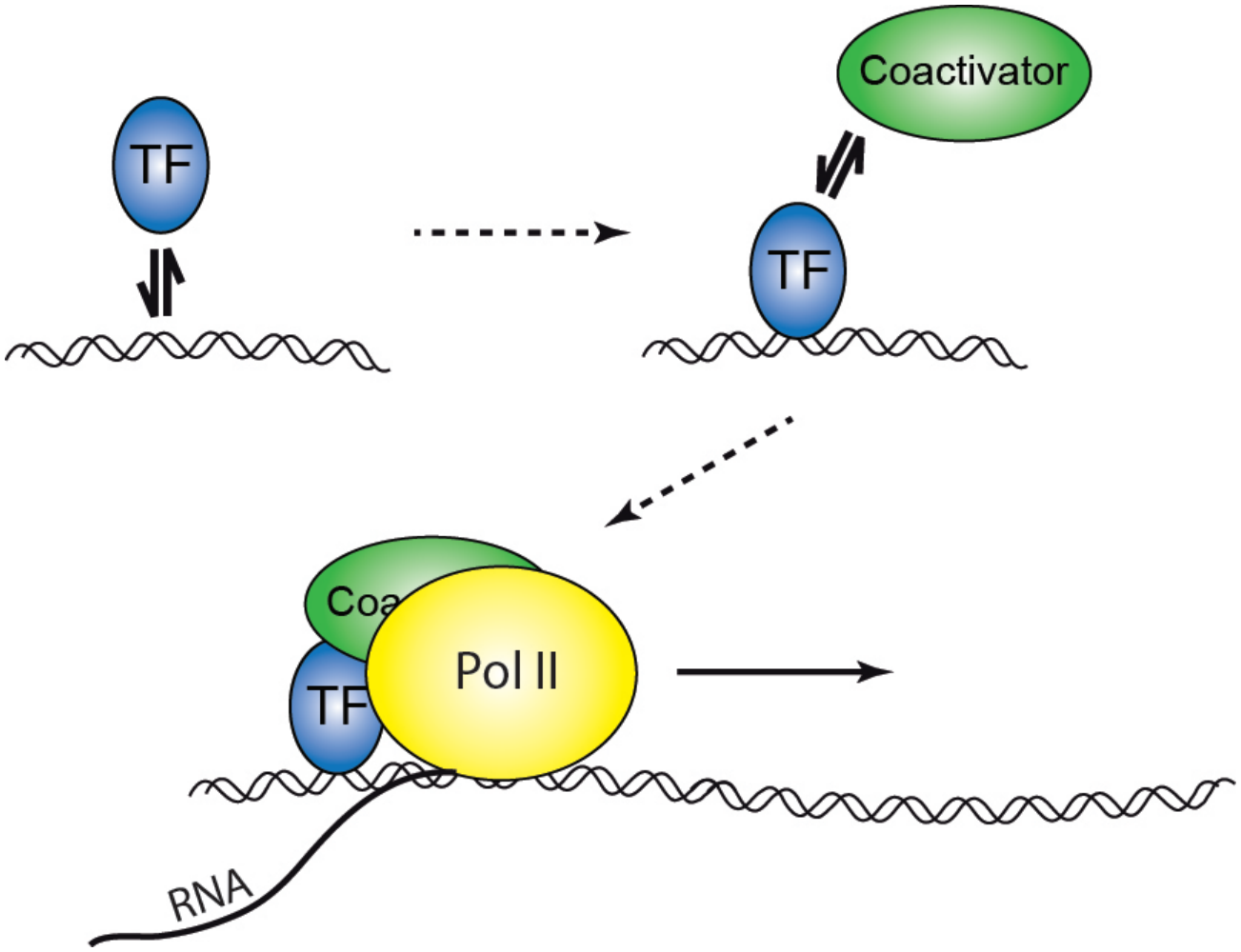
Bayesian Statistics for Genetics

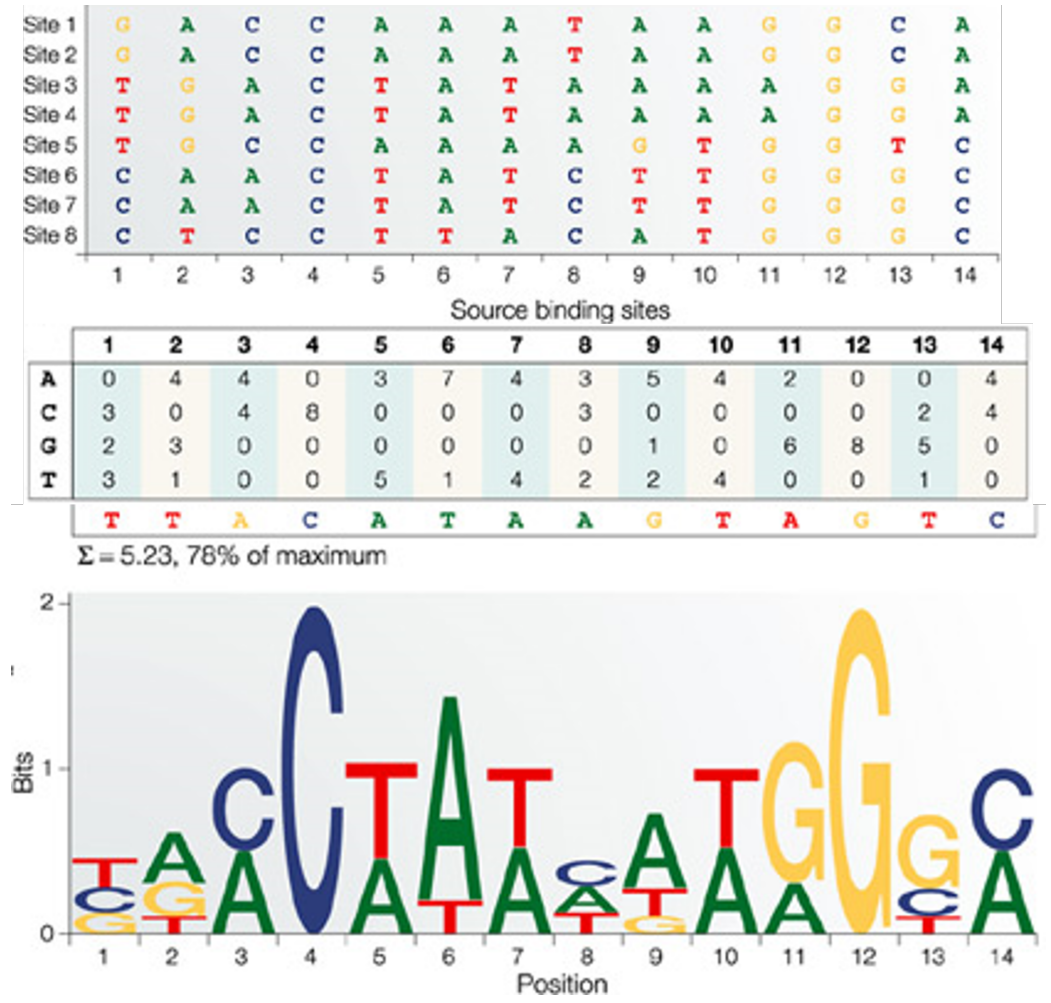
Lecture 6: Modeling DNA sequence motifs

June, 2024

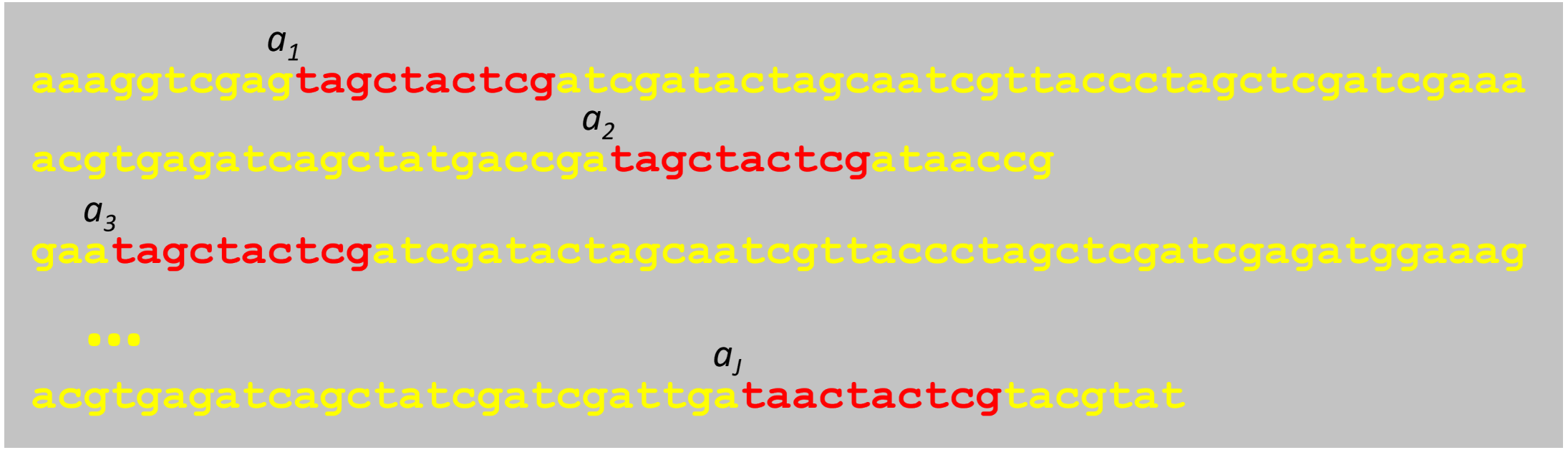
Transcription regulation



Modeling sequence motifs



Modeling Motifs



DNA sequence data $\boldsymbol{R} = (\boldsymbol{R}_1, \boldsymbol{R}_2, ..., \boldsymbol{R}_J)$

Motif alignment model

- Alignment variable $\boldsymbol{A} = \{a_1, a_2, ..., a_J\}$
- Every background position (non-motif part) follows a common multinomial distribution with parameter $\Theta_0 = \{\theta_{0A}, \theta_{0C}, \theta_{0G}, \theta_{0T}\}$
- Every base i inside the motif follows a specific multinomial distribution with parameter $\theta_i = \{\theta_{iA}, \theta_{iC}, \theta_{iG}, \theta_{iT}\}$

Likelihood

The likelihood of observing $\boldsymbol{R}$ given all the parameters can be written as

$$P(\boldsymbol{R}|\theta_0, \Theta, A) = \prod_j \prod_{k=1}^{4} \theta_{0k}^{h_k(R_j)} \prod_{i=1}^{w} \prod_{k=1}^{4} (\frac{\theta_{ik}}{\theta_{0k}})^{h_k(a_j-1+i)}$$

This is a *mixture model**, i.e., Sequence data generated from two distinct distributions.

What are they?

* see Lawrence *et al* Science 1993, also Liu *et al* JASA 1995

Statistical model & algorithm to fit it

We aim to learn the joint posterior of multinomial parameters Θ and alignment $\mathbf{A}$, i.e. $\mathbb{P}[\,\Theta, \boldsymbol{A}|R, \theta_0\,]$. Using Gibbs sampling, we can do this via

- $\mathbb{P}[\,\boldsymbol{A}|\Theta, R, \theta_0\,]$
- $\mathbb{P}[\,\Theta|\boldsymbol{A}, R, \theta_0\,]$, a.k.a. the *full conditionals*

As an algorithm:

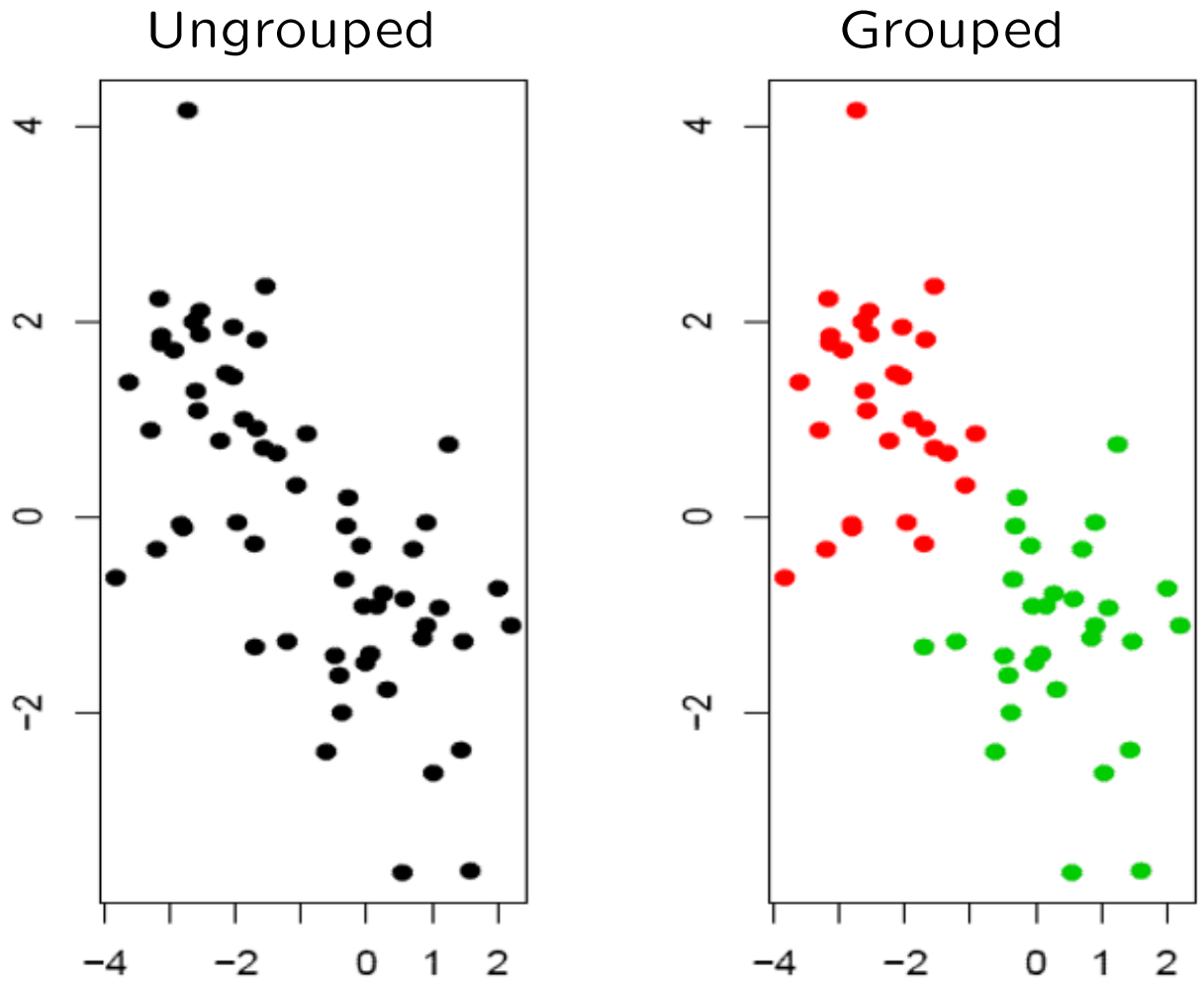
1. Initialize $\Theta, \boldsymbol{A}$ by choosing random starting positions

2. Iterate the following steps many times;

    - Randomly or systematically choose a sequence to exclude
    - Carry out the predictive-updating step to update the starting position
    - Stop when there are no more observable changes in likelihood

Clustering

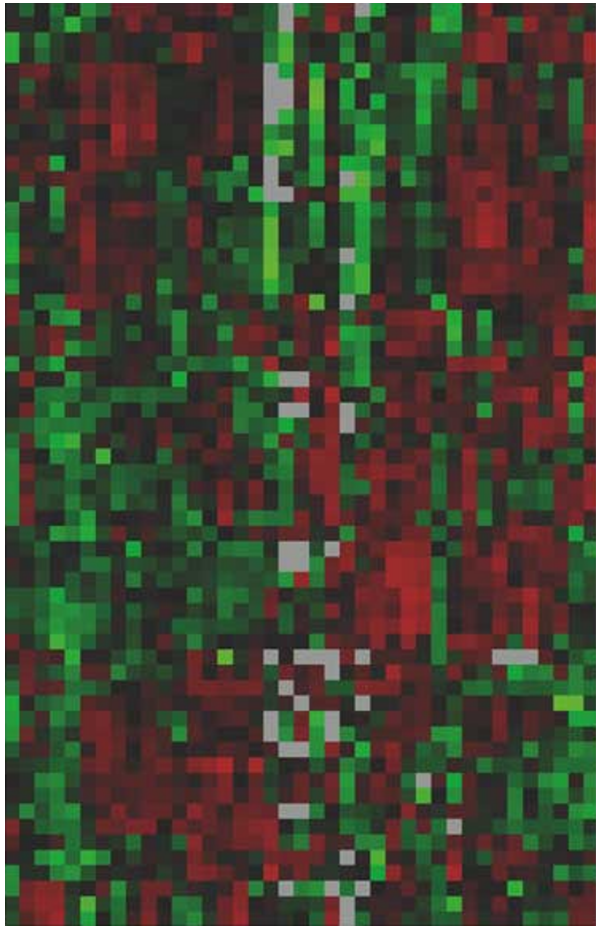
The goal is to group objects according to their similarity, a.k.a. *unsupervised learning*;

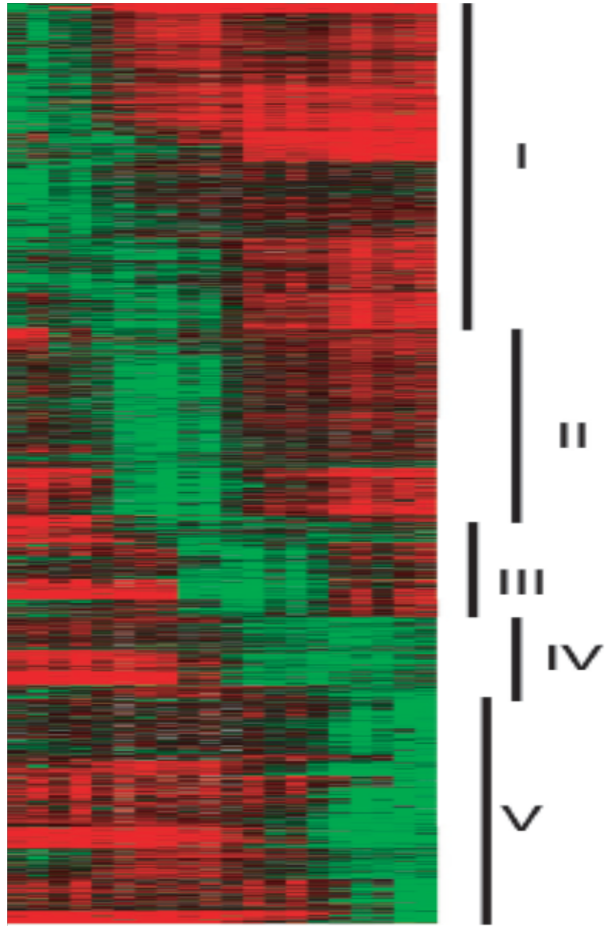


Clustering

This approach is very popular in genetics and genomics:

- Population structure
- Disease subtypes
- Co-regulated genes
- Separate cell types
- Cladistics: classify species





Why cluster?

By clustering *genes* we can:

- Identify groups of possibly co-regulated genes (e.g. in conjunction with sequence data)
- Identify typical temporal or spatial gene expression patterns (e.g. cell cycle data)
- Arrange a set of genes in an order that is not *totally* meaningless

By clustering *samples* we can

- Do quality control: detect experimental artifacts/bad hybridizations, label switches, etc
- Check whether samples are grouped according to known categories
- Identify new classes of biological samples (e.g. tumor subtypes)

Existing clustering methods

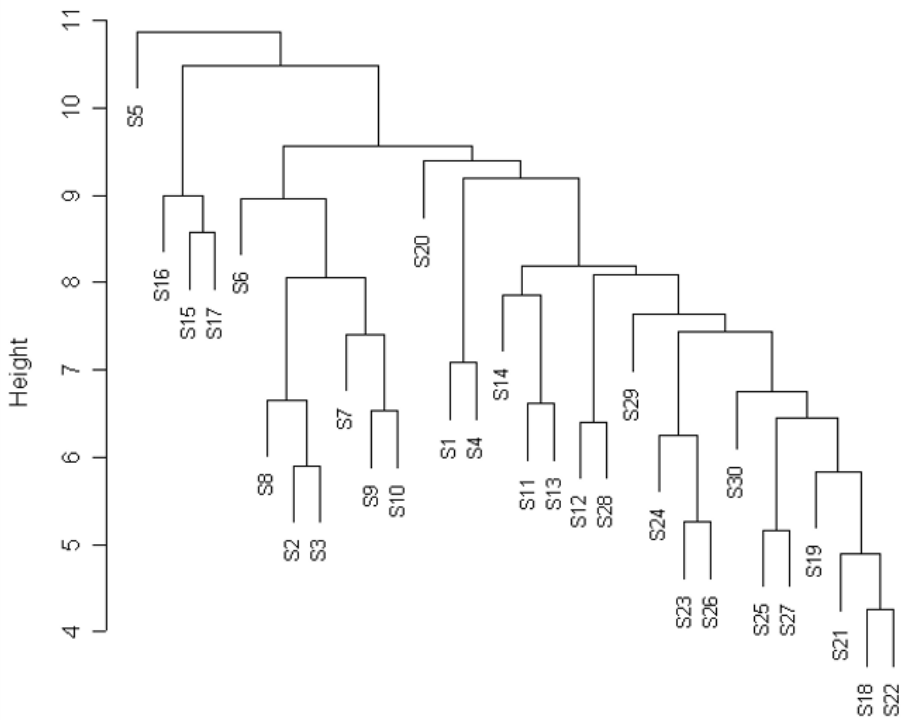
Clustering methods generally rely on two components:

- **Distance measure:** Quantification of (dis-)similarity of objects:
  - Euclidean distance
  - Manhattan distance
  - Correlation distance

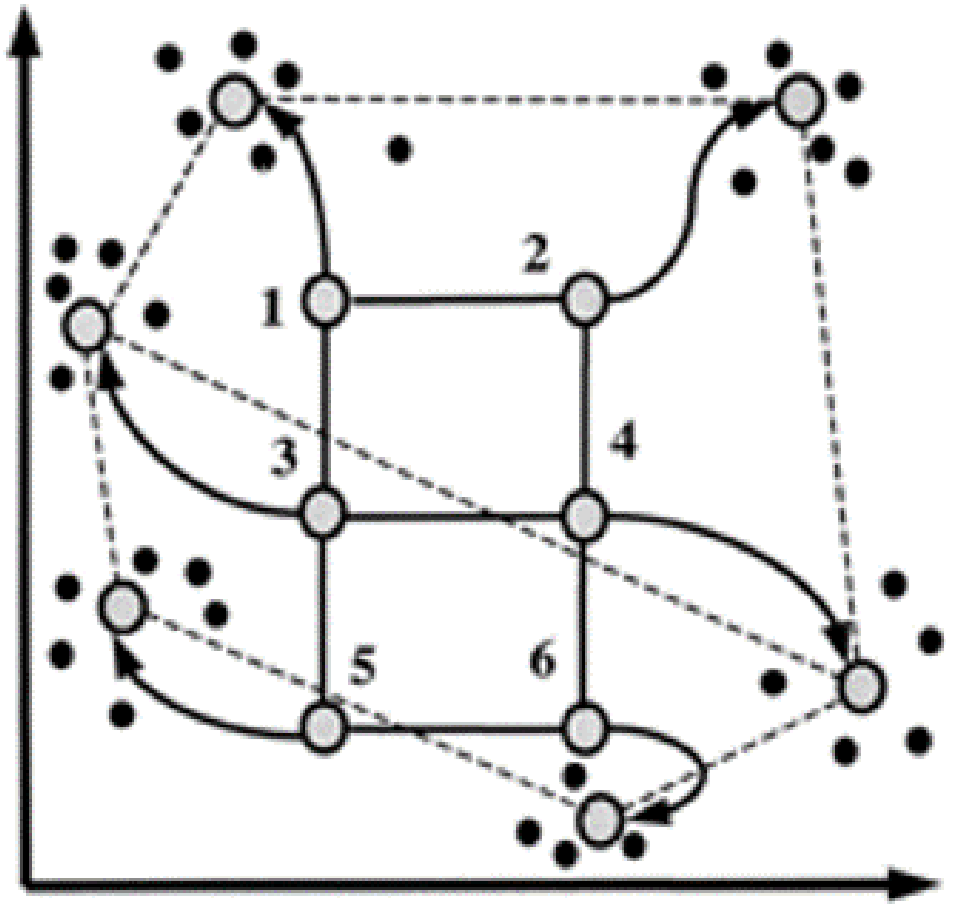
- **Cluster algorithm:** A procedure to group objects, aiming for small within-cluster distances, large between-cluster distances:
  - Hierarchical clustering
  - K-means
  - Self Organizing Map

Existing clustering methods

Hierarchical clustering



Self Organizing Map (SOM)

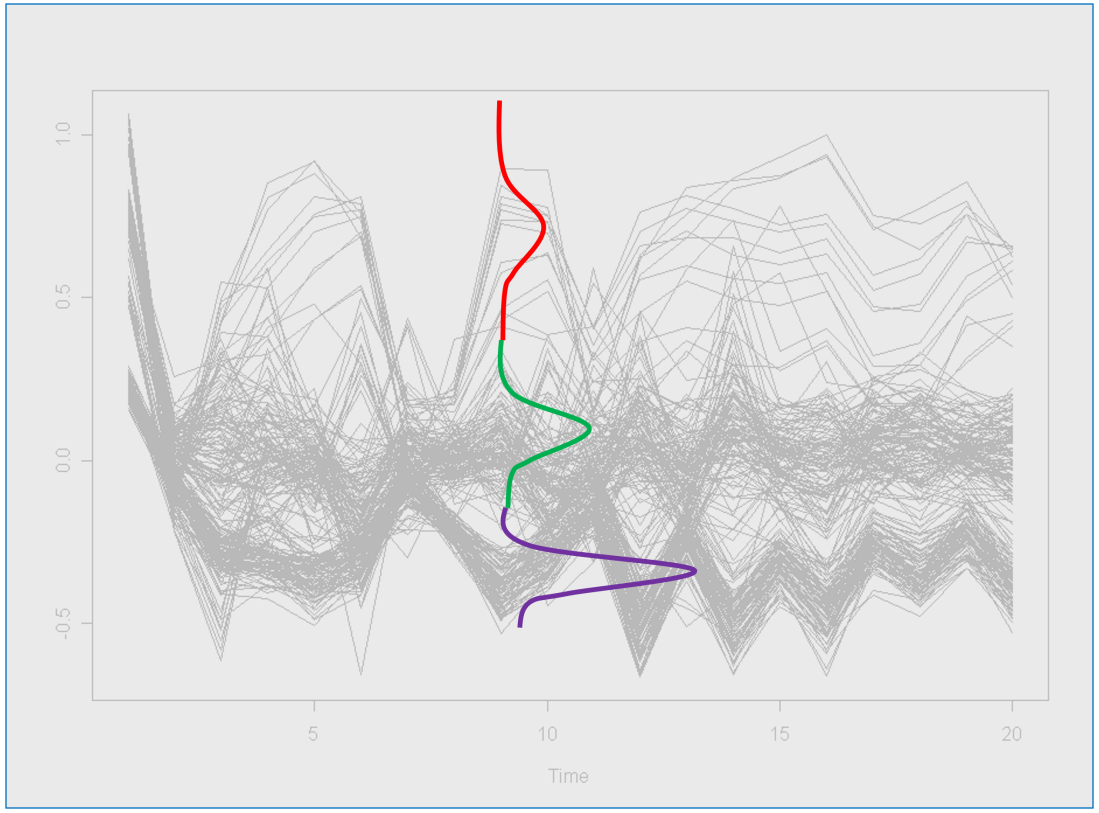


Model-based clustering

An alternative is to specify a *finite mixture model**, where group membership is an unknown parameter. The model for data X states

$$\mathbb{P}[\, X|\Theta, \Lambda\,] = \sum_{i=1}^{n} \sum_{k=1}^{K} \lambda_k p(x_i|\theta_k),$$

where number of clusters K is determined using the Bayesian Information Criterion (BIC), and clustering is then performed using the EM algorithm.

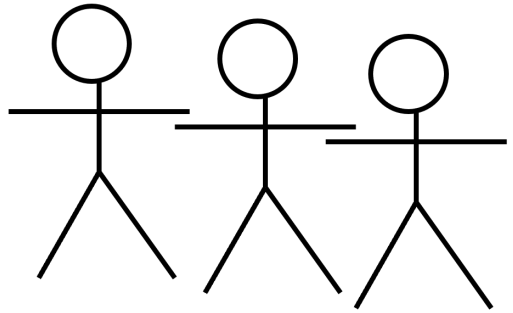


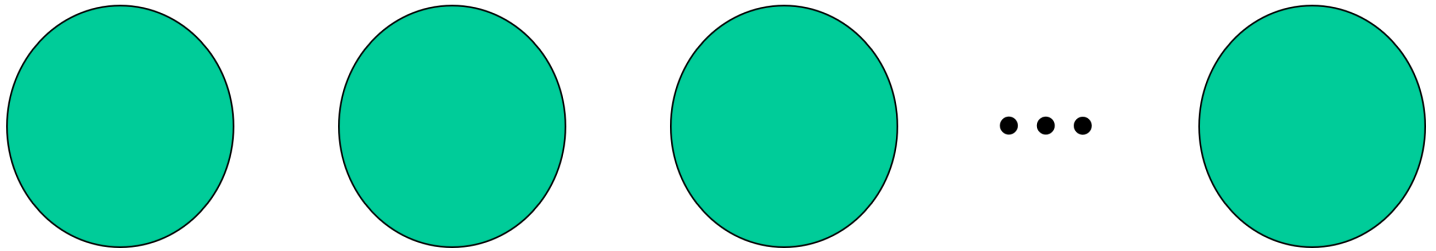
* see Banfield & Raftery, Biometrics 1993, Yeung *et al* Bioinformatics 2001

Dirichlet process mixture model

To limit the impact of choosing K based on the data, we can implement an *infinite mixture model* – essentially averaging over many potential K.

The best-known approach for this uses a *Chinese restaurant process:**



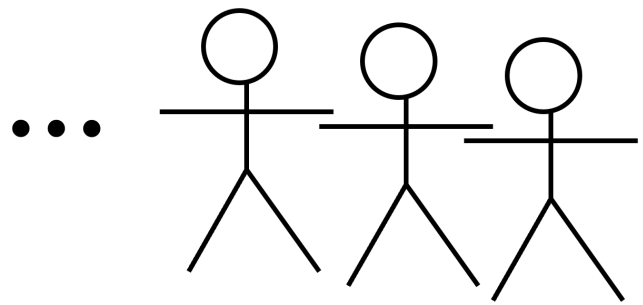


* Developed by Jim Dubins & Lester Pitman, details in Aldous (1985). Pitman (1996)

Dirichlet process mixture model

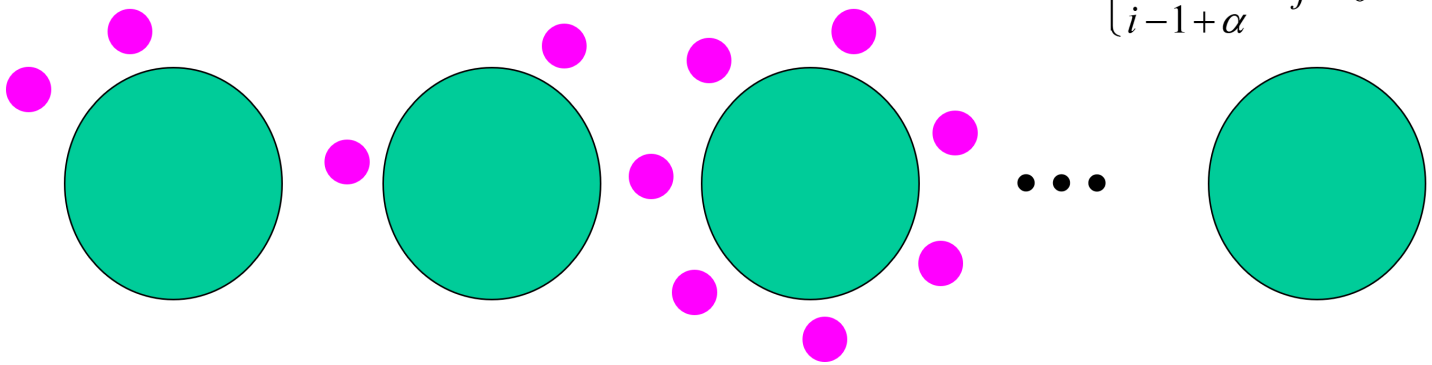
To limit the impact of choosing K based on the data, we can implement an *infinite mixture model* – essentially averaging over many potential K.

The best-known approach for this uses a *Chinese restaurant process:**



The probability of joining these tables:

$$P(E(i)=j \mid E(1),\ldots,E(i-1)) = \begin{cases} \dfrac{n_k}{i-1+\alpha} & j=k \\ \dfrac{\alpha}{i-1+\alpha} & j=0 \end{cases}$$



* Developed by Jim Dubins & Lester Pitman, details in Aldous (1985), Pitman 1996)

About Dirichlet Process

- Let $(\Theta, \boldsymbol{B})$ be a measurable space, G_0 be a probability measure on the space, and [FIXME symbol] be a positive real number
- A Dirichlet process is any distribution of a random probability measure G over $(\Theta, \boldsymbol{B})$ such that, for all finite partitions $(A_1, ..., A_r)$ of Θ,

$$(G(A_1), ..., G(A_r)) \sim Dirichlet(\alpha G_0(A_1), ..., \alpha G_0(A_r))$$

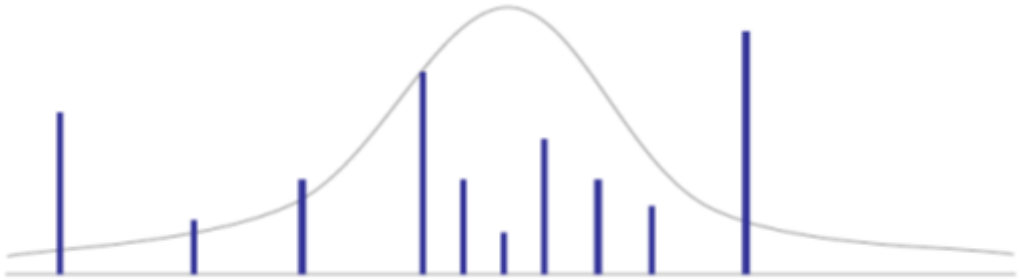
- Draws G from DP are generally not distinct
- The number of distinct values grows with $O(\log n)$

General scheme

$$
\begin{array}{lcll}
G & \sim & DP(\alpha, G_0) & \\
\theta_i & \sim & G, & i = 1, ..., n \\
y_i & \sim & p(\theta_i), & i = 1, ..., n
\end{array}
$$

- Sample $\theta_1 \sim G_0$,
- Sample $\theta_2 \sim \frac{1}{1+\alpha} I_{\theta_1} + \frac{\alpha}{1+\alpha} G_0$,

  ⋮
- Sample $\theta_n \sim \frac{n_1}{N-1+\alpha} I_{\theta_1} + \frac{n_2}{n+1+\alpha} I_{\theta_2} + \cdots \frac{n_K}{N-1+\alpha} I_{\theta_K} + \frac{\alpha}{1+\alpha} G_0$,



Related topics

- Polya urn process
- Stick breaking
- Infinite mixture model
- Bayesian nonparametric model
- Pioneers: Thomas Ferguson, David Blackwell, ...

Real example: clustering DNA motifs

- Supposed we have collected n motifs of equal width.
- We want to explore how many motif patterns we can find from them.
- Model: product multinomial distributions
  - motifs within a cluster follow the same distribution
  - each cluster is represented by a distinct distribution
- No need to specify a cluster number
- Inference can be conducted using MCMC

* References for bioinformatics applications: clustering motifs (2003); clustering gene expression data (2006)

Prior and Posterior

- $\mathbf{X} = \{x_{ij}\}$ denotes the DNA motif data (motif i, position j),
- $\mathbf{E} = \{E(i)\}$ is indicator of cluster membership for motif i. This is parameter of interest.
- Prior for $\mathbf{E}$:

$$P(E(i) = j | E(1), ..., E(i-1), E(i+1), ..., E(n)) = \begin{cases} \frac{n_k}{i-1+\alpha}, & j = k \\ \frac{\alpha}{i-1+\alpha}, & j = 0 \end{cases}$$

- Posterior for $\mathbf{E}$:

$$P(E(i) = j | E(1), ..., E(i-1), E(i+1), ..., E(n)) \propto \begin{cases} n_k P(X | E(i) = j), & j = k \\ \alpha P(X | E(i) = 0), & j = 0 \end{cases}$$

Algorithm

- Initialization: randomly assign genes into an arbitrary number of K_0 clusters $1 \leq K_0 \leq N$.
- For each gene i, perform the following reassignment:
  - Remove gene i from its current cluster, given the current assignment of all the other genes, calculate the probability of this gene joining each of the existing cluster as well as being alone.
  - Assign gene i to the $K + 1$ possible clusters according to probabilities. Update indicator variable $E(i)$ based on the assignment.
  - Repeat the above two steps for every gene, and repeat for a large number of rounds until convergence.

Summary

- Model-based clustering is based on probability distribution assumption
- Ideal for handling noisy data
- Computationally efficient: no need to calculate pairwise distances
- Providing statistical inference is straightforward
- Dirchlet Process-based clustering enables to determine the number of clusters automatically