Application of Hierarchical Clustering to Find Expression Modules in Cancer

T. M. Murali

August 18, 2008

Innovative Application of Hierarchical Clustering

- *A module map showing conditional activity of expression modules in cancer*, Eran Segal, Nir Friedman, Daphne Koller and Aviv Regev, Nature Genetics 36, 1090–1098, 2004
- Analyse gene expression data to find groups of genes expressed in concert between different cancers.
- Use hierarchical clustering innovatively.

Goals

- Move away from standard approach: determine genes that respond (based on cut-off) and study these genes further.
- Develop method that can analyse large numbers (1000s) of samples across multiple conditions.

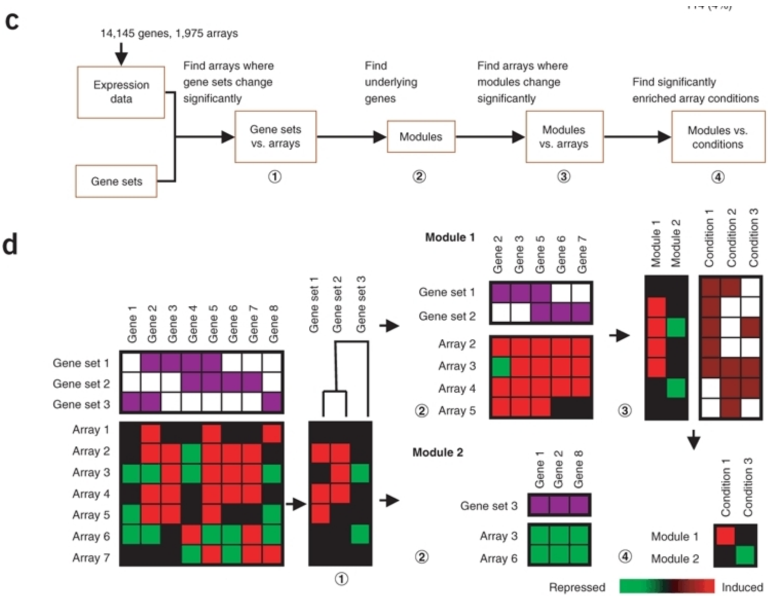
Goals

- Move away from standard approach: determine genes that respond (based on cut-off) and study these genes further.
- Develop method that can analyse large numbers (1000s) of samples across multiple conditions.
- Patterns of co-expression across all conditions are not very interesting.

Goals

- Move away from standard approach: determine genes that respond (based on cut-off) and study these genes further.
- Develop method that can analyse large numbers (1000s) of samples across multiple conditions.
- Patterns of co-expression across all conditions are not very interesting.
- Compute *gene modules*: groups of genes that show concerted behaviour across multiple conditions.
- Specifically, Segal et al. associate with each gene module, a set of samples in which the module is up-regulated and a set of samples in which the module is down-regulated.

Key Steps



c

14,145 genes, 1,975 arrays

Expression data

Gene sets

Find arrays where gene sets change significantly

Gene sets vs. arrays ①

Find underlying genes

Modules ②

Find arrays where modules change significantly

Modules vs. arrays ③

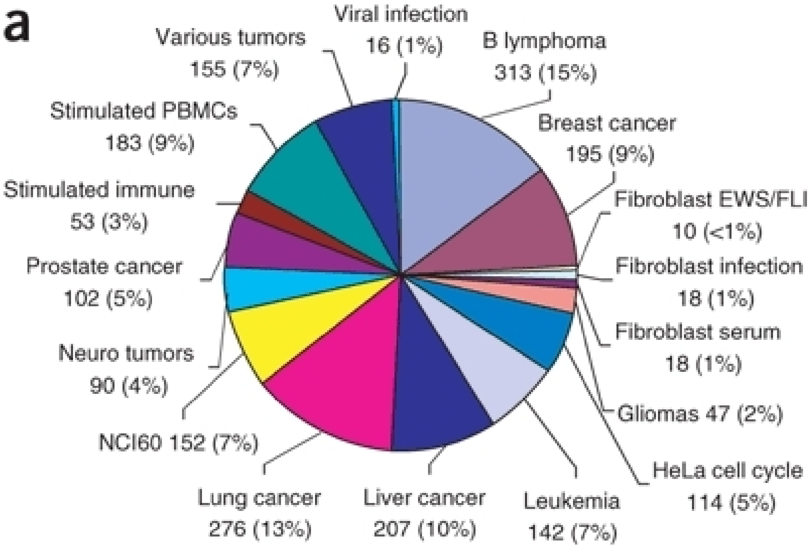
Find significantly enriched array conditions

Modules vs. conditions ④

Key Steps

- Group genes into predefined *gene sets*, e.g., groups of genes with the same functional annotation.
- Convert gene-by-array matrix into gene-set-by-array matrix.
- Hierarchically cluster gene sets in this matrix.
- Identify "interesting" gene set clusters (nodes) in the tree.
- In each gene set cluster, remove genes not expressed consistently with the cluster.

Gene Expression Data Sets



Data Normalisation

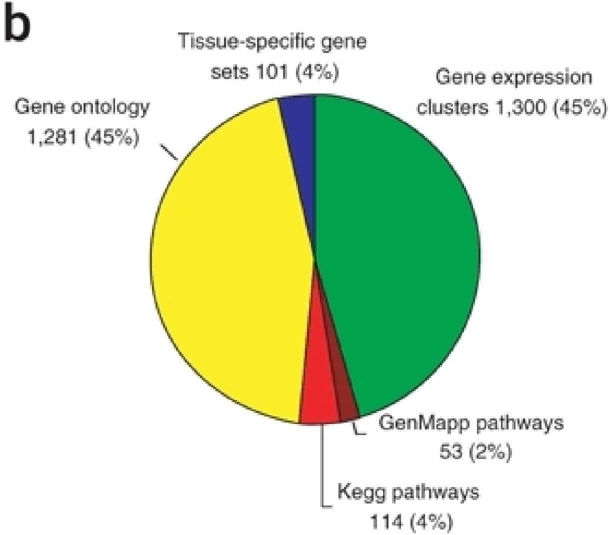
Data Normalisation

- Needed because some arrays measure “absolute” value of gene expression and others measure “relative” values.
- Affymetrix microarrays: take logarithm to the base-2 and zero transform within data set.
- cDNA microarrays:

Data Normalisation

- Needed because some arrays measure “absolute” value of gene expression and others measure “relative” values.
- Affymetrix microarrays: take logarithm to the base-2 and zero transform within data set.
- cDNA microarrays: zero transform within data set.

Pre-defined Genes Sets



Computing Gene-Set-By-Array Matrix

- Goal is to construct a gene-set-by-array matrix.
- For each gene set-array pair, find an “average” expression value of that gene set in that array.

Computing Gene-Set-By-Array Matrix

- Goal is to construct a gene-set-by-array matrix.
- For each gene set-array pair, find an "average" expression value of that gene set in that array.
- A gene is *induced* (respectively, *repressed* in an array if its change in expression is ≥ 2 (respectively, ≤ 2).
- For each gene set-array pair, compute the fraction of genes induced or repressed.
- Use these values in the gene-set-by-array matrix.

Computing Significant Entries in the Gene-Set-By-Array Matrix

- Many entries in the gene-set-by-array matrix may not be statistically significant.

Computing Significant Entries in the Gene-Set-By-Array Matrix

- Many entries in the gene-set-by-array matrix may not be statistically significant.
- For a given array, fraction of induced genes in a gene set may be close to the fraction of induced genes in the array.

Computing Significant Entries in the Gene-Set-By-Array Matrix

- Many entries in the gene-set-by-array matrix may not be statistically significant.
- For a given array, fraction of induced genes in a gene set may be close to the fraction of induced genes in the array.
- Statistical test: for a given array, is the fraction of induced genes in a gene set much larger than the fraction of induced genes in the entire array?

Computing Significant Entries in the Gene-Set-By-Array Matrix

- Many entries in the gene-set-by-array matrix may not be statistically significant.
- For a given array, fraction of induced genes in a gene set may be close to the fraction of induced genes in the array.
- Statistical test: for a given array, is the fraction of induced genes in a gene set much larger than the fraction of induced genes in the entire array?
- Compute the p-value (statistical significance) of the fraction. Exercise.
- Do so for every gene-set-array pair.
- Use false discovery rate correction to account for multiple hypotheses testing.
- Replace insignificant entries by 0.

Hierarchical Clustering

- Start from a gene-set-by-array matrix containing fraction of induced/repressed genes. Fraction is negative if repressed.
- Apply bottom-up hierarchical clustering.
- Vector at internal node is

Hierarchical Clustering

- Start from a gene-set-by-array matrix containing fraction of induced/repressed genes. Fraction is negative if repressed.
- Apply bottom-up hierarchical clustering.
- Vector at internal node is average of vectors at descendant leaves.

Hierarchical Clustering

- Start from a gene-set-by-array matrix containing fraction of induced/repressed genes. Fraction is negative if repressed.
- Apply bottom-up hierarchical clustering.
- Vector at internal node is average of vectors at descendant leaves.
- Which nodes do we select as clusters in the tree?

Hierarchical Clustering

- Start from a gene-set-by-array matrix containing fraction of induced/repressed genes. Fraction is negative if repressed.
- Apply bottom-up hierarchical clustering.
- Vector at internal node is average of vectors at descendant leaves.
- Which nodes do we select as clusters in the tree?
  - Associate each interior node with Pearson correlation between the two children.
  - Cluster $\equiv$ node whose Pearson correlation differs by more than 0.05 from the Pearson correlation of its parent.

Turning Clusters into Modules

- Each cluster is a gene set and a set of arrays.

Turning Clusters into Modules

- Each cluster is a gene set and a set of arrays.
  - The gene set in a cluster is the union of descendant gene sets (leaves).
  - The arrays in a cluster are only those that are induced or repressed in the cluster.
- Module $\equiv$ Cluster minus genes whose expression is not consistent with the rest of the cluster.

Testing Consistency of a Gene with a Gene Set

- Let g be the gene and G be the gene set.

Testing Consistency of a Gene with a Gene Set

- Let g be the gene and G be the gene set.
- Let I (respectively, R) be the set of arrays in which G is significantly induced (respectively, repressed).
- For an array a in I (or R), let p_a be the fraction of genes that are induced (or repressed) by two-fold or more in a.

Testing Consistency of a Gene with a Gene Set

- Let g be the gene and G be the gene set.
- Let I (respectively, R) be the set of arrays in which G is significantly induced (respectively, repressed).
- For an array a in I (or R), let p_a be the fraction of genes that are induced (or repressed) by two-fold or more in a.
- Measure extent to which g's expression changed by more (or less) than two-fold in the arrays in I (or R):

Testing Consistency of a Gene with a Gene Set

- Let g be the gene and G be the gene set.
- Let I (respectively, R) be the set of arrays in which G is significantly induced (respectively, repressed).
- For an array a in I (or R), let p_a be the fraction of genes that are induced (or repressed) by two-fold or more in a.
- Measure extent to which g's expression changed by more (or less) than two-fold in the arrays in I (or R):

$$\text{Score}(g) = \sum_{a \in I | g \text{ is induced in } a} -\log(p_a) + \sum_{a \in R | g \text{ is repressed in } a} -\log(p_a)$$

Testing Consistency of a Gene with a Gene Set

- Let g be the gene and G be the gene set.
- Let I (respectively, R) be the set of arrays in which G is significantly induced (respectively, repressed).
- For an array a in I (or R), let p_a be the fraction of genes that are induced (or repressed) by two-fold or more in a.
- Measure extent to which g's expression changed by more (or less) than two-fold in the arrays in I (or R):

$$\text{Score}(g) = \sum_{a \in I | g \text{ is induced in } a} -\log(p_a) + \sum_{a \in R | g \text{ is repressed in } a} -\log(p_a)$$

- An array contributes to the score only if g is consistent with G in the array.
- Larger contribution from arrays with fewer induced genes.
- Compute statistical significance of this score.

Computing Statistical Significance of Score(g)

$$\text{Score}(g) = \sum_{a \in I | g \text{ is induced in } a} -\log(p_a) + \sum_{a \in R | g \text{ is repressed in } a} -\log(p_a)$$

Computing Statistical Significance of Score(g)

$$\text{Score}(g) = \sum_{a \in I | g \text{ is induced in } a} -\log(p_a) + \sum_{a \in R | g \text{ is repressed in } a} -\log(p_a)$$

- Null hypothesis: genes in each array are randomly permuted, i.e., the p_a induced genes in an array $a \in I$ are chosen randomly.

Computing Statistical Significance of Score(g)

$$\text{Score}(g) = \sum_{a \in I | g \text{ is induced in } a} -\log(p_a) + \sum_{a \in R | g \text{ is repressed in } a} -\log(p_a)$$

- Null hypothesis: genes in each array are randomly permuted, i.e., the p_a induced genes in an array $a \in I$ are chosen randomly.
- Each element in Score(g) is an independent binary random variable.
- Random variable takes the value $-\log(p_a)$ with probability p_a and the value 0 with the probability $1 - p_a$.

Computing Statistical Significance of Score(g)

$$\text{Score}(g) = \sum_{a \in I | g \text{ is induced in } a} -\log(p_a) + \sum_{a \in R | g \text{ is repressed in } a} -\log(p_a)$$

- Null hypothesis: genes in each array are randomly permuted, i.e., the p_a induced genes in an array $a \in I$ are chosen randomly.
- Each element in Score(g) is an independent binary random variable.
- Random variable takes the value $-\log(p_a)$ with probability p_a and the value 0 with the probability $1 - p_a$.
- Under the null hypothesis, Score(g) has mean $\sum_{a \in I \cup R} -p_a \log p_a$ and variance $\sum_{a \in I \cup R} p_a(1 - p_a)\log^2 p_a$.

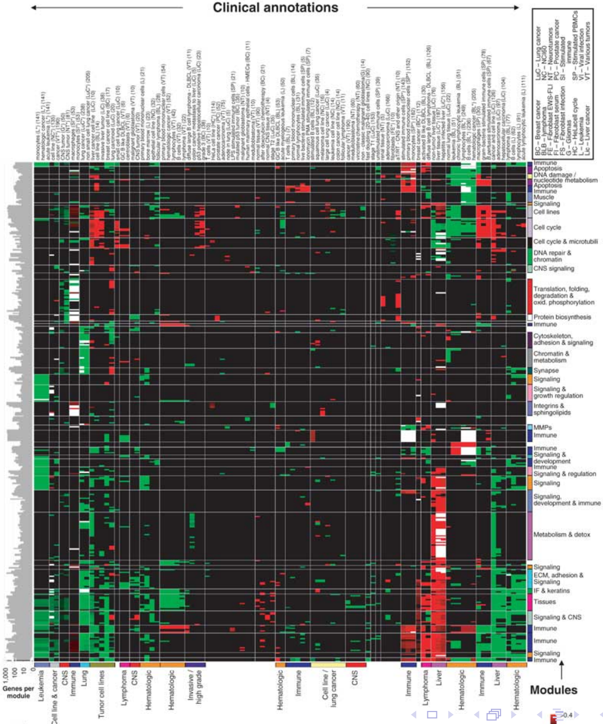
Computing Statistical Significance of Score(g)

$$\text{Score}(g) = \sum_{a \in I | g \text{ is induced in } a} - \log(p_a) + \sum_{a \in R | g \text{ is repressed in } a} - \log(p_a)$$

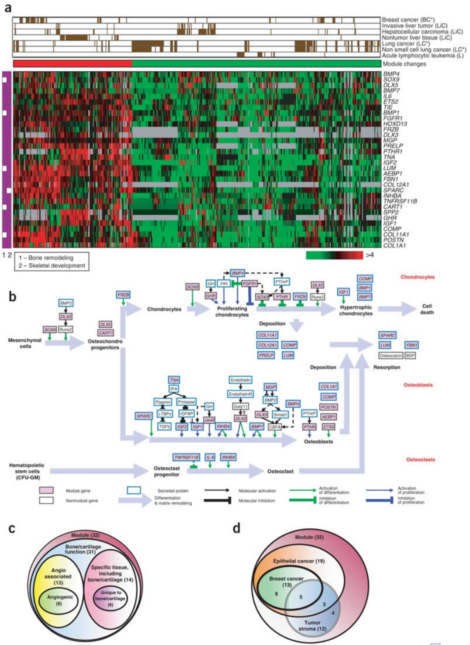
- Null hypothesis: genes in each array are randomly permuted, i.e., the p_a induced genes in an array $a \in I$ are chosen randomly.
- Each element in Score(g) is an independent binary random variable.
- Random variable takes the value $-\log(p_a)$ with probability p_a and the value 0 with the probability $1 - p_a$.
- Under the null hypothesis, Score(g) has mean $\sum_{a \in I \cup R} -p_a \log p_a$ and variance $\sum_{a \in I \cup R} p_a(1 - p_a) \log^2 p_a$.
- Suppose we observe a score of t. What is the probability of achieving a score of t or higher under the null hypothesis? Exercise.

Further Analysis

- Statistical significance of computed modules using leave-one-out cross validation (read supplement).
- Compute enrichment of clinical annotations of the arrays in a module.
- Visualisation of modules.
- Literature-based analysis of modules


Clinical annotations
BC – Breast cancer
BLB – lymphoma
FE – Fibroblast EWS-FLI
FI – Fibroblast serum
FS – Fibroblast infection
G – Gliomas
HC – HeLacell cycle
L – Leukemia
Lic – Liver cancer
LuC – Lung cancer
NC – NCI60
NT – Neurotumors
PC – Prostate cancer
SI – Stimulated immune
SP – Stimulated PBMCs
VI – Viral infection
VT – Various tumors
Immune
Apoptosis
DNA damage / nucleotide metabolism
Apoptosis
Immune
Muscle
Signaling
Cell lines
Cell cycle
Cell cycle & microtubili
DNA repair & chromatin
CNS signaling
Translation, folding, degradation & oxid. phosphorylation
Protein biosynthesis
Immune
Cytoskeleton, adhesion & signaling
Chromatin & metabolism
Synapse
Signaling
Signaling & growth regulation
Integrins & sphingolipids
MMPs
Immune
Immune
Signaling & development
Immune
Signaling & regulation
Signaling
Signaling, development & immune
Metabolism & detox
Signaling
ECM, adhesion & Signaling
IF & keratins
Tissues
Signaling & CNS
Immune
Immune
Signaling
Immune
Modules
1,000
100
10
0
Genes per module
Leukemia
Cell line & cancer
CNS
Immune
Lung
Tumor cell lines
Lymphoma
CNS
Hematologic
Hematologic
Invasive / high grade
Hematologic
Immune
Cell line / lung cancer
CNS
Immune
Lymphoma
Liver
Hematologic
Immune
Liver
Hematologic
0.4

Bone Osteoblastic Module



Conclusions

- Used pre-defined gene sets to drive hierarchical clustering algorithm.
- Remove genes from a cluster of gene sets if the gene's expression profile deviates from the cluster.
- Automatically decide which arrays are part of a module.
- Natural segue into lectures on biclustering where we will automatically decide which arrays *and* which genes to include in a bicluster.

Software Exercise

1. Register for and download Genomica.
2. Use Genomica to compute a module map for the sample data set.
3. Download human Entrez Gene gene sets and gene expression data.
4. Run Genomica on these data sets.
5. Change parameters and run Genomica again. Are the results different?

Computational Exercises

1. In case of d_{min}, show that the hierarchical clustering algorithm returns the minimum spanning tree.
2. How can we measure the "useful" biological knowledge that a cluster contains?
3. Given an array, the set of genes induced in that array, and a specific gene set, devise a statistical test to determine if the number of induced genes in the gene set is (much) larger than the number of induced genes in the entire array.
4. Under the null hypothesis, Score(g) has mean $\sum_{a \in I \cup R} -p_a \log p_a$ and variance $\sum_{a \in I \cup R} p_a(1-p_a) \log^2 p_a$. Suppose we observe a score of t. What is the probability of achieving a score of t or higher under the null hypothesis?
5. How will you modify Genomica to accept a new dataset without performing all computations from scratch?