資料分群技術
及其在生物資訊資料分析之應用

Bi-Ru Dai （戴碧如 博士）
Assistant Professor
Department of Computer Science and Information Engineering
National Taiwan University of Science and Technology
Outlines

- What is Cluster Analysis?
- Types of Data in Cluster Analysis
- Major Clustering Methods
- Biclustering for Gene Expression Data
Outlines

- What is Cluster Analysis?
- Types of Data in Cluster Analysis
- Major Clustering Methods
- Biclustering for Gene Expression Data

Data Mining: Concepts and Techniques
Jiawei Han and Micheline Kamber
What Is Cluster Analysis

- Cluster: a collection of data objects
  - Similar to one another within the same cluster
  - Dissimilar to the objects in other clusters
- Cluster analysis
  - Finding similarities between data according to the characteristics found in the data and grouping similar data objects into clusters
- Unsupervised learning: no predefined classes
- Typical applications
  - As a stand-alone tool to get insight into data distribution
  - As a preprocessing step for other algorithms

Source: “Data Mining: Concepts and Techniques”, Jiawei Han and Micheline Kamber
Quality: What Is Good Clustering?

- A good clustering method will produce high quality clusters with
  - high intra-class similarity
  - low inter-class similarity
- The quality of a clustering result depends on both the similarity measure used by the method and its implementation
- The quality of a clustering method is also measured by its ability to discover some or all of the hidden patterns

Source: “Data Mining: Concepts and Techniques”, Jiawei Han and Micheline Kamber
Requirements of Clustering in Data Mining

- Scalability
- Ability to deal with different types of attributes
- Ability to handle dynamic data
- Discovery of clusters with arbitrary shape
- Minimal requirements for domain knowledge to determine input parameters
- Able to deal with noise and outliers
- Insensitive to order of input records
- High dimensionality
- Incorporation of user-specified constraints
- Interpretability and usability

Source: “Data Mining: Concepts and Techniques”, Jiawei Han and Micheline Kamber
Outlines

- What is Cluster Analysis?
- Types of Data in Cluster Analysis
- Major Clustering Methods
- Biclustering for Gene Expression Data

Data Mining: Concepts and Techniques
Jiawei Han and Micheline Kamber
Data Structures

- Data matrix
  - (two modes)

\[
\begin{bmatrix}
  x_{11} & \ldots & x_{1f} & \ldots & x_{1p} \\
  \vdots & \ddots & \vdots & \ddots & \vdots \\
  x_{i1} & \ldots & x_{if} & \ldots & x_{ip} \\
  \vdots & \ddots & \vdots & \ddots & \vdots \\
  x_{n1} & \ldots & x_{nf} & \ldots & x_{np}
\end{bmatrix}
\]

- Dissimilarity matrix
  - (one mode)

\[
\begin{bmatrix}
  0 & & & \\
  d(2,1) & 0 & & \\
  d(3,1) & d(3,2) & 0 & \\
  \vdots & \vdots & \vdots & \\
  d(n,1) & d(n,2) & \ldots & 0
\end{bmatrix}
\]

Source: “Data Mining: Concepts and Techniques”, Jiawei Han and Micheline Kamber
Type of Data in Clustering Analysis

- Interval-scaled variables
- Binary variables
- Nominal, ordinal, and ratio variables
- String variables
- Vector Objects

Modified from: “Data Mining: Concepts and Techniques”, Jiawei Han and Micheline Kamber
Interval-valued variables

- Standardize data
  - Calculate the mean absolute deviation:
    \[
    s_f = \frac{1}{n} (|x_{1f} - m_f| + |x_{2f} - m_f| + \ldots + |x_{nf} - m_f|)
    \]
    where
    \[
    m_f = \frac{1}{n} (x_{1f} + x_{2f} + \ldots + x_{nf})
    \]
  - Calculate the standardized measurement (z-score)
    \[
    z_{if} = \frac{x_{if} - m_f}{s_f}
    \]
- Using mean absolute deviation is more robust than using standard deviation

Source: “Data Mining: Concepts and Techniques”, Jiawei Han and Micheline Kamber
Similarity and Dissimilarity Between Objects

D\textbf{istances} are normally used to measure the \textbf{similarity} or \textbf{dissimilarity} between two data objects.

Some popular ones include: \textit{Minkowski distance}.

\[
d(i, j)=\sqrt[q]{{\left(|x_{i1}-x_{j1}|^q +|x_{i2}-x_{j2}|^q +\cdots +|x_{ip}-x_{jp}|^q\right)}^{1/q}}
\]

where \( i = (x_{i1}, x_{i2}, \ldots, x_{ip}) \) and \( j = (x_{j1}, x_{j2}, \ldots, x_{jp}) \) are two \( p \)-dimensional data objects, and \( q \) is a positive integer.

If \( q = 1 \), \( d \) is Manhattan distance

\[
d(i, j)=|x_{i1}-x_{j1}|+|x_{i2}-x_{j2}|+\cdots+|x_{ip}-x_{jp}|
\]

Source: “Data Mining: Concepts and Techniques”, Jiawei Han and Micheline Kamber
Similarity and Dissimilarity Between Objects (Cont.)

- If $q = 2$, $d$ is Euclidean distance:
  \[
  d(i,j) = \sqrt{|x_{i_1} - x_{j_1}|^2 + |x_{i_2} - x_{j_2}|^2 + \ldots + |x_{i_p} - x_{j_p}|^2}
  \]

- Properties
  - $d(i,j) \geq 0$
  - $d(i,i) = 0$
  - $d(i,j) = d(j,i)$
  - $d(i,j) \leq d(i,k) + d(k,j)$

- Also, one can use weighted distance, parametric Pearson product moment correlation, or other disimilarity measures.

Source: “Data Mining: Concepts and Techniques”, Jiawei Han and Micheline Kamber
Binary Variables

- A contingency table for binary data

<table>
<thead>
<tr>
<th></th>
<th>Object $i$</th>
<th>Object $j$</th>
<th>sum</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>$a$</td>
<td>$b$</td>
<td>$a+b$</td>
</tr>
<tr>
<td>0</td>
<td>$c$</td>
<td>$d$</td>
<td>$c+d$</td>
</tr>
<tr>
<td>sum</td>
<td>$a+c$</td>
<td>$b+d$</td>
<td>$p$</td>
</tr>
</tbody>
</table>

- Distance measure for symmetric binary variables:

$$d (i, j) = \frac{b + c}{a + b + c + d}$$

- Distance measure for asymmetric binary variables:

$$d (i, j) = \frac{b + c}{a + b + c}$$

- Jaccard coefficient (similarity measure for asymmetric binary variables):

$$sim_{Jaccard} (i, j) = \frac{a}{a + b + c}$$

Source: “Data Mining: Concepts and Techniques”, Jiawei Han and Micheline Kamber
Dissimilarity between Binary Variables

Example

<table>
<thead>
<tr>
<th>Name</th>
<th>Gender</th>
<th>Fever</th>
<th>Cough</th>
<th>Test-1</th>
<th>Test-2</th>
<th>Test-3</th>
<th>Test-4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Jack</td>
<td>M</td>
<td>Y</td>
<td>N</td>
<td>P</td>
<td>N</td>
<td>N</td>
<td>N</td>
</tr>
<tr>
<td>Mary</td>
<td>F</td>
<td>Y</td>
<td>N</td>
<td>P</td>
<td>N</td>
<td>P</td>
<td>N</td>
</tr>
<tr>
<td>Jim</td>
<td>M</td>
<td>Y</td>
<td>P</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
</tr>
</tbody>
</table>

- gender is a symmetric attribute
- the remaining attributes are asymmetric binary
- let the values Y and P be set to 1, and the value N be set to 0

\[ d \left( \text{jack} , \text{mary} \right) = \frac{0 + 1}{2 + 0 + 1} = 0.33 \]

\[ d \left( \text{jack} , \text{jim} \right) = \frac{1 + 1}{1 + 1 + 1} = 0.67 \]

\[ d \left( \text{jim} , \text{mary} \right) = \frac{1 + 2}{1 + 1 + 2} = 0.75 \]

Source: “Data Mining: Concepts and Techniques”, Jiawei Han and Micheline Kamber
Nominal Variables

- A generalization of the binary variable in that it can take more than 2 states, e.g., red, yellow, blue, green
- Method 1: Simple matching
  - \( m \): # of matches, \( p \): total # of variables
  
  \[
  d(i,j) = \frac{p - m}{p}
  \]
- Method 2: use a large number of binary variables
  - creating a new binary variable for each of the \( M \) nominal states

Source: “Data Mining: Concepts and Techniques”, Jiawei Han and Micheline Kamber
Ordinal Variables

- An ordinal variable can be discrete or continuous
- Order is important, e.g., rank
- Can be treated like interval-scaled
  - replace $x_{if}$ by their rank $r_{if} \in \{1, \ldots, M_f\}$
  - map the range of each variable onto [0, 1] by replacing $i$-th object in the $f$-th variable by
    \[
    z_{if} = \frac{r_{if} - 1}{M_f - 1}
    \]
  - compute the dissimilarity using methods for interval-scaled variables

Source: "Data Mining: Concepts and Techniques", Jiawei Han and Micheline Kamber
Ratio-Scaled Variables

- **Ratio-scaled variable**: a positive measurement on a nonlinear scale, approximately at exponential scale, such as $Ae^{Bt}$ or $Ae^{-Bt}$

- **Methods**:
  - treat them like interval-scaled variables—*not a good choice!* (why?—the scale can be distorted)
  - apply logarithmic transformation
    \[ y_{if} = \log(x_{if}) \]
  - treat them as continuous ordinal data treat their rank as interval-scaled

Source: “Data Mining: Concepts and Techniques”, Jiawei Han and Micheline Kamber
String Variables

- The **edit distance** between two strings of characters is the **minimum number of operations** (point mutations) required to transform one of them into the other.

- **Insertion**
  - mile → smile

- **Deletion**
  - good → god

- **Substitution**
  - there → these

Example of DNA sequences:

…AATTAGACAGGCATGATGG…

…AATTAGGCAGGCATGGTGA…
Vector Objects

- Vector objects: keywords in documents, gene features in micro-arrays, etc.
- Broad applications: information retrieval, biologic taxonomy, etc.
- Cosine measure

\[
s(\vec{X}, \vec{Y}) = \frac{\vec{X}^t \cdot \vec{Y}}{|\vec{X}| |\vec{Y}|},
\]

\(\vec{X}^t\) is a transposition of vector \(\vec{X}\), \(|\vec{X}|\) is the Euclidean normal of vector \(\vec{X}\),
Outlines

- What is Cluster Analysis?
- Types of Data in Cluster Analysis
- Major Clustering Methods
- Biclustering for Gene Expression Data

Data Mining: Concepts and Techniques
Jiawei Han and Micheline Kamber
Major Clustering Approaches

- **Partitioning approach:**
  - Construct various partitions and then evaluate them by some criterion, e.g., minimizing the sum of square errors
  - Typical methods: k-means, k-medoids, CLARANS

- **Hierarchical approach:**
  - Create a hierarchical decomposition of the set of data (or objects) using some criterion
  - Typical methods: Diana, Agnes, BIRCH, ROCK, CAMELEON

- **Density-based approach:**
  - Based on connectivity and density functions
  - Typical methods: DBSCAN, OPTICS, DenClue

Source: “Data Mining: Concepts and Techniques”, Jiawei Han and Micheline Kamber
Partitioning Algorithms: Basic Concept

- **Partitioning method:** Construct a partition of a database $D$ of $n$ objects into a set of $k$ clusters, s.t., min sum of squared distance

$$
\sum_{m=1}^{k} \sum_{t_{mi} \in K_m} (C_m - t_{mi})^2
$$

- Given a $k$, find a partition of $k$ clusters that optimizes the chosen partitioning criterion
  - Global optimal: exhaustively enumerate all partitions
  - Heuristic methods: $k$-means and $k$-medoids algorithms
    - $k$-means (MacQueen’67): Each cluster is represented by the center of the cluster
    - $k$-medoids or PAM (Partition around medoids) (Kaufman & Rousseeuw’87): Each cluster is represented by one of the objects in the cluster
The *K-Means* Clustering Method

- Given $k$, the *k-means* algorithm is implemented in four steps:
  - Partition objects into $k$ nonempty subsets
  - Compute seed points as the centroids of the clusters of the current partition (the centroid is the center, i.e., *mean point*, of the cluster)
  - Assign each object to the cluster with the nearest seed point
  - Go back to Step 2, stop when no more new assignment

Source: “Data Mining: Concepts and Techniques”, Jiawei Han and Micheline Kamber
The *K-Means* Clustering Method

- **Example**

  1. Arbitrarily choose *K* objects as initial cluster centers.
  2. Assign each object to the most similar center.
  3. Update the cluster means.
  4. Reassign objects to the most similar center.
  5. Update the cluster means.
  6. Reassign objects to the most similar center.

Source: “Data Mining: Concepts and Techniques”, Jiawei Han and Micheline Kamber
The *K-Medoids* Clustering Method

- Find *representative* objects, called *medoids*, in clusters
- *PAM* (Partitioning Around Medoids, 1987)
  - starts from an initial set of medoids and iteratively replaces one of the medoids by one of the non-medoids if it improves the total distance of the resulting clustering
  - *PAM* works effectively for small data sets, but does not scale well for large data sets
- *CLARA* (Kaufmann & Rousseeuw, 1990)
- *CLARANS* (Ng & Han, 1994): Randomized sampling
- Focusing + spatial data structure (Ester et al., 1995)

Source: “Data Mining: Concepts and Techniques”, Jiawei Han and Micheline Kamber
A Typical K-Medoids Algorithm (PAM)

K=2

Do loop

Until no change

Arbitrary choose k object as initial medoids

Assign each remaining object to nearest medoids

Randomly select a nonmedoid object, \( O_{\text{random}} \)

Swapping \( O \) and \( O_{\text{random}} \)

If quality is improved.

Total Cost = 20

Total Cost = 26

Source: “Data Mining: Concepts and Techniques”, Jiawei Han and Micheline Kamber
Hierarchical Clustering

- Use distance matrix as clustering criteria. This method does not require the number of clusters $k$ as an input, but needs a termination condition.

Source: “Data Mining: Concepts and Techniques”, Jiawei Han and Micheline Kamber
Dendrogram: Shows How the Clusters are Merged

Decompose data objects into a several levels of nested partitioning (tree of clusters), called a dendrogram.

A clustering of the data objects is obtained by cutting the dendrogram at the desired level, then each connected component forms a cluster.

Source: “Data Mining: Concepts and Techniques”, Jiawei Han and Micheline Kamber
Typical Alternatives to Calculate the Distance between Clusters

- Single link
- Complete link
- Average
- Centroid
- Medoid

Modified from: “Data Mining: Concepts and Techniques”, Jiawei Han and Micheline Kamber
Typical Alternatives to Calculate the Distance between Clusters (Cont.)

- **Single link**: smallest distance between an element in one cluster and an element in the other
  
  - $d(K_i, K_j) = \min(t_{ip}, t_{jq})$
Typical Alternatives to Calculate the Distance between Clusters (Cont.)

- **Complete link**: largest distance between an element in one cluster and an element in the other

  \[ d(K_i, K_j) = \max(t_{ip}, t_{jq}) \]
Typical Alternatives to Calculate the Distance between Clusters (Cont.)

- **Average**: avg distance between an element in one cluster and an element in the other
  - \[ d(K_i, K_j) = \text{avg}(t_{ip}, t_{jq}) \]
Typical Alternatives to Calculate the Distance between Clusters (Cont.)

- **Centroid**: distance between the centroids of two clusters,
  \[ d(K_i, K_j) = d(C_i, C_j) \]
Typical Alternatives to Calculate the Distance between Clusters (Cont.)

- **Medoid**: distance between the medoids of two clusters,
  - $d(K_i, K_j) = d(M_i, M_j)$
  - Medoid: one chosen, centrally located object in the cluster
Density-Based Clustering Methods

- Clustering based on density (local cluster criterion), such as density-connected points
- Major features:
  - Discover clusters of arbitrary shape
  - Handle noise
  - One scan
  - Need density parameters as termination condition
- Several interesting studies:
  - **DBSCAN**: Ester, et al. (KDD’96)
  - **DENCLUE**: Hinneburg & D. Keim (KDD’98)
  - **CLIQUE**: Agrawal, et al. (SIGMOD’98) (more grid-based)

Source: “Data Mining: Concepts and Techniques”, Jiawei Han and Micheline Kamber
Density-Based Clustering: Basic Concepts

- Two parameters:
  - \textit{Eps}: Maximum radius of the neighbourhood
  - \textit{MinPts}: Minimum number of points in an Eps-neighbourhood of that point

- \( N_{Eps}(p) := \{q \text{ belongs to } D \mid \text{dist}(p,q) \leq Eps\} \)

- Directly density-reachable: A point \( p \) is directly density-reachable from a point \( q \) w.r.t. \( Eps, MinPts \) if
  - \( p \) belongs to \( N_{Eps}(q) \)
  - core point condition:
    \[ |N_{Eps}(q)| \geq MinPts \]

Source: “Data Mining: Concepts and Techniques”, Jiawei Han and Micheline Kamber
Density-Reachable and Density-Connected

- **Density-reachable:**
  - A point $p$ is **density-reachable** from a point $q$ w.r.t. $Eps, MinPts$ if there is a chain of points $p_1, \ldots, p_n, p_1 = q, p_n = p$ such that $p_{i+1}$ is directly density-reachable from $p_i$.

- **Density-connected**
  - A point $p$ is **density-connected** to a point $q$ w.r.t. $Eps, MinPts$ if there is a point $o$ such that both, $p$ and $q$ are density-reachable from $o$ w.r.t. $Eps$ and $MinPts$.

Source: “Data Mining: Concepts and Techniques”, Jiawei Han and Micheline Kamber
DBSCAN: Density Based Spatial Clustering of Applications with Noise

- Relies on a density-based notion of cluster: A cluster is defined as a maximal set of density-connected points.
- Discovers clusters of arbitrary shape in spatial databases with noise.

Source: "Data Mining: Concepts and Techniques", Jiawei Han and Micheline Kamber
DBSCAN: Sensitive to Parameters

Figure 8. DBScan results for DS1 with MinPts at 4 and Eps at (a) 0.5 and (b) 0.4.

Figure 9. DBScan results for DS2 with MinPts at 4 and Eps at (a) 5.0, (b) 3.5, and (c) 3.0.

Source: “Data Mining: Concepts and Techniques”, Jiawei Han and Micheline Kamber
Outlines

- What is Cluster Analysis?
- Types of Data in Cluster Analysis
- Major Clustering Methods
- Biclustering for Gene Expression Data

Biclustering algorithms for biological data analysis: a survey
Madeira, S.C.; Oliveira, A.L.;
IEEE/ACM Transactions on Computational Biology and Bioinformatics,
Volume 1, Issue 1, Jan.-March 2004 Page(s):24 - 45
## Gene Expression Data

<table>
<thead>
<tr>
<th>Gene 1</th>
<th>Condition 1</th>
<th>...</th>
<th>Condition ( j )</th>
<th>...</th>
<th>Condition ( m )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( a_{11} )</td>
<td>...</td>
<td>...</td>
<td>( a_{ij} )</td>
<td>...</td>
<td>( a_{1m} )</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>Gene ( i )</td>
<td>( a_{i1} )</td>
<td>...</td>
<td>( a_{ij} )</td>
<td>...</td>
<td>( a_{im} )</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>Gene ( n )</td>
<td>( a_{n1} )</td>
<td>...</td>
<td>( a_{nj} )</td>
<td>...</td>
<td>( a_{nm} )</td>
</tr>
</tbody>
</table>

- **Row:**
  - gene

- **Column:**
  - condition

- **Element:**
  - expression level of a gene under a specific condition
What is Biclustering?

- **Bicluster**
  - A subset of rows that exhibit similar behavior across a subset of columns, and vice versa.

- **Biclustering**
  - Also named coclustering, bidimensional clustering, subspace clustering

- **Clustering vs. biclustering**
  - global model vs. local model

- **Goal**
  - To identify subgroups of genes and subgroups of conditions
  - By performing simultaneous clustering of both rows and columns
Notation

- Given the matrix $A = (X, Y)$
  - $X$: set of rows
  - $Y$: set of columns
- A bicluster is a submatrix $(I, J)$
  - $I$ is a subset of rows $X$
  - $J$ is a subset of the columns $Y$
- $a_{ij}$
  - The value in the matrix $A$ corresponding to row $i$ and column $j$
Bicluster Types

- Biclusters with
  - constant values
  - constant values on rows or columns
  - coherent values
  - coherent evolutions
Bicluster Type (a)

- Constant Bicluster

\[ a_{ij} = \mu + \eta_{ij}, \]

where \( \eta_{ij} \) is the noise associated with the real value \( \mu \) of \( a_{ij} \).

- A subset of genes with similar expression values within a subset of conditions.

We will omit the noise term in the following slides and introduce the formulas of perfect biclusters.

<table>
<thead>
<tr>
<th>1.0</th>
<th>1.0</th>
<th>1.0</th>
<th>1.0</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.0</td>
<td>1.0</td>
<td>1.0</td>
<td>1.0</td>
</tr>
<tr>
<td>1.0</td>
<td>1.0</td>
<td>1.0</td>
<td>1.0</td>
</tr>
<tr>
<td>1.0</td>
<td>1.0</td>
<td>1.0</td>
<td>1.0</td>
</tr>
<tr>
<td>1.0</td>
<td>1.0</td>
<td>1.0</td>
<td>1.0</td>
</tr>
</tbody>
</table>
Bicluster Type (b)

- **Constant rows**
  
  $a_{ij} = \mu + \alpha_i,$
  
  $a_{ij} = \mu \times \alpha_i,$
  
  where $\mu$ is the typical value within the bicluster and $\alpha_i$ is the adjustment for row $i \in I$

- A subset of genes with similar expression values across a subset of conditions, allowing the expression levels to differ from gene to gene

<table>
<thead>
<tr>
<th>1.0</th>
<th>1.0</th>
<th>1.0</th>
<th>1.0</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.0</td>
<td>2.0</td>
<td>2.0</td>
<td>2.0</td>
</tr>
<tr>
<td>3.0</td>
<td>3.0</td>
<td>3.0</td>
<td>3.0</td>
</tr>
<tr>
<td>4.0</td>
<td>4.0</td>
<td>4.0</td>
<td>4.0</td>
</tr>
</tbody>
</table>
Bicluster Type (c)

- **Constant columns**
  \[ a_{ij} = \mu + \beta_j, \]
  \[ a_{ij} = \mu \times \beta_j, \]
  where \( \mu \) is the typical value within the bicluster and \( \beta_j \) is the adjustment for column \( j \in J \).

- A subset of conditions within which a subset of genes present similar expression values, allowing the expression levels to differ from condition to condition.
Bicluster Type (d)

- **Coherent values** (additive model)

\[ a_{ij} = \mu + \alpha_i + \beta_j, \]

where \( \mu \) is the typical value within the bicluster, \( \alpha_i \) is the adjustment for row \( i \in I \), and \( \beta_j \) is the adjustment for column \( j \in J \).

- A subset of genes and a subset of conditions with coherent values on both rows and columns
Coherent values
(multiplicative model)

\[ a_{ij} = \mu \times \alpha_i \times \beta_j, \]

where \( \mu \) is the typical value within the bicluster, \( \alpha_i \) is the adjustment for row \( i \in I \), and \( \beta_j \) is the adjustment for column \( j \in J \)
Bicluster Type (f)

- Overall coherent evolution

- Coherent evolution
  - Coherent behaviors regardless of the exact numeric values in the data matrix

- A subset of conditions that have the same effects on a subset of genes
Bicluster Type (g)

- **Coherent evolution on the rows**

- A subset of genes that are upregulated or downregulated across a subset of conditions without taking into account their actual expression levels

<table>
<thead>
<tr>
<th></th>
<th>S1</th>
<th>S1</th>
<th>S1</th>
<th>S1</th>
</tr>
</thead>
<tbody>
<tr>
<td>S2</td>
<td>S2</td>
<td>S2</td>
<td>S2</td>
<td>S2</td>
</tr>
<tr>
<td>S3</td>
<td>S3</td>
<td>S3</td>
<td>S3</td>
<td>S3</td>
</tr>
<tr>
<td>S4</td>
<td>S4</td>
<td>S4</td>
<td>S4</td>
<td>S4</td>
</tr>
</tbody>
</table>
Bicluster Type (h)

- Coherent evolution on the columns

- A subset of conditions that have always the same or opposite effects on a subset of genes
Bicluster Type (i)

- Coherent evolution on the columns

- Order-preserving submatrix
  - There is a permutation of its columns under which the sequences of values in every row is strictly increasing

<table>
<thead>
<tr>
<th></th>
<th>70</th>
<th>13</th>
<th>19</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>49</td>
<td>40</td>
<td>49</td>
<td>35</td>
<td></td>
</tr>
<tr>
<td>40</td>
<td>20</td>
<td>27</td>
<td>15</td>
<td></td>
</tr>
<tr>
<td>90</td>
<td>15</td>
<td>20</td>
<td>12</td>
<td></td>
</tr>
</tbody>
</table>
**Bicluster Type (j)**

- **Coherent sign changes on rows and columns**

- A subset of genes and a subset of conditions with coherent positive and negative changes relatively to a normal value

<p>| | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>↑</td>
<td>↑</td>
<td>↓</td>
<td>←</td>
</tr>
<tr>
<td>↓</td>
<td>←</td>
<td>↑</td>
<td>→</td>
</tr>
<tr>
<td>←</td>
<td>→</td>
<td>↑</td>
<td>→</td>
</tr>
<tr>
<td>→</td>
<td>→</td>
<td>↑</td>
<td>→</td>
</tr>
</tbody>
</table>
Bicluster Structure

- One bicluster vs. $K$ biclusters

- **Exclusive** vs. nonexclusive
  - A row or a column can only belong to one bicluster

- **Overlapping** vs. nonoverlapping
  - A particular pair (row, column) belongs to more than one bicluster

- **Exhaustive** vs. nonexhaustive
  - Every row and every column belongs to at least one bicluster
Bicluster Structure (a)

- Single bicluster
Bicluster Structure (b)

- **Exclusive row and column biclusters**

- Every row and every column in the matrix belongs exclusively to one of the $K$ biclusters
Bicluster Structure (c)

- Checkerboard structure
- Nonoverlapping and nonexclusive biclusters where each row in the data matrix belongs to exactly $K$ biclusters
Bicluster Structure (d)

- **Exclusive rows biclusters**

- Rows can only belong to one bicluster, while columns can belong to several biclusters
Bicluster Structure (e)

- **Exclusive columns biclusters**

- Columns can only belong to one bicluster, while rows can belong to several biclusters
Bicluster Structure (f)

- Nonoverlapping biclusters with tree structure
Bicluster Structure (g)

- Nonoverlapping nonexclusive biclusters
Bicluster Structure (h)

- Overlapping biclusters with hierarchical structure
- Either the biclusters are disjoint or one includes the other
Bicluster Structure (i)

- Arbitrarily positioned overlapping biclusters