Microarray analysis at a glance – from low-level data processing to data analysis

Olga Troyanskaya

Many of the slides about SMD borrowed/modified from Gavin Sherlock et al. (Stanford)

Admin

- Slides, readings, announcements are at:

http://www.cs.princeton.edu/courses/archive/fall03/cs597F/

Sign up for talks (sing-up going around)

Fill out survey (going around)

Microarray analysis at a glance

- **Data Storage & Retrieval**
- Filtering
- Normalization
- Missing value estimation
- Analysis – unsupervised or supervised
- Visualization

Purpose of a microarray DB

Data management

Integration with basic analysis tools

Integration with external information
- consolidation
- data integration

Publication of Results

Example: Stanford Microarray Database (SMD)

- Data management
  - Storage, archiving and data viewing tools.
- Integration with analysis tools and external information.
  - Clustering, partitioning and output of data for other use. Linkage with SGD and GO.
- Publication of results
  - Provide data, images, analysis and connections with biological resources. Linkage with SGD.

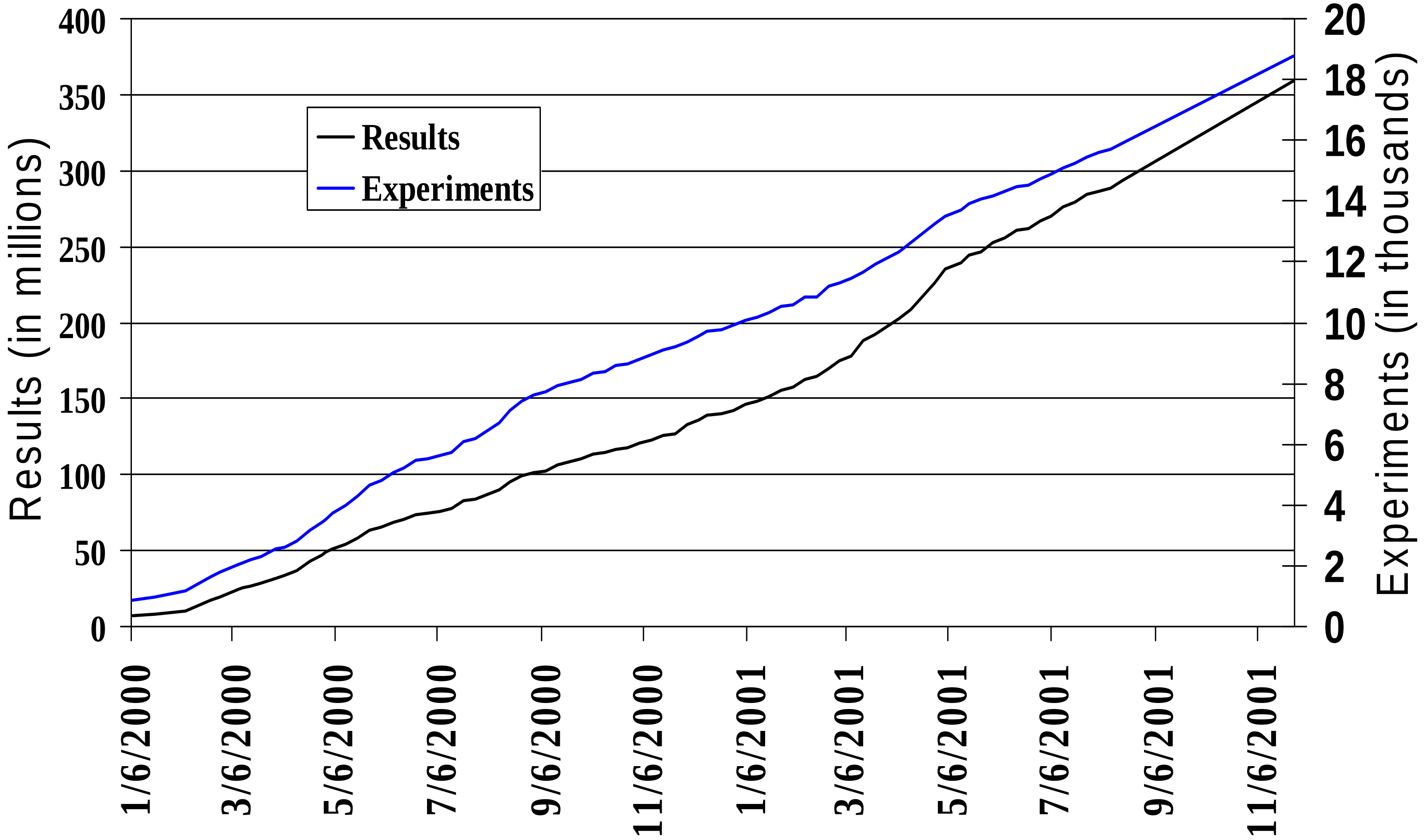
SMD provides:

- Storage of both the raw and normalized data from microarray experiments, as well as their corresponding image files.
- Interfaces for data retrieval, analysis, visualization, and organization.
- A means of associating meaningful information, both biological and methodological, with the experiment. This includes annotation of the arrayed samples, the probe(s), the materials and methods, and the experimental context (groupings).

Scale of the problem by the end of 2001

- 500 slides (experiments) per week
- >40,000 spots per slide
- 1 billion spots/year!
- Uncertain number of organisms to be included.
- 750 GB in TIFF images per year, and growing

Growth of SMD



As of November 27, 2001

SMD Built from Components

- Oracle DBMS
- Web interface via Perl CGI and DBI
- TIFFs and primary data archived to tape and Magneto-optical disks
- GIF pseudocolor images stored outside DBMS
- Microarray data stored in 24 core tables
- External datasets currently in 34 tables

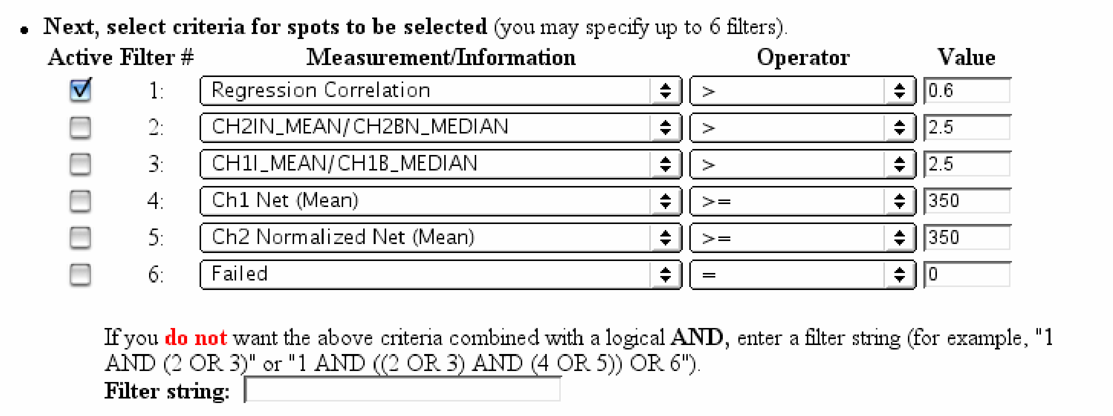
Design challenges, an example

- Need to consider at least two levels of identifier:
  - Physical DNA (SUID) - should track with sequence, though sequence is not always known
  - Genetic Entity to which DNA maps (LocusID)
    - can dynamically change => need regular communication with NIH databases for updating
    - requires that SUID can be easily mapped to the LocusID
- Access issues

Microarray analysis at a glance

- Data Storage & Retrieval
- **Filtering**
- Normalization
- Missing value estimation
- Analysis – unsupervised or supervised
- Visualization

Data Filtering



- **Next, select criteria for spots to be selected** (you may specify up to 6 filters).

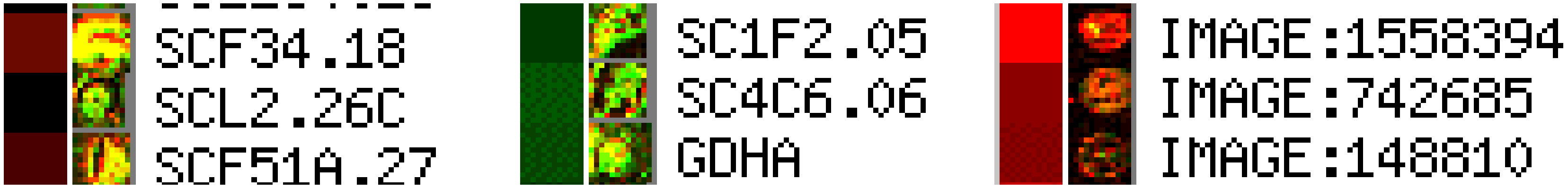
Active Filter #		Measurement/Information	Operator	Value
☑	1:	Regression Correlation	>	0.6
□	2:	CH2IN_MEAN/CH2BN_MEDIAN	>	2.5
□	3:	CH1I_MEAN/CH1B_MEDIAN	>	2.5
□	4:	Ch1 Net (Mean)	>=	350
□	5:	Ch2 Normalized Net (Mean)	>=	350
□	6:	Failed	=	0

If you **do not** want the above criteria combined with a logical **AND**, enter a filter string (for example, "1 AND (2 OR 3)" or "1 AND ((2 OR 3) AND (4 OR 5)) OR 6").

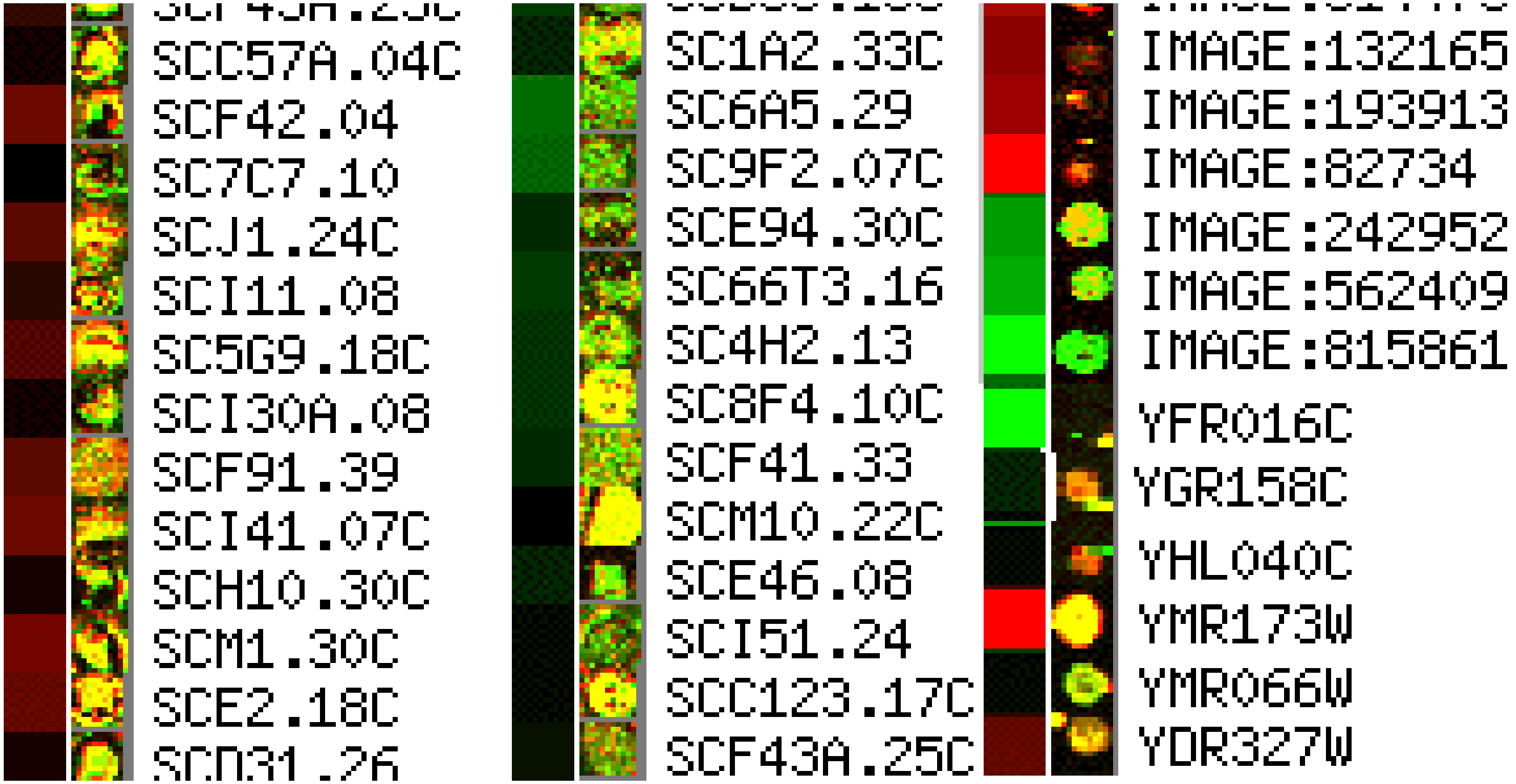
Filter string:

- Goals:
  - Extract only experiment/gene subsets of interests
  - Extract only “accurate” data points
- Various filtering criteria:
  - Manual
  - Fluorescence distribution
  - Level of expression in each channel
- Filters can be combined using logical operators

Why worry?
Spots with low regression correlation



Challenge – How can we differentiate between data and noise on image level?



Microarray analysis at a glance

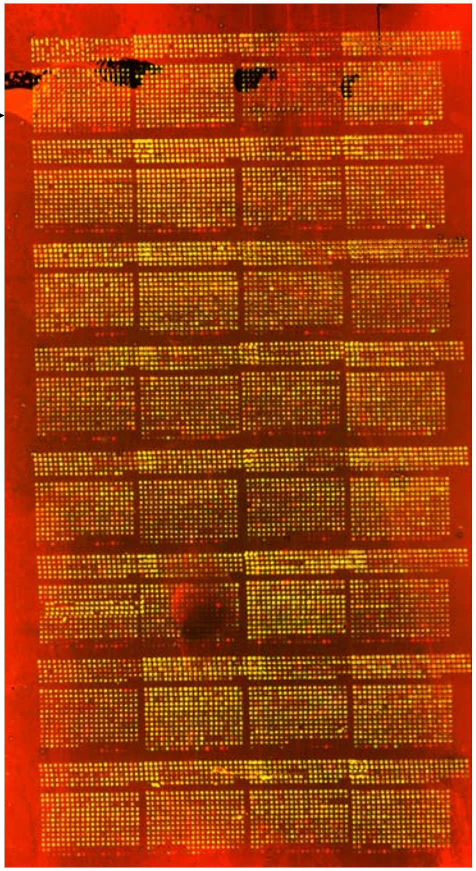
- Data Storage & Retrieval
- Filtering
- **Normalization**
- Missing value estimation
- Analysis – unsupervised or supervised
- Visualization

Data Normalization: Definition

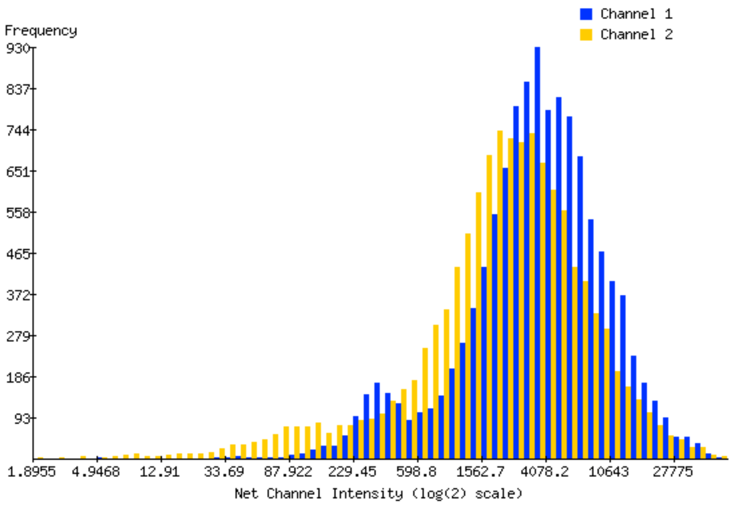
- Normalization is an attempt to compensate for systematic bias in data
- Normalization attempts to remove the impact of non-biological influences on biological data:
  - Balance fluorescent intensities of the two dyes
  - Adjust for differences in experimental conditions (b/w replicate gene expression experiments)
- Normalization allows to compare data from one experiment to another (after removing experiment-specific biases)

Normalization: Sources of Systematic Bias

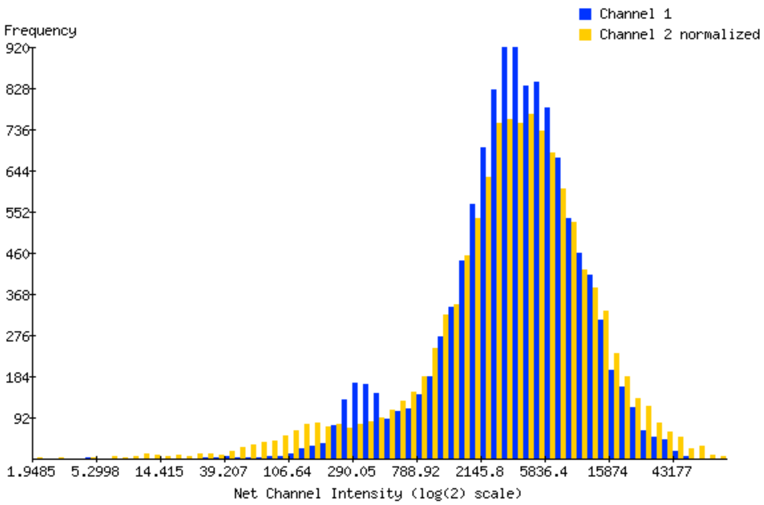
- Different labeling efficiencies or dye effects (two-channel arrays)
- Scanner malfunction
- Differences in concentration of DNA on arrays (plate effects)
- Printing or tip problems
- Uneven hybridization
- Batch bias
- Experimenter issues



Normalization: Effects on Intensity



Non-normalized



Normalized

Same mRNA hybridized in both channels

Microarray analysis at a glance

- Data Storage & Retrieval
- Filtering
- Normalization
- **Missing value estimation – next class**
- Analysis – unsupervised or supervised
- Visualization

Microarray analysis at a glance

- Data Storage & Retrieval
- Filtering
- Normalization
- Missing value estimation – next class
- **Analysis – unsupervised or supervised**
- Visualization

Clustering in gene expression world – the basics

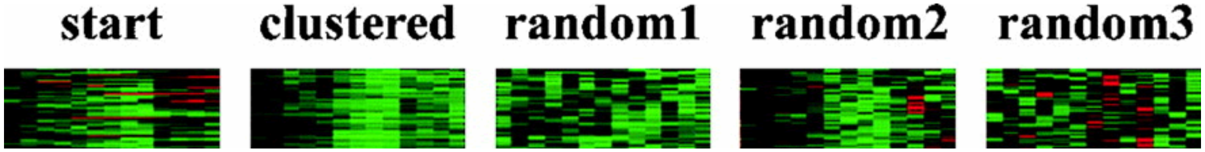
Why cluster?

- “Guilt by association” => if unknown gene ***i*** is similar in expression to known gene ***j***, maybe they are involved in the same/related pathway
- Dimensionality reduction: datasets are too big to be able to get information out without reorganizing the data

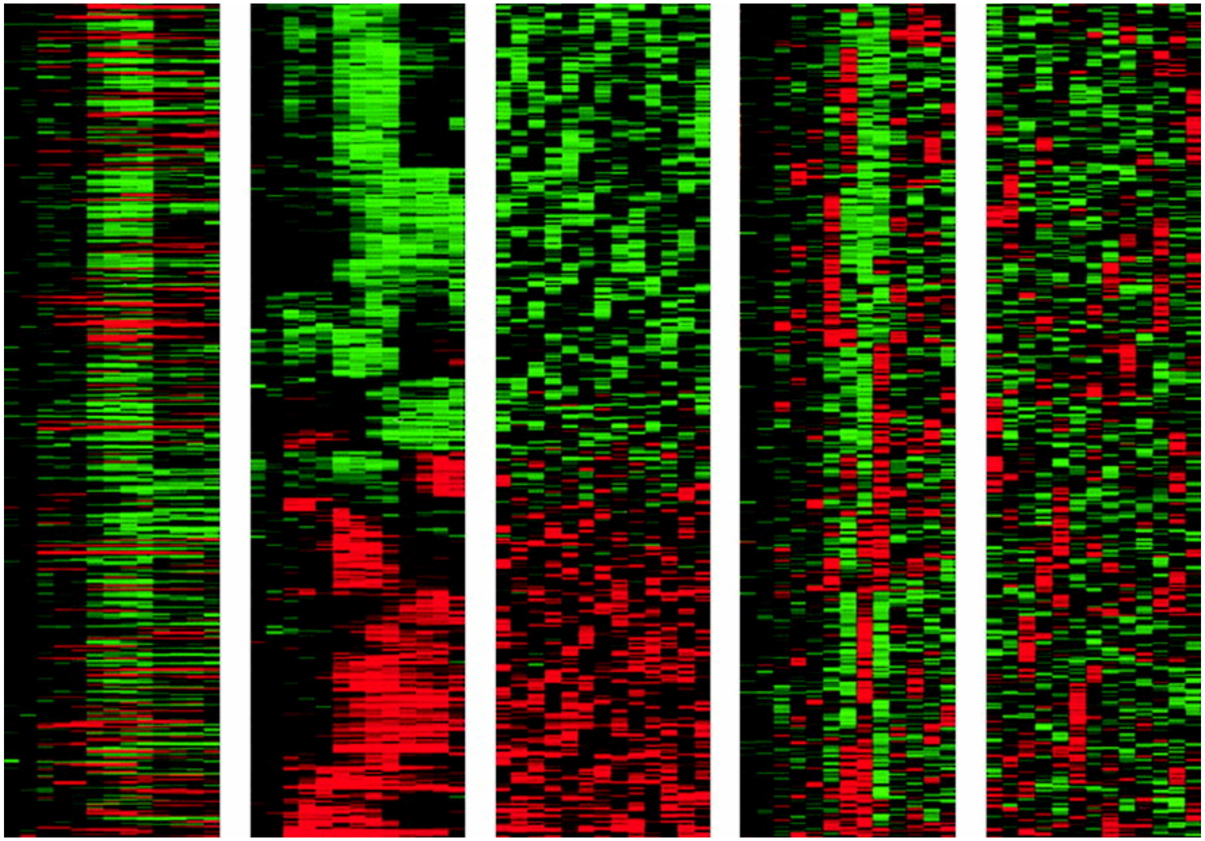
What is clustering?

- Reordering of gene (or experiment) expression vectors in the dataset so that similar patterns are next to each other (or in separate groups)

Clustering Random vs Biological Data



Challenge – when is clustering “real”?



From *Eisen MB, et al, PNAS 1998 95(25):14863-8*

K-means clustering

- 1. Define k = number of clusters
- 2. Randomly initialize a seed vector for each cluster
- 3. Go through all genes, and assign each gene to the cluster which it is most similar to
- 4. Recalculate all seed vectors as means (or medians) of patterns of each cluster
- 5. Repeat 3&4 until <stop condition>

K-means clustering: stop conditions

- Until the change in seed vectors is < <constant>
- Until all genes get assigned to the same partition twice in a row
- Until some minimal number of genes (e.g. 90%) get assigned to the same partition twice in a row

K-means: problems

- Have to set ***k*** ahead of time
- Each gene only belongs to 1 cluster
- One cluster has no influence on the others (one dimensional clustering)
- Genes assigned to clusters on the basis of all experiments

Defining ***k*** (# of clusters)

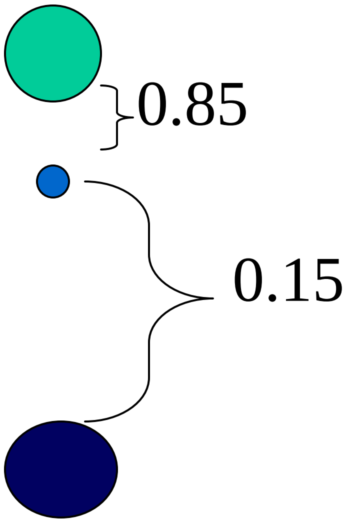
- Gap statistic
  - Find k at which within-cluster variation is min
  - Plot difference between real and random data's within-cluster variation, choose max difference point
- Leave-one out cross-validation
  - quality of clusters higher if less within-cluster variation on the "test" array
- Resampling based methods

Can a gene belong to N clusters?

- Fuzzy clustering: each gene’s relationship to a cluster is probabilistic
- Gene can belong to many clusters
- More biologically realistic,

but harder to get to work well/fast

- Harder to interpret

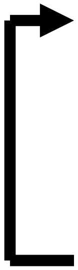


Self Organizing Maps (SOM)

- Similar to k-means
- BUT: allow clusters to influence each other

Self-organizing maps algorithm

- 1. Partition data (e.g. 3x2 grid)
- 2. Randomly choose “seed” vectors for each partition (length = # experiments)

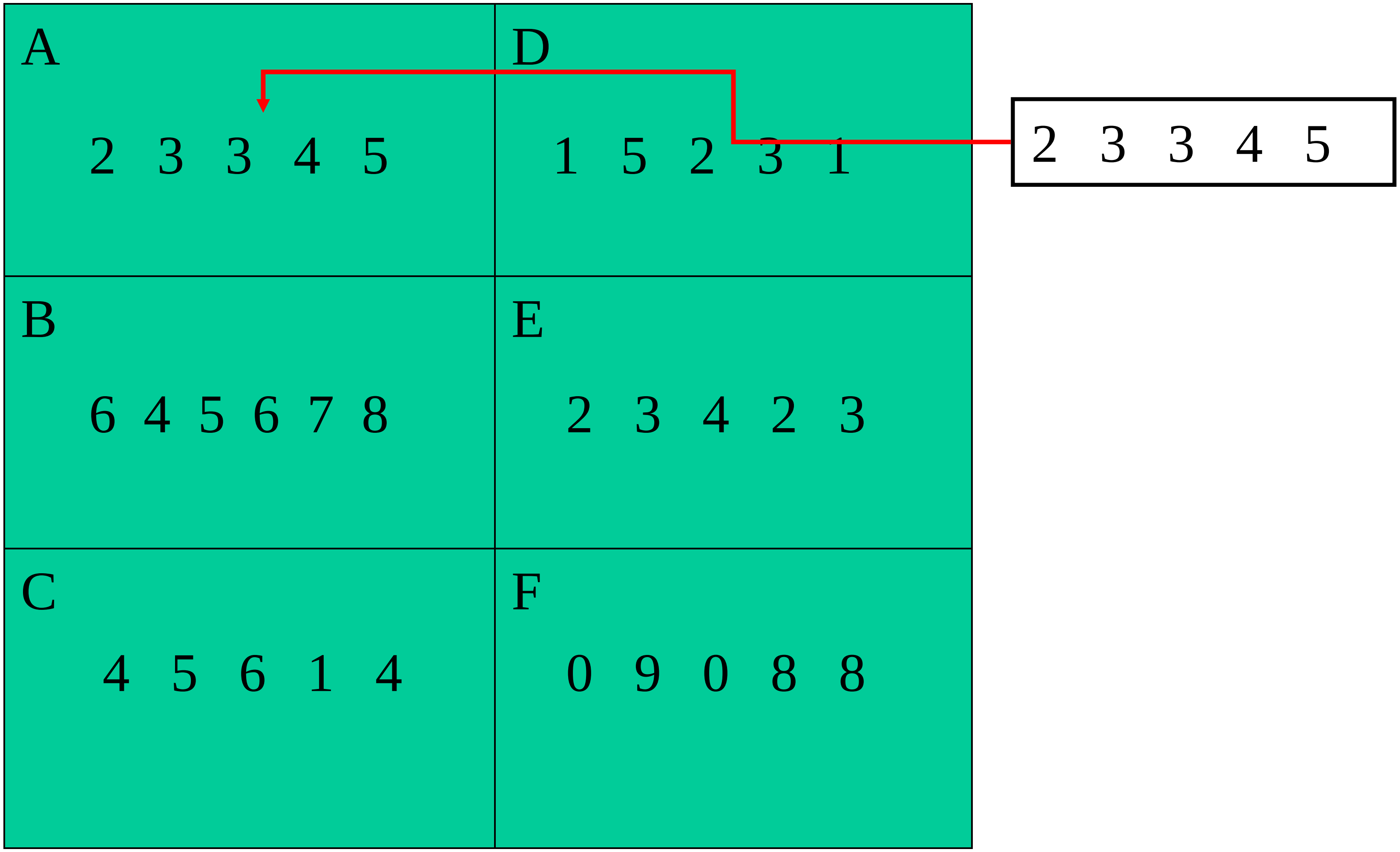


- 3. Pick a gene at random (e.g. gene ***i***, see which partition it is most similar to (e.g. partition A), and modify A’s seed vector to be more similar to gene ***i***
- 4. Now modify neighboring partitions of A to be more similar to A
- 5. After map “settles down”, assign each gene to the most similar partition

1. Initialize the seeds for each partition

A 1 2 3 4 5	D 1 5 2 3 1
B 6 4 5 6 7 8	E 2 3 4 2 3
C 4 5 6 1 4	F 0 9 0 8 8

2. Pick a gene at random, and adjust the closest partition



A

2 3 3 4 5

D

1 5 2 3 1

B

6 4 5 6 7 8

E

2 3 4 2 3

C

4 5 6 1 4

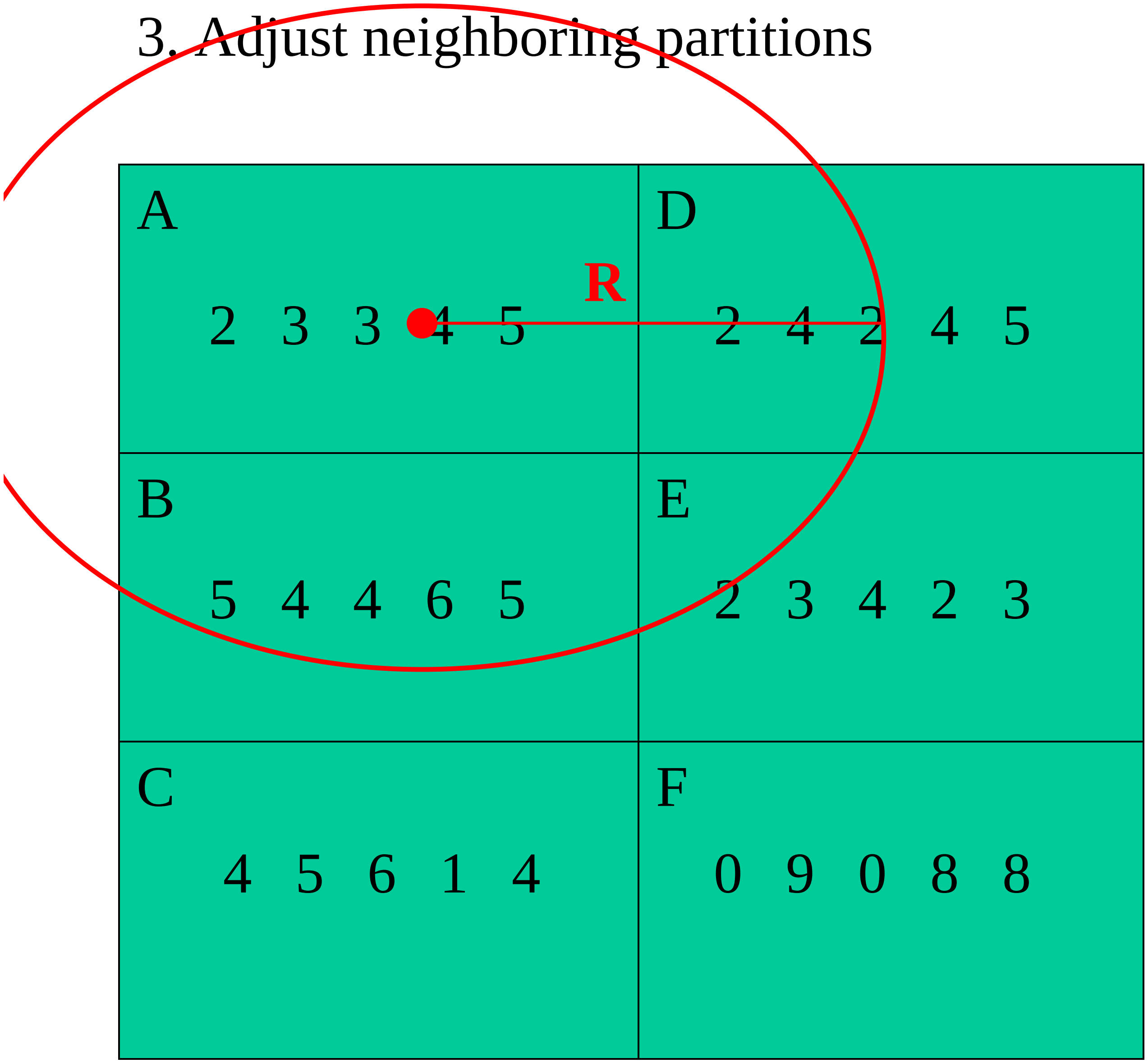
F

0 9 0 8 8

2 3 3 4 5

Iteration 1.

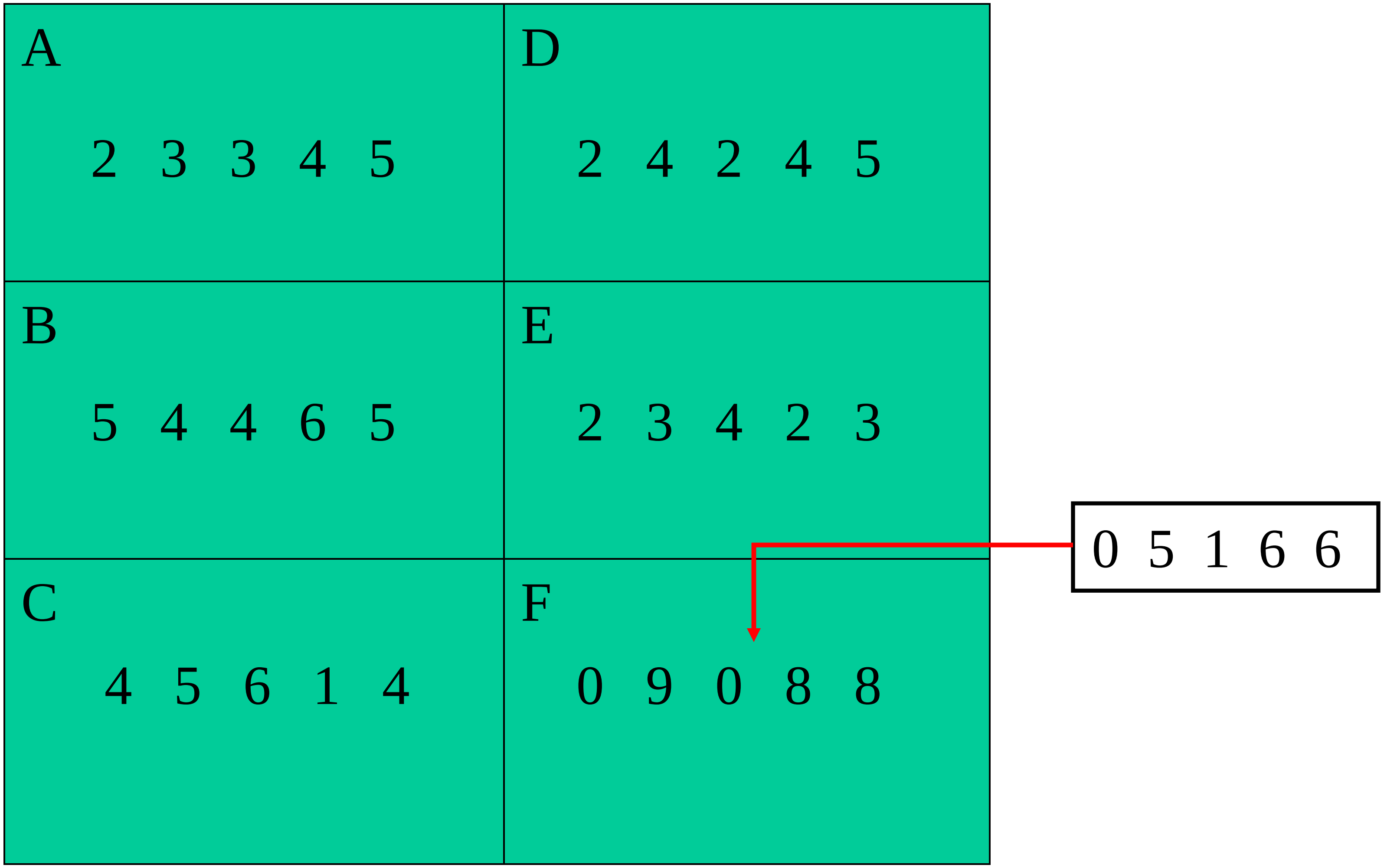
3. Adjust neighboring partitions



A 2 3 3 4 5	D 2 4 2 4 5
B 5 4 4 6 5	E 2 3 4 2 3
C 4 5 6 1 4	F 0 9 0 8 8

Iteration 1.

2. Pick a gene at random, and adjust the closest partition



A 2 3 3 4 5	D 2 4 2 4 5
B 5 4 4 6 5	E 2 3 4 2 3
C 4 5 6 1 4	F 0 9 0 8 8

0 5 1 6 6

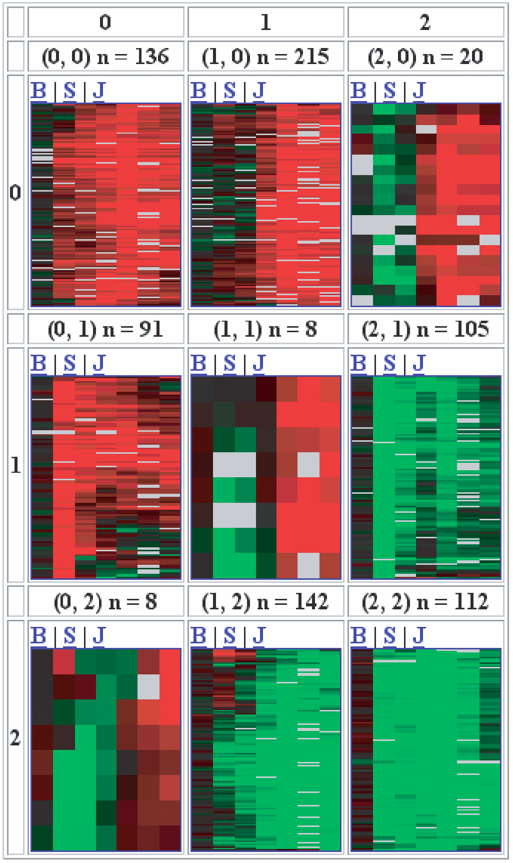
Iteration 2.

Self-organizing maps iterations

- At higher iterations, smaller R
- At higher iterations, smaller change to partition seeds

- => the map “settles down”

Self Organizing Maps: Result



- SOMs result in genes being assigned to partitions of most similar genes
- Neighboring partitions are more similar to each other than they are to distant partitions

SOM: problems

- Have to set ***n*** and ***m*** ahead of time
- Each gene only belongs to 1 cluster
- Genes assigned to clusters on the basis of all experiments

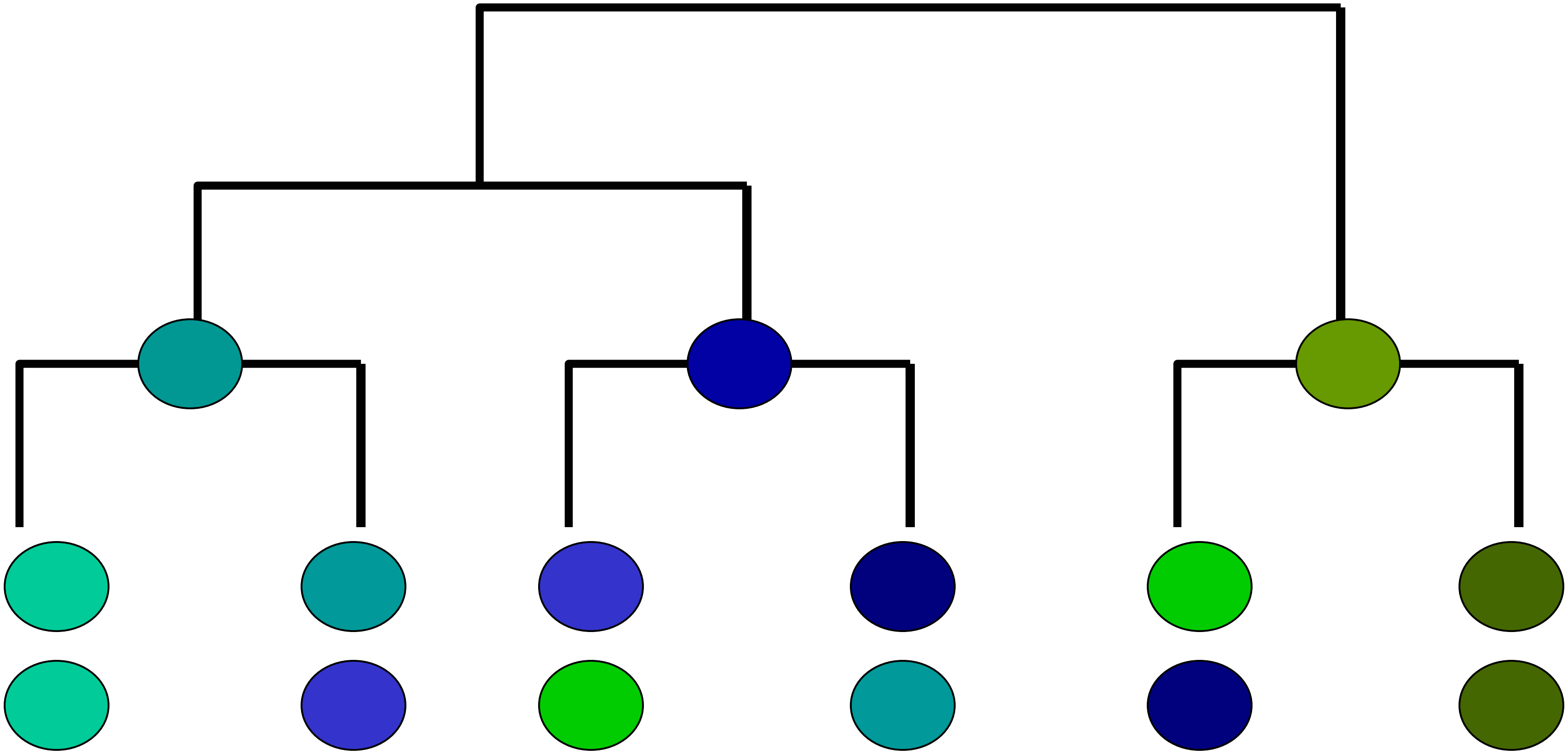
Hierarchical clustering

- Imposes hierarchical structure on all of the data
- Easy visualization of similarities and differences between genes (experiments) and clusters of genes (experiments)

How does Hierarchical Clustering work?

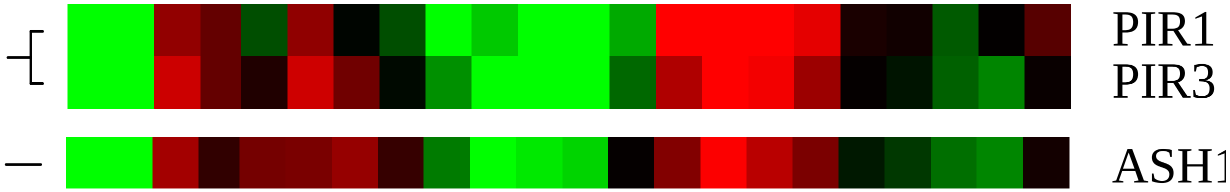
1. Compare all expression patterns to each other.
2. Join patterns that are the most similar out of all patterns.
3. Compare joined patterns to all other un-joined patterns.
4. Go to step 2, and repeat until all patterns are joined.

Hierarchical Clustering



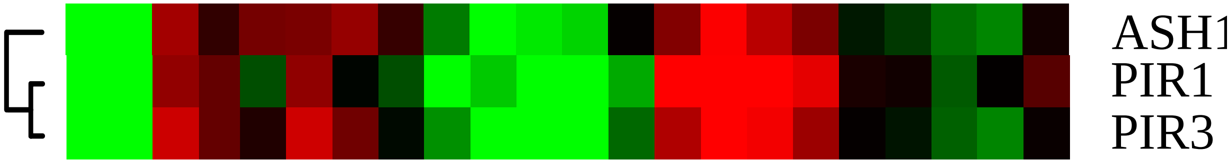
Optimizing node order

- Consider:



- Is Ash1’s expression most similar to Pir1, or Pir2?

- Flip when joining to make most similar patterns adjacent:



Hierarchical clustering: problems

- Hard to define distinct clusters
- Genes assigned to clusters on the basis of all experiments
- Optimizing node ordering hard (finding the optimal solution is NP-hard)
- Can be driven by one strong cluster – a problem for gene expression b/c data in row space is often highly correlated
- Hard to partition into distinct clusters

Choice of distance metric is important

- Treat data for a gene as a vector
- Distance metric important:
  - Linear: Euclidean distance, or Pearson correlation
  - Nonlinear: Spearman…

$$d_{x,y} = \sqrt{\frac{\sum_{j=1}^{n} (x_j - y_j)^2}{n}}$$

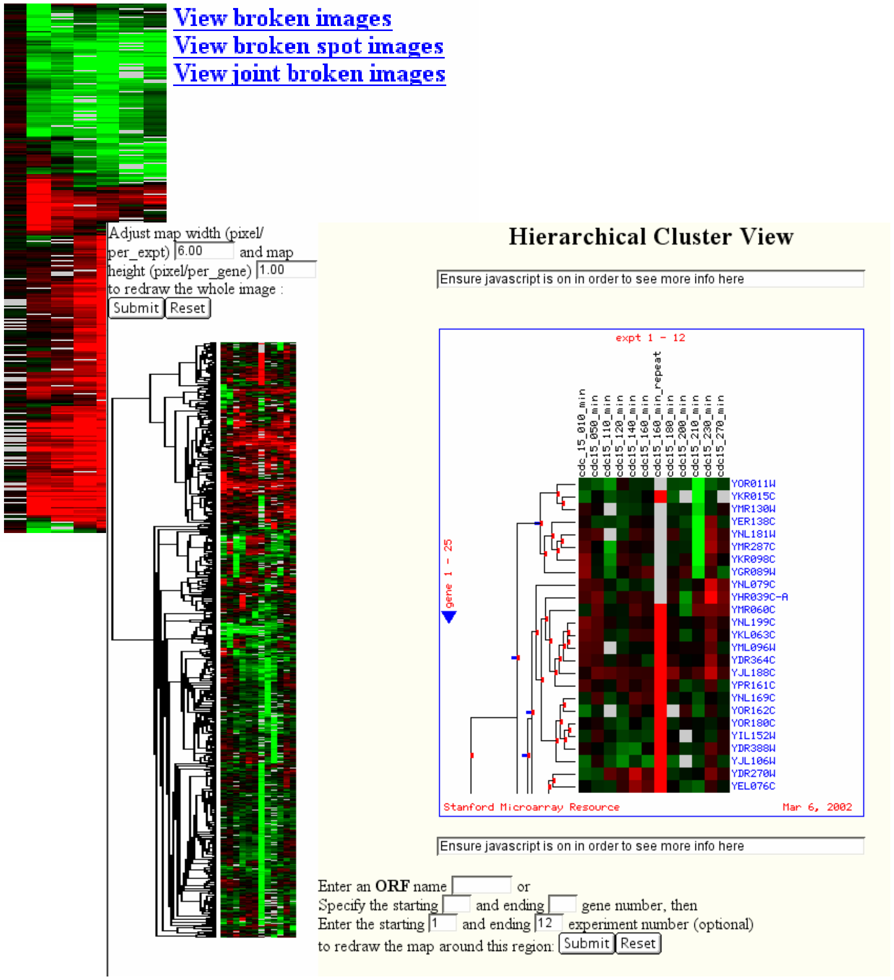
$$d_{x,y} = \frac{1}{n}\sum_{i=1}^{n}\left(\frac{x_i - \bar{x}}{\sigma_x}\right)\left(\frac{y_i - \bar{y}}{\sigma_y}\right)$$

EVALUATION:
Clustering (supervised or unsupervised)

- a new brilliant algorithm is not enough – how does it compare?
- No external standard on real data =>
  - Can use synthetic datasets
  - Beware of assumptions (e.g. normality)
- Internal standards – lots of research in this area!

A difference between a useful bioinformatics advance and a non-relevant publication is most often EVALUATION!

Clustering: Visualization



View broken images
View broken spot images
View joint broken images

Adjust map width (pixel/per_expt) 6.00 and map height (pixel/per_gene) 1.00 to redraw the whole image :
Submit Reset

Hierarchical Cluster View

Ensure javascript is on in order to see more info here

Ensure javascript is on in order to see more info here

Enter an **ORF** name ▭ or
Specify the starting ▭ and ending ▭ gene number, then
Enter the starting 1 and ending 12 experiment number (optional)
to redraw the map around this region: Submit Reset

- **Lots of Visualization and HCI challenges:**
- Lots of data
- Dynamic navigation
- Simultaneous display of different data types
- Simultaneous display of different zoom levels for data
- Dynamic links to other databases

Visualization often critical for late-stage biological analysis!

End of class 2