Clustering

Jaak Vilo
Data Mining Course
University of Tartu
vilo at ut . ee

Unsupervised vs. Supervised

Find groups inherent to data (clustering) Find a “classifier” for known classes
Clustering – it’s “easy” (for humans)

Clustering cont…
Analysis of biological samples with microarrays

Ratio quantitation

<table>
<thead>
<tr>
<th>Cy3</th>
<th>Cy5</th>
</tr>
</thead>
<tbody>
<tr>
<td>[red circle]</td>
<td>[green circle]</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>LogRatio 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>8</td>
</tr>
<tr>
<td>2</td>
</tr>
<tr>
<td>1</td>
</tr>
<tr>
<td>0.5</td>
</tr>
<tr>
<td>0.125</td>
</tr>
</tbody>
</table>
Image analysis basics

On both channels:
Identify the grid
Identify the spot

Spot foreground
Spot background

\[ \text{FG} - \text{BG} = \text{Intensity} \]

Many pixels – take mean or median

From microarray images to gene expression data

Raw data
Array scans

Intermediate data
Image quantifications

Final data
Samples

Spot
Spot/Image quantifications

Gene
Gene expression levels
Gene expression database – a conceptual view:

Using gene expression data matrix

- **Comparing genes** by comparing rows in the matrix, e.g., finding groups of co-regulated and potentially functionally related genes
- **Comparing samples** by comparing columns. For instance, finding genes affected by a toxin, toxin classification and prediction of toxicity effects
These gene expression vectors of log(ratio) values can be used to construct an expression matrix.

<table>
<thead>
<tr>
<th></th>
<th>Expt 1</th>
<th>Expt 2</th>
<th>Expt 3</th>
<th>Expt 4</th>
<th>Expt 5</th>
<th>Expt 6</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gene₁</td>
<td>-1.2</td>
<td>-0.5</td>
<td>0</td>
<td>0.25</td>
<td>0.75</td>
<td>1.4</td>
</tr>
<tr>
<td>Gene₂</td>
<td>0.2</td>
<td>-0.5</td>
<td>1.2</td>
<td>-0.25</td>
<td>-1.0</td>
<td>1.5</td>
</tr>
<tr>
<td>Gene₃</td>
<td>1.2</td>
<td>0.5</td>
<td>0</td>
<td>-0.25</td>
<td>-0.75</td>
<td>-1.4</td>
</tr>
<tr>
<td>etc.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- This is often represented as a red/green colored matrix.

**Expression Matrix**

The Expression Matrix is a representation of data from multiple microarray experiments.

- Each element is a log ratio, usually \( \log_2(Cy5/Cy3) \).
- Black indicates a log ratio of zero, i.e., \( Cy5 \) and \( Cy3 \) are very close in value.
- Green indicates a negative log ratio, i.e., \( Cy5 < Cy3 \).
- Red indicates a positive log ratio, i.e., \( Cy5 > Cy3 \).
- Gray indicates missing data.
Tumor classification:
1) class prediction  2) class discovery

Gölbub et al, Science Oct 15th 1999
- 38 samples of acute myeloic leukemia (AML) and acute lymphoblastic leukemia (ALL)
- 6817 genes
- classifier built based on 50 best correlated genes
- tested on 34 new samples, 29 of them predicted accurately
Distance measures:
which two profiles are similar to each other?

- Euclidean, Manhattan etc.

- Correlation, angle, etc.

- Rank correlation
  1. 
  2. 
  3. 

- Time warping

Distance measures

- How to formally describe which objects are “close” to each other, and which are not
- More than one way to define distances.
- Distance is a metric, if
  - $d(X,X) = 0$
  - $d(X,Y) = d(Y,X) \geq 0$
  - $d(A,B) \leq d(A,C) + d(C, B)$
Some standard distance measures

Euclidean distance
\[ d(f, g) = \sqrt{\sum_{i=1}^{c} (f_i - g_i)^2} \]

Euclidean squared
\[ d(f, g) = \sum_{i=1}^{c} (f_i - g_i)^2 \]

Manhattan (city-block)
\[ d(f, g) = \sum_{i=1}^{c} |f_i - g_i| \]

Average distance
\[ d(f, g) = \frac{1}{c} \sum_{i=1}^{c} (f_i - g_i)^2 \]

Pearson correlation
\[ d(f, g) = 1 - \frac{\sum_{i=1}^{c} (f_i - \overline{f})(g_i - \overline{g})}{\sqrt{\sum_{i=1}^{c} (f_i - \overline{f})^2 \sum_{i=1}^{c} (g_i - \overline{g})^2}} \]

If means of each row are 0, then it becomes:
\[ d(f, g) = 1 - \frac{\sum_{i=1}^{c} f_i g_i}{\sqrt{\sum_{i=1}^{c} f_i^2 \sum_{i=1}^{c} g_i^2}} = 1 - \cos \Theta \]
Chord distance

\[ d(f, g) = \sqrt{2(1 - \frac{\sum_{i=1}^{c} f_i g_i}{\sqrt{\sum_{i=1}^{c} f_i^2 \sum_{i=1}^{c} g_i^2}})} \]

\[ d(f, g) = \sqrt{2(1 - \cos \Theta)} \]

Euclidean distance between two vectors whose length has been normalized to 1

Legendre & Legendre: Numerical Ecology 2nd ed.

Rank correlation

\[ d(f, g) = 1 - \frac{6 \sum_{i=1}^{c} (\text{rank}_{fi} - \text{rank}_{gi})}{c(c^2 - 1)} \]

Sort and rank - smallest has rank 1, next 2, etc.

Equal values have a rank that is average of the supposed to be ranks

\begin{align*}
  f &= 3 \quad 17 \quad 12 \quad 12 \quad 8 \\
  \text{rank} &= 1 \quad 5 \quad 3.5 \quad 3.5 \quad 2
\end{align*}
Edit distance

- Strings X, Y
- How many edit operations (delete or add a character, replace a character) are needed to get a Y from X?
- Levenshtein metric
- Effective Dynamic Programming algorithms (see Text Algorithms course)

Kolmogorov Complexity

- Text x
- \( K(x) \) – shortest program \( x^* \) to produce x
- \( K(x|y) \) – shortest program to produce x given y
- Mutual Information
  - \( I(x:y) = K(x)+K(y) - K(x,y) \)
Information Content Distance

\[ d(x, y) = \frac{\max\{K(x \mid y^*), K(y \mid x^*)\}}{\max\{K(x), K(y)\}} \approx \frac{|C(xy) - C(x)|}{|C(y)|} \]

- \( C(X) = \text{gzip } -c \text{ file}_X \mid \text{wc } -c \)

Rakendused

- Genoomide võrdlemine
- Autorsuse tuvastamine
- Keelte suguluse määramine
Hierarchical clustering

All against all distance matrix
Distance: Euclidean, average linkage
Hierarchical clustering

All against all distance matrix
Distance: Euclidean, average linkage

\[ D(1:2:3, 4:5) = 4.5 \]
Hierarchical clustering

1. All against all distance matrix
2. Linkage strategy – identify closest clusters and merge

Performance:
$O(dn^2)$

Cluster matrices:

Keep joining together two closest clusters by using the:
- Minimum distance => Single linkage
- Maximum distance => Complete linkage
- Average distance => Average linkage
  (UPGMA, WPGMA)
WPGMA, UPGMA

(Un)Weighted Pair Group Method with Arithmetic Mean

\[ d(C, AB) = \frac{(d(C, A) + d(C, B))}{2} \]

WPGMA: \[ d(C, AB) = \frac{(d(C, A) + d(C, B))}{2} \]

UPGMA: \[ d(C, AB) = \frac{(d(C, A) \times |A| + d(C, B) \times |B|)}{|A| + |B|} \]

Centroid

(Un)Weighted Pair Group Method with Arithmetic Mean

\[ d(C, AB) = \frac{(d(C, A) + d(C, B))}{2} \]
The method used in this example is called WPGMA (weighted pair group method using arithmetic averaging) because the distance between clusters is calculated as a simple average. [For example, in the last step the WPGMA distance between (AB) and C+(DE) = (55 + 90) / 2 = 72.5]. Though computationally easier, when there are unequal numbers of taxa in the clusters, the distances in the original matrix do not contribute equally to the intermediate calculations, and the final result is therefore said to be weighted.

A more commonly used method is UPGMA (unweighted PGMA), in which averages are weighted by the number of taxa in each cluster. [The calculation is slightly more complicated. For example, in the last step the UPGMA distance between (AB) and C+(DE) = (55 + 2x90) / 3 = 78.33]. As a result, each distance contributes equally to the final result, which is therefore said to be unweighted.

Running time for hierarchical clustering

Data size vs. Time in seconds

<table>
<thead>
<tr>
<th>Data Size</th>
<th>Distances 10 attrib</th>
<th>Clustering 10,100, 1000 dim</th>
<th>Distances 10 attrib</th>
</tr>
</thead>
<tbody>
<tr>
<td>10K</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>15K</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>20K</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Note: The running time is measured in seconds, with a 1 minute and 5 minute mark.

Graph showing the relationship between data size and running time for hierarchical clustering.
Limits of standard clustering

- Hierarchical clustering is (very) good for visualization (first impression) and browsing
- Speed for modern data sets remains relatively slow (minutes or even hours)
- ArrayExpress database needs some faster analytical tools
- Hard to predict number of clusters (=>Unsupervised)
Speeding up

• Have to avoid All-against-All distance calculations

• Use clustering that doesn’t require all
  • K-means
  • SOM
  • …
  • But … we don’t know the nr. of clusters…

Approximate distances

• Triangle inequality for metrics

\[ |d(A, B) - d(B, C)| \leq d(A, C) \leq d(A, B) + d(B, C) \]
Large example

Dataset: Gene expression data
- \( N = 6000 \) datapoints in 80 dimensions;
- \( N \times (N-1)/2 \approx 18 \text{ million} \) pairs of datapoints.

Heuristic algorithm
- calculated 1.4 million distances, i.e. \(~7\%\);
- found the 10000 most similar pairs.

1.4 million randomly chosen distances contain on the average only 700 of these 10000 pairs.

Results

Dataset: \( N=6000 \)

Heuristic hierarchical clustering using
a) random distances (7% calculated)
b) heuristic small distances (1% calculated)

had about the same quality!
6200 genes, 80 exp.

Monitor size 1600x1200 pixels

Laptop: 800x600
6200 genes, 80 exp.

Monitor size 1600x1200
Laptop: 800x600

“COLLAPSE”
75 subtrees

Developed and implemented in Expression Profiler in October 2000 by Jaak Vilo

Heat map color schema design
K-means

1. Guess K centres

2. Assign obj to clusters

3. Move C to gravity centres

Alizadeh et al., Nature 403:503-11, 2000
K-means

- New centers - center of gravity for a cluster
- Cluster objects closest to a center

* Start clustering by choosing K centers randomly
* Iterate clustering step until no cluster changes
* Deterministic, might get "stuck" in local minimum

K-means clustering
K-medoids

- Choose the cluster center to be one of the existing objects.
- Why?
  - If more complex data or distance measure the “Real” center could not be found easily
  - Instead of trying to “invent” – use one of the existing objects, whatever the distance measure
Self Organising Maps (SOM)

MxN matrix of neurons, each representing “a cluster”
Object X is put to Wi, to which it is most similar.
Wi and its near surrounding is changed to resemble X more
Train, train, train…

Problem - there is no clear objective function to map D-dimenisonal data to 2 dimensions...

SOM topologies

• Different topologies of the nodes

• E.g. Linear – n nodes (Eisen et.al)
SOTA tree

- Top-down hierarchical clustering
- At each node division 2-node SOM
- Divide recursively

Some examples of SOM application

The ET-MAP (Chen, 1997)
- A hierarchical set of category map (visual directories) - uses a ‘land use’ metaphor to display over 110,000 entertainment-related web pages listed by Yahoo.
Some examples of SOM application

WebSOM: Lagus et al. (1996); Honkela et al. (1998) - HUT NN Research Centre

- SOM analysis technique to map thousands of articles posted on Usenet newsgroups

Some examples of SOM application

World Poverty Map: Kaski et al. (1997)

[Map of the world showing poverty levels]
Clustering etc. algorithms

- Hierarchical clustering methods + visualisation
- K-means, Self Organising Maps (SOM)
- SOTA trees (Self Organising Maps + Tree)
- Fuzzy, EM (object can belong to several clusters)
- Graph theory (cliques, strongly connected components)
- Similarity search: X -> Y s.t. d(X,Y)< 0.3
- Model based (rediscover distributions)
- Planar embeddings, Multidimensional scaling
- Principal Component Analysis
- Correspondence analysis
- Independent Component Analysis
Similarity searches

Query: “cyc1”  
(cyc1, activator for cyc1, repressor for cyc1)

=> 3 genes + 10 most similar ones for each

= 3 “clusters”
Similarity searches

Expand a tight cluster by other most similar genes:

Components of
Expression Profiler
http://ep.ebi.ac.uk/

Expression data

EPCLUST
Expression data

EP:GO
GeneOntology

SEQLOGO

URLMAP
provide links

PATMATCH
visualise patterns

External data, tools pathways, function, etc.

GENOMES
sequence, function, annotation

SPEXS
discover patterns
Getting data into EP

Expression Profiler: **EPCLUST**

* DATA → SELECT/FILTER → FOLDER → ANALYZE *

- URL: `http://host/data.txt`
- `http://host/cgi?id=D&d=ratio`

A “CLUSTER”

- GeneOntology
- Pathways
- Databases
- SPEXS
- Other tools
- ...

URLMAP
Data selection/filtering

- Many methods applicable
- Interface problems harder than methods
- How thoroughly anybody can go through all the criteria?
- Standard filters?
- Missing values
- Data randomization and generation

Complex data selection queries
EPCLUST features

- Hierarchical and K-means clustering
- Many distance measures
- Similarity searches (!)
- Data rescaling (log, root, etc), normalization
- Interactive image maps (PNG,GIF)
- User data upload
- WWW-based, no Java => client platform independent, also runs on thin clients
- Data matrix transposing (clustering of columns)
- Data randomization
- Select from alternative annotations for genes
- Sequence clustering (dependent on distance measures for sequences)
- Connection to URLMAP, providing links to other tools and databases

EP “old” way

PERL

require "$LIBDIR/Clustering.pl";
print "\t";
for $i (1..$N)
    $value += func();
    print "\t$value \n"
}
print "\n";
system( "bin.cluster -f f1 -p 2.3 ...");
open( FILES , "$DIR/outputfile.txt");
 Hierarchical clustering output

• Simple Web UI: Basic Architecture
  • XML Component Descriptions & XSLT Rendering
  • Chainable Components

Web Interface (Services/UI/etc.) -> Request Response
XSLT Processor

EP Component (EPC) XML
EPC Rendering XSL

EP Database
Internal and 3rd party Components

ArrayExpress
EP File-system

External Services Access

R (S-SPLUS)
EPC XML Components

• Inputs
  • Grouped into sections/subsections
  • Input names, type, validation type
  • Used for rendering

• Dynamic data
  • External service access (e.g. via Perl)
  • EP DB/file system access, etc.

• Outputs
  • Output name, type, validation type
  • Output format (e.g. regular expression)

• Action
  • URL (or other definition)
  • EPC target IDs

```xml
<?xml version="1.0" encoding="UTF-8" ?>
  <component id="3" title="Hierarchical CI">
    <section title="Available datasets" id="">
      <shorthand>Get one of the datasets</shorthand>
      <dynamic_list name="DATANAME" call="
        <shorthand>Get one of the datasets</shorthand>
        <dynamic_input type="radio" name="
          <shorthand>Choose the clustering</shorthand>
          </dynamic_input>
        </section>
```

Expression Profiler (component interface)
**Transformed with XSL**

- Accessing EP DB for definition of dynamic input elements

---

**BGA data input screen, where a user inputs a gene expression dataset and vector defining known or predefined classes of each sample**

---

**3D visualisation of the resulting first three axes of sample co-ordinates from a BGA analysis**
Design any heat-map coloring scheme

Heat map color schema design

To create a customised colour scheme, you will need to first create a colour palette and, then, create the discretisation.

1. Click on the colour squares in the colour map on the left in the order of the colours in the desired palette, from left to right.
To create a familiar Green-Black-Red palette, click on a green square, followed by a black and a red square. You'll see the palette being formed below the colour cube map.

2. Pick a discretisation type (linear/exponential/harmonic/histogram), and enter the number of slots into which you'd like to discretise the palette.

3. Click on the "CREATE DISCRETISATION" link. You'll see the palette broken up into chunks of colour. Linear discretisation will contain equal size chunks, and others will have proportionately varied colour stretches.

4. Click on the "Save" button to save this palette in the main colour options screen.
Latest Developments

- Other [clustering] methods
  - K-Medoids (Meelis Kull, Jaak Vilo, Tartu)
  - Gene ordering (Karlis Freivalds, Riga)
  - Principal Components Analysis (Aedin Culhane, Cork)
  - Correspondence Analysis (Aedin)

- Supervised methods
  - Between Group Analysis (Aedin Culhane et al, 2002)

- EP Architecture
  - Metadata completely in the database
  - Users, groups, user/group projects: folders, subfolders
  - Data sharing between users/groups

- Integration with R (S-PLUS)
  - Descriptive Statistics (mean, median, σ (std. dev.))
  - Histograms, QQ-plots, etc.
  - Integrate w/Bioconductor (started – in BGA)

Future Work

- Other [clustering] methods
  - Two way clustering
  - Fast/distributed clustering methods
  - Cluster comparison methods

- Supervised methods
  - Decision tree algorithms
  - Attribute/instance filters

- EP Architecture
  - Possible integration with EMBOSS/Talisman
  - Better support for export of gene groups/clusters
  - Better integration framework with other tools

- Integration with R (S-PLUS)
  - Integrate w/Bioconductor and other packages
**Biclustering**

- Usual clustering algorithms are based on global similarities of rows or columns of an expression data matrix.
- But the similarity of the expression profiles of a group of genes may be restricted to certain experimental conditions.
- Goal of biclustering: identify “homogeneous” submatrices.
- Difficulties: computational complexity, assessing the statistical significance of results
- Example: Tanay et al. 2002.

---

**Correspondence analysis: Interpretation**

- Shows both gene-vectors and condition-vectors as dots in the plane
- Genes that are nearby are similar
- Conditions that are nearby are similar
- When genes and conditions point in the same direction then the gene is up-regulated in that condition.
Spellman et al took several samples per time-point and hybridized the RNA to a glass chips with all yeast genes.
Choose experiments where coexpressed
Choose genes which correspond to profile

M O D U L E S

Ihmels, J., et.al Nature Genetics vol 31, August 2002
Ihmels, J., Friedlander, G., Bergmann, S., Sarig, O., Yaniv, Z., Barkai, N.

*Nature Genetics* vol 31, august 2002

**Revealing modular organization in the yeast transcriptional network**

http://www.weizmann.ac.il/home/barkai
Large-scale expression data

- Combining data from cell-cycle and stress conditions
- Multiple sets of specific conditions
- Diverse conditions

Large-scale expression data challenges: context-specific regulation

- Irrelevant conditions contribute noise
- Combinatorial regulation
- Redundancy
- Worry about redundant context

Gene A and Gene B
- A and B in the same function
- A and B in different functions
Signature Algorithm

Choose experiments where coexpressed
Choose genes which correspond to profile

Input of the signature algorithm

Identify co-regulated core
Remove unrelated genes
Add additional genes from the genome
Signature Algorithm

Gene Score:

\[ s_g = \frac{1}{|s|} \sum_{c \in S} s_c E_g^c \]

Significant

Noise Reduction Example

Amino-acid Biosynthesis 119 genes

TGACTC: binding site of GCN4
Master regulator of Aa. biosynthesis

132 genes
Examples

- Module related to glycolysis
- Module of phosphate metabolism-related genes
- Module of genes regulated by a transcription factor (or have at least the same site present in upstreams)
- Yeast homologues to genes in *E.coli*
  TCA cycle => yeast TCA cycle
How many randomly added genes destroys a module?

If genes are not from the module (random) then adding more random ones destroys the signature
Idea of the algorithm

- Use prior knowledge about genes related
- Identify under which CONDITIONS they are related
- Identify which (possibly other) GENES are related under those conditions (throw away some, find new ones)
Examples

• Module related to glycolysis
• Module of phosphate metabolism-related genes
• Module of genes regulated by a transcription factor (or have at least the same site present in upstreams)
• Yeast homologues to genes in *E.coli*
  TCA cycle => yeast TCA cycle
How many randomly added genes destroys a module?

If genes are not from the module (random) then adding more random ones destroys the signature.