Statistical Analysis of Microarray Data

Clustering

Jacques van Helden

Jacques.van-Helden@univ-amu.fr
Aix-Marseille Université, France
Technological Advances for Genomics and Clinics
(TAGC, INSERM Unit U1090)
http://jacques.van-helden.perso.luminy.univ-amu.fr/

FORMER ADDRESS (1999-2011)
Université Libre de Bruxelles, Belgique
Bioinformatique des Génomes et des Réseaux (BiGRe lab)
http://www.bigre.ulb.ac.be/
Contents

- Data sets
- Distance and similarity metrics
- K-means clustering
- Hierarchical clustering
- Evaluation of clustering results
Introduction to clustering

- Clustering is an *unsupervised* approach
  - Class discovery: starting from a set of objects, group them into classes, without any prior knowledge of these classes.

- There are many clustering methods
  - hierarchical
  - k-means
  - self-organizing maps (SOM)
  - knn
  - ...

- The results vary drastically depending on
  - clustering method
  - similarity or dissimilarity metric
  - additional parameters specific to each clustering method (e.g. number of centres for the k-mean, agglomeration rule for hierarchical clustering, ...)
Statistical Analysis of Microarray Data

Data sets
DeRisi et al published the first article describing a full-genome monitoring of gene expression data.

This article reported an experiment called “diauxic shift” with 7 time points.

Initially, cells are grown in a glucose-rich medium.

As time progresses, cells
- Consume glucose -> when glucose becomes limiting
  - Glycolysis stops
  - Gluconeogenesis is activated to produce glucose
- Produce by-products -> the culture medium becomes polluted/
  - Stress response

Cell cycle data

- Spellman et al. (1998)
- Time profiles of yeast cells followed during cell cycle.
- Several experiments were regrouped, with various ways of synchronization (elutriation, cdc mutants, …)
- ~800 genes showing a periodic patterns of expression were selected (by Fourier analysis)

Gene expression data – response to environmental changes

- Gasch et al. (2000), 173 chips (stress response, heat shock, drugs, carbon source, …)

Gene expression data - carbon sources

- Gasch et al. (2000), 173 chips (stress response, heat shock, drugs, carbon source, ...)
- We selected the 13 chips with the response to different carbon sources.

Data standardization and filtering

- For the cell cycle experiments, genes had already been filtered in the original publication. We used the 800 selected genes for the analysis.
- For the diauxic shift and carbon source experiments, each chip contained >6000 genes, most of which are un-regulated.
- Standardization
  - We applied a chip-wise standardization (centring and scaling) with robust estimates (median and IQR) on each chip.
- Filtering
  - Z-scores obtained after standardization were converted
    - to P-value (normal distribution)
    - to E-value (= P-value * N)
  - Only genes with an E-value < 1 were retained for clustering.
Filtering of carbon source data

Gene expression profiles

Chip-wise standardization

Z-scores

Threshold filtering

Profiles of regulated genes

<table>
<thead>
<tr>
<th>ORF</th>
<th>ethanol</th>
<th>galactose</th>
<th>glucose</th>
<th>mannose</th>
<th>raffinose</th>
<th>sucrose</th>
<th>fructose</th>
<th>galactose</th>
<th>glucose</th>
<th>mannose</th>
<th>raffinose</th>
<th>sucrose</th>
</tr>
</thead>
<tbody>
<tr>
<td>YAL066W</td>
<td>0.71</td>
<td>-1.87</td>
<td>-1.15</td>
<td>-4.90</td>
<td>-1.85</td>
<td>-3.81</td>
<td>-3.34</td>
<td>-0.96</td>
<td>-3.41</td>
<td>-1.04</td>
<td>0.36</td>
<td>-1.55</td>
</tr>
<tr>
<td>YAR008W</td>
<td>-2.70</td>
<td>0.36</td>
<td>0.03</td>
<td>-4.90</td>
<td>-0.94</td>
<td>-0.53</td>
<td>0.64</td>
<td>-0.53</td>
<td>-2.57</td>
<td>-0.73</td>
<td>0.38</td>
<td>1.75</td>
</tr>
<tr>
<td>YAR071W</td>
<td>-5.43</td>
<td>-1.22</td>
<td>2.73</td>
<td>-0.44</td>
<td>-0.24</td>
<td>3.24</td>
<td>6.69</td>
<td>1.10</td>
<td>-5.21</td>
<td>1.39</td>
<td>-0.70</td>
<td>0.22</td>
</tr>
<tr>
<td>YBL005W</td>
<td>1.40</td>
<td>3.05</td>
<td>3.97</td>
<td>4.92</td>
<td>1.18</td>
<td>5.52</td>
<td>-0.53</td>
<td>0.79</td>
<td>-0.84</td>
<td>-1.00</td>
<td>1.12</td>
<td>2.26</td>
</tr>
<tr>
<td>YEL015W</td>
<td>4.00</td>
<td>0.28</td>
<td>-3.46</td>
<td>-3.65</td>
<td>-2.38</td>
<td>-4.94</td>
<td>-3.26</td>
<td>-4.84</td>
<td>0.59</td>
<td>-3.76</td>
<td>-1.82</td>
<td>1.98</td>
</tr>
<tr>
<td>YBL043W</td>
<td>3.91</td>
<td>-1.16</td>
<td>-4.89</td>
<td>-4.90</td>
<td>-1.61</td>
<td>-4.76</td>
<td>4.47</td>
<td>-6.97</td>
<td>-0.61</td>
<td>-6.67</td>
<td>-7.12</td>
<td>0.78</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>
Statistical Analysis of Microarray Data

Hierarchical clustering
In 1998, Eisen et al. implemented a software tool called Cluster, which combines hierarchical clustering and heatmap visualization. They applied it to extract clusters of co-expressed genes from various types of expression profiles.

Clustering with gene expression data

**Gene expression profiles** → **Chip-wise standardization** → **Z-scores** → **Threshold filtering** → **Profiles of regulated genes**

### Carbon sources

<table>
<thead>
<tr>
<th>ORF</th>
<th>Ethanol</th>
<th>Galactose</th>
<th>Glucose</th>
<th>Mannose</th>
<th>Raffinose</th>
<th>Sucrose</th>
<th>Fructose</th>
<th>Galactose vs ref.pool</th>
<th>Glucose vs ref.pool</th>
<th>Mannose vs ref.pool</th>
<th>Raffinose vs ref.pool</th>
<th>Sucrose vs ref.pool</th>
</tr>
</thead>
<tbody>
<tr>
<td>YAL066W</td>
<td>0.71</td>
<td>-1.87</td>
<td>-1.15</td>
<td>-4.90</td>
<td>-1.85</td>
<td>-3.81</td>
<td>-3.34</td>
<td>-0.96</td>
<td>-0.96</td>
<td>0.36</td>
<td>-1.55</td>
<td>-0.87</td>
</tr>
<tr>
<td>YAR008W</td>
<td>-2.70</td>
<td>0.36</td>
<td>0.03</td>
<td>-4.90</td>
<td>-0.94</td>
<td>-0.53</td>
<td>0.64</td>
<td>-0.53</td>
<td>-0.73</td>
<td>0.38</td>
<td>-1.75</td>
<td>-0.55</td>
</tr>
<tr>
<td>YAR071W</td>
<td>-5.43</td>
<td>-1.22</td>
<td>2.73</td>
<td>-0.44</td>
<td>-0.24</td>
<td>3.24</td>
<td>-6.99</td>
<td>1.10</td>
<td>-5.21</td>
<td>1.39</td>
<td>-0.70</td>
<td>0.22</td>
</tr>
<tr>
<td>YAR071W</td>
<td>1.40</td>
<td>3.05</td>
<td>3.97</td>
<td>4.92</td>
<td>1.18</td>
<td>5.52</td>
<td>-0.53</td>
<td>0.79</td>
<td>-0.84</td>
<td>-1.00</td>
<td>1.12</td>
<td>-2.26</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>
Hierarchical clustering on gene expression data

Gene expression profiles → Chip-wise standardization → Z-scores


Distance matrix:

YAL066W 0.00 6.82 12.99 16.33 11.64 17.39 36.41 32.52 36.07 12.00 ...
YAR008W 6.82 0.00 11.70 13.69 12.58 18.18 37.51 33.46 37.18 12.36 ...
YAR071W 12.99 11.70 0.00 13.32 21.77 26.62 42.48 38.48 42.15 21.09 ...
YBL005W 16.33 13.69 13.32 0.00 19.52 25.04 44.95 41.16 44.62 17.86 ...
YBL015W 11.64 12.58 21.77 19.52 0.00 8.51 34.47 30.79 33.77 6.46 ...
YBL043W 17.39 18.18 26.62 25.04 8.51 0.00 31.74 28.64 30.90 11.13 ...
YBR018C 36.41 37.51 42.48 44.95 34.47 31.74 0.00 5.12 4.66 35.84 ...
YBR019C 32.52 33.46 38.48 41.16 30.79 28.64 5.12 0.00 4.81 32.58 ...
YBR020W 36.07 37.18 42.15 44.62 33.77 30.90 4.66 4.81 0.00 35.63 ...
YBR054W 12.00 12.36 21.09 17.86 6.46 11.13 35.84 32.58 35.63 0.00 ...
...
...
...
...
...
...
...
...

Threshold filtering → Profiles of regulated genes

Pairwise distance calculation → Distance matrix
Hierarchical clustering on gene expression data

1. Gene expression profiles
2. Chip-wise standardization
3. Z-scores
4. Threshold filtering
5. Profiles of regulated genes
6. Pairwise distance calculation
7. Distance matrix
8. Tree building
9. Tree
Hierarchical clustering is an aggregative clustering method
- takes as input a distance matrix
- progressively regroups the closest objects/groups

One needs to define a (dis)similarity metric between two groups. There are several possibilities
- **Average linkage**: the average distance between objects from groups A and B
- **Single linkage**: the distance between the closest objects from groups A and B
- **Complete linkage**: the distance between the most distant objects from groups A and B

### Algorithm
1. Assign each object to a separate cluster.
2. Find the pair of clusters with the shortest distance, and regroup them in a single cluster
3. Repeat (2) until there is a single cluster

The result is a tree, whose intermediate nodes represent clusters
- N objects $\rightarrow$ N-1 intermediate nodes

Branch lengths represent distances between clusters
Isomorphism on a tree

In a tree, the two children of any branch node can be swapped. The result is an isomorphic tree, considered as equivalent to the initial one.

The two trees shown here are equivalent, however
- Top tree: leaf 1 is far away from leaf 2
- Bottom tree: leaf 1 is neighbour from leaf 2

The vertical distance between two nodes does NOT reflect their actual distance!

The distance between two nodes is the sum of branch lengths.
Hierarchical clustering on gene expression data

Gene expression profiles → Chip-wise standardization → Z-scores

- Threshold filtering → Profiles of regulated genes
- Pairwise distance calculation → Distance matrix
- Hierarchical clustering → Tree
- Tree cut → Clusters
The choice of the agglomeration rule has a strong impact on the structure of a tree resulting from hierarchical clustering.

Those four trees were built from the same distance matrix, using 4 different agglomeration rules.

The clustering order is completely different.

Single-linkage typically creates nesting clusters ("Matryoshka dolls").

Complete and Ward linkage create more balanced trees.

Note: the matrix was computed from a matrix of random numbers. The subjective impression of structure are thus complete artifacts.
Golub 1999 - Impact of the linkage method (Euclidian distance for all the trees)
Golub 1999 - Effect of the distance metrics (complete linkage for all the trees)
Gene clustering highlights groups of genes with similar expression profiles.
Biclustering consists in clustering the rows (genes) and the columns (samples) of the data set.

This reveals some subgroups of samples.

With the golub 1999 data set

- The AML and ALL patients are clearly separated at the top level of the tree
- There are apparently two clusters among the AML samples.
Biclustering consists in clustering the rows (genes) and the columns (samples) of the data set.

This reveals some subgroups of samples.

With the golub 1999 data set
- The AML and ALL patients are clearly separated at the top level of the tree
- There are apparently two clusters among the ALL samples. Actually these two clusters correspond to distinct cell subtypes: T and B cells, respectively.
Impact of distance metrics and agglomeration rules

- **Euclidian**
  - Single
  - Average
  - Complete
  - Ward

- **Correlation**
  - Single
  - Average
  - Complete
  - Ward

- **Dot product**
  - Single
  - Average
  - Complete
  - Ward
Golub 1999 - Pruning the tree

golub; Euclidian distance; Ward linkage

pruned tree, k=8
Impact of the linkage method

Carbon sources; z-score > 4.8; single linkage; Euclidian distance

Carbon sources; z-score > 4.8; average linkage; Euclidian distance

Carbon sources; z-score > 4.8; complete linkage; Euclidian distance

Carbon sources; z-score > 4.8; ward linkage; Euclidian distance
Impact of the distance metric - complete linkage

Carbon sources; z-score > 4.8; complete linkage; Euclidian distance

Carbon sources; z-score > 4.8; complete linkage; Dot product

Carbon sources; z-score > 4.8; complete linkage; Correlation
Impact of the distance metric - single linkage
Carbon sources; Euclidian average
Pruning and cutting the tree

- The tree can be cut at level $k$ (starting from the root), which creates $k$ clusters
- A $k$-group partitioning is obtained by collecting the leaves below each branch of the pruned tree
Statistical Analysis of Microarray Data

K-means clustering
Clustering around mobile centres

- The number of centres (k) has to be specified a priori

Algorithm

1. Arbitrarily select k initial centres
2. Assign each element to the closest centre
3. Re-calculate centres (mean position of the assigned elements)
4. Repeat (2) and (3) until one of the stopping conditions is reached
   - the clusters are the same as in the previous iteration
   - the difference between two iterations is smaller than a specified threshold
   - the max number of iterations has been reached
Mobile centres example - initial conditions

- Two sets of random points are randomly generated
  - 200 points centred on (0,0)
  - 50 points centred on (1,1)
- Two points are randomly chosen as seeds (blue dots)
Mobile centres example - first iteration

Step 1
- Each dot is assigned to the cluster with the closest centre
- Centres are recalculated (blue star) on the basis of the new clusters
Mobile centres example - second iteration

At each step,
- points are re-assigned to clusters
- centres are re-calculated

Cluster boundaries and centre positions evolve at each iteration
Mobile centres example - after 3 iterations

At each step,
- points are re-assigned to clusters
- centres are recalculated

Cluster boundaries and centre positions evolve at each iteration
Mobile centres example - after 4 iterations

At each step,
- points are re-assigned to clusters
- centres are recalculated

Cluster boundaries and centre positions evolve at each iteration

iter.max = 4 ; iterations = 4
Mobile centres example - after 5 iterations

At each step,
- points are re-assigned to clusters
- centres are re-calculated

Cluster boundaries and centre positions evolve at each iteration

```
iter.max = 5 ; iterations = 5
```
Mobile centres example - after 6 iterations

At each step,
- points are re-assigned to clusters
- centres are recalculated

Cluster boundaries and centre positions evolve at each iteration
Mobile centres example - after 10 iterations

After some iterations (6 in this case), the clusters and centres do not change anymore.
Mobile centres example - random data

initial conditions

iter.max = 1 ; iterations = 1

iter.max = 2 ; iterations = 2

iter.max = 3 ; iterations = 3

iter.max = 4 ; iterations = 4

iter.max = 5 ; iterations = 5

iter.max = 6 ; iterations = 6

iter.max = 10 ; iterations = 6

iter.max = 15 ; iterations = 6
K-means clustering is a variant of clustering around mobile centres. After each assignation of an element to a centre, the position of this centre is recalculated. The convergence is much faster than with the basic mobile centre algorithm. After 1 iteration, the result might already be stable. K-means is time- and memory-efficient for very large data sets (e.g. thousands of objects).
Clustering with gene expression data

- Clustering can be performed in two ways
  - Taking genes as objects and conditions/cell types as variables
  - Taking conditions/cell types as objects and genes as variables

- Problem of dimensionality
  - When genes are considered as variables, there are many more variables than objects
  - Generally, only a very small fraction of the genes are regulated (e.g. 30 genes among 6,000)
  - However, all genes will contribute equally to the distance metrics
  - The noise will thus affect the calculated distances between conditions

- Solution
  - Selection of a subset of strongly regulated genes before applying clustering to conditions/cell types
K-means clustering is a variant of clustering around mobile centres. After each assignment of an element to a centre, the position of this centre is recalculated. The convergence is much faster than with the basic mobile centre algorithm: after 1 iteration, the result might already be stable. K-means is time- and memory-efficient for very large data sets (e.g. thousands of objects).
Diauxic shift: k-means clustering on all genes
Diauxic shift: k-means clustering on filtered genes
Diauxic shift: k-means clustering on permuted filtered genes
Cell cycle data: K-means clustering
Cell cycle data: K-means clustering, permuted data
Carbon sources: K-means clustering
Golub - K-means clustering
K-means clustering - summary

- **Strengths**
  - Simple to use
  - Fast
  - Can be used with very large data sets

- **Weaknesses**
  - The choice of the number of groups is arbitrary
  - The results vary depending on the initial positions of centres
  - The R implementation is based on Euclidian distance, no other metrics are proposed

- **Solutions**
  - Try different values for k and compare the result
  - For each value of k, run repeatedly to sample different initial conditions

- **Weakness of the solution**
  - Instead of one clustering, you obtain hundreds of different clustering results, totaling thousands of clusters, how to decide among them
Evaluation of clustering results
How to evaluate the result?

- It is very hard to make a choice between the multiple possibilities of distance metrics, clustering algorithms and parameters.
- Several criteria can be used to evaluate the clustering results
  - **Consensus**: using different methods, comparing the results and extracting a consensus
  - **Robustness**: running the same algorithm multiple times, with different initial conditions
  - Bootstrap
  - Jack-knife
  - Test different initial positions for the k-means
  - **Biological relevance**: compare the clustering result to functional annotations (functional catalogs, metabolic pathways, ...)


Comparing two clustering results

- If two methods return partitions of the same size, their clusters can be compared in a confusion table.
- Optimal correspondences between clusters can be established (permuting columns to maximize the diagonal).
- The consistency between the two classifications can then be estimated with the hit rate.
- Example:
  - Carbon source data, comparison of k-means and hierarchical clustering.

### Comparison Table

<table>
<thead>
<tr>
<th></th>
<th>h1</th>
<th>h2</th>
<th>h3</th>
<th>h4</th>
<th>h5</th>
<th>h6</th>
<th>h7</th>
<th>Sum</th>
</tr>
</thead>
<tbody>
<tr>
<td>k-means clustering</td>
<td>18</td>
<td>2</td>
<td>14</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>35</td>
</tr>
<tr>
<td>hierarchical clustering</td>
<td>4</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>4</td>
</tr>
<tr>
<td>k-means clustering</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>10</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>10</td>
</tr>
<tr>
<td>hierarchical clustering</td>
<td>0</td>
<td>0</td>
<td>10</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>19</td>
</tr>
<tr>
<td>k-means clustering</td>
<td>0</td>
<td>40</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>9</td>
<td>59</td>
</tr>
<tr>
<td>hierarchical clustering</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>5</td>
<td>0</td>
<td>0</td>
<td>19</td>
</tr>
<tr>
<td>k-means clustering</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>4</td>
<td>0</td>
<td>4</td>
</tr>
<tr>
<td>hierarchical clustering</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>Sum</td>
<td>42</td>
<td>14</td>
<td>12</td>
<td>22</td>
<td>24</td>
<td>15</td>
<td>4</td>
<td>133</td>
</tr>
</tbody>
</table>

### Correspondence Table

<table>
<thead>
<tr>
<th></th>
<th>h1</th>
<th>h2</th>
<th>h3</th>
<th>h4</th>
<th>h5</th>
<th>h6</th>
<th>h7</th>
</tr>
</thead>
<tbody>
<tr>
<td>k-means clustering</td>
<td>k4</td>
<td>k3</td>
<td>k5</td>
<td>k1</td>
<td>k2</td>
<td>k7</td>
<td>k6</td>
</tr>
<tr>
<td>h1</td>
<td>18</td>
<td>2</td>
<td>14</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>h2</td>
<td>4</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>h3</td>
<td>0</td>
<td>0</td>
<td>10</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>h4</td>
<td>0</td>
<td>10</td>
<td>0</td>
<td>40</td>
<td>0</td>
<td>0</td>
<td>9</td>
</tr>
<tr>
<td>h5</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>2</td>
<td>12</td>
<td>0</td>
<td>5</td>
</tr>
<tr>
<td>h6</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>4</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>h7</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Sum</td>
<td>22</td>
<td>12</td>
<td>24</td>
<td>42</td>
<td>14</td>
<td>4</td>
<td>15</td>
</tr>
</tbody>
</table>

- Matches: 84
- Hit rate: 63.2%
- Mismatches: 49
- Error rate: 36.8%
The bootstrap consists in repeating $r$ times (for example $r=100$) the clustering, using each time

- Either a different subset of variables
- Or a different subset of objects

The subset of variables is selected randomly, with resampling (i.e. the same variable can be present several times, whilst other variables are absent.

On the images the tree is colored according to the reproducibility of the branches during a 100-iterations bootstrap.