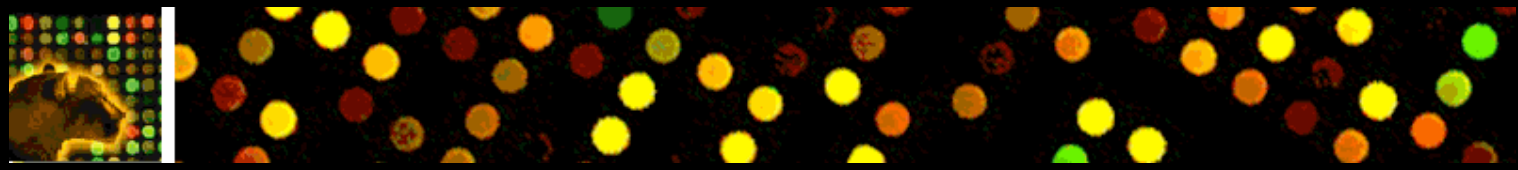
# PUMAdb
# Basic Data Analysis

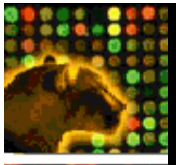


John Matese

October, 2008

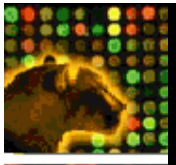
# User Help: Tutorials and Workshops

- Help & FAQ
  - http://puma.princeton.edu/help/
  - http://puma.princeton.edu/help/FAQ.shtml
- Tutorials
  - http://puma.princeton.edu/help/tutorials_subpage.shtml
  - Ideas? Email array@genomics.princeton.edu
- Hybridization & Scanning Individual Instruction
  - Email dstorton@molbio.princeton.edu



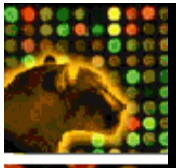
# PUMAdb Data Analysis

- Concepts of data manipulation
  - Data normalization
  - Data filtering
  - Data centering
  - Data clustering
- Using the Database's Analysis Pipeline
  - Gene Selection and Annotation
  - Data Filtering
  - Data Retrieval
  - Gene Filtering
  - Clustering and Image Generation
- Other Things You Should Know…
  - Repository
    - Synthetic Genes
  - Java TreeView
  - GO Term Finder



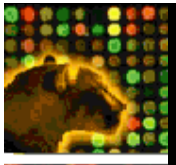
# Concepts of data manipulation

- Data normalization
- Data filtering
- Data centering
- Data clustering



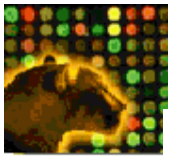
# Concepts of Data Manipulation

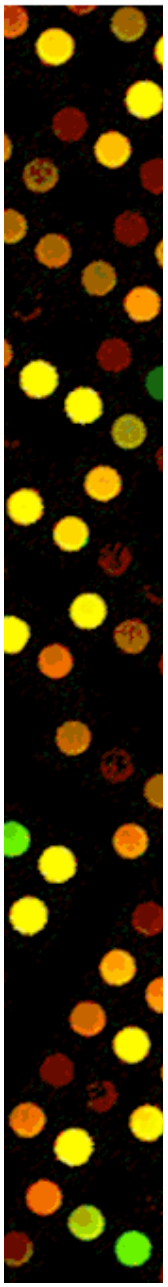
- Data normalization
  - Transforms data for cross-array comparison, by eliminating or compensating for some biases.
- Data filtering
  - Removes unreliable or uninteresting data.
- Data centering
  - Transforms data for within-vector comparisons (gene or array).
- Data clustering
  - Identify and reveal patterns within the data.



# What is Normalization?

- Normalization is an attempt to correct for systematic bias in data.
- Normalization allows you to compare data from one array to another.





Pool of Cell Lines

Tumor

Different amounts of starting material.

mRNA

Different amounts of RNA in each channel

cDNA

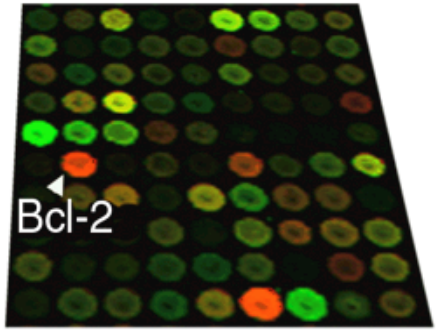
Bcl-2

Differential labeling efficiency of dyes

Differential efficiency of hybridization over slide surface.

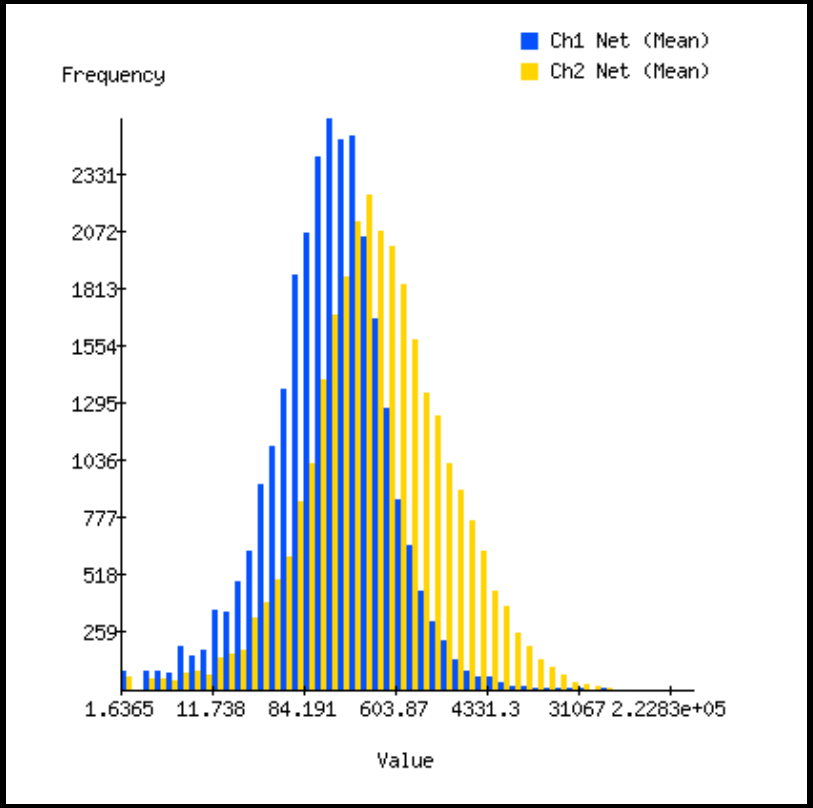
DNA microarray

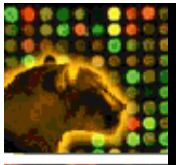
Differential efficiency of scanning in each channel.



# Consequences of Biases

Plotting the frequency of raw (pre-normalized) intensities reveals these differential effects between the two channels.





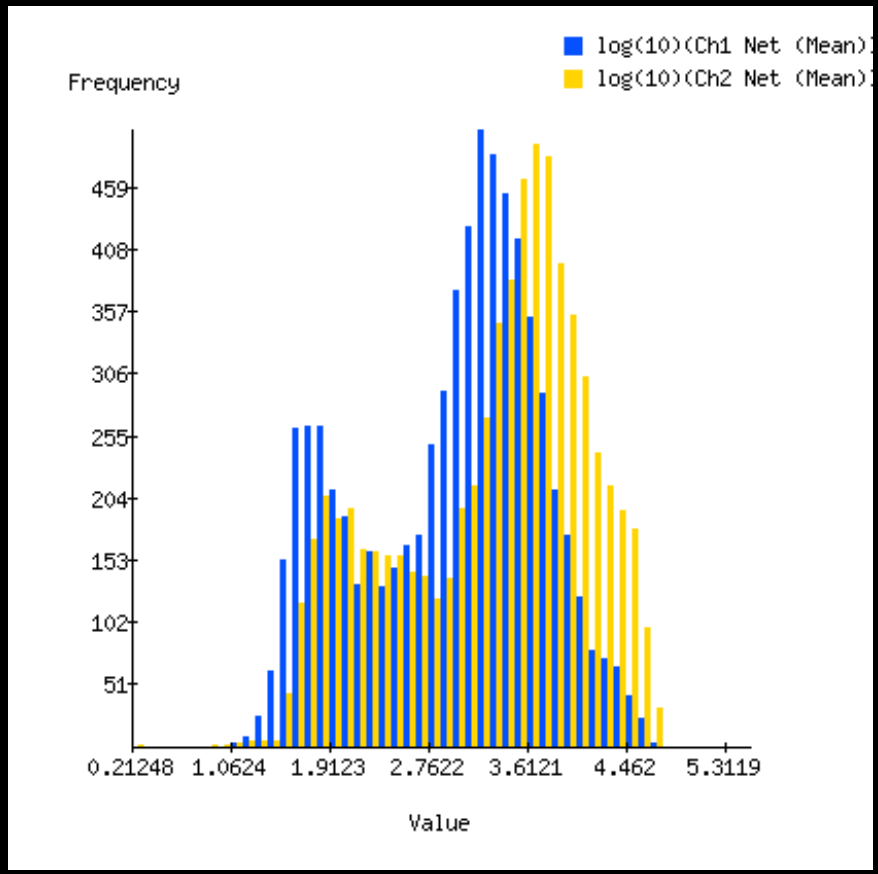
# How do we deal with this?

Normalization:

- *In general*, an assumption is made that the average gene does not change.
- For some experimental designs, this may not be an appropriate assumption…
- The number of ‘reporters’ (clones or genes) you are assaying will affect this.

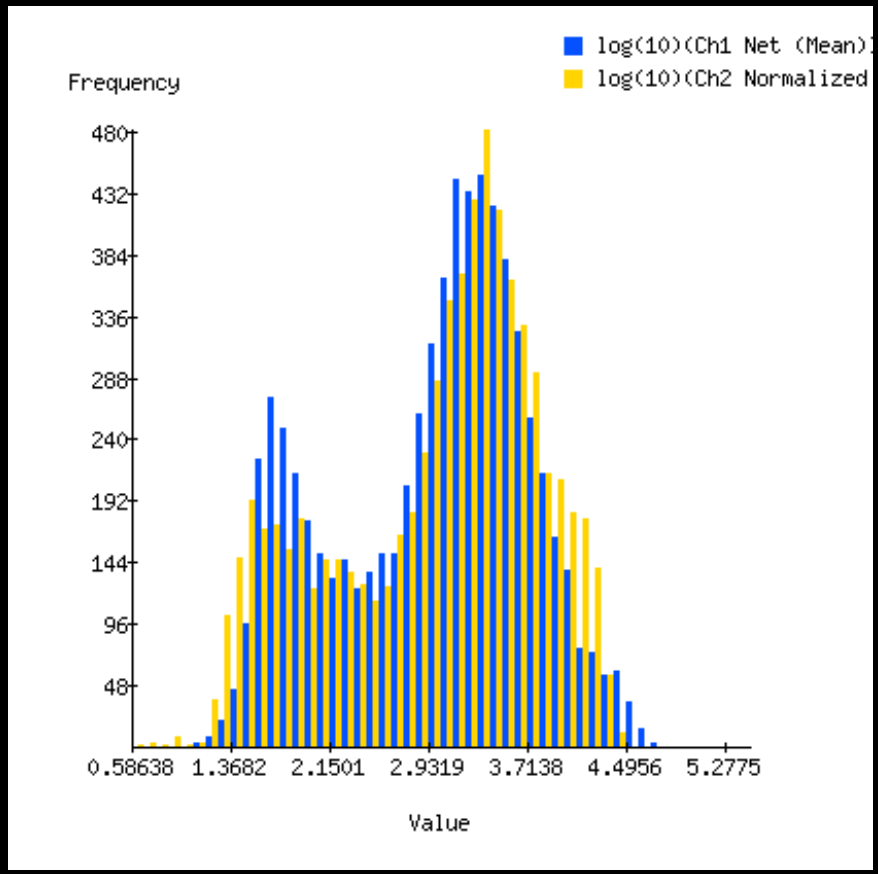
# Normalization: Channel biases

**Before Normalization…**



# Normalization: Channel biases

**After Normalization…**

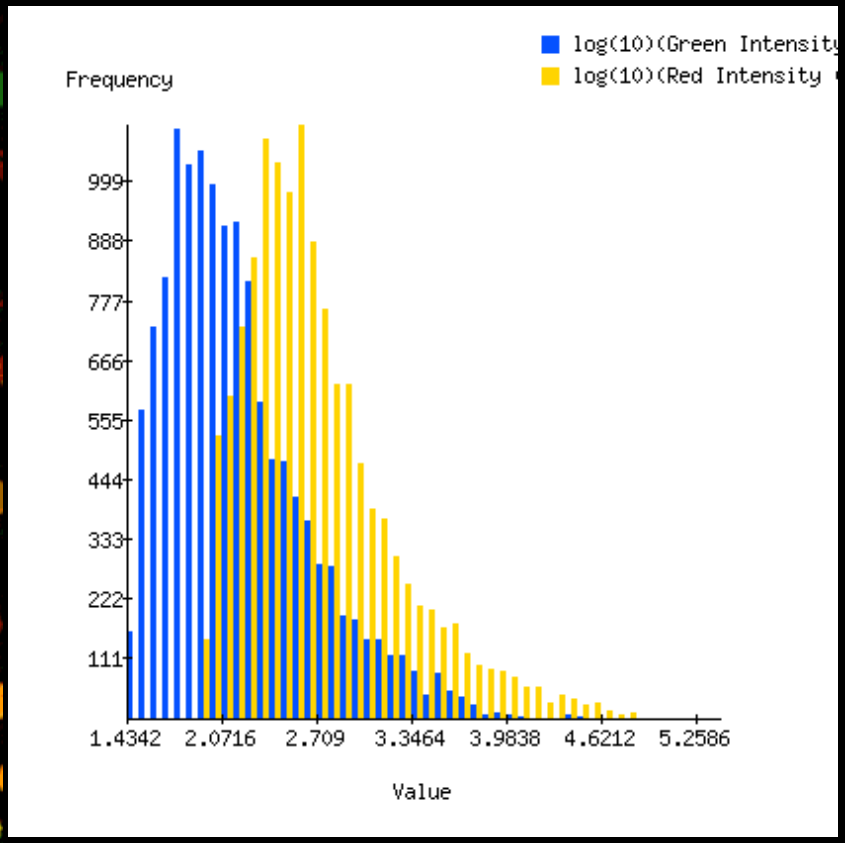


# Total Intensity Normalization

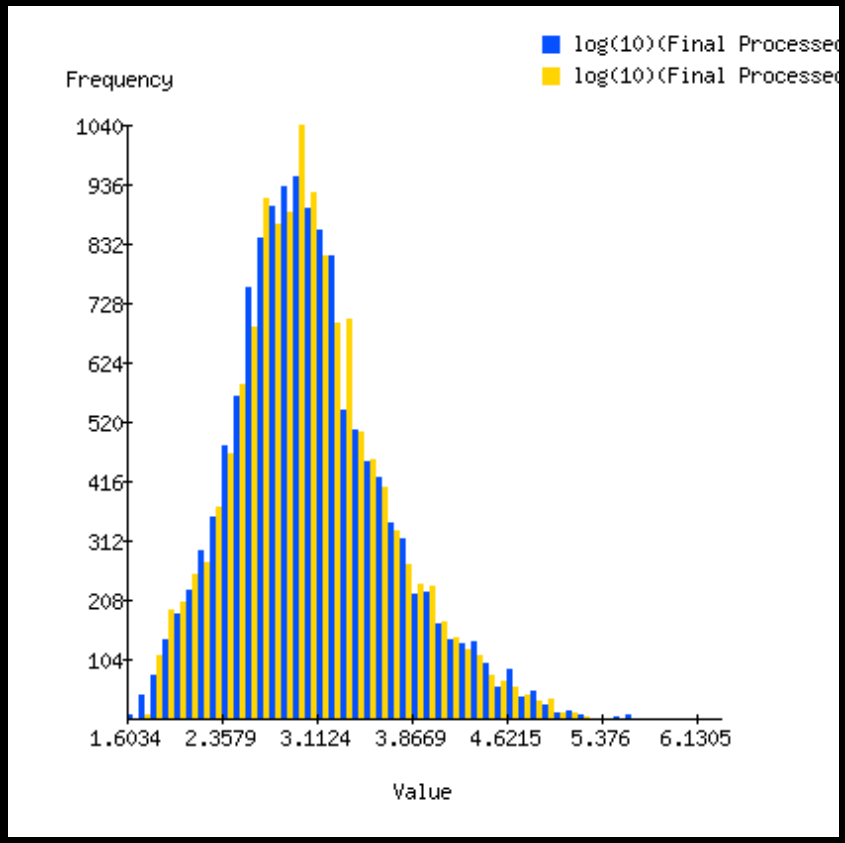
- For those spots that are thought to be well measured, calculate mean or median log ratio.
- Use this as a normalization factor to adjust all log ratios (linear).
- Equivalent to assuming same total intensity in both channels.
- Basic example (our software):
  - two simple methods for selection of well measured spots: pixel-by-pixel regression, and foreground over background intensity.
  - calculates normalized values for all channel 2 measurements, and ratios.

# Normalization: Agilent

**Plotting log(10)(Green Intensity (median)) and log(10)(Red Intensity (median))**




**Plotting log(10)(Final Processed Green Intensity) and log(10)(Final Processed Red Intensity)**



# Normalization: Agilent

The dye-normalized signal is calculated by multiplying the background-subtracted signal by the dye normalization factor:

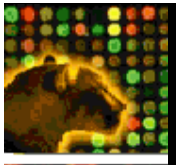
$$DyeNormSignal = BGSubSignal \times DNF$$

where *DNF=LinearDyeNormFactor*, when linear dye normalization method is used

OR

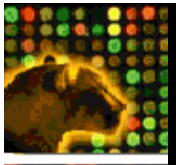
where *DNF=LinearDyeNormFactor* × *LOWESSDyeNormFactor*

when (Linear&)LOWESS dye normalization method is used.



# Concepts of data manipulation

- Data normalization
- Data filtering
- Data centering
- Data clustering

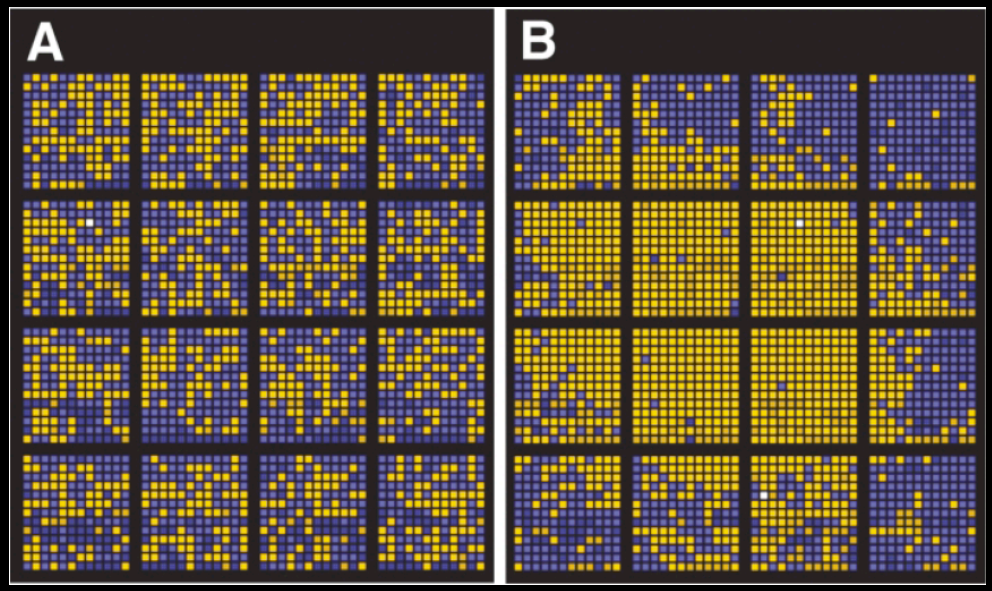


# Signal or Noise?

*Just because you are capable of making 40,000 measurements does NOT mean they all are worthwhile and should be trusted!*

- Artifacts happen (spatial bias)
- Poor array designs
- Each assay has its own sensitivity
- Even well measured spots can potentially be uninteresting in a biological context (i.e. no variability)

# Spatial Bias

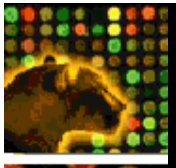



A. Neither spatial bias nor suspected plate bias
B. Strong spatial bias (poor hybridization)

Excerpted from *Gollub J, et al.(2003) Nucleic Acids Research, Vol. 31, No. 1* ***94-96***

# Plate Bias

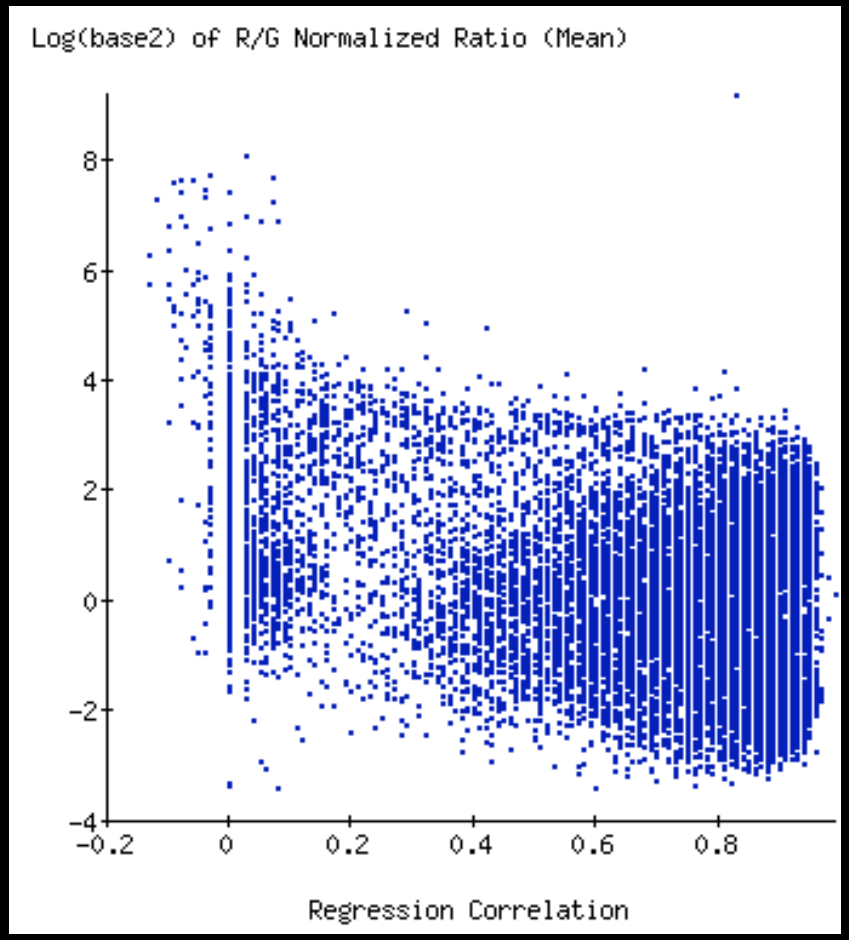
- When the contents of some plates are intrinsically different from the others
  - Plate source
  - Platesample nature
  - Biosequence source
- For example, a multi-organism array design…



# Data Filtering

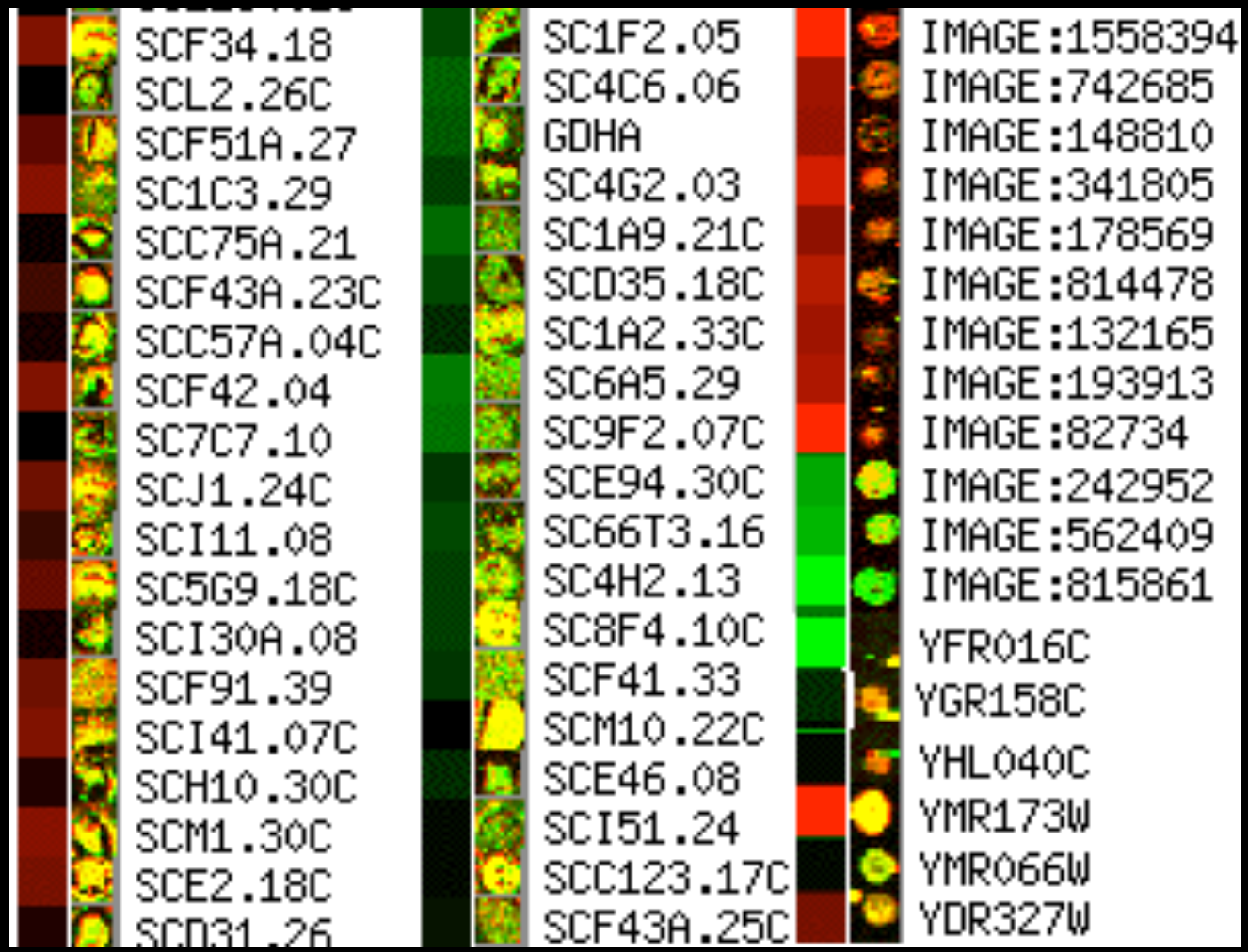
- Data you trust (spots)
  - Signal-to-noise
  - Uniformity within spot (pixel-pixel regression)
  - Reasonable expectations in context of spot population
- Data of interest (“genes”)
  - Meaningful changes in group of assays
  - Patterns

# Data Filtering: Regression Correlation

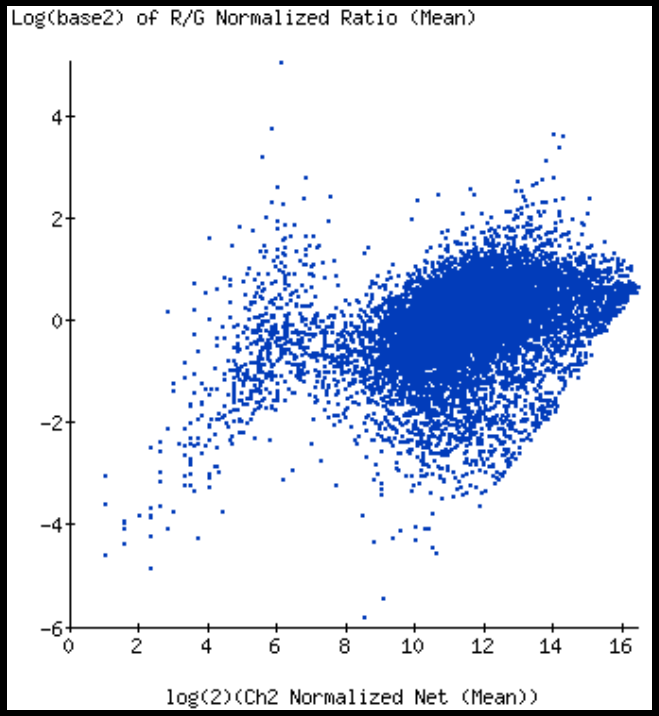


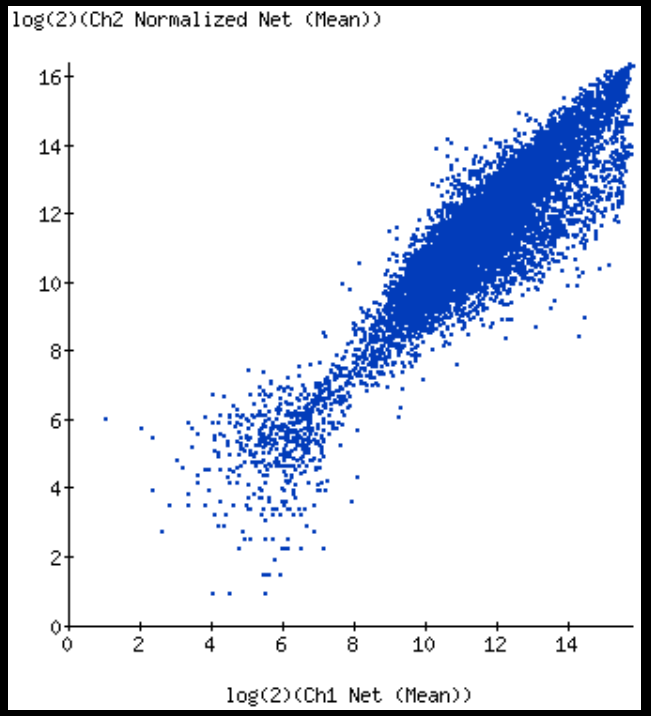

- Plot filter field (here regression correlation) against test field (log ratio).
- Log ratios should center around 0.
- Here, the log ratios appear to diverge below a regression correlation of about 0.4 - 0.6.

# Spots with low regression correlation



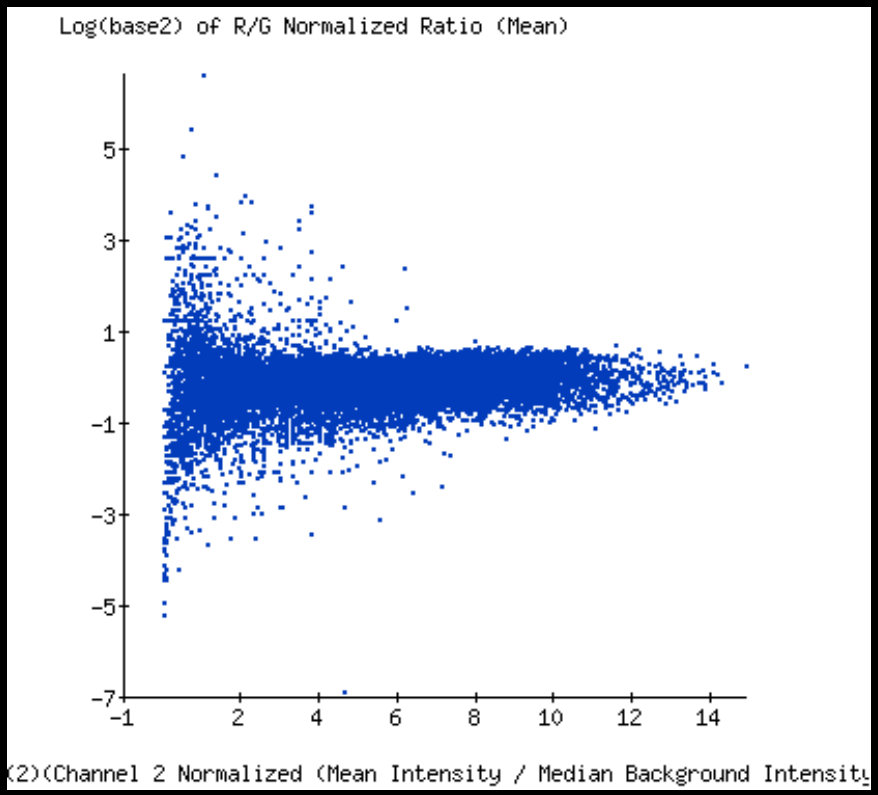
# Data Filtering: Intensity Cutoff




- More than one way to look at a fish.

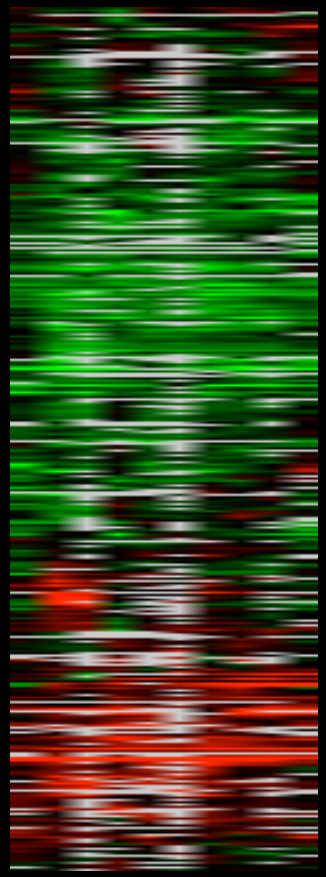
# Data Filtering: Foreground to Background Intensity Ratios



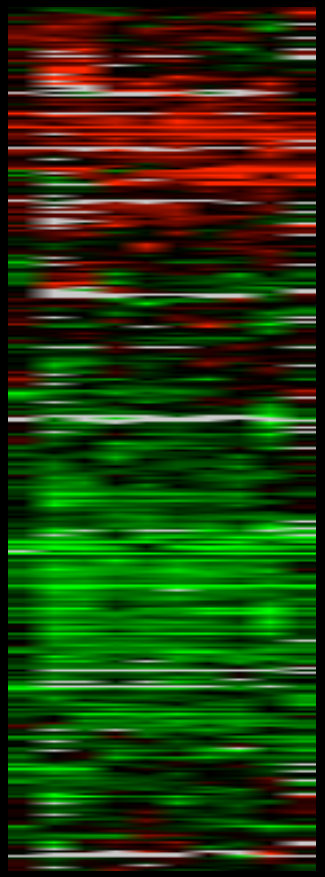

- FG/BG (log scale) versus log ratio
- Data center around 0
- Impose cutoff at 2.5 (linear) to eliminate “flare” at low relative intensity.

# Data Filtering: Intensity to Background Ratios

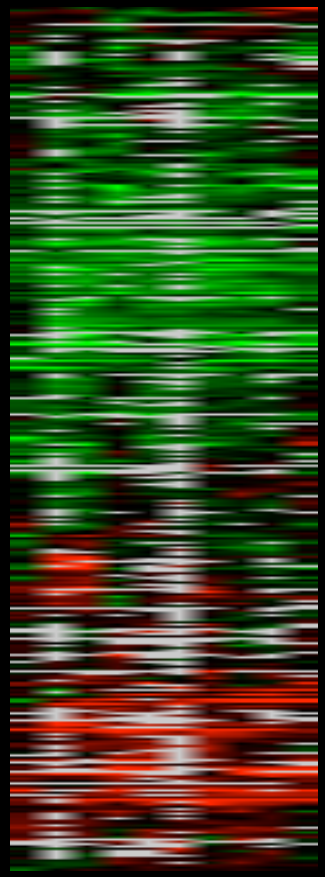
Red Channel (Ch2)



Green Channel (Ch1)



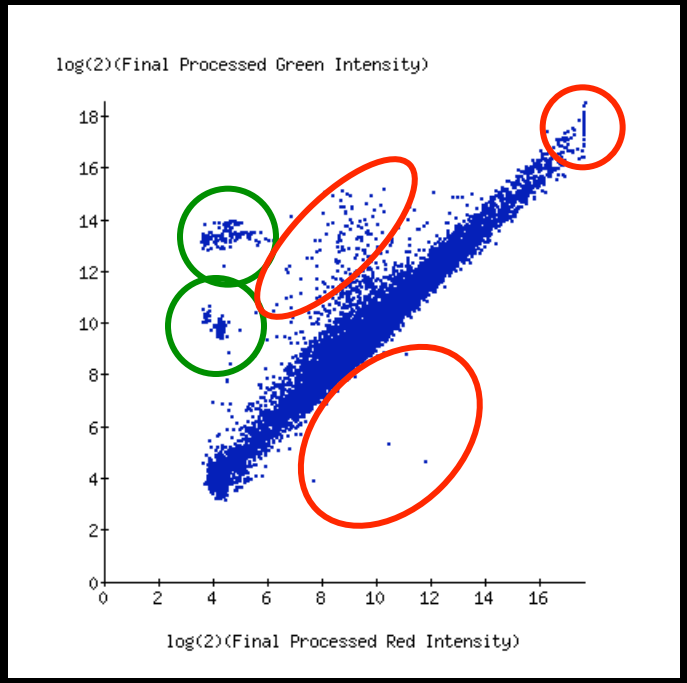
Both Channels



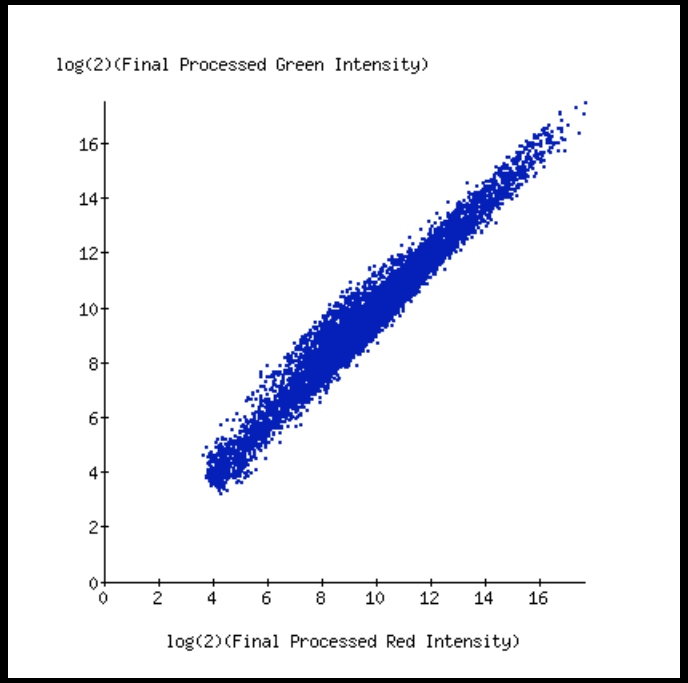
# Data Filtering: Agilent's Outliers

- Boolean flags ; "called" for all features (and local background) in both channels.
- NonUniformity Outliers - Whether a feature (or background) is "non-uniform", i.e. if the pixel noise exceeds a threshold established for a "uniform" feature (or background).
- Population Outliers - Using population statistics (probes with replicate features), outliers are called if where signal is less than a lower threshold or exceeds an upper threshold determined using the interquartile range of the population

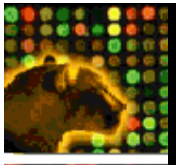
# Data Filtering: Agilent's Outliers




Raw data
(no filters)




Outliers and
controls
filtered away

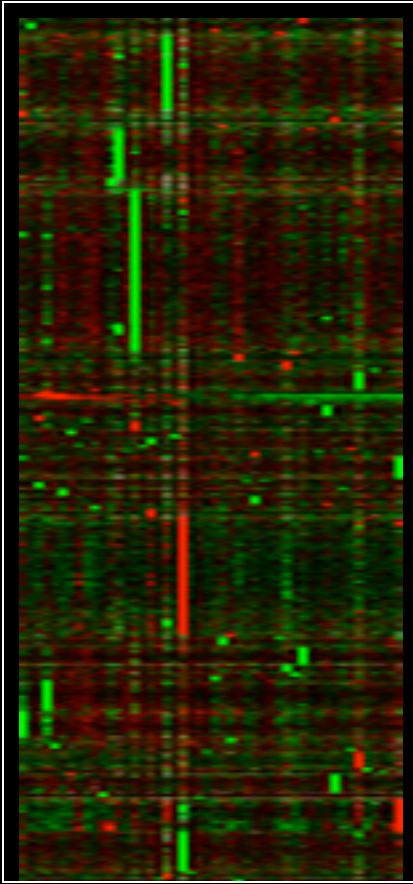


# Population Outliers

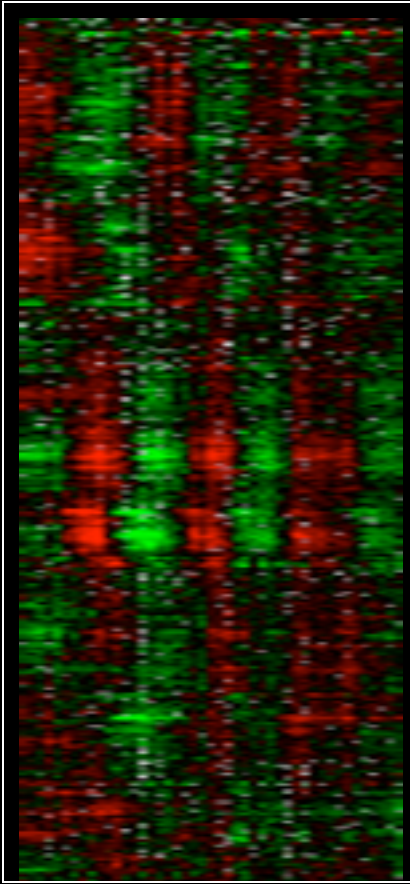
"*Agilent outliers remove not just good data, but sometimes, they remove my best data*" - Maitreya Dunham

# Data filtering : spot, vector, and context

Data from a cell cycle experiment, using 40000 feature microarrays on 48 distinct samples (timepoints of a synchronized cell culture)



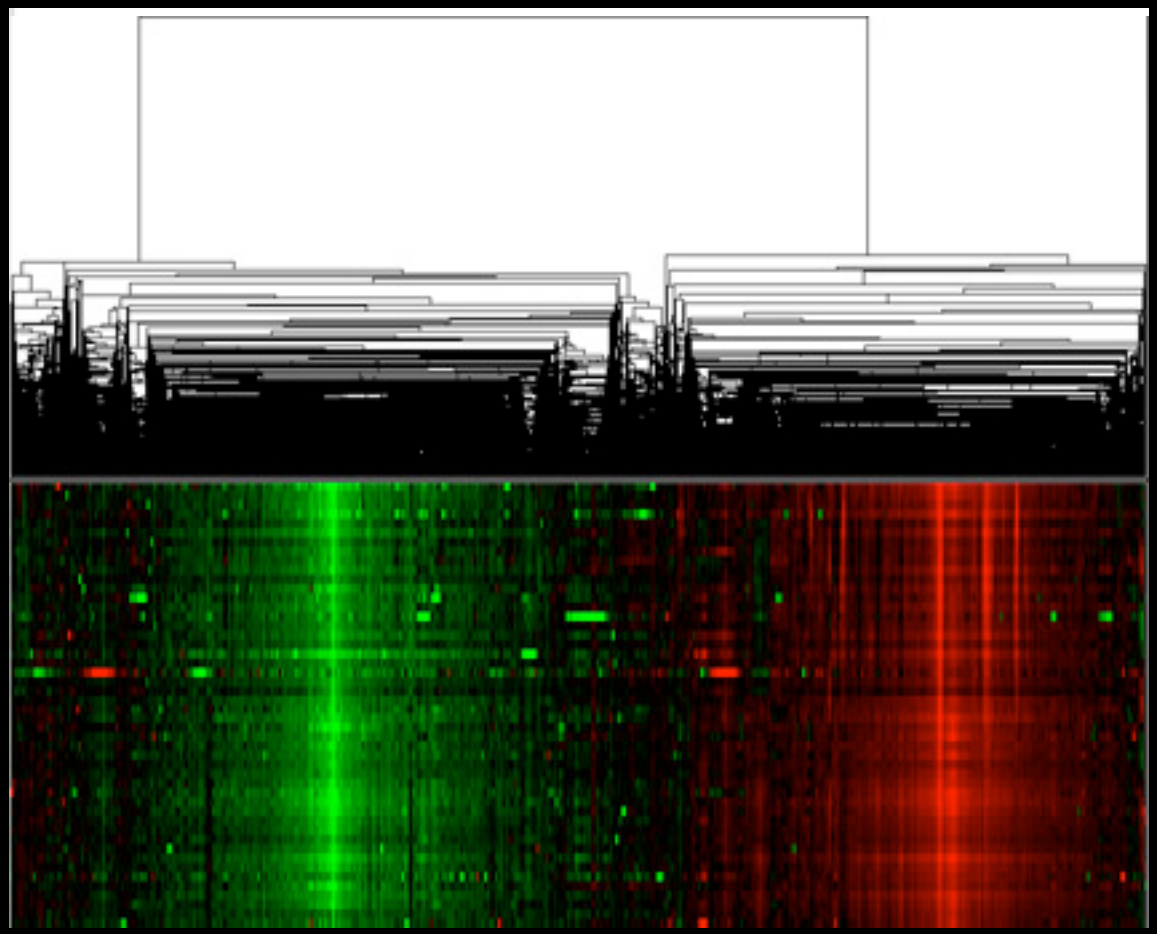
- No quality filters performed
- Rows filtered by arbitrary cut-off of ±1.5 fold
- ~4500 genes, 48 arrays
- data centered and clustered

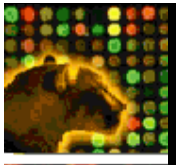


- Spot-quality filtered
- 80% good data in rows or columns
- Rows filtered for periodicity (fourier transformed)
- ~900 genes, 45 arrays
- data centered and clustered

Raw data from *Whitfield, M.L., et al. Mol Biol Cell, 2002. 13(6): p. 1977-2000*

# Needle in a haystack…





# Data Filtering: Conclusion

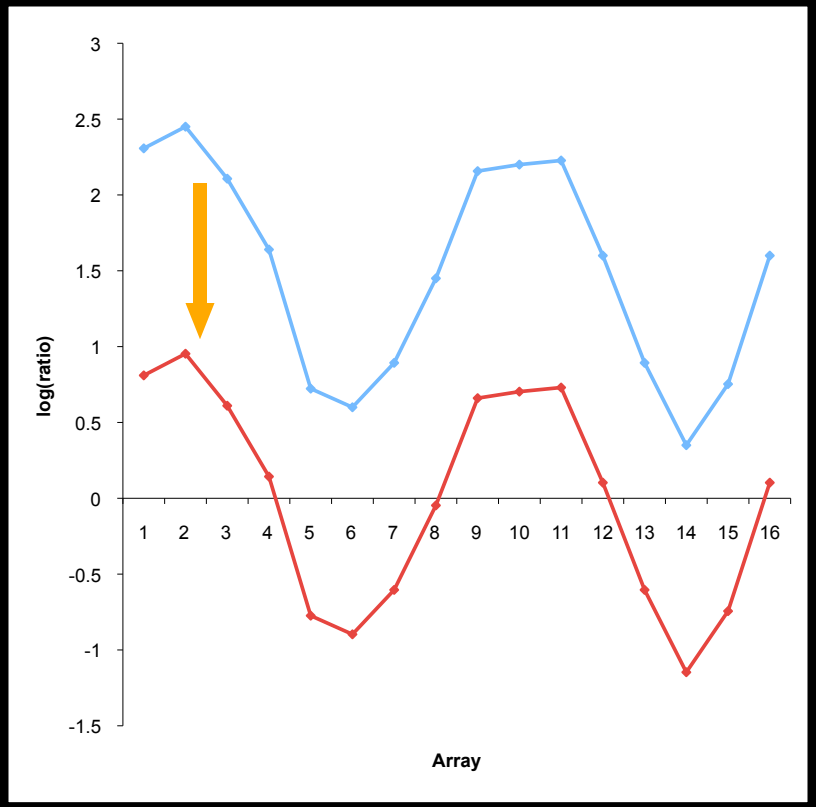
- Don’t keep data you can’t trust!
- Things to look out for…
  - Spatial biases
  - Poor overall signal intensity
  - Poor signal to background
  - Poor fluorescence uniformity across feature
  - Population outliers or other aberrations
  - Systematic gene and array (row and column) problems
- Filter for contextual interest as well

# Concepts of data manipulation

- Data normalization
- Data filtering
- Data centering
- Data clustering

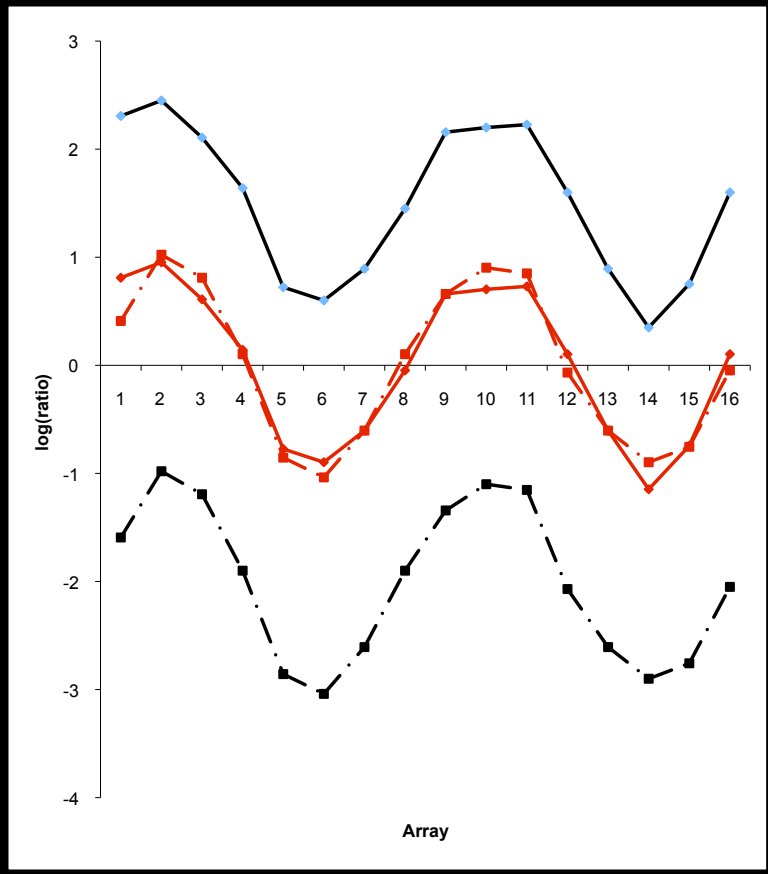
# Data Centering

- Centering sets the average value of a vector to zero.
- This results in a loss of some information, but may reveal important patterns.



# Data Centering

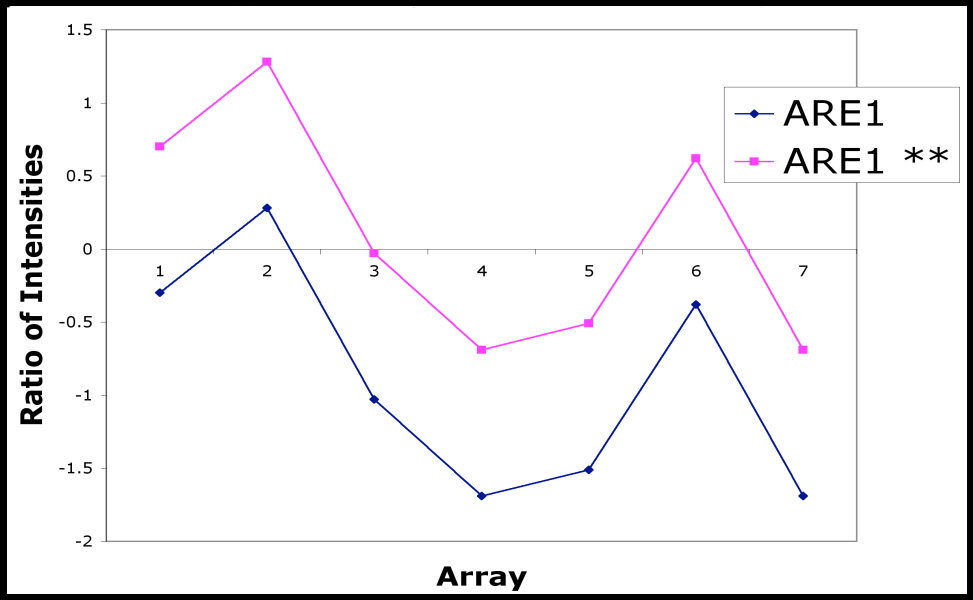
- Centering is useful when the actual value of the ratio is not meaningful (e.g. when using a **common reference**, like a pool of cell lines).
- Centering is generally **not** appropriate when using a biologically meaningful control sample, such as a matched, untreated sample, or a zero timepoint.



# Data Centering

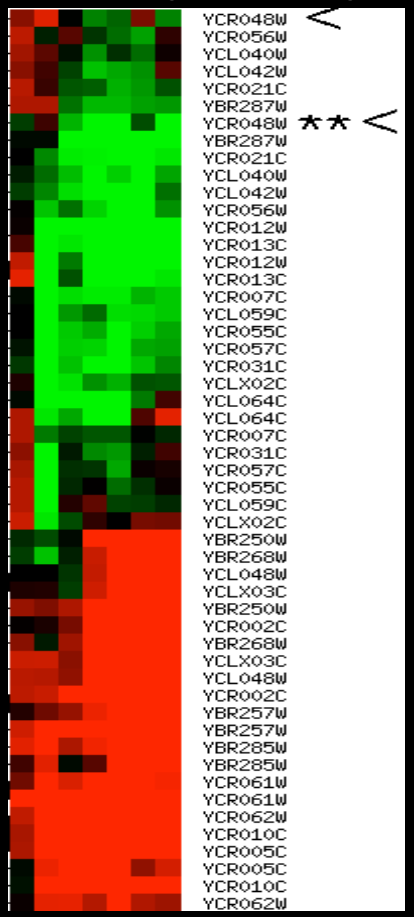
- To illustrate how centering affects data, a small dataset has been duplicated, adding a constant to each row indicated by asterisks (**)

| | A | B | C | D | E | F | G | H |
|---|---|---|---|---|---|---|---|---|
| 1 | NAME | 0 | 0.5 | 2 | 5 | 7 | 9 | 11 |
| 2 | YCL048W | 0.01 | -0.01 | -0.25 | 1.09 | 2.26 | 2.67 | 3.03 |
| 3 | YCL048W ** | 1.01 | 0.99 | 0.75 | 2.09 | 3.26 | 3.67 | 4.03 |
| 4 | | | | | | | | |

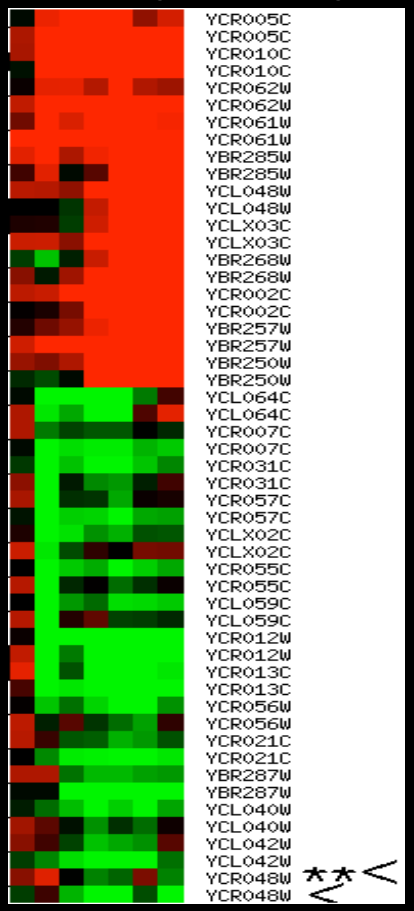


# Data Centering: Effects of Different Strategies

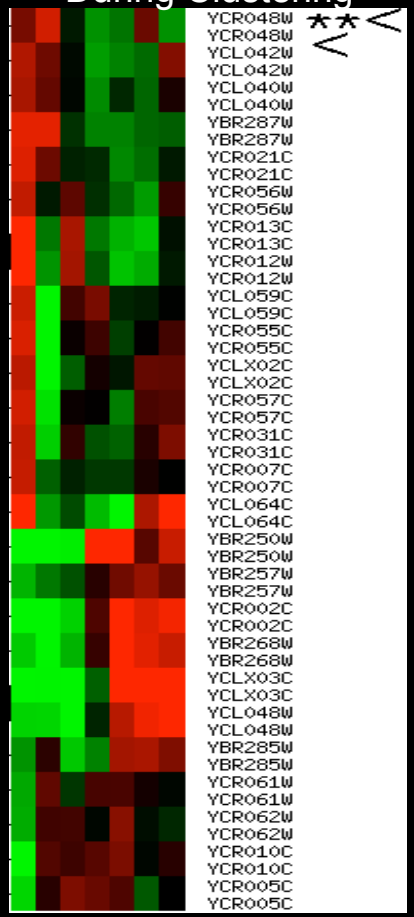
Uncentered Data,
No Centering Metric
During Clustering




Uncentered Data,
Centering Metric
During Clustering

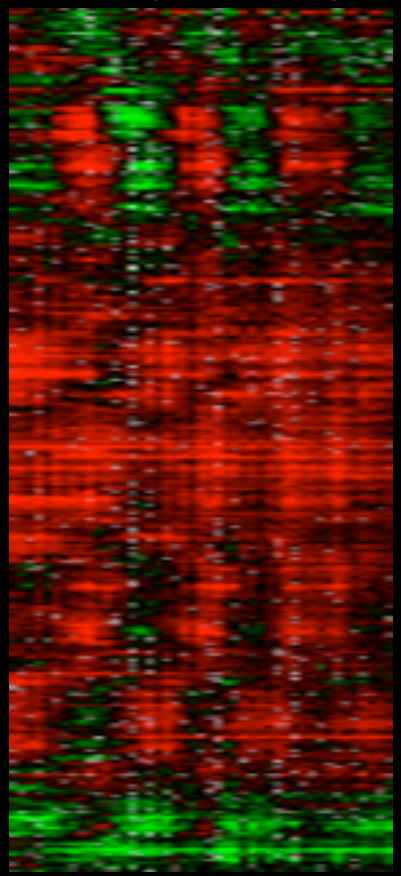



Centered Data,
No Centering Metric
During Clustering

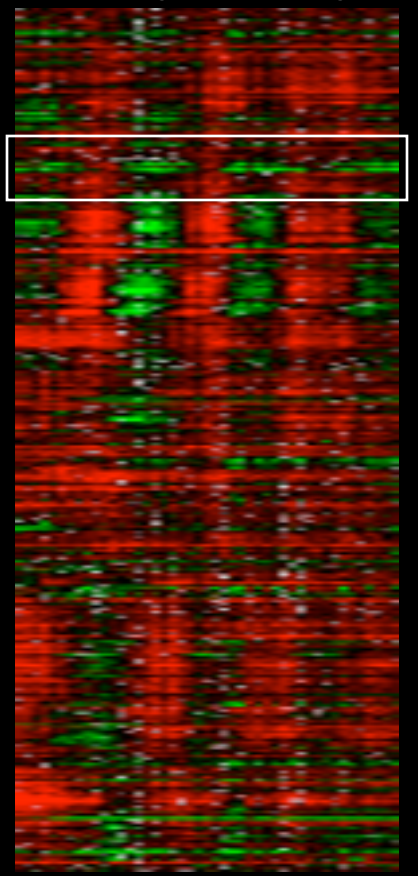


# Data Centering: Actual Dataset

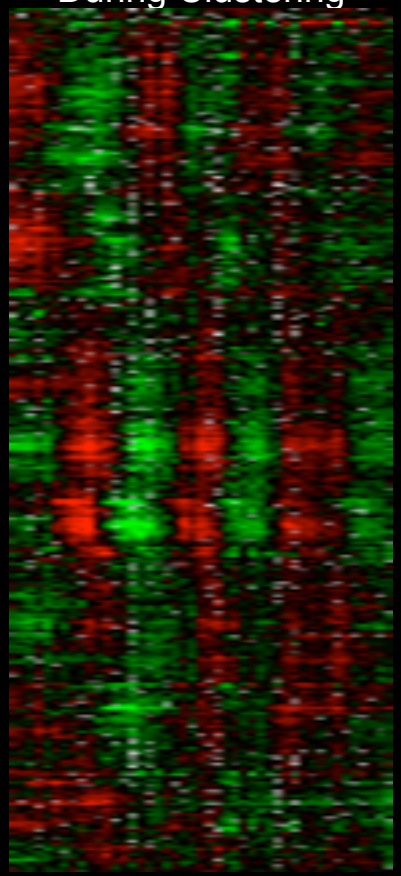
Uncentered Data,
No Centering Metric
During Clustering



Uncentered Data,
Centering Metric
During Clustering



Centered Data,
No Centering Metric
During Clustering



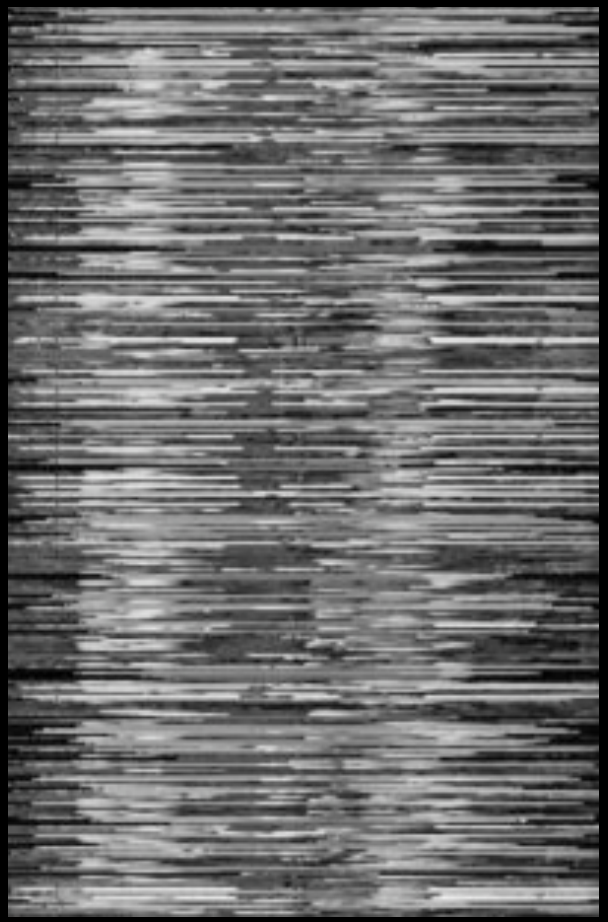
Raw data from *Whitfield, M.L., et al. Mol Biol Cell, 2002. 13(6): p. 1977-2000*

# Clustering Algorithms

- In microarray studies, we often use clustering algorithms to help us identify patterns in complex data.
- For example, we can randomize the data used to represent this painting and see if clustering will help us visualize the pattern.

# Clustering Algorithms

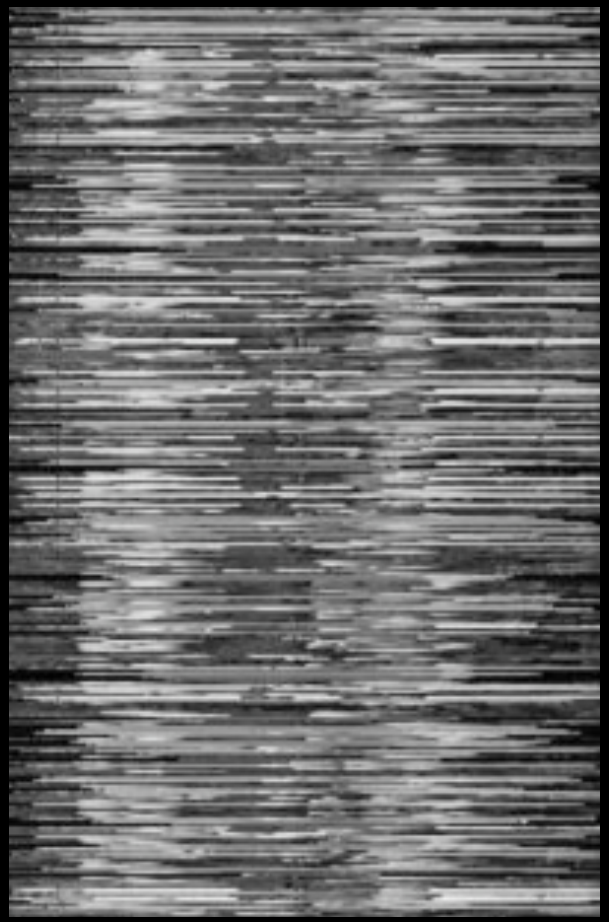
?



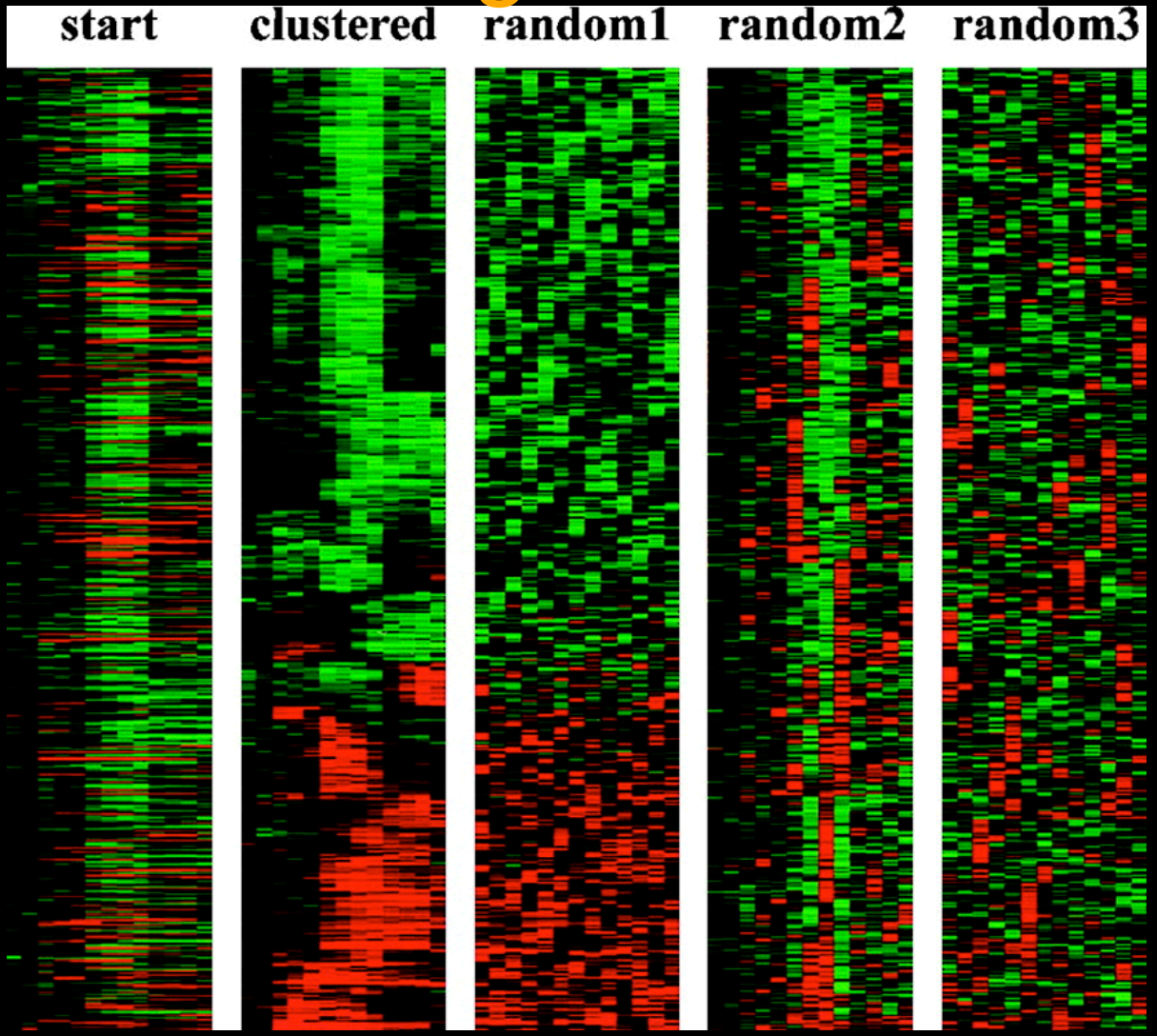
The painting is “sliced” into rows which are then randomized.

# Clustering Algorithms

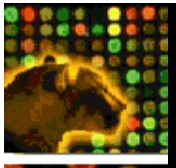
Rows ordered by hierarchical clustering with nodes flipped to optimize ordering



# Clustering Random vs. Biological Data




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



# How does clustering work?

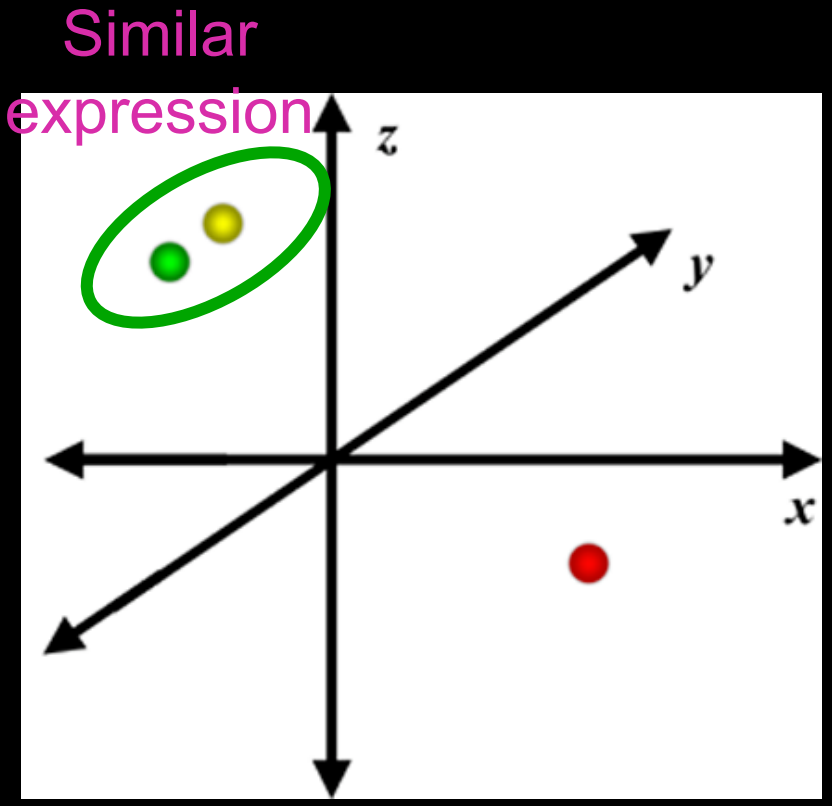
1. Compare all expression patterns to each other.
2. Join patterns that are the most similar out of all patterns.
3. Compare all joined and unjoined patterns.
4. Go to step 2, and repeat until all patterns are joined.

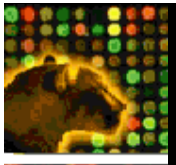
# How do we compare expression profiles?

- Treat expression data for a gene as a multidimensional vector.
- Decide on a distance metric to compare the vectors.

# Expression Vectors

- Crucial concept for understanding clustering
- Each gene is represented by a vector where coordinates are its values (log(ratio)) in each experiment
  - $x = \log(\text{ratio})_{\text{expt1}}$
  - $y = \log(\text{ratio})_{\text{expt2}}$
  - $z = \log(\text{ratio})_{\text{expt3}}$
  - etc.





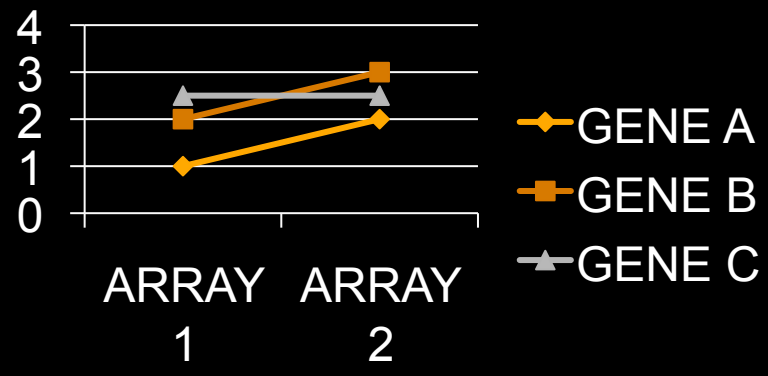
# Distance Metrics

- Distances are measured “between” expression vectors
- Distance metrics define the way we measure distances
- Many different ways to measure distance:
  - Euclidean distance
  - Pearson correlation coefficient(s)
  - Others (Manhattan distance, Mutual information, Kendall’s Tau, etc.)
- Each has different properties and can reveal different features of the data

# Euclidean distance

- The **Euclidean** distance metric detects similar vectors by identifying those that are **closest in space.**
- In this example, A and C are closest to one another.

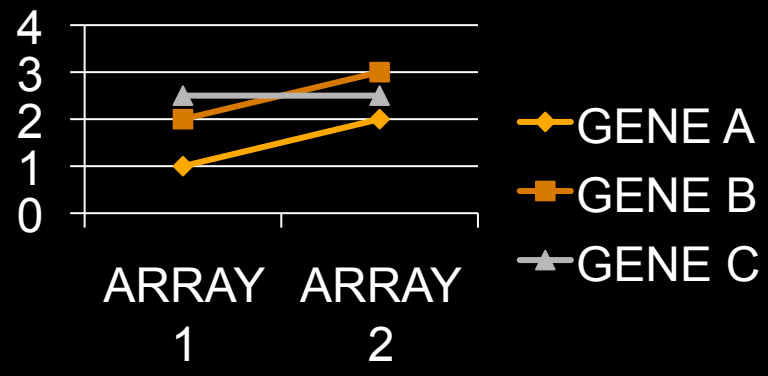
| NAME | ARRAY 1 | ARRAY 2 |
|---|---|---|
| GENE A | 1 | 2 |
| GENE B | 3 | 3 |
| GENE C | 2.5 | 2.5 |



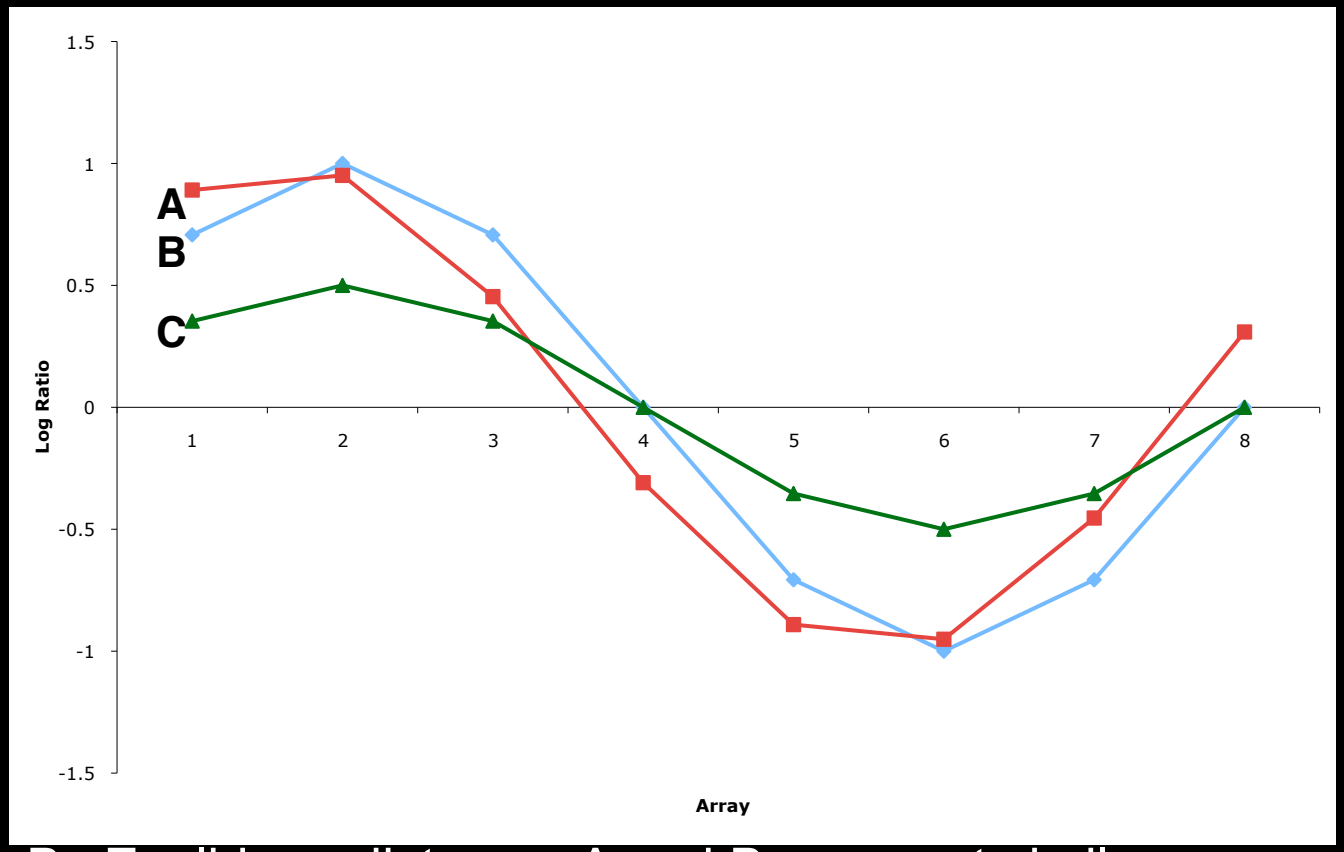
# Pearson correlation

- The **Pearson** correlation disregards the magnitude of the vectors but instead **compares their directions**.
- In this example, Gene A and Gene B have the same slope, so would be most similar to each other.

| NAME | ARRAY 1 | ARRAY 2 |
|---|---|---|
| GENE A | 1 | 2 |
| GENE B | 3 | 3 |
| GENE C | 2.5 | 2.5 |



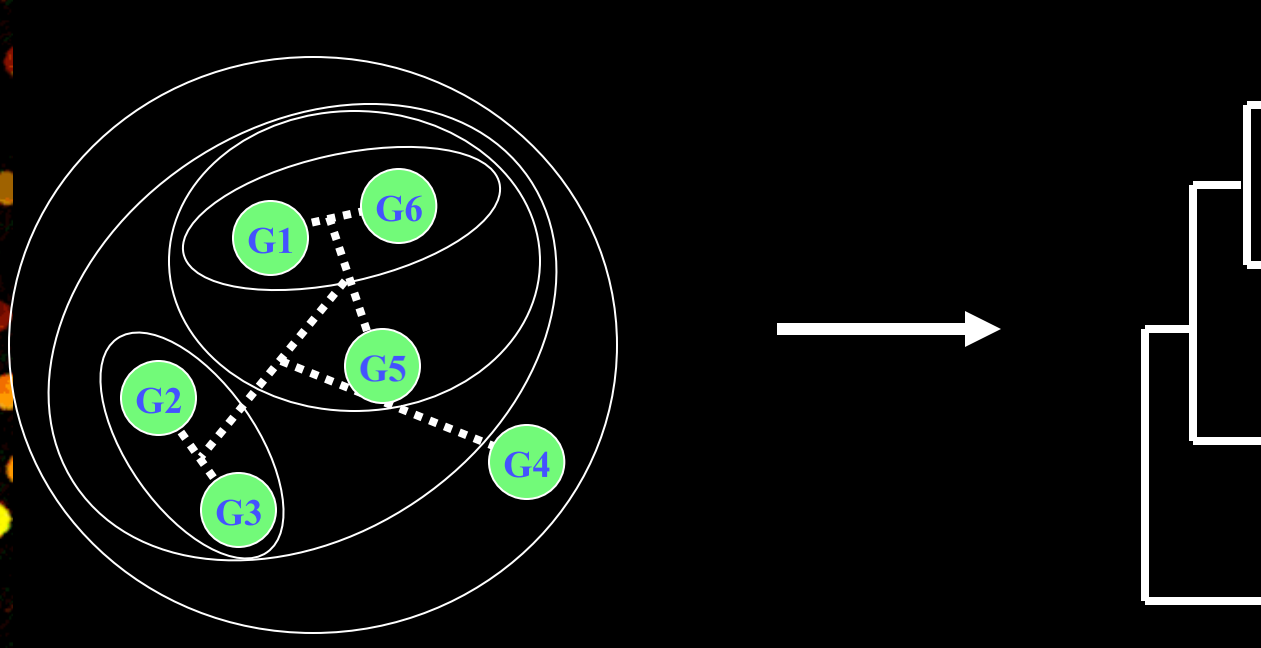
# Distance Metric: Pearson vs. Euclidean




- By Euclidean distance, A and B are most similar.
- By Pearson correlation, A and C are most similar.

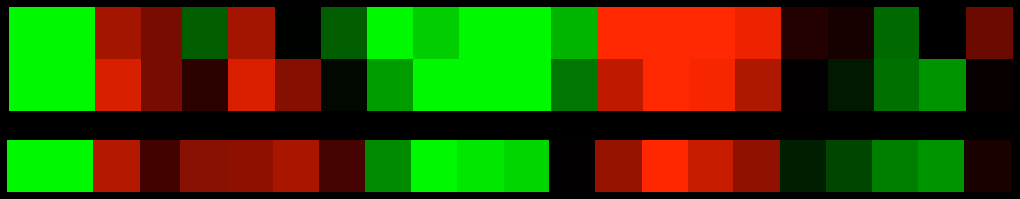
# Hierarchical Clustering

1. Calculate the distance between all genes. Find the smallest distance. If several pairs share the same similarity, use a predetermined rule to decide between alternatives.
2. Fuse the two selected clusters to produce a new cluster that now contains at least two objects. Calculate the distance between the new cluster and all other clusters.
3. Repeat steps 1 and 2 until only a single cluster remains.
4. Draw a tree representing the results.

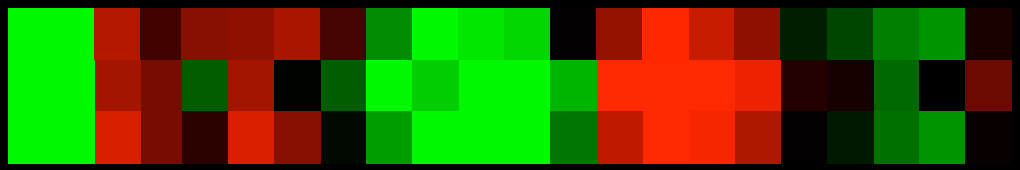


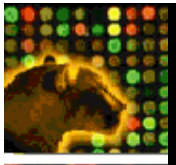
# Clustering: Optimizing node order

- When joining a gene vector to another, it is important to think about the order in which the nodes are joined.

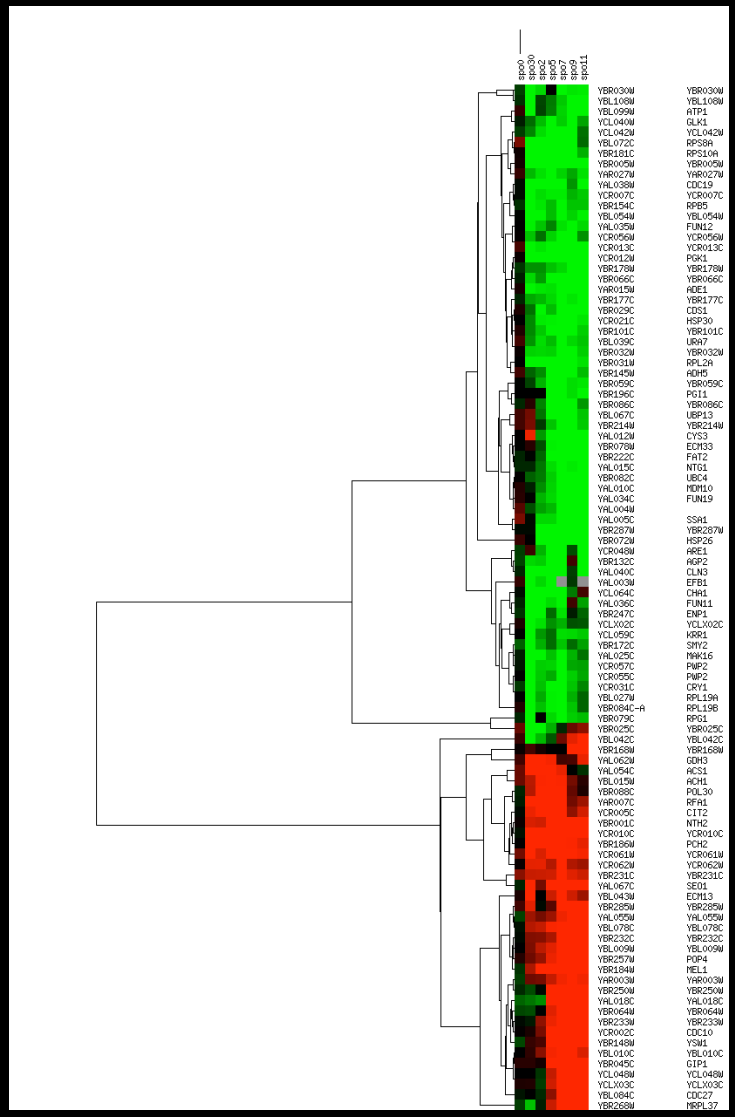


- In this example, ASH1 is allegedly most similar to PIR1, so their patterns are displayed adjacent to one another.



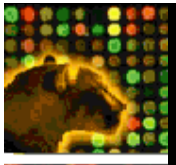


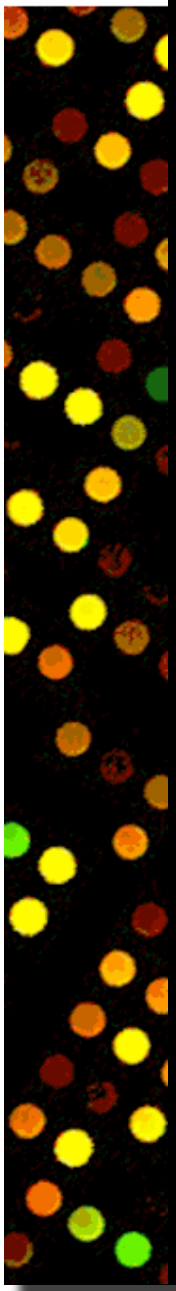
# And we finally get a cluster…

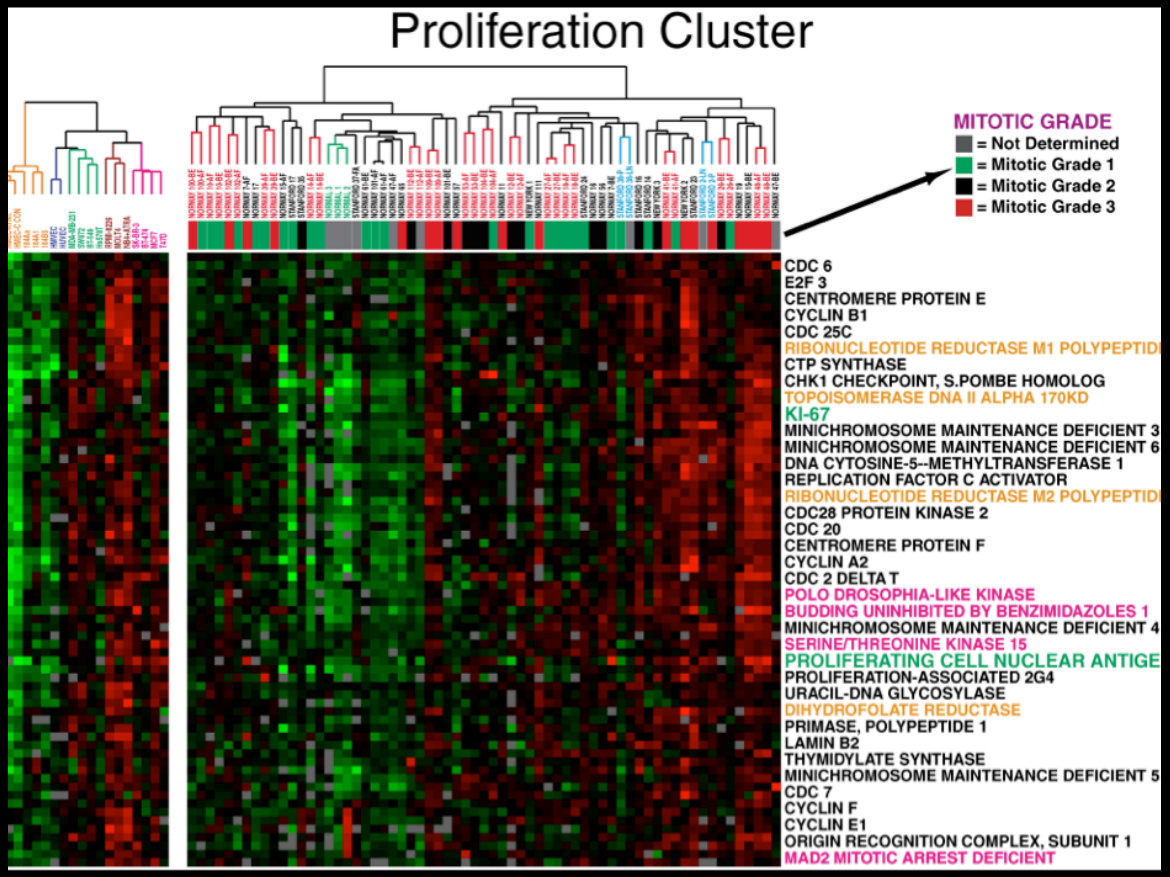


# Clustering: Two-way clustering

- Just as gene patterns are clustered, array patterns can be clustered.
- All the data points for an array can be used to construct a vector for that array and the vectors of multiple arrays can be compared.



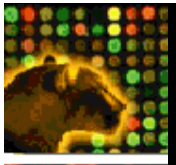





## Clustering: Two-way Clustering

Two-way clustering can help show which samples are most similar, as well as which genes.

From *Perou CM, et al. 2000. Nature 406:747-52*



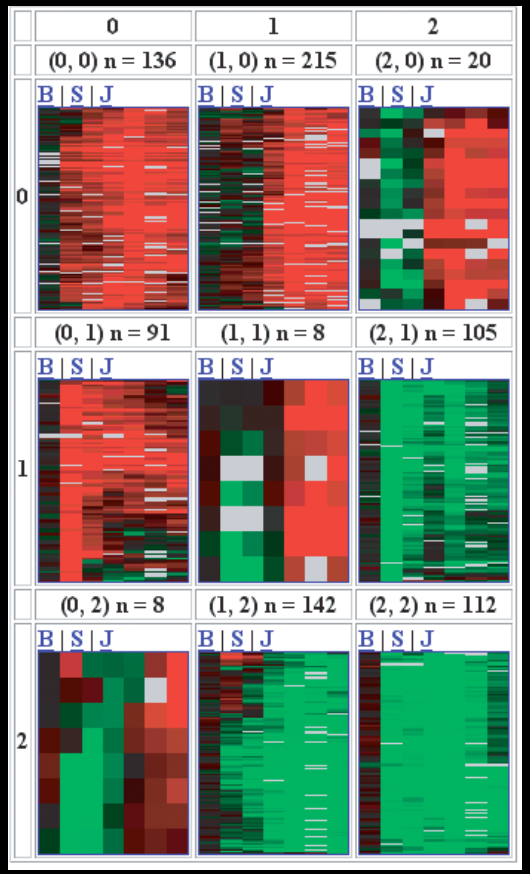
# So is clustering the solution?

- Advantages:
  - Simple
  - Easy to implement
  - Easy to visualize
- Disadvantages:
  - Can lead to incorrect/incomplete conclusions
  - Discarding of subtleties in 2-way clustering
  - May be driven by strong sub-clusters

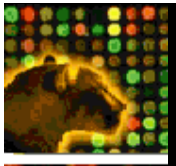
# Clustering: Partitioning Methods

- Split data up into smaller, more homogenous sets
- Should avoid artifacts associated with incorrectly joining dissimilar vectors
- Can cluster each partition independently of others
- Self-Organizing Maps is one partitioning method

# Clustering: Self Organizing Maps




- 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.



# The $64,000 question

- How many partitions do I use?
  - Ask a statistician
    - Tibshirani R, et al. (2000) **Estimating the number of clusters in a dataset via the Gap statistic**
    - http://www-stat.stanford.edu/~tibs/ftp/gap.pdf
  - Ask us, and we'll say trial and error ;-)
    - The ideal outcome is a single expression pattern in each partition, and each partition distinct from the others.

# Working with PUMAdb

- Assay Retrieval
- Using the Database Analysis Pipeline
  - Gene Selection and Annotation
  - Data Filtering
  - Data Retrieval, and reports
  - Gene Filtering
  - Clustering and Image Generation
- Other Things You Should Know…
  - Repository (specifically, Synthetic Genes)
  - Java TreeView

# Assay retrieval : Search software

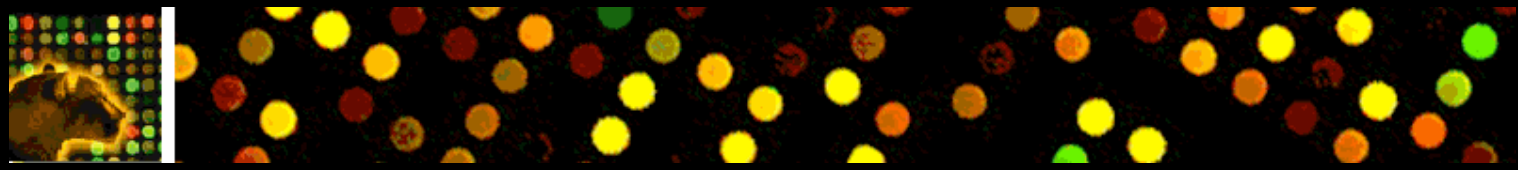
**Use 'Basic Search' to browse/retrieve:**

- a single Publication
- a single Experiment set
  * your personal sets
  * others', if viewable
- a single Experimental category

**Use 'Advanced Search' to perform:**

- A boolean search
  * by Experimenter
  * by Category
  * by Subcategory
- A search by Print
- A search by arraylist

# Demo : Assay retrieval



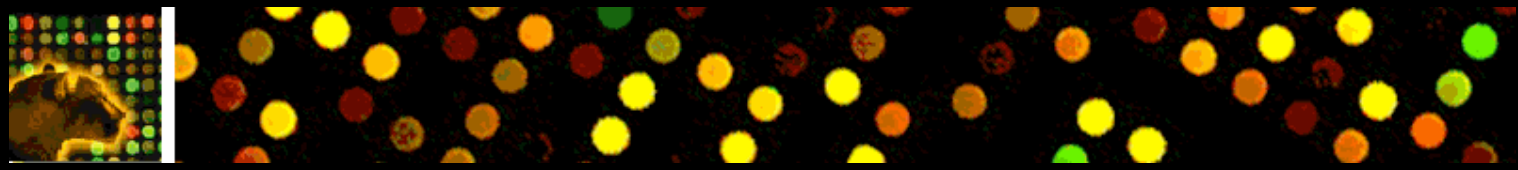
Live demonstration at

PUMAdb - http://puma.princeton.edu

# Data Processing and Clustering

- Experiment Selection
- Gene Selection and Annotation
- Data Filtering
- Data Retrieval
- Gene Filtering
- Clustering and Image Generation

# Demo : pipeline and tools

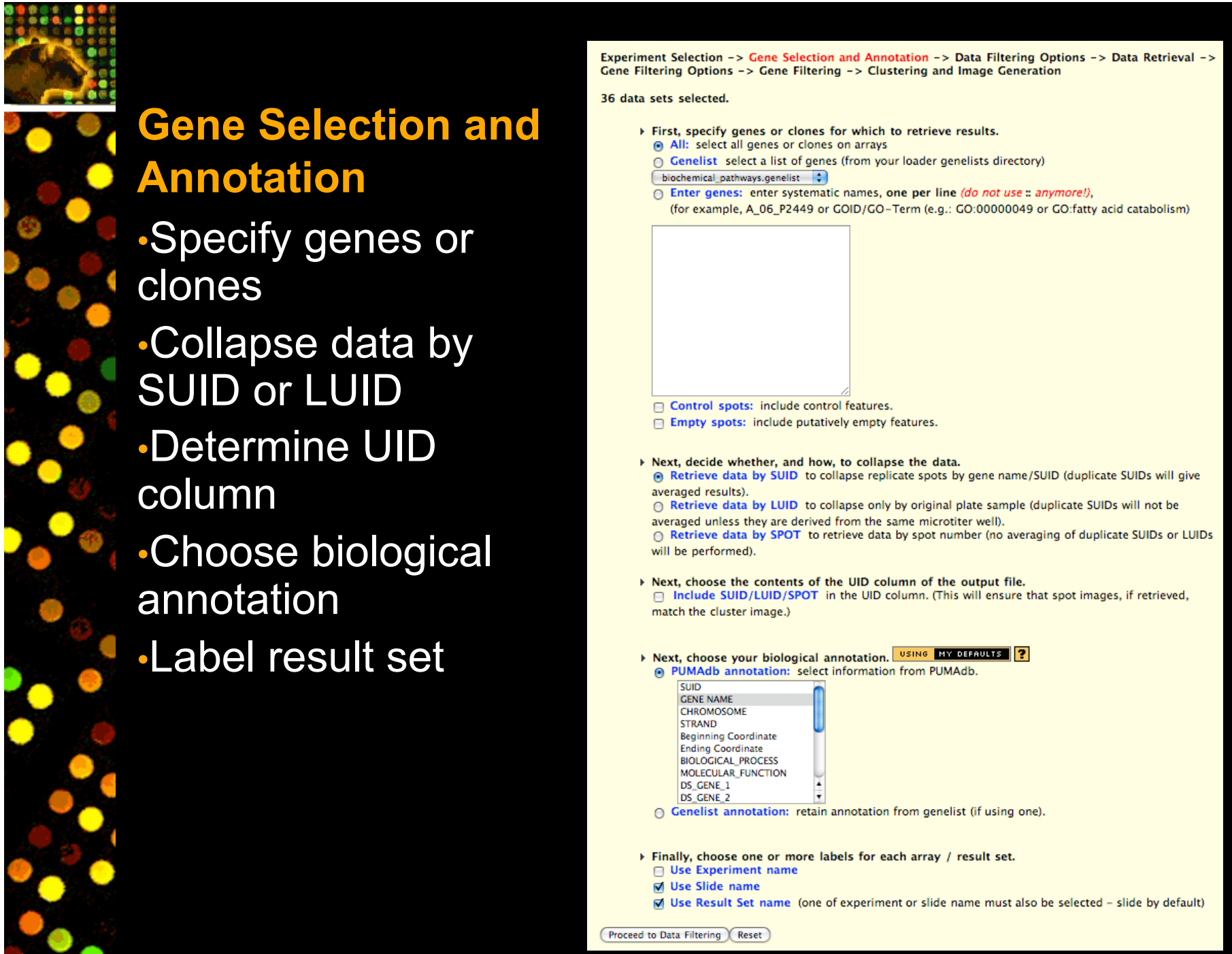


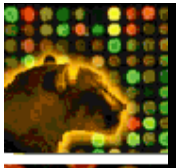
Live demonstration at

PUMAdb - http://puma.princeton.edu

# Gene Selection and Annotation

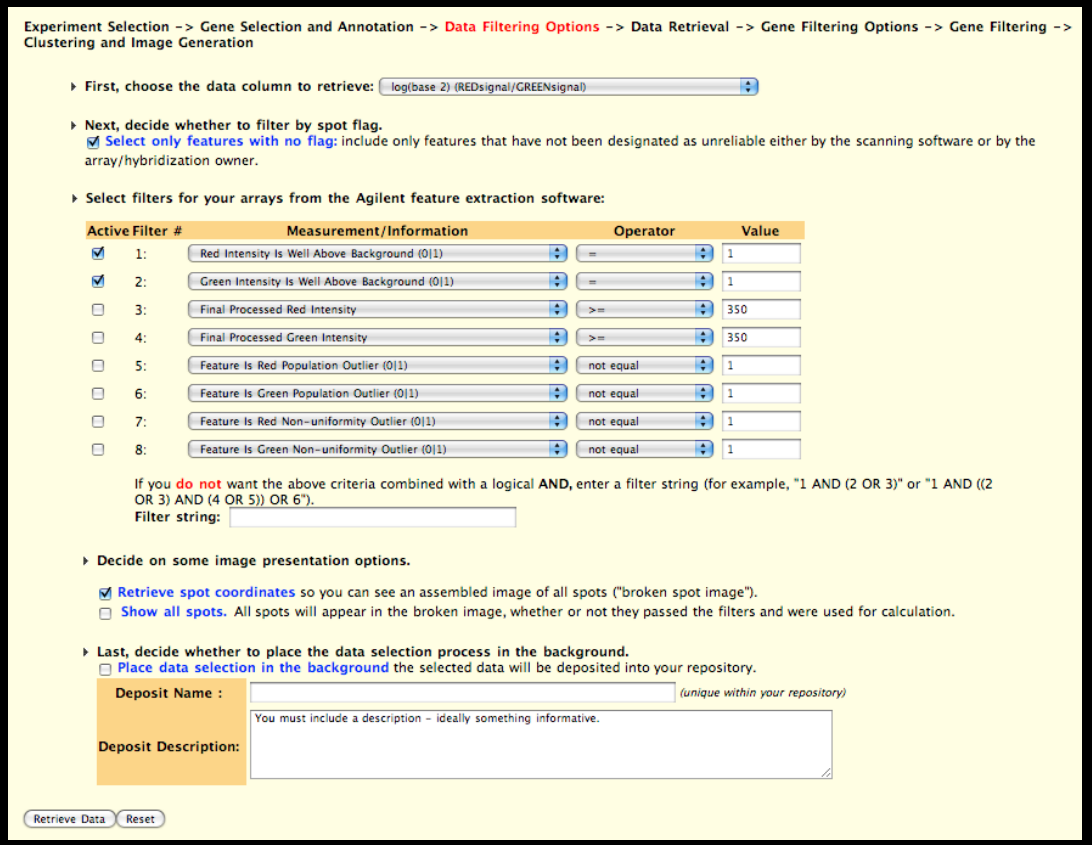
- Specify genes or clones
- Collapse data by SUID or LUID
- Determine UID column
- Choose biological annotation
- Label result set



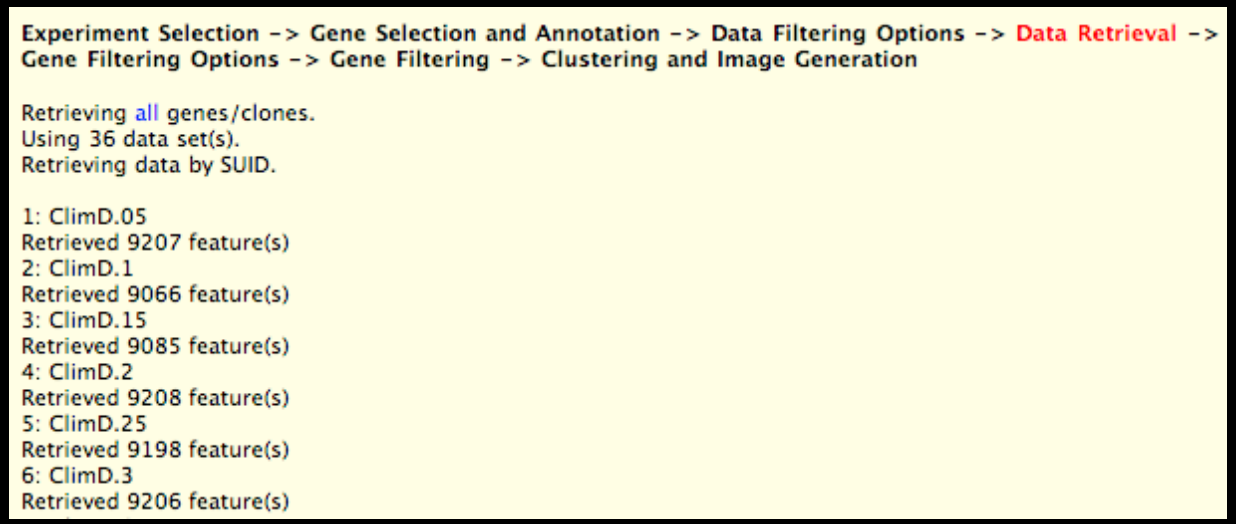


# Data Filtering

- Choose data column to retrieve
- Elect to invert reverse dye replicates
- Elect to filter by spot flag
- Select spot criteria for filtering
- Define image presentation options
- Add to repository in background? (for large datasets)



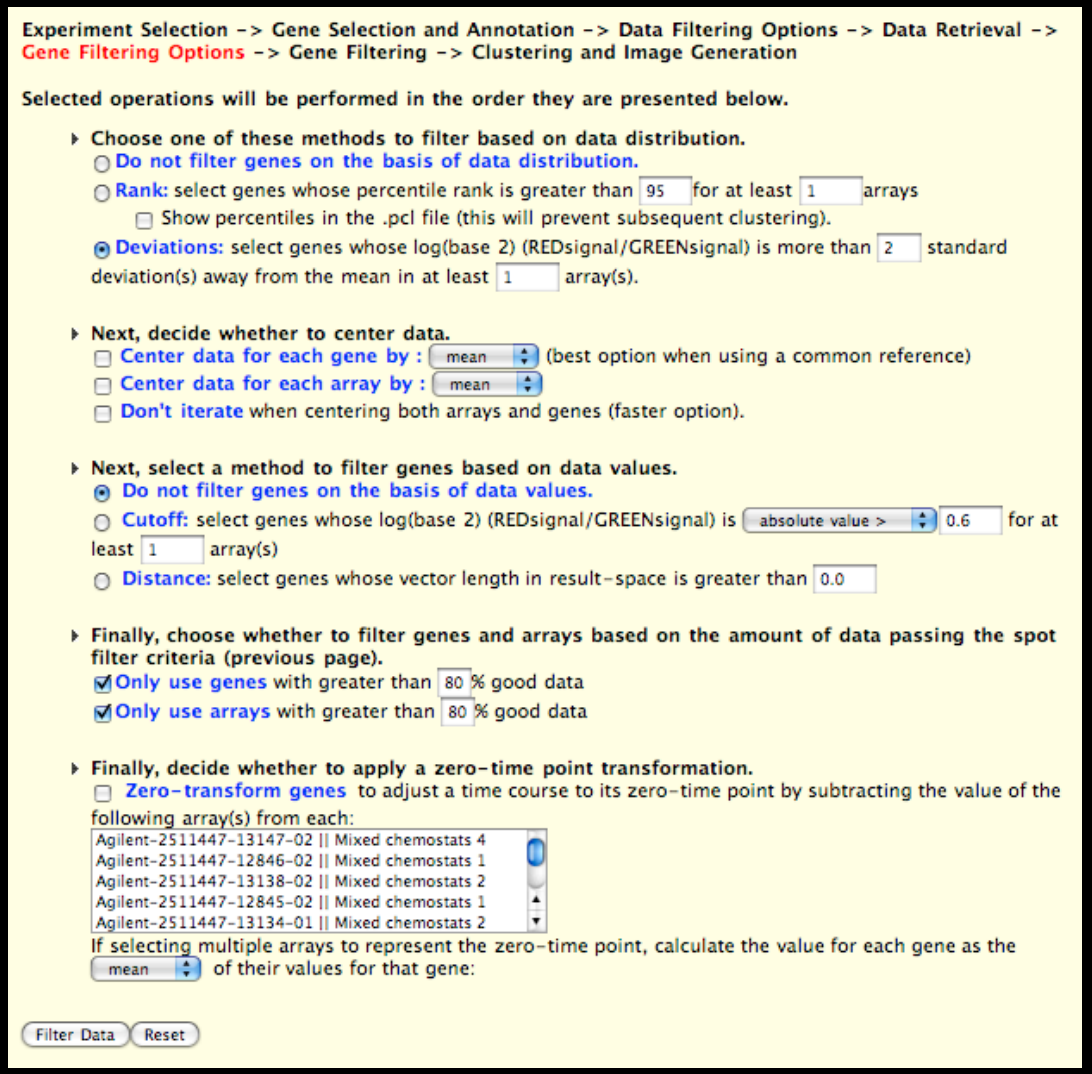
# Data Retrieval




- General results and progress
- PreClustering (.pcl) file
- Data retrieval summary report
- Option to deposit data in repository

# Gene Filtering

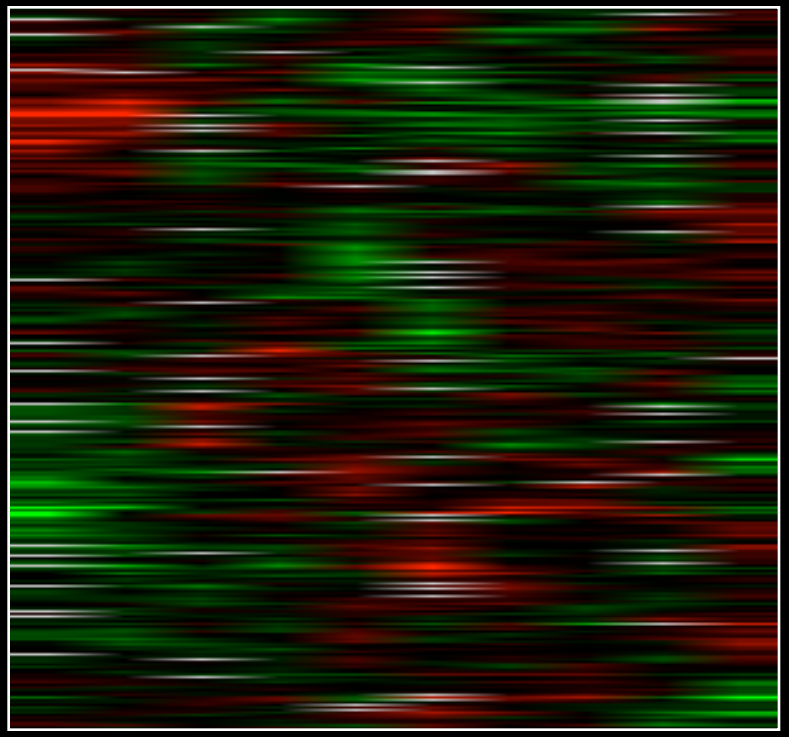
- Transform single-channel data
- Filter genes based on data distribution
- Data centering
- Filter genes based on data values
- Filter genes and arrays based on spot filter criteria
- Zero-transformation



# Spot Filtering vs. Gene Filtering

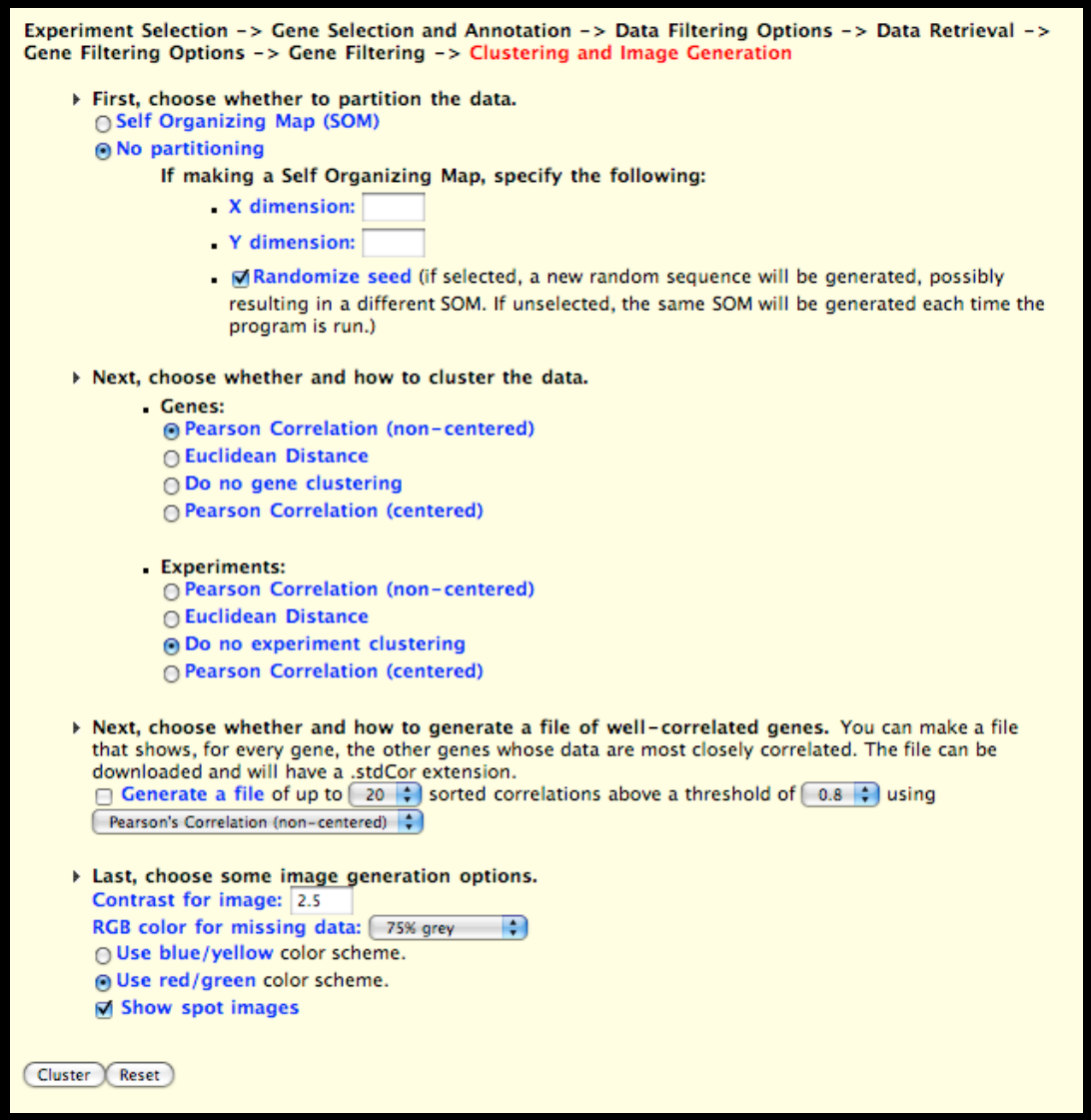
Spot filters remove individual data points. That means there will be more missing (gray) data.

Gene filters remove the genes that do not meet the filter criteria often enough. This reduces the number of genes/rows (and untrusted or uninteresting data).



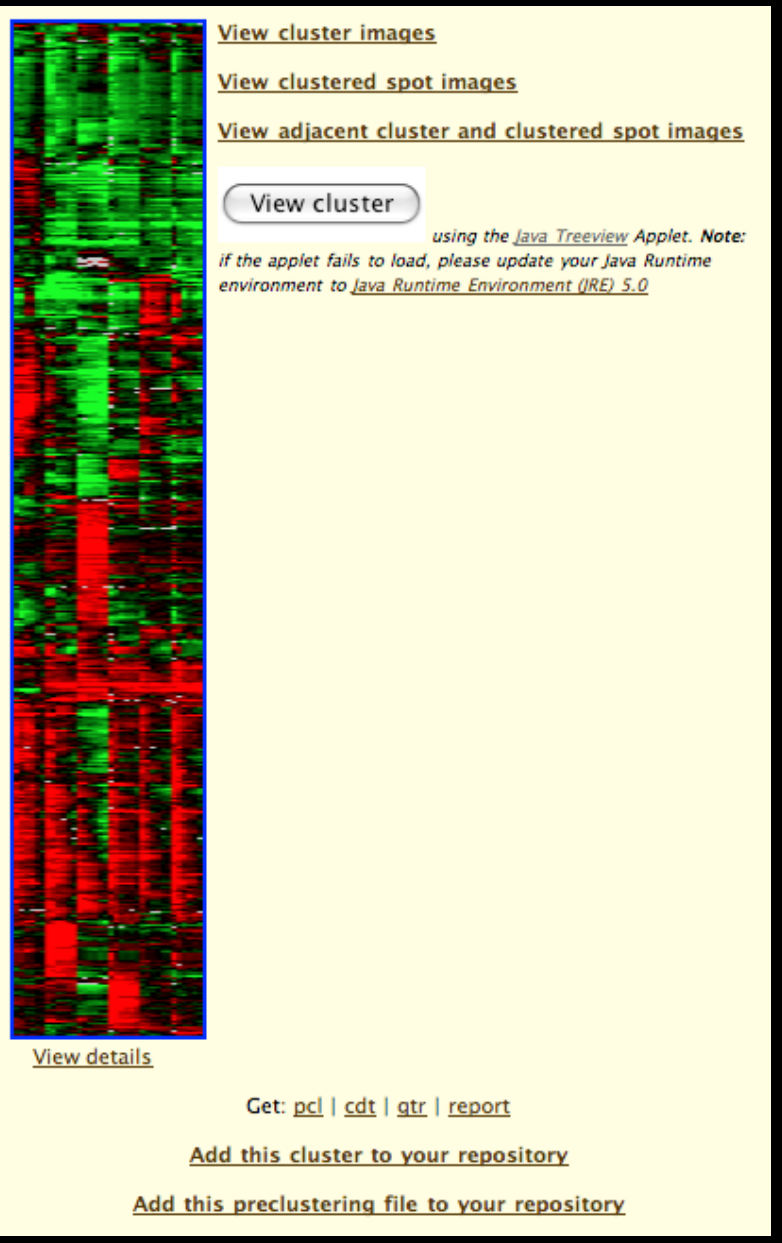
# Clustering and Image Generation

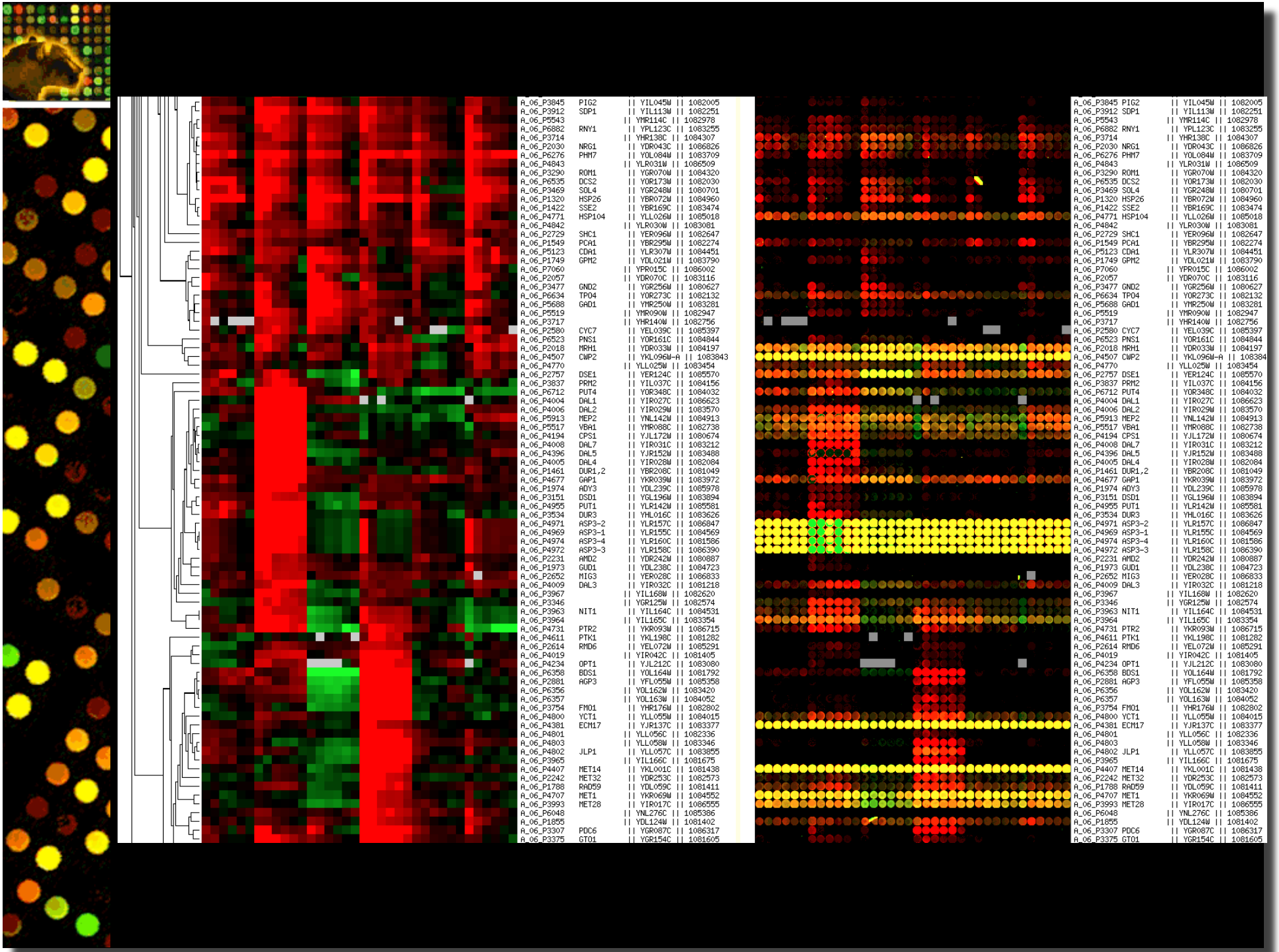
- Partitioning options
- Clustering metric selections
- Correlated genes
- Image generation options



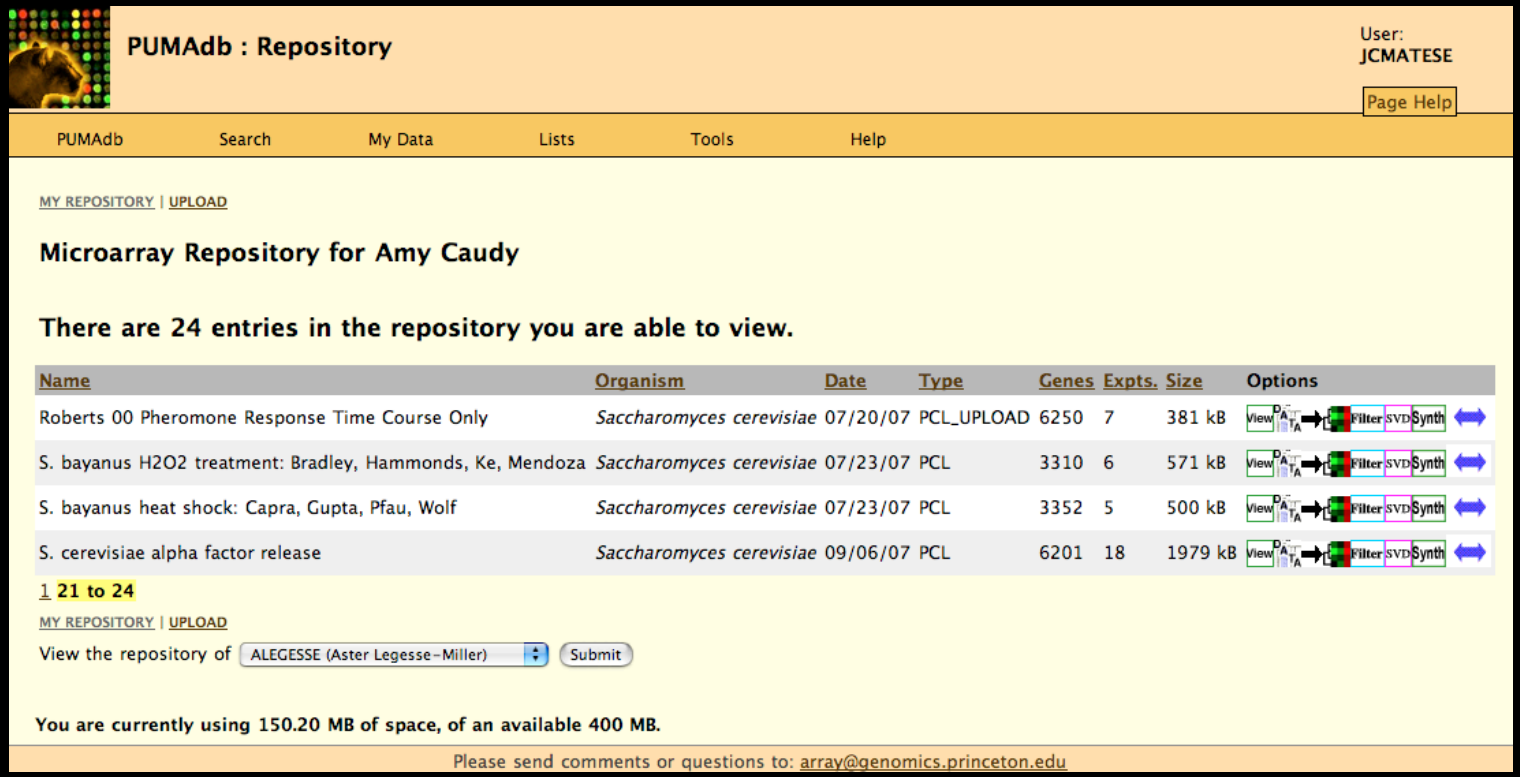
## Clustering and Image Generation

- Cluster images
  - ratios
  - spots
  - both, adjacent
- Basic visualization applets
- Data files for client applications
- Deposit to repository



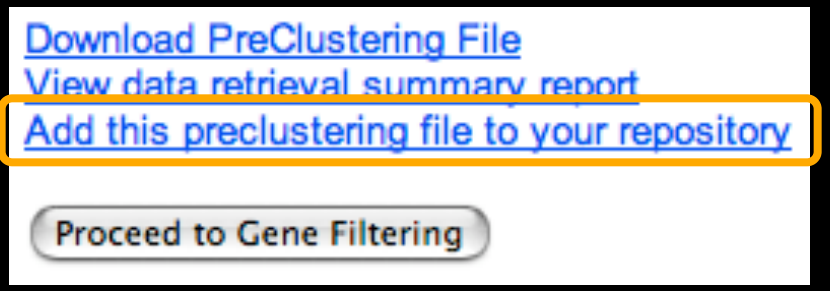

A_06_P3845 PIG2 || YIL045W || 1082005
A_06_P3912 SDP1 || YIL113W || 1082251
A_06_P5543 || YMR114C || 1082978
A_06_P6882 RNY1 || YPL123C || 1083255
A_06_P3714 || YHR138C || 1084307
A_06_P2030 NRG1 || YDR043C || 1086826
A_06_P6276 PHM7 || YOL084W || 1083709
A_06_P4843 || YLR031W || 1086509
A_06_P3290 ROM1 || YGR070W || 1084320
A_06_P6535 DCS2 || YOR173W || 1082030
A_06_P3469 SOL4 || YGR248W || 1080701
A_06_P1320 HSP26 || YBR072W || 1084960
A_06_P1422 SSE2 || YBR169C || 1083474
A_06_P4771 HSP104 || YLL026W || 1085018
A_06_P4842 || YLR030W || 1083081
A_06_P2729 SHC1 || YER096W || 1082647
A_06_P1549 PCA1 || YBR295W || 1082274
A_06_P5123 CDA1 || YLR307W || 1084451
A_06_P1749 GPM2 || YDL021W || 1083790
A_06_P7060 || YPR015C || 1086002
A_06_P2057 || YDR070C || 1083116
A_06_P3477 GND2 || YGR256W || 1080627
A_06_P6634 TPO4 || YOR273C || 1082132
A_06_P5688 GAD1 || YMR250W || 1083281
A_06_P5519 || YMR090W || 1082947
A_06_P3717 || YHR140W || 1082756
A_06_P2580 CYC7 || YEL039C || 1085397
A_06_P6523 PNS1 || YOR161C || 1084844
A_06_P2018 MRH1 || YDR033W || 1084197
A_06_P4507 CWP2 || YKL096W-A || 1083843
A_06_P4770 || YLL025W || 1083454
A_06_P2757 DSE1 || YER124C || 1085570
A_06_P3837 PRM2 || YIL037C || 1084156
A_06_P6712 PUT4 || YOR348C || 1084032
A_06_P4004 DAL1 || YIR027C || 1086623
A_06_P4006 DAL2 || YIR029W || 1083570
A_06_P5913 MEP2 || YNL142W || 1084913
A_06_P5517 VBA1 || YMR088C || 1082738
A_06_P4194 CPS1 || YJL172W || 1080674
A_06_P4008 DAL7 || YIR031C || 1083212
A_06_P4396 DAL5 || YJR152W || 1083488
A_06_P4005 DAL4 || YIR028W || 1082084
A_06_P1461 DUR1,2 || YBR208C || 1081049
A_06_P4677 GAP1 || YKR039W || 1083972
A_06_P1974 ADY3 || YDL239C || 1085978
A_06_P3151 DSD1 || YGL196W || 1083894
A_06_P4955 PUT1 || YLR142W || 1085581
A_06_P3534 DUR3 || YHL016C || 1083626
A_06_P4971 ASP3-2 || YLR157C || 1086847
A_06_P4969 ASP3-1 || YLR155C || 1084569
A_06_P4974 ASP3-4 || YLR160C || 1081586
A_06_P4972 ASP3-3 || YLR158C || 1086390
A_06_P2231 AMD2 || YDR242W || 1080887
A_06_P1973 GUD1 || YDL238C || 1084723
A_06_P2652 MIG3 || YER028C || 1086833
A_06_P4009 DAL3 || YIR032C || 1081218
A_06_P3967 || YIL168W || 1082620
A_06_P3346 || YGR125W || 1082574
A_06_P3963 NIT1 || YIL164C || 1084531
A_06_P3964 || YIL165C || 1083354
A_06_P4731 PTR2 || YKR093W || 1086715
A_06_P4611 PTK1 || YKL198C || 1081282
A_06_P2614 RMD6 || YEL072W || 1085291
A_06_P4019 || YIR042C || 1081405
A_06_P4234 OPT1 || YJL212C || 1083080
A_06_P6358 BDS1 || YOL164W || 1081792
A_06_P2881 AGP3 || YFL055W || 1085358
A_06_P6356 || YOL162W || 1083420
A_06_P6357 || YOL163W || 1084052
A_06_P3754 FMO1 || YHR176W || 1082802
A_06_P4800 YCT1 || YLL055W || 1084015
A_06_P4381 ECM17 || YJR137C || 1083377
A_06_P4801 || YLL056C || 1082336
A_06_P4803 || YLL058W || 1083346
A_06_P4802 JLP1 || YLL057C || 1083855
A_06_P3965 || YIL166C || 1081675
A_06_P4407 MET14 || YKL001C || 1081438
A_06_P2242 MET32 || YDR253C || 1082573
A_06_P1788 RAD59 || YDL059C || 1081411
A_06_P4707 MET1 || YKR069W || 1084552
A_06_P3993 MET28 || YIR017C || 1086555
A_06_P6048 || YNL276C || 1085386
A_06_P1855 || YDL124W || 1081402
A_06_P3307 PDC6 || YGR087C || 1086317
A_06_P3375 GT01 || YGR154C || 1081605

# Repository

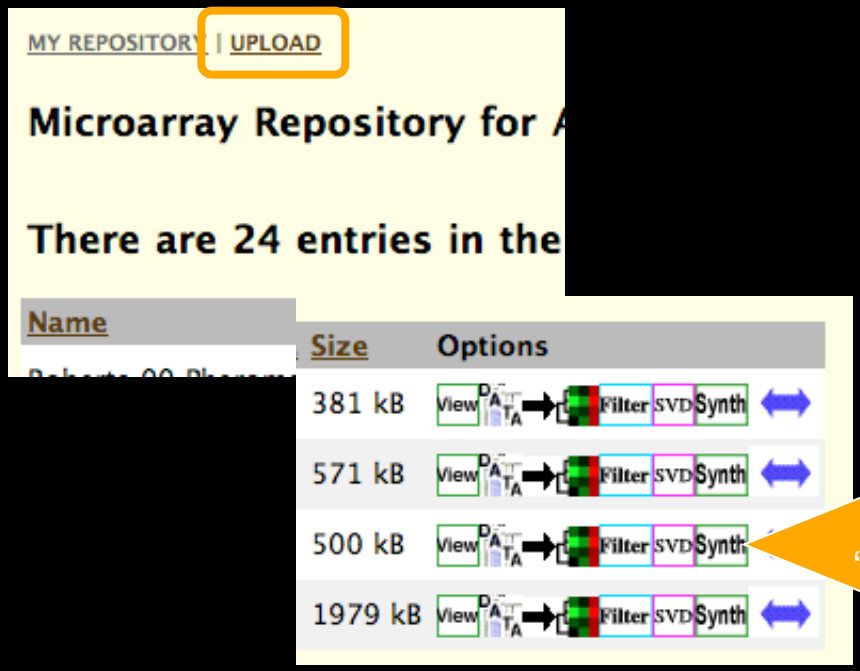



- Deposit the results of an analysis
- Re-enter the pipeline (filter, cluster)
- Download locally
- SVD Analysis
- “Synthetic gene” transformation

# Depositing Data into your Respository




- Deposit from data retrieval “pipeline”

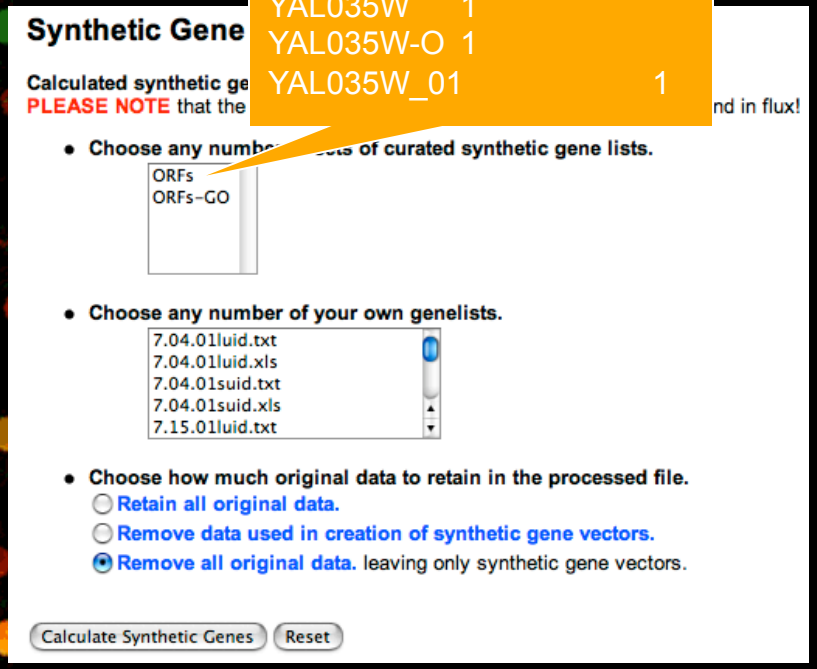



- Upload from desktop

“Synthetic gene transformation”

# “Synthetic” Gene Transformation

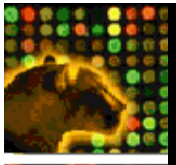
```
# NAME YAL035W
# ANNOTATION FUN12
NAME          WEIGHT
11374_AT      1
A_06_P1059                  1
YAL035W       1
YAL035W-O     1
YAL035W_01                  1
```




- A "synthetic gene" is a group of "reporters" (clones, oligos, ORFs, etc.), together with some arithmetic method of combining their expression vectors.
- Can be used to either “translate” reporter names (e.g. *S. bayanus* orthologous cosmid ids, Agilent ID to systematic ORF name) or combine reporters from different platforms into a representative gene.
- Specialty lists could potentially be used to capture the behavior of a class of genes, such as all proteases, or all genes in a given cytoband.
- Choose handling of original data: retain, remove, or retain unused data.

# External Analysis Tools

- Bioconductor
- TIGR Mev
- Java Treeview
- Gene Ontology (GO) and its application

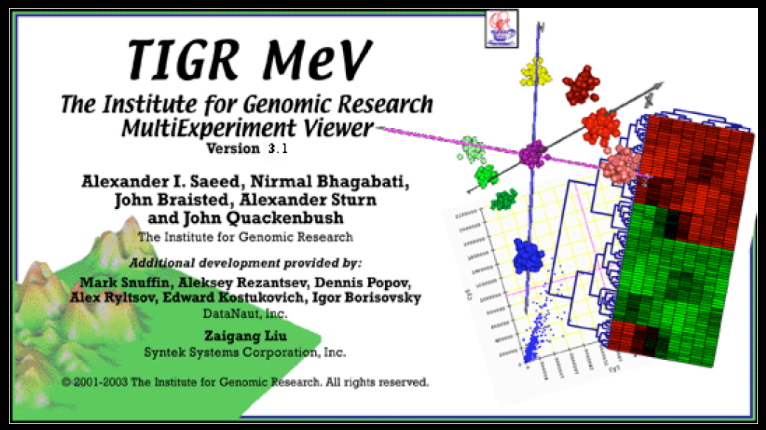


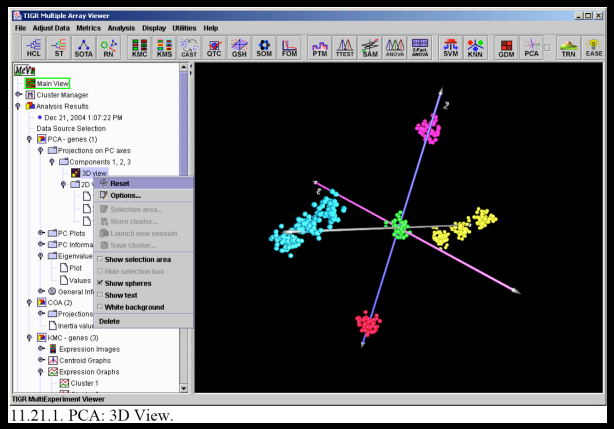
# BioConductor

- An “open source and open development software project
for the analysis and comprehension of genomic data.”
- R and the R package system
- Packages available for pre-processing Affymetrix and cDNA array data
- Real-time associations to biological metadata from GenBank, LocusLink, PubMed, etc.

http://www.bioconductor.org/

# TIGR MeV



11.21.1. PCA: 3D View.

- MeV = MultiExperiment Viewer
- Analysis plug-ins
  - Hierarchical clustering
  - Support trees
  - Self-organizing Maps
  - K-Means Clustering
  - Gene shaving
  - Principal components analysis
  - Support Vector machines
  - T-Tests
  - ANOVA
  - and more…

*"MeV is a versatile microarray data analysis tool, incorporating sophisticated algorithms for clustering, visualization, classification, statistical analysis and biological theme discovery."*

# Java TreeView

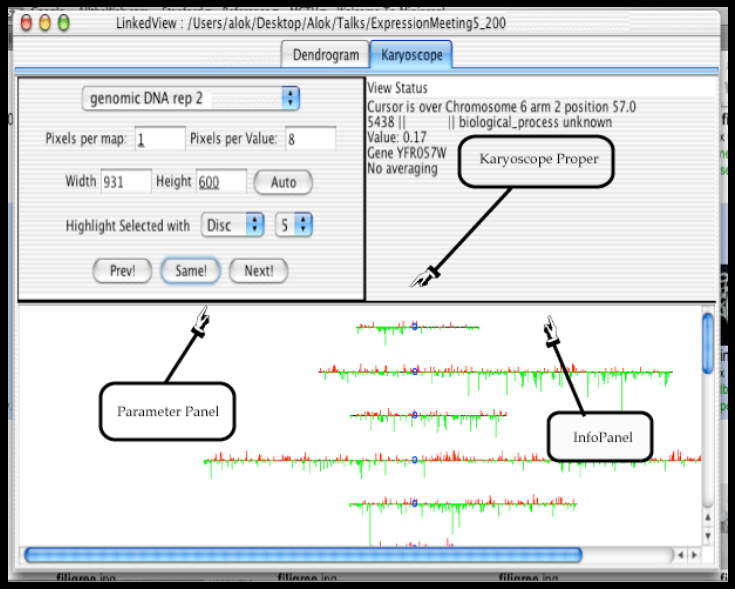

Saldanha, A.J. Bioinformatics, 2004. **20**(17): p. 3246-8.

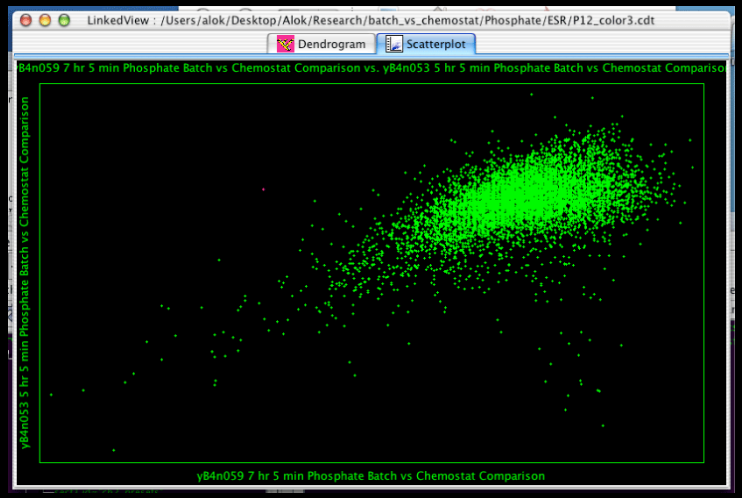
http://jtreeview.sourceforge.net/

… and for those Affymetrix and Agilent sequence identifiers

http://puma.princeton.edu/help/treeview_url/

# Java TreeView : Additional Features




Originally just a dendrogram view, but now also supports a Karyoscope View, ScatterView, and more…

# Gene Ontology (GO)

- "a collaborative effort to address the need for consistent descriptions of gene products in different databases"
- Controlled vocabularies describing a gene product's biological process, molecular function, and cellular component
- Vocabularies are structured (directed acyclic graphs)

*http://www.geneontology.org/*

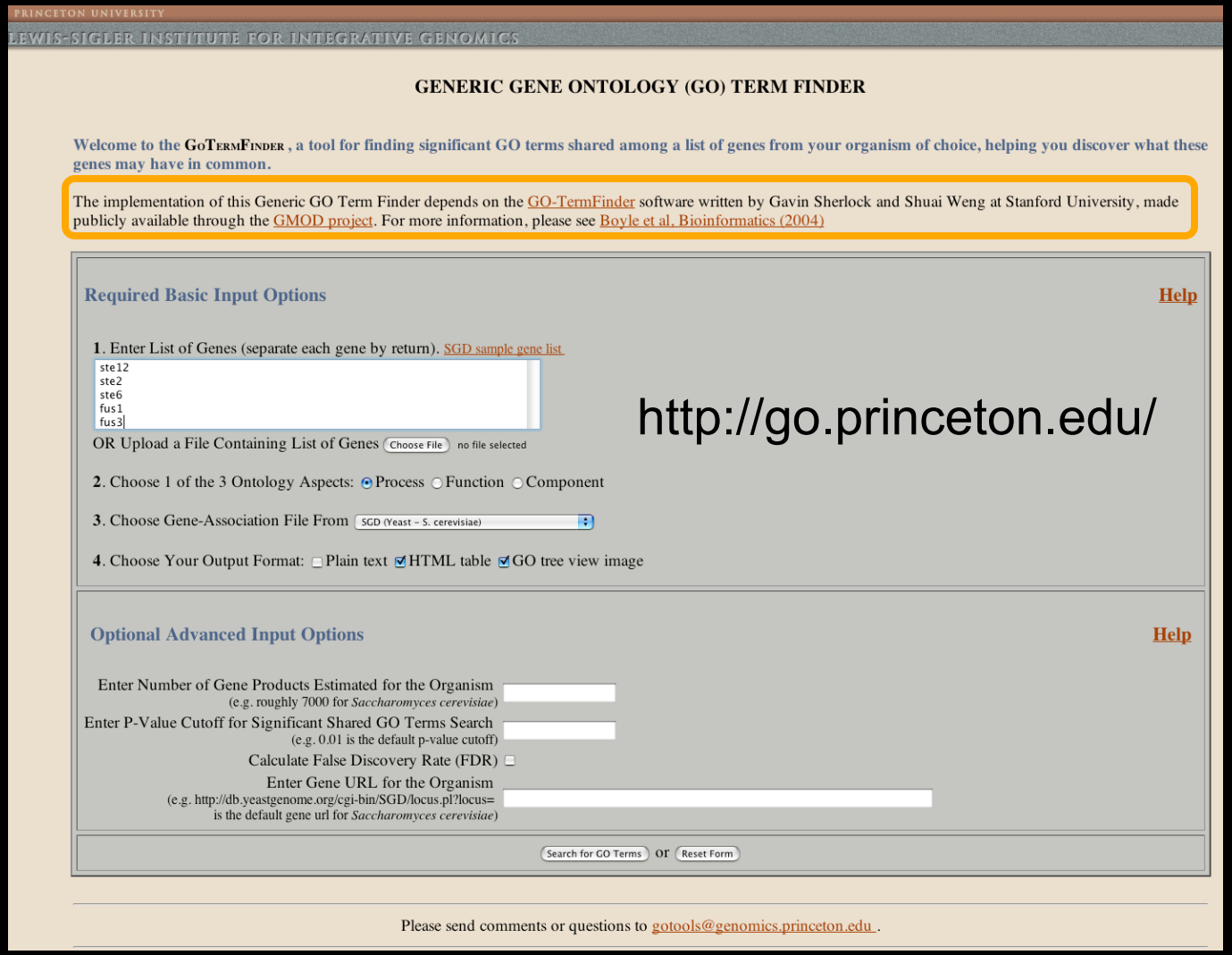
# Many, many array (genelist) applications using the GO

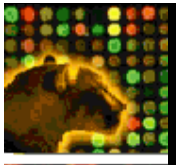
- Over 40 at last count …
- FatiGO, GoSurfer, GOMiner, L2L, NetAffx, Spotfinder, etc.
- GO TermFinder
  - a standalone version resides at :

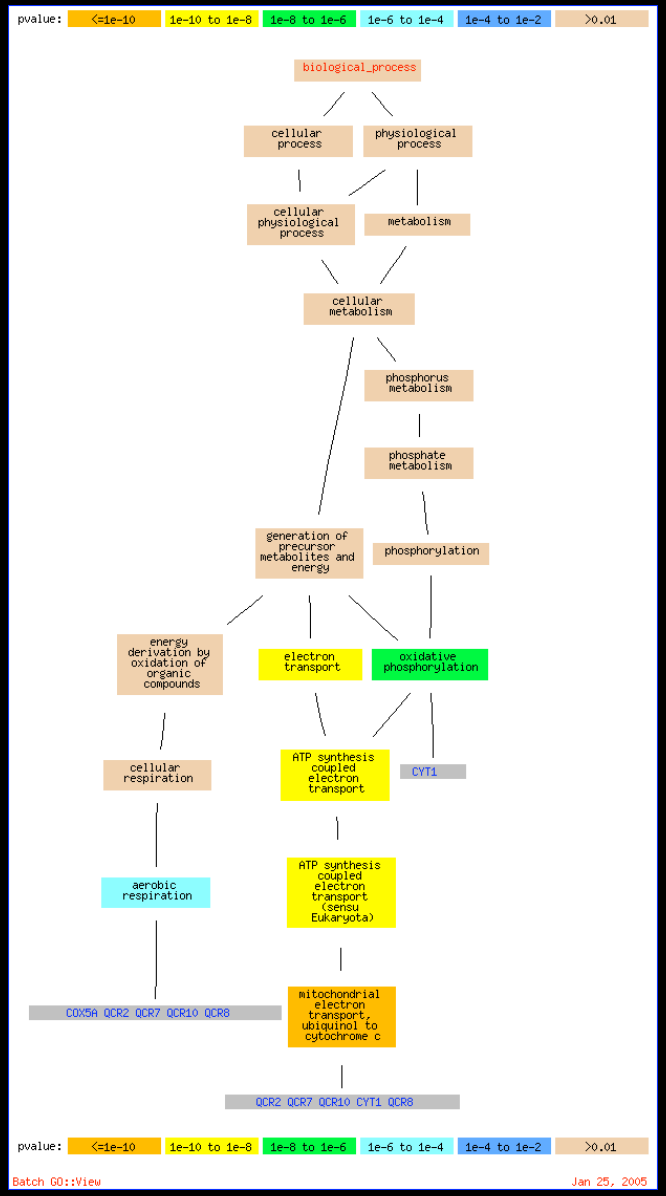
    *http://go.princeton.edu/*

# GO-Termfinder implementation

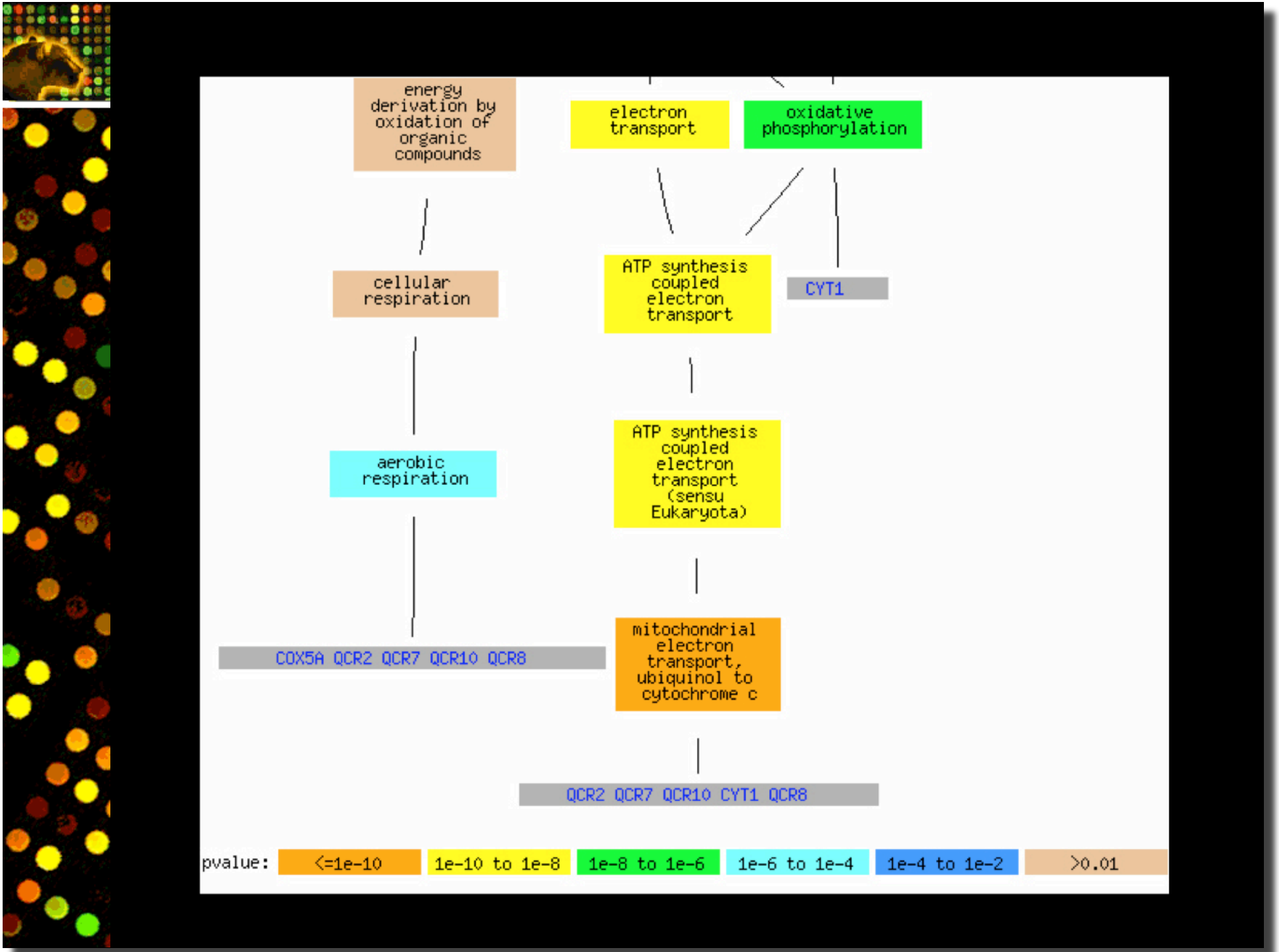
- ✓ Finds enrichment of GO terms used within a list of genes
- ✓ Utilizes code and algorithm described in: Boyle et al (2004) Bioinformatics
- ✓ Works for any species with GO annotations
- ✓ Publicly available over the web








1) Export cluster constituents from Treeview
2) Submit to GO Termfinder
3) Discover significant GO terms that are shared among a list of genes


energy derivation by oxidation of organic compounds
electron transport
oxidative phosphorylation
cellular respiration
ATP synthesis coupled electron transport
CYT1
aerobic respiration
ATP synthesis coupled electron transport (sensu Eukaryota)
COX5A QCR2 QCR7 QCR10 QCR8
mitochondrial electron transport, ubiquinol to cytochrome c
QCR2 QCR7 QCR10 CYT1 QCR8
pvalue:
<=1e-10
1e-10 to 1e-8
1e-8 to 1e-6
1e-6 to 1e-4
1e-4 to 1e-2
>0.01

# Acknowledgements


- PUMAdb, Lewis-Sigler Institute
  - http://puma.princeton.edu
  - Fan Kang, Laurie Kramer, Mark Schroeder, John Wiggins
- SMD
  - http://smd.stanford.edu
  - Catherine Ball, Gavin Sherlock, and staff