User Help: Tutorials and Workshops

• Help & FAQ
  – http://puma.princeton.edu/help/
  – http://puma.princeton.edu/help/FAQ.shtml

• Tutorials
  – 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.
Tumor Pool of Cell Lines

Differential labeling efficiency of dyes

Differential efficiency of hybridization over slide surface.

mRNA

cDNA

Bcl-2

Different amounts of starting material.

Different amounts of RNA in each channel

Differential labeling efficiency of dyes

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…

![Histogram of log10 Channel biases](graph.png)
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 $\log_{10}(\text{Green Intensity (median)})$ and $\log_{10}(\text{Red Intensity (median)})$

Plotting $\log_{10}(\text{Final Processed Green Intensity})$ and $\log_{10}(\text{Final Processed Red Intensity})$
The dye-normalized signal is calculated by multiplying the background-subtracted signal by the dye normalization factor:

\[ \text{DyeNormSignal} = \text{BGSubSignal} \times \text{DNF} \]

where \( \text{DNF} = \text{LinearDyeNormFactor} \), when linear dye normalization method is used

OR

where \( \text{DNF} = \text{LinearDyeNormFactor} \times \text{LOWESSDyeNormFactor} \)

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)

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

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

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.
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

| start | clustered | random1 | random2 | random3 |

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.

<table>
<thead>
<tr>
<th>NAME</th>
<th>ARRAY 1</th>
<th>ARRAY 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>GENE A</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>GENE B</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>GENE C</td>
<td>2.5</td>
<td>2.5</td>
</tr>
</tbody>
</table>
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.

<table>
<thead>
<tr>
<th>NAME</th>
<th>ARRAY 1</th>
<th>ARRAY 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>GENE A</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>GENE B</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>GENE C</td>
<td>2.5</td>
<td>2.5</td>
</tr>
</tbody>
</table>
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.

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
  – 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

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

Retrieving all genes/clones.
Using 36 data set(s).
Retrieving data by SUID.

1: ClimD.05
Retrieved 9207 feature(s)
2: ClimD.1
Retrieved 9066 feature(s)
3: ClimD.15
Retrieved 9085 feature(s)
4: ClimD.2
Retrieved 9208 feature(s)
5: ClimD.25
Retrieved 9198 feature(s)
6: ClimD.3
Retrieved 9206 feature(s)

- 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

Experiment Selection $\rightarrow$ Gene Selection and Annotation $\rightarrow$ Data Filtering Options $\rightarrow$ Data Retrieval $\rightarrow$ Gene Filtering Options $\rightarrow$ Gene Filtering $\rightarrow$ Clustering and Image Generation

- First, choose whether to partition the data.
  - Self Organizing Map (SOM)
  - No partitioning
    - If making a Self Organizing Map, specify the following:
      - X dimension: 
      - Y dimension: 
      - Randomize seed (if selected, a new random sequence will be generated, possibly resulting in a different SOM. If unselected, the same SOM will be generated each time the program is run.)

- Next, choose whether and how to cluster the data.
  - Genes:
    - Pearson Correlation (non-centered)
    - Euclidean Distance
    - Do no gene clustering
    - Pearson Correlation (centered)
  - Experiments:
    - Pearson Correlation (non-centered)
    - Euclidean Distance
    - Do no experiment clustering
    - Pearson Correlation (centered)

- Next, choose whether and how to generate a file of well-correlated genes. You can make a file that shows, for every gene, the other genes whose data are most closely correlated. The file can be downloaded and will have a .txtCor extension.
  - Generate a file of up to 20 sorted correlations above a threshold of 0.8 using Pearson's Correlation (non-centered)

- Last, choose some image generation options.
  - Contrast for image: 2.5
  - RGB color for missing data: 75% grey
  - Use blue/yellow color scheme.
  - Use red/green color scheme.
  - Show spot images
Clustering and Image Generation

• Cluster images
  • ratios
  • spots
  • both, adjacent
• Basic visualization applets
• Data files for client applications
• Deposit to repository
Deposit the results of an analysis
Re-enter the pipeline (filter, cluster)
Download locally
SVD Analysis
“Synthetic gene” transformation
Depositing Data into your Repository

- Deposit from data retrieval “pipeline”
- Upload from desktop

“Synthetic gene transformation”
“Synthetic” Gene Transformation

- 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

- **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."
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

The implementation of this Generic GO Term Finder depends on the GO-TermFinder software written by Gavin Sherlock and Shuai Weng at Stanford University, made publicly available through the GMOD project. For more information, please see Boyle et al, Bioinformatics (2004)

http://go.princeton.edu/
1) Export cluster constituents from Treeview

2) Submit to GO Termfinder

3) Discover significant GO terms that are shared among a list of genes
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