{\boldsymbol{\mu}} \sum_i \left(\mathbf{x}_i - \mathbf{u}\right)^2$$

Solution is the centroid of the x_i

If We Know Cluster Centers

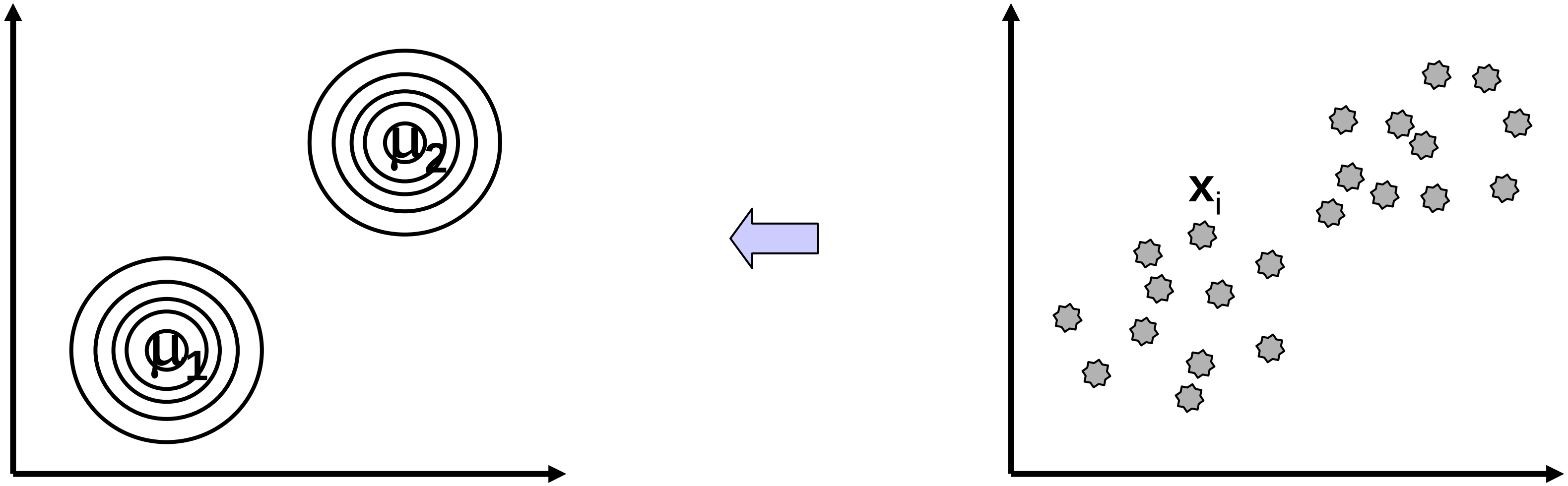
Need to estimate labels for the data



Distance measure for K-means

$$\arg\max_{k} \; \mathrm{P}_{k}\left(\mathbf{x}_i \mid \boldsymbol{\mu}_i\right) = \arg\max_{k} \; \frac{1}{\sqrt{2\pi}} \exp\left\{ -\frac{\left(\mathbf{x}_i - \mathbf{u}_k\right)^2}{2} \right\} = \arg\min_{k} \left(\mathbf{x}_i - \mathbf{u}_k\right)^2$$

What If We Have Neither?



An idea:

1. **Imagine we start with some u_k^0**
2. **We *could* calculate the most likely labels for x_i^0 given these u_k^0**
3. **We *could* then use these labels to choose u_k^1**
4. **And iterate (to convergence)**

Expectation Maximization (EM)

1. **Initialize parameters**

2. **E Step Estimate probability of hidden labels , Q, given parameters and sequence**

$$Q = P(label_i \mid x, u_k^{t-1})$$

3. **M Step Choose new parameters to maximize expected likelihood of parameters given Q**

$$u_k^{\,t} = \arg\max_{u} E_Q\left[\log P(labels \mid x, u_k^{\,t-1})\right]$$

4. **Iterate**

P(x|Model) *guaranteed* to increase each iteration

Expectation Maximization (EM)

Remember the basic idea!

1. **Use model to estimate (distribution of) missing data**
2. **Use estimate to update model**
3. **Repeat until convergence**

Model is the gaussian distributions

Missing data are the data point labels

Revisiting K-Means

Generative Model Perspective

1. Initialize K centers $\mathbf{u}_k$

2. Assign each $\mathbf{x}_i$ the label of the nearest center, where the distance between $\mathbf{x}_i$ and $\mathbf{u}_k$ is

$$d_{i,k} = \left(\mathbf{x}_i - \boldsymbol{\mu}_k\right)^2$$

⇨ **The most likely label k for a point $\mathbf{x}_i$**

3. Move the position of each $\mathbf{u}_k$ to the centroid of the points with that label

⇨ **Maximum likelihood parameter μ_k given most likely label**

4. Iterate

Revisiting K-Means

1. Initialize K centers $\mathbf{u}_k$

2. Assign each $\mathbf{x}_i$ the label of the nearest center, where the distance between $\mathbf{x}_i$ and $\mathbf{u}_k$ is

$$d_{i,k} = \left(\mathbf{x}_i - \boldsymbol{\mu}_k\right)^2$$

3. Move the position of each $\mathbf{u}_k$ to the centroid of the points with that label

4. Iterate

Generative Model Perspective

1. Initialize parameters

2. E Step Estimate most likely missing label given previous parameter

3. M Step Choose new parameters to maximize likelihood of parameters given estimated labels

4. Iterate

Revisiting K-Means

This is analogous to Viterbi Learning from HMMs

1. Initialize K centers $\mathbf{u}_k$
2. Assign each $\mathbf{x}_i$ the label of the nearest center, where the
3.

that label

4. Iterate

Analogy with HMM is to use Viterbi to find most likely missing *path* labels

(see Durbin book)

1. Initialize parameters
2. E Step Estimate most likely missing label given previous parameter
3. M Step Choose new parameters to maximize likelihood of parameters given estimated labels
4. Iterate

Revisting Fuzzy K-Means

Recall that instead of assigning each point x_i to a label k, we calculate the probability of each label for that point (fuzzy membership):

$$\mathrm{P}(\mathrm{label\ K} \,|\, \mathbf{x}_{\mathrm{i}}, \boldsymbol{\mu}_k)$$

Recall that given a set of x_i, all with label k, we select a new μ_k with the update:

Looking at case b=1

$$\boldsymbol{\mu}_k(n+1) = \sum_{\mathrm{x_i\ \ with\ label\ j}} \mathbf{x}_{\mathrm{i}}\ \mathrm{P}(\boldsymbol{\mu}_k \,|\, \mathbf{x}_{\mathrm{i}}) \Big/ \sum_{\mathrm{x_i\ \ with\ label\ j}} \mathrm{P}(\boldsymbol{\mu}_k \,|\, \mathbf{x}_{\mathrm{i}})^{b}$$

It can be shown that this update rule follows from assuming the gaussian mixture generative models and performing *Expectation-Maximization*

Revisiting Fuzzy K-Means

This is analogous to <u>Baum Welch from HMMs</u>

1. Initialize K centers $\mathbf{u}_k$
2. For each point calculate the probability of membership for each category

$$\text{P(label K} \mid \mathbf{x}_\text{i}, \boldsymbol{\mu}_k)$$

3. Move the position of each $\mathbf{u}_k$ to the weighted centroid :

$$\boldsymbol{\mu}_k(n+1) = \sum_{\text{x}_\text{i} \text{ with label j}} \mathbf{x}_\text{i} \, \text{P}(\boldsymbol{\mu}_k \mid \mathbf{x}_\text{i})^b \Big/ \sum_{\text{x}_\text{i} \text{ with label j}} \text{P}(\boldsymbol{\mu}_k \mid \mathbf{x}_\text{i})^b$$

4. Iterate

1. Initialize parameters
2. E Step Estimate <u>probability over missing labels</u> given previous parameter
3. M Step Choose new parameters to <u>maximize expected likelihood</u> of parameters given estimated labels
4. Iterate

K-Means, Viterbi learning & EM

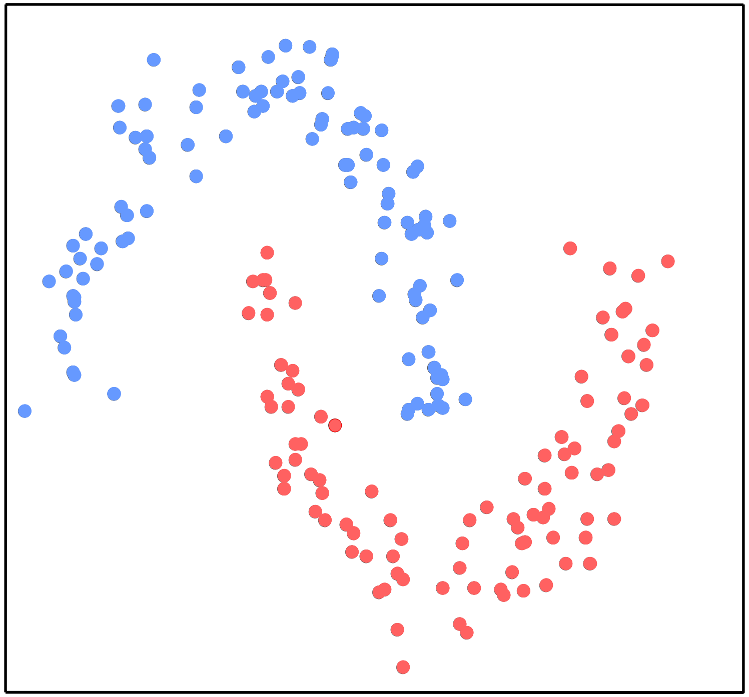
K-Means and Fuzzy K-means are two related methods that can be seen performing unsupervised learning on a gaussian mixture model

Reveal assumptions about underlying data model

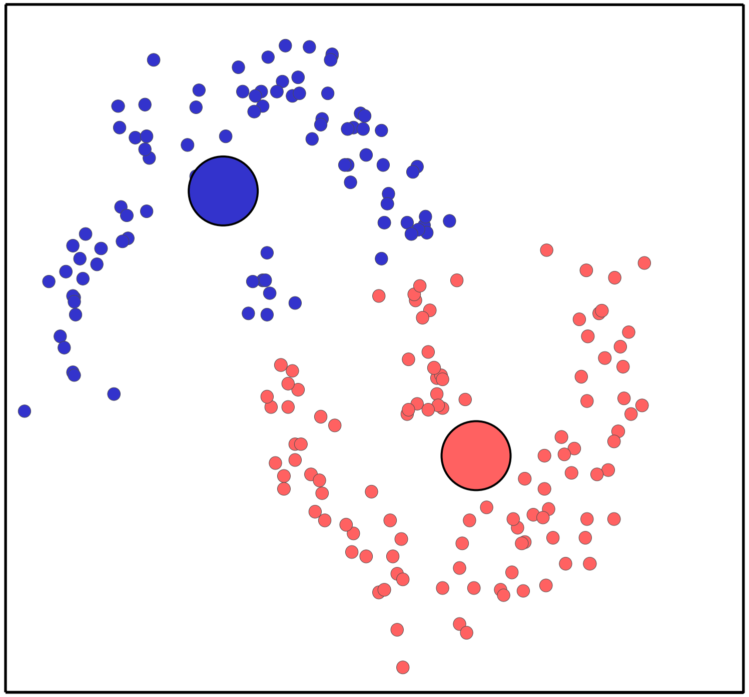
Can relax assumptions by relaxing constraints on model

- Including explicit covariance matrix
- Relaxing assumption that all gaussians are equally likely

Implications: Non-globular Clusters



Actual Clustering



K-means (K = 2)

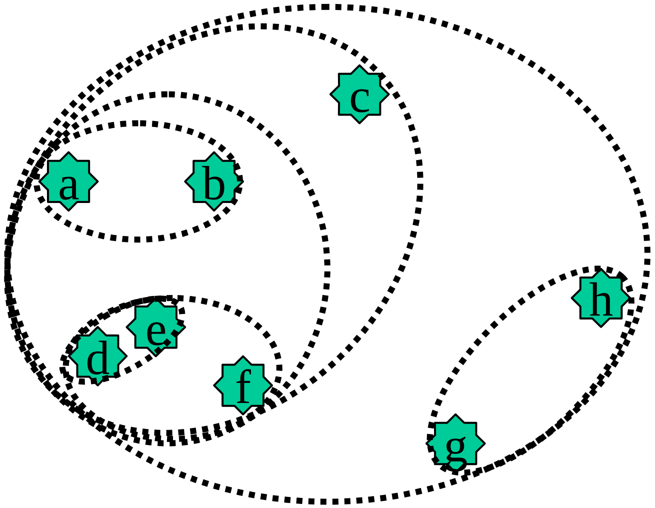
But How Many clusters?

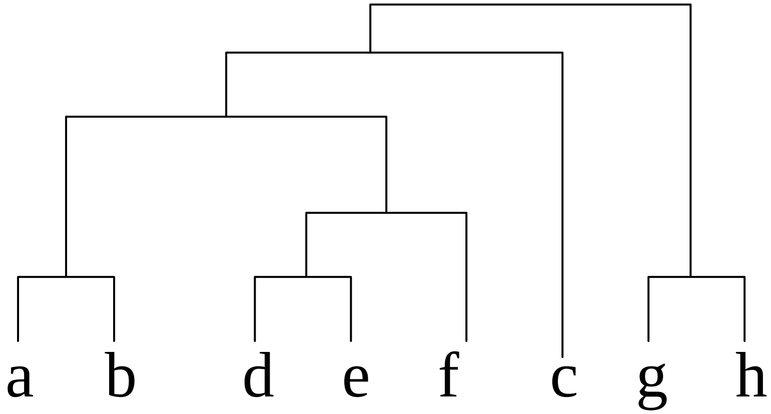
- How do we select K?
  - We can always make clusters “more compact” by increasing K
  - e.g. What happens is if K=number of data points?
  - What is a meaningful improvement?
- Hierarchical clustering side-steps this issue

Hierarchical clustering

Most widely used algorithm for expression data

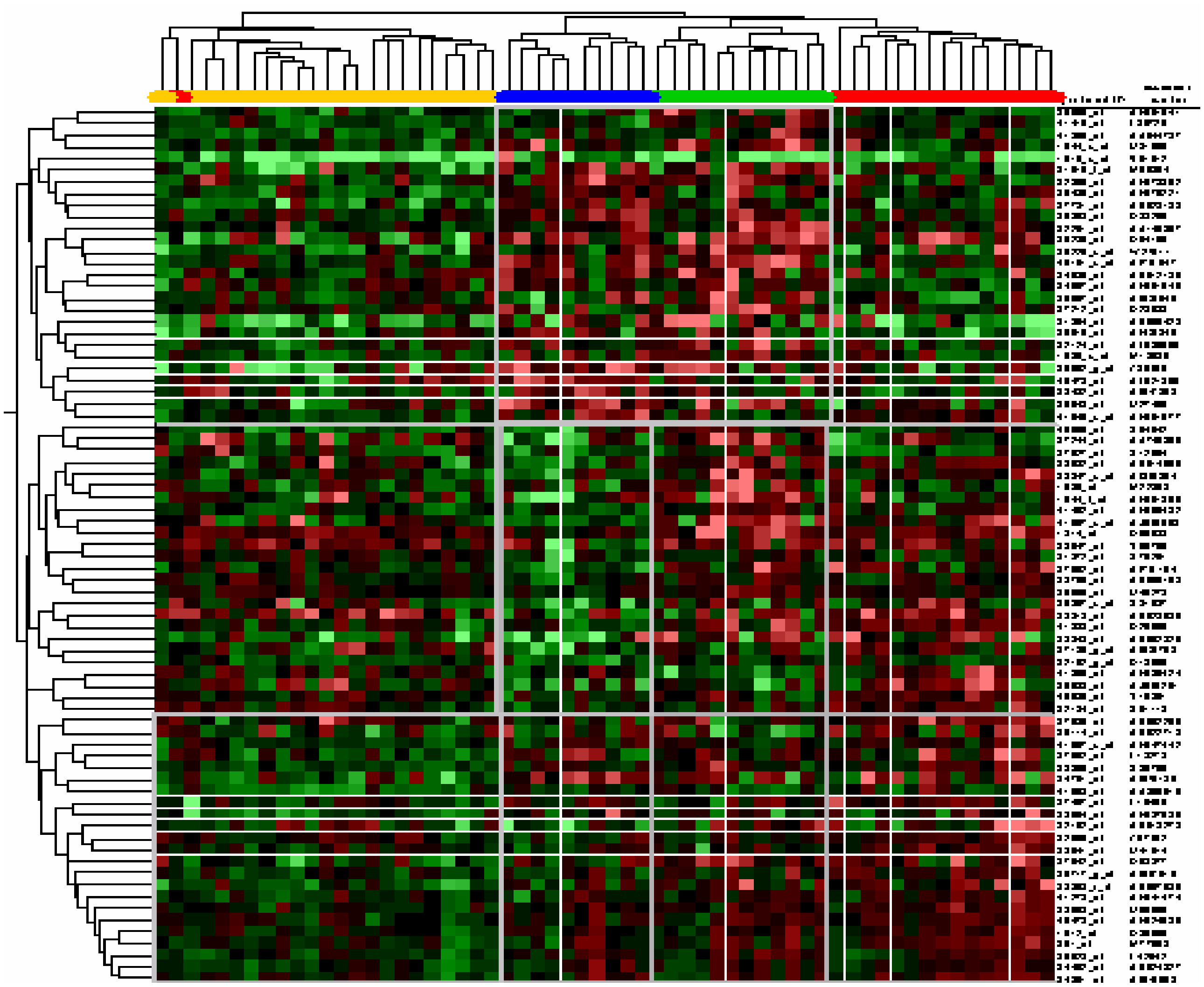
- Start with each point in a separate cluster
- At each step:
  - Choose the pair of **closest clusters**
  - Merge





Phylogeny (UMPGA)

Visualization of results



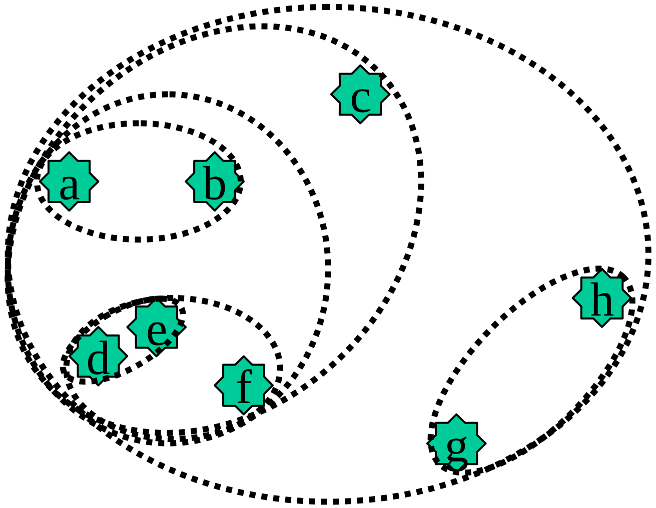
Hierarchical clustering

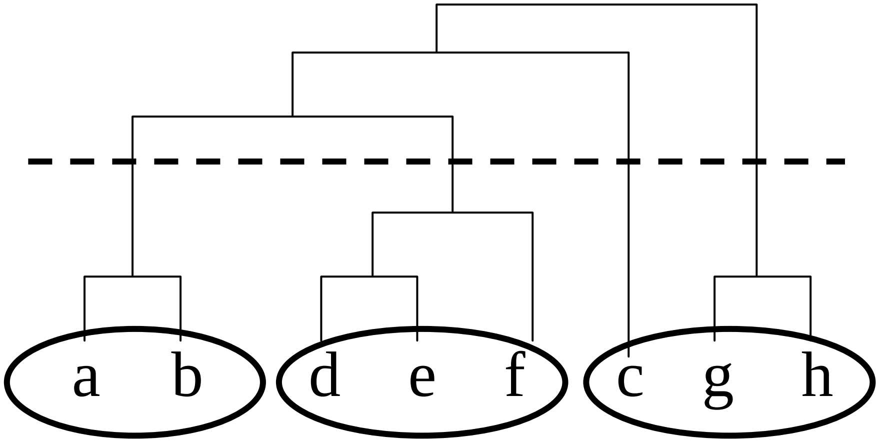
Avoid needing to select number of clusters

Produces clusters at all levels

We can always select a “cut level” to create disjoint clusters

But how do we define distances between clusters?

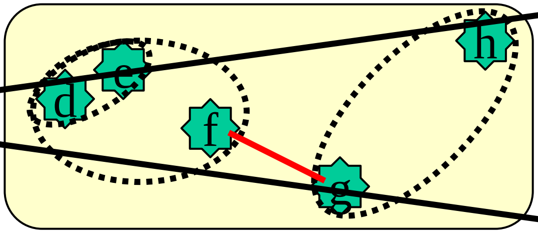




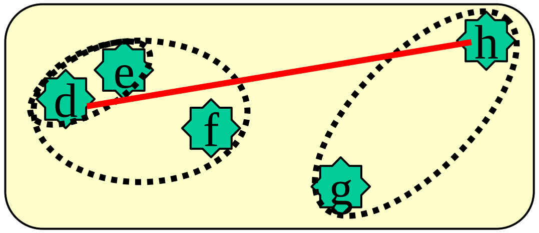
slide credits: M. Kellis

Distance between clusters

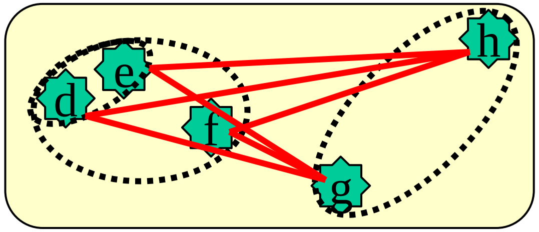
- ~~$CD(X,Y)=\min_{x \in X, y \in Y} D(x,y)$~~
  ~~*Single-link method*~~



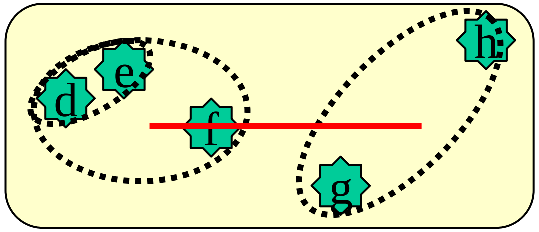
- $CD(X,Y)=\max_{x \in X, y \in Y} D(x,y)$
  Complete-link method



- $CD(X,Y)=\text{avg}_{x \in X, y \in Y} D(x,y)$
  Average-link method



- $CD(X,Y)=D(\ \text{avg}(X)\ ,\ \text{avg}(Y)\ )$
  Centroid method



(Dis)Similarity Measures

Image removed due to copyright restrictions.

Table 1, Gene expression similarity measures. D'haeseleer, Patrik. "How Does Gene Expression Clustering Work?" *Nature Biotechnology* 23 (2005): 1499-1501.

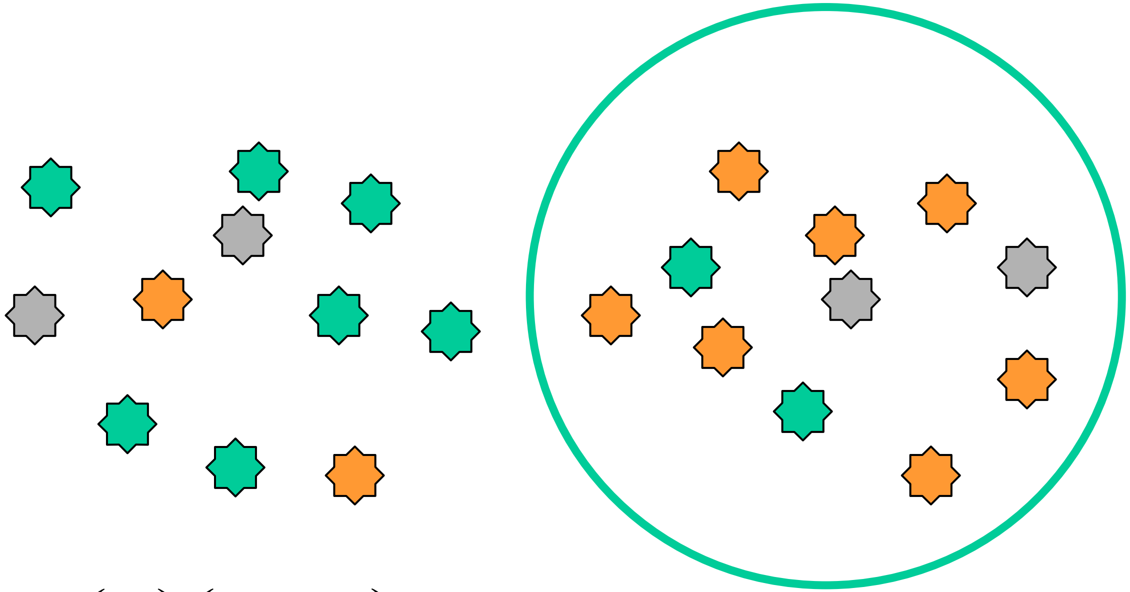
D’haeseleer (2005) Nat Biotech

Evaluating Cluster Performance

In general, it depends on your goals in clustering

- Robustness
  - Select random samples from data set and cluster
  - Repeat
  - Robust clusters show up in all clusters
- Category Enrichment
  - Look for categories of genes “over-represented” in particular clusters
  - Also used in Motif Discovery

Evaluating clusters – Hypergeometric Distribution



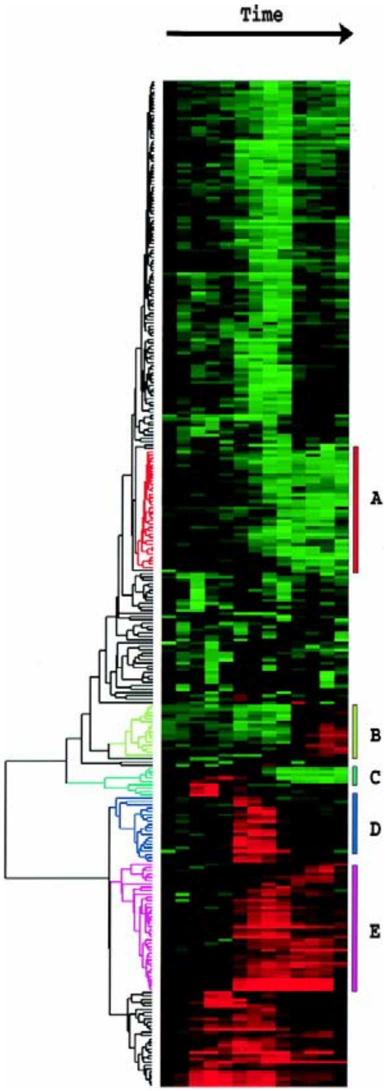
$$P(pos \geq r) = \sum_{m \geq r} \frac{\binom{p}{m}\binom{N-p}{k-m}}{\binom{N}{k}}$$

P-value of uniformity in computed cluster

Prob that a randomly chosen set of k experiments would result in m positive and k-m negative

- N experiments, p labeled +, (N-p) –
- Cluster: k elements, m labeled +
- P-value of *single* cluster containing k elements of which at least r are +

Similar Genes Can Cluster



Clustered 8600 human genes using expression time course in fibroblasts

(A) Cholesterol biosynthesis

(B) Cell cycle

(C) Immediate early response

(D) Signalling and angiogenesis

(E) Wound healing

Eisen, Michael et al. "Cluster Analysis and Display of Genome-wide Expression Patterns." *PNAS* 95, no. 25 (1998): 14863-14868.

(Eisen (1998) PNAS)

Clusters and Motif Discovery

Expression from 15 time points during yeast cell cycle

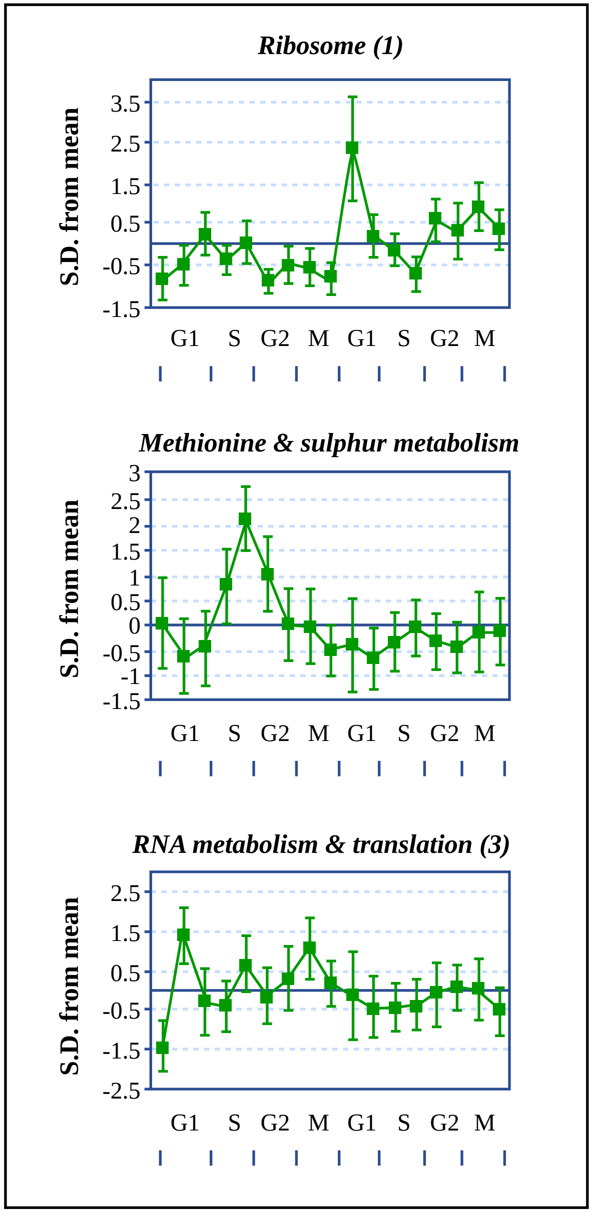


Figure by MIT OpenCourseWare.

Tavazoie & Church (1999)

Next Lecture

The other side of the coin… Classification