

# Acuity 4.0

MICROARRAY INFORMATICS SOFTWARE

## User's Guide

Part Number 2500-0144 Rev F April 2005 Printed in USA

Copyright 2005 Axon Instruments / Molecular Devices Corp.

No part of this manual may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, microfilming, recording, or otherwise, without written permission from Molecular Devices, Corp.

QUESTIONS? See Axon's Knowledge Base: <http://support.axon.com>



## **VERIFICATION**

THIS PROGRAM IS EXTENSIVELY TESTED BEFORE DISTRIBUTION.  
NEVERTHELESS, RESEARCHERS SHOULD INDEPENDENTLY VERIFY ITS  
PERFORMANCE USING KNOWN DATA.



# Table of Contents

<b>Chapter 1 Installation .....</b>	<b>1</b>
Computer Requirements.....	1
Minimum Client or Server Requirements .....	1
Recommended Server Requirements .....	2
Recommended Client Requirements.....	2
Installation.....	3
Client and Database.....	4
Client Only .....	4
Connecting the Security Key.....	4
Starting Acuity .....	5
<b>Chapter 2 Introduction.....</b>	<b>7</b>
Analysis.....	7
Visualizations.....	9
Database .....	10
<b>Chapter 3 Analysis Algorithms: Theory and Use.....</b>	<b>11</b>
The Fundamental Assumption .....	11
What Clustering Shows.....	12
What PCA Shows.....	13
PCA or Clustering? .....	14
Which Clustering Method Should I Use on My Data? .....	14
Using Cluster Analysis.....	15
Hierarchical Cluster Analysis.....	16
Correlation Coefficients .....	17

K-Means and K-Medians Cluster Analysis .....	24
Gap Statistic Analysis .....	26
Self-Organizing Maps Analysis .....	27
Gene Shaving .....	29
Algorithm Details .....	30
Principal Components Analysis .....	30
Normalization.....	32
Origins of variability .....	32
Normalization Methods.....	32
Linear, Ratio Normalization.....	34
Lowess Normalization.....	35
Robust Multichip Analysis (RMA) .....	38
Background Correction .....	38
Normalization.....	38
Summarization .....	40
Algorithm Complexity .....	41
References .....	42
<b>Chapter 4 Tutorial.....</b>	<b>47</b>
Introduction .....	47
Starting Acuity and Connecting To a Database .....	48
Starting Acuity .....	48
Connecting To A Database.....	48
Changing Your Password.....	49
Forgotten Passwords .....	49
Importing Microarray Data .....	49
The Acuity Interface.....	50
Common Tasks.....	50
Project Tree .....	51
Performing Analyses .....	63
Normalization.....	64
Creating a Dataset .....	66
Preparing a Dataset for Analysis .....	68
Finding Differentially Expressed Genes .....	71

Hierarchical Clustering .....	78
Visualizing Principal Components Analysis and SOMs Together.....	84
Web Links and Pathways .....	85
Reproducing the Published Results.....	86
Summary .....	86
Feedback .....	87
<b>Technical Assistance.....</b>	<b>89</b>
<b>Customer License Agreement .....</b>	<b>91</b>
<b>Licensing Notice.....</b>	<b>93</b>





# Chapter 1

## Installation

### Computer Requirements

Because Acuity is a client/server application, there will be slightly different requirements for client and server computers. In general, server computers that store data and run the database should have larger hard disks and faster hard disk access; depending on the analyses being performed, client computers need more RAM and faster processors.

### Minimum Client or Server Requirements

If you intend to run the Acuity client and server on the same computer, the following is a minimum configuration:

- IBM-AT compatible computer with a Pentium 1 GHz or faster processor
- Windows 98 or ME operating system (dual-boot systems are not recommended)
- 256 MB RAM
- Hard disk with 10 GB free (for data storage)
- CD-ROM drive

## 2 • Installation

- 1024×768 display system with 65K colors
- Internet Explorer 5.0 or higher
- USB port

### **Recommended Server Requirements**

Please discuss your server requirements with your computer vendor. Server specifications depend strongly on the number of simultaneous users to support. We recommend the following:

- Windows 2000 or 2003 Server operating system (dual-boot systems are not recommended)
- 768 MB RAM or more
- SCSI or Firewire hard disk with 60 GB or more free (for data storage)
- Back up device (*e.g.*, hard disk, tape)
- USB port
- Fast network card

We do not recommend using Windows NT because it does not support the use of USB dongles.

For managing multiple users, Windows 2003 Server has better security and stability than Windows 2000.

### **Recommended Client Requirements**

The following is a recommended configuration for a client computer:

- IBM-AT compatible computer with a 2.0 GHz or faster processor
- Windows 2000 or Windows XP operating system (dual-boot systems are not recommended)

- 768 MB RAM or more
- 1280×1024 display system with 16M colors
- Internet Explorer 5.0 or higher
- Fast network card

As with all performance measures, choose your system configurations based on the types of analyses that you perform.

For example, hierarchical clustering uses large amounts of RAM on the client side, while gene shaving uses large amounts of processor time.

If you are routinely opening large datasets (say, more than 200 microarrays) but using fast analyses like self-organizing maps or K-Means, a fast hard disk on the server and fast server access is more valuable than a fast processor on the client.

## Installation

For complete step-by-step installation instructions, please consult the accompanying Acuity installation document.

Acuity consists of both client software and database software.

Before installing the Acuity database you need to install Microsoft SQL Server 2000, which you purchase independently of Acuity. Once SQL Server 2000 is installed, proceed with installing Acuity.

The Acuity installation CD also includes MSDE, the free single-user version of Microsoft SQL Server 2000. You can use MSDE in place of SQL Server 2000 for a single-user installation. MSDE has a database size-limit of 2 GB.

To run the Acuity installer, double-click “setup.exe” on the Acuity CD.

Alternatively, from the *Start* menu select “Run”, and type “x:\setup.exe” where “x” is the drive letter of your CD-ROM drive.

The install program offers the following installation options:

### **Client and Database**

Select this option if you want this machine to be a database server and to run Acuity. You need to have SQL Server 2000 already installed and running.

### **Client Only**

Select this option if you want to install Acuity on this machine, but not the database. You will have to connect to a database on another machine. You do not need to have SQL Server 2000 installed either to install or to run Acuity in this mode.

For step-by-step instructions on the installation of Acuity, please refer to the accompanying installation documents.

## **Connecting the Security Key**

The hardware protection key ('dongle') that is shipped with Acuity can be attached to any computer on your network, but we recommend that you attach it to the server computer.

For a local, single-user installation, it is sufficient to attach the dongle to a USB port on the computer with Acuity.

For a multi-user installation on a separate server computer, you need to install network software to support the key. This is explained in the Acuity installation document.

## Starting Acuity

After the successful installation of the software, you will find the entry “Axon Laboratory” in your list of Programs in the Start menu, and there will be two new icons on your desktop. There is an “Acuity 4.0” entry in your Axon Laboratory group, and an icon on your desktop. Both of these shortcuts will start Acuity.



# Chapter 2

## Introduction

Acuity from Axon Instruments / Molecular Devices, is a fully featured microarray expression informatics software package that has the following features:

### Analysis

- Hierarchical clustering with many different similarity metrics.
- Self-organizing maps (SOMs) with many different similarity metrics.
- Order dendrograms with SOMs, PCA.
- K-Means and K-Medians with many different similarity metrics.
- Gap Statistic to estimate optimal number of K-Means and K-Medians clusters.
- Principal components analysis.
- Gene Shaving.
- Find similar expression profile with the following similarity metrics.
- Find similar expression profile to user-defined profile.
- Variable selection with diagonal linear and quadratic discriminant analysis.

- Robust Multichip analysis (RMA) of Affymetrix probe-level data.
- Import and display full annotation data in an unlimited number of columns.
- Import and export gene lists.
- Import and export datasets.
- Import chromosome data.
- Substance lists and associated colors.
- Union and intersection of lists.
- Normalization wizard, including ratio-based normalization, wavelength-based normalization, print-tip lowess normalization with options for centering and scaling data, normalization to time points and samples.
- Statistics calculated for replicate microarrays, including mean, median, coefficient of variation, standard deviation, maximum, minimum.
- Significance statistics (p-values) calculated by Two-Sample Student's t-Test or Mann-Whitney test, and corrections for multiple hypothesis testing by Bonferroni, Step-Down Bonferroni, Hochberg, Sidak, Step-Down Sidak, and Benjamini-Hochberg methods.
- Multiple group comparisons by one-way ANOVA.
- Support for dye-swap microarrays in datasets.
- Column arithmetic on any data column.
- Multiple column transformations on datasets (row and column centering and scaling).
- Image display and integration with data tables, scatter plots and all other visualizations in Acuity.
- Lasso selection on images.
- User-definable flag values.



- Scripting engine for customizable analysis through VBScript, JavaScript or ActiveX objects.
- Analysis queuing.
- Fully integrated with GenePix Pro.
- Web links for unlimited access to web-based databases, including pathways.
- Create datasets from completely general database queries across all microarrays and all annotations in the database.
- Construct ontologies from imported gene ontology information.
- Merge microarrays.
- Apply GAL file to microarrays.

## **Visualizations**

- Dendrograms.
- 2-D interactive plots.
- Animated interactive 3-D scatter plots.
- Chromosome visualization.
- Normalization Viewer shows unnormalized and normalized data in the same window in scatter plots or histograms.
- M v A plots, including lowess print-tip smoothing curves.
- Line graphs of any microarray parameter.
- Scatter plots of any GPR or other microarray data type, or any analysis data type, such as p-value or correlation coefficient.
- Color tables.

- Export any visualization as PDF, BMP or WMF.
- Export animated 3D scatter plots as AVI.

## **Database**

- Support for Microsoft SQL Server 2000 and Oracle 9.
- ODBC-compliant.
- Full client-server model for effortless local, LAN or remote TCP/IP access.
- Tools for creating and managing users and groups.
- Users with read only, read-write or lab head permissions.
- Advanced database search tools.
- Advanced database management tools, such as a database optimizer to rebuild database table indices.
- Organize substance annotations into warehouses and genomes.
- True copy and paste of microarrays in the database.
- Attach (import) any file type to a microarray or a dataset in the database.
- Database backup and restore utility (SQL Server only).
- Database Recycle Bin to permanently delete or restore deleted data.
- Compact database tool to minimize database size on disk.
- Universal text file import, including GenePix Pro 5.0 GPR files and Affymetrix CEL, CHP and CDF files.
- Includes Microsoft SQL Server 2000 Desktop Edition.

## Chapter 3

# Analysis Algorithms: Theory and Use

Acuity employs a number of advanced algorithms for microarray analysis. This chapter explains the uses of these analyses, their limitations, and how to interpret their results.

### The Fundamental Assumption

In microarray data analysis (more specifically, in time course experiments), we make one fundamental assumption:

Genes that are expressed together share common functions.

From this assumption, we infer the following, which is sometimes called ‘guilt by association’:

*We can suggest possible roles for genes of unknown function based on their temporal association with genes of known function.*

For experiments in which the microarrays are derived from different tissue samples, instead of the same sample at different times, we can formulate the guilt by association statement as:

We can categorize samples of unknown physiological state based on their association with samples of known physiological state.

The central analytical task, then, is to group together substances or microarrays based on the similarity of their expression profiles. This is, we want to reduce the

complexity of the data so that large-scale trends and structure are revealed. Once we have a sense of the large-scale structure, we can investigate the fine structure in these trends further.

There is a wide variety of well-known mathematical and statistical techniques that can be brought to bear on such data reduction problems. The two main methods used in Acuity are Principal Components Analysis (PCA) and cluster analysis.

Principal Components Analysis is a data reduction technique. It reduces the complexity of a dataset by deriving a small number of variables (components) from the data. The investigator then examines the behavior of substances on a small number of these components, instead of the behavior across many microarrays.

Cluster Analysis is a grouping technique. It reduces the complexity of datasets by partitioning data into a small number of sets. The investigator can then examine the behavior of each set, which is representative of the data in it, instead of the behavior of each substance or microarray.

## **What Clustering Shows**

Consider clustering a dataset of 6000 substances into, say, 16 clusters. You begin with 6000 different expression profiles, and you end up with 16 ‘representative’ expression profiles. Yet within each cluster there are quite marked differences among expression profiles. The clustering method ignores the differences. It effectively throws away 5984 expression profiles, and keeps 16. One hopes that the information thrown away is less useful than the information retained and highlighted. If you ask the clustering algorithm to find 17 clusters instead of 16, then suddenly some substances that were in the same cluster are in different clusters. Every clustering procedure tries to provide a summary of the dataset, but it does this by throwing away information.

What is an argument *for* the efficacy of clustering? If an organism has 6000 genes, and one does an experiment on the organism, the 6000 genes do not act independently. On the contrary, significant numbers of them are acting in concert.

## What PCA Shows

Principal Components Analysis provides a low-dimensional summary of the dataset. If you have a dataset that has three columns, you can think of the value in each column as being a coordinate in a dimension, and so each row has a position in 3-dimensional space. Substances close in that 3-dimensional space have similar expression profiles, while substances far apart have dissimilar profiles. However, because most datasets have significantly more than three columns, we use PCA to reduce the dimensionality so that the dataset is easier to visualize.

In almost any dataset, some dimensions (*e.g.*, the values of substances on some microarrays) contribute less to the overall variance in the sample than other dimensions. Biologically, we might say that there are some microarrays on which most features are over- or under-expressed, while on other microarrays most spots have very little change in expression.

One way of thinking about Principal Components Analysis is that it removes the microarrays (dimensions) on which there is not much happening, leaving only the dimensions that contribute most to the variance. Furthermore, it orders the dimensions from those contributing most to those contributing least to the variance in the dataset. The components that are graphed in the PCA Select Components dialog box are the remaining dimensions, transformed in a way that maximizes the difference between each dimension.

If one has  $n$  columns in a purely random dataset, each column explains  $(100/n)$  % of the variance of the dataset. When looking at a PCA result, therefore, the only significant components are those that explain more than  $(100/n)$  % of the variance. In the case of the seven Diauxic arrays, this number is  $100/7$  or about 14 %. Double-click on the PCA display to see the variance explained by each component.

What we look for in a PCA display are points clumped together. By being together, they have similar expression profiles; by being slightly separated from other points, they form a distinct group.

## **PCA or Clustering?**

The difference between Principal Components Analysis and clustering is that there is much less arbitrariness in PCA. While a clustering technique always has to make a somewhat arbitrary decision about which cluster to assign a substance to, Principal Components Analysis is more likely to produce an informative representation of the real structure of the data.

Why is clustering arbitrary? Fundamentally, each clustering technique puts similar substances together, so each clustering technique must decide on some mathematical measure of similarity. One reason why there are so many different clustering techniques is that there are many different measures of similarity. Furthermore, there is no sense of any one measure of similarity being ‘better’ than another. Different cluster techniques partition the data differently, and so all partitions are arbitrary.

The disadvantage of PCA is that it rarely partitions the data into distinct sets: there are few sets of substances in the PCA space that are completely separated from all other sets. But that is the point of PCA: it shows that clustering methods usually make arbitrary decisions about membership.

## **Which Clustering Method Should I Use on My Data?**

Users unfamiliar with clustering techniques usually want answers to the following questions:

- ‘Which clustering method should I use?’
- ‘How many clusters should I find?’
- ‘Which similarity metric should I use?’

Each of these questions presupposes the existence of a single best clustering method that will reveal all and only the intrinsic structure in the data.

There is no such method.

The way to use Acuity is to use all the clustering and data reduction methods together. By doing this, you yourself will quickly discover the patterns in the data. However, some metrics are more appropriate for some analyses than others.

For K-Means Analysis, the Euclidean Squared metric usually the most appropriate, as for that metric the cluster centroids are arithmetic averages of the points in each cluster. Minimizing the Euclidean Squared distance of the cluster's points to the cluster's centroid naturally gives the centroid as the arithmetic average.

On the other hand, for K-Medians Analysis the cluster centroids are the medians of the points in each cluster. Minimizing the City Block distance of the cluster's points to the cluster's centroid naturally gives the centroids as the median in this case.

## Using Cluster Analysis

Acuity includes both hierarchical and non-hierarchical clustering methods. Use these methods for different experimental tasks.

Use hierarchical clustering when you are interested in the relationship of each substance or array to every other substance or array. For example, if your experiment is an attempt to classify tumor subtypes from a large number of tissue samples, where there is one tissue sample per array, then you would use hierarchical clustering, because you want to identify the tumor subgroups, but you also want to see the degrees of similarity among the subgroups. For example, some tumor types might be subtypes of a more general tumor type.

You also use hierarchical clustering where you have little or no prior knowledge of how the data will be clustered, as hierarchical clustering does not set the number of clusters to form before the analysis begins.

A disadvantage of hierarchical clustering is that it clusters substances into a single structure and pairs each substance with one other, when several regulatory pathways may be present in biological systems and expressed substances can participate in more than one pathway.

Hierarchical clustering is a relatively slow method, and it requires a very large amount of computer memory (RAM) compared to the non-hierarchical techniques.

Use non-hierarchical clustering when you are interested in separating substances into distinct classes, but you are not as interested in relationships between the classes. For example, you might be interested in pathogen-induced expression across a genome, and you want to identify groups of genes by function. In such a case, the functions of different groups may not be related, so there is no sense in looking at the similarity of different groups.

As non-hierarchical clustering forces the data into a user-defined number of clusters, you also use it when you have some *a priori* idea about the number of clusters that you want to form. For example, you might think that the response to the pathogen occurs in three main phases (say, early, middle and late), and you want to see genes clustered into these three temporal groups.

Alternatively, you may perform a hierarchical cluster analysis to identify the number of main clusters in a sample, and then instruct the non-hierarchical clustering method to find that many clusters.

Gene Shaving is unique among clustering techniques because it groups together both positively and negatively correlated substances. That is, it ignores the sign of a correlation, and looks only at the shape.

Apart from Gene Shaving, non-hierarchical clustering is very fast and uses little memory. It is therefore suitable for large datasets (say, > 100 microarrays). Gene Shaving is unsuitable for very large datasets.

## Hierarchical Cluster Analysis

Hierarchical cluster analysis produces the familiar tree structures called dendrograms. The hierarchical nature of the tree means that clusters that are not linked together at one degree of similarity are linked together at a lesser degree of similarity. Eventually, all objects in the tree are linked together.



In a hierarchical cluster, the number of clusters is not set before the analysis: it is derived from the analysis.

To create a hierarchical cluster one must specify a similarity metric and a linkage method. The similarity metric is used to form data points into clusters, while the linkage method is used to join clusters to form a tree.

Similarity can be expressed mathematically in many different ways. Acuity employs three different classes of similarity metrics, based on correlation coefficients, distance measures and binary measures.

Binary measures tend to produce trees with much less structure than those produced by either correlation coefficients or distance measures. Large numbers of substances are grouped together at the same degree of similarity. This has a number of immediate advantages: the overall structure of the data is revealed, and the clustering is much quicker. Binary metrics are not very useful for expression studies where one is looking at continuously varying levels of expression. They may be useful in studies such as Comparative Genomic Hybridization (CGH) where one is looking for the presence or absence of genomic DNA.

## Correlation Coefficients

### Pearson

This is the familiar Pearson's correlation coefficient.

#### *Centered*

Includes the forced assumption that the mean of a row is zero (*i.e.*, the mean of the row is subtracted from each value).

#### *Absolute*

Uses the absolute value of the Pearson correlation. When using this method, correlated and anti-correlated genes are clustered together.

## Spearman's rho

Spearman's rho is similar to Pearson's correlation coefficient except that it is calculated on ranks rather than data values. That is, when calculating the correlation, the actual numbers don't matter, just their order within the set.

## Kendall's tau

Like Spearman's rho, Kendall's tau is a rank-based correlation coefficient.

Both Spearman's rho and Kendall's tau are superior to Pearson's correlation coefficient when there are significant outliers in the data.

## Distance Measures

Distance measures are based on common measures of physical distance. There are different metrics for continuous data and binary data.

### *Continuous Data*

#### *Euclidean Squared*

$$d_{ij} = \sum_k (x_{ik} - x_{jk})^2$$

#### *City Block (Manhattan)*

$$d_{ij} = \sum_k |x_{ik} - x_{jk}|$$

The City Block measure and the Euclidean measure give similar results, but the City Block is less affected by extreme outliers (as the values are not squared).

#### *Canberra*

$$d_{ij} = 0, \text{ when } x_{ik} = x_{jk} = 0;$$

$$d_{ij} = \sum_k |x_{ik} - x_{jk}| / (|x_{ik}| + |x_{jk}|), \text{ otherwise.}$$

The Canberra metric is self-standardizing. This metric can be unstable when there are many values near zero, which happens with log-transformed array data. It usually applies to non-negative data.

*Bray-Curtis*

$$d_{ij} = 0, \text{ when } x_{ik} = x_{jk} = 0;$$

$$d_{ij} = \sum_k |x_{ik} - x_{jk}| / (x_{ik} + x_{jk}), \text{ otherwise.}$$

Bray-Curtis is usually applied to non-negative data only.

*Soergel*

$$d_{ij} = 0, \text{ when } x_{ik} = x_{jk} = 0;$$

$$d_{ij} = \sum_k |x_{ik} - x_{jk}| / \max(x_{ik}, x_{jk}), \text{ otherwise.}$$

The Soergel metric applies to on-negative data. For binary data (0, 1 values) it gives the same result as the Jaccard Metric.

**Binary Data**

Metrics for binary data are generated by considering the table of co-occurrences of two substances over a set of  $p$  arrays:

		Substance $i$	
		Positive	Negative
Substance $j$	Positive	a	b
	Negative	c	d

From this table we can see that on  $a$  of the  $p$  arrays, both substance  $i$  and substance  $j$  are positive; for  $d$  of the  $p$  arrays, both substances are negative; for  $c$  of the arrays, substance  $i$  is positive and substance  $j$  is negative; and on  $b$  of the arrays, substance  $j$  is negative and substance  $i$  is positive. We also know that  $a + b + c + d = p$ .

From such a table one can generate a number of distance measures:

*Simple Matching*

$$d_{ij} = (b + c) / p$$

This is the ratio of mismatches to the total number of values.

*Jaccard*

$$d_{ij} = (b + c) / (a + b + c)$$

This is the ratio of mismatches to the total number, excluding joint absences.

*Bray-Curtis*

$$d_{ij} = (b + c) / (2a + b + c)$$

This is the ratio of mismatches to the total number (weighted to joint matches), excluding joint absences.

## **Some Background Information on the Metrics**

### ***Geometric Metrics***

Both the Euclidean Squared metric and the City Block metric have geometric interpretations. Loosely speaking, the Euclidean Squared Metric gives clusters that are sphere shaped, while the City Block metric gives clusters that are diamond shaped.

For K-Means Analysis, the Euclidean Squared metric usually the most appropriate, as for that metric the cluster centroids are arithmetic averages of the points in each cluster. Minimizing the Euclidean Squared distance of the cluster's points to the cluster's centroid naturally gives the centroid as the arithmetic average.

On the other hand, for K-Medians Analysis the cluster centroids are the medians of the points in each cluster. Minimizing the City Block distance of the cluster's points to the cluster's centroid naturally gives the centroids as the median in this case.

### ***Population Metrics***

The Bray-Curtis and the Canberra metrics were originally devised for measuring populations in different habitats. As such, these metrics are

most applicable when the data points have non-negative values (as would be the case for population counts).

For these metrics, the  $p$  elements in a point are the population counts for different species. Comparison of two points is in effect the comparing of the different population counts, species by species.

### ***Binary Metrics***

These metrics are applicable in the case of binary data, where for instance the presence of an attribute is signaled by a 1, and the absence by a 0.

The Simple Binary Matching metric simply records the number of times that the elements of one point differ from those of another. It gives equal weighting to the case of an attribute being present in both points, and the case of an attribute being absent in both points. For cases where we are more interested in the common presence of an attribute than in its absence, the Jaccard metric is the preferred metric. This is reflected in the fact that it does not use the  $d$  value in its calculation.

## **Linkage**

Different linkage methods can produce clusters with very different shapes. Choose a linkage method based on any *a priori* structure in the data, or experiment with different linkage methods.

### ***Average Linkage***

Average linkage forms clusters according to the rule: a case is joined to an existing cluster if it has the same level of similarity as an arithmetic average of the levels of similarities of all members of the existing cluster. In other words, the distance between ‘average neighbors’ determines the distance between clusters. This method tends to be equally good with data that are in long chains or in clumps.

### ***Single Linkage***

Single linkage forms clusters according to the rule: a case is joined to an existing cluster if it has the same level of similarity as at least one member of the existing cluster. In other words, the distance between ‘nearest neighbors’ determines the distance between clusters. This method tends to produce long chain-shaped trees.

### ***Complete Linkage***

Complete linkage forms clusters according to the rule: a case is joined to an existing cluster if it is within a certain level of similarity to all members of the existing cluster. In other words, the distance between ‘furthest neighbors’ determines the distance between clusters. This method tends to produce trees where clusters are clumped together, so it may not be appropriate if the data are in fact in long chains.

## **Clustering Substances and Arrays**

Acuity gives you the options of clustering substances, or arrays, or both substances and arrays in the one process. You would never cluster both substances and arrays in a time series experiment, because the arrays in such an experiment are already intrinsically ordered. In an experiment where each array corresponds to a tissue sample from a different patient, for example, you would cluster on both substances and arrays.

With any method that reduces the dimension of the data, however, finer structure can be lost. For example, suppose the expression of some subset of genes divides the samples in an informative way, correlating with the rate of proliferation of tumor cells, for example, whereas another subset of genes divides the samples a different way, representing the immune response, for example. Then methods such as two-way hierarchical clustering, which seek a single reordering of the samples for all genes, cannot find such structure.

## Optimizing Branch Swapping in Dendrograms with SOMs and Principal Components

For a tree containing  $n$  items, there are  $2^{k-1}$  different ways of ordering the tree that are consistent with the results of the clustering algorithm. Some orderings of trees reveal the tree structure better than other orderings. For this reason, you can manually swap the branches under a node by selecting the node, then selecting the *Swap Branches* command from the right mouse menu on the dendrogram.

Manual branch swapping is a trial-and-error process that is extremely time-consuming; on large dendrograms, it is practically impossible. However, in Acuity you can use the cluster order from a Self-Organizing Map analysis, or scores from a Principal Components Analysis to produce an optimal ordering of the tree.

Because Principal Components Analysis scores both substances and microarrays, you can sort both substance trees and microarray trees with PCA scores.

Self-Organizing Maps in Acuity cluster substances only, so if using a SOM to swap branches, you can sort the substances tree only. For best results with SOMs, sort trees with an  $n \times 1$  or  $1 \times n$  SOM, as SOMs of those dimensions produce a single unambiguous order.

### Algorithm Details

Hierarchical clustering in Acuity uses a well-known algorithm that has been optimized for speed on large data sets. In purely theoretical terms, hierarchical clustering scales for speed as  $n^2$ , where  $n$  is the number of rows (substances).

The algorithm proceeds differently depending on the amount of computer memory (RAM) that is available. If there is not enough memory to perform the entire calculation at once, the computational task is re-cast so that memory is never exceeded. In the latter case, the calculation is much slower than in the former.

## K-Means and K-Medians Cluster Analysis

K-Means and K-Medians cluster analyses fall into the category of non-hierarchical cluster analysis, along with Gene Shaving and Self-Organizing Maps.

K-Means clustering partitions the data into a set of mutually exclusive and exhaustive groups (that is, every observation is in one and only one group). The number of groups is chosen a priori. The benefit of using K-Means analysis is that instead of looking at the response of, say, 30,000 substances across arrays, we look instead at the response of a much smaller set of clusters (perhaps ten or so). It may be informative to examine which substances have been allocated to which cluster.

For large numbers of substances, K-Means is very much faster than a hierarchical cluster analysis. K-Means works by optimizing a quality criterion, generally involving the within-cluster sums of squares. Since the problem of allocating a large number of substances (~30,000) to a small number of clusters (~10) is a huge combinatorial task, the algorithm works with a set of heuristics. This means that we can be fairly confident of obtaining a reasonable solution, but have little chance of obtaining the ‘best’ solution. Since, however, the best solution is dependent on the optimality criterion, and there are no compelling reasons for choosing one criterion over another, the best solution is not really a meaningful target.

The consequence of these considerations is that there is no right or wrong solution using K-Means. There are many different variants of the algorithm, and they will be more or less useful under different types of pathological data. Users must never be uncritical in their acceptance of K-Means results, but the procedure will frequently show interesting patterns in data. Similar comments could be made about hierarchical clustering.

K-Medians is the same as K-Means, except that each cluster is approximated by the median of its members, rather than the mean.

### Algorithm Details

The algorithm for K-Means clustering is based on the original papers by Hartigan *et al.* (1975,1979) and later work by Linde *et al.* (1980). The basic idea of the algorithm is to begin by estimating  $n$  initial cluster centroids. The



elements of the data set are then assigned to the cluster with the nearest centroid, and the values of the centroids are updated according to the current elements in the cluster. The assignment and updating steps are iterated until the centroids only shift by some minimal amount between iterations.

Variations on the algorithm exist because of the different techniques possible for initializing the centroid values. In Acuity initialization is not done explicitly. Rather, the algorithm begins by computing one centroid of all elements to be clustered as a starting centroid. That centroid is then randomly perturbed slightly above and below its value and these two perturbed centroids become the initial values for a K-Means iterated procedure for estimating 2 clusters for the data. When those 2 cluster centroids have converged, they are used to initialize 4 (perturbed) cluster centroids, and so on, doubling the number of centroids at each step. This initialization process is called ‘splitting’. When the number of clusters desired is not a power of 2, the final splitting step involves using only a subset of perturbed centroids from the previous step.

By employing this splitting method, the initialization of the centroids at each split will be much more likely to provide good starting values for converging to a good local optimum (close to the global optimum), as opposed to a poor local optimum. The latter may occur if the centroids are initialized in a more random manner.

However, because the initial perturbations are random, K-Means cluster solutions are not entirely reproducible. That is, running the same analysis on the same dataset twice produces slightly different results. This is not an error or a drawback of the algorithm. Rather, it demonstrates the fundamental arbitrariness of any cluster solution. Cluster membership is always affected by assumptions that one makes in the implementation.

The K-Means and K-Medians algorithms in Acuity 4.0 use a similar set of metrics as are available for hierarchical clustering. For more information on the metrics, see their description above in the Hierarchical Clustering section.

For K-Means Analysis, the Euclidean Squared metric is usually the most appropriate, as for that metric the cluster centroids are arithmetic averages of

the points in each cluster. Minimizing the Euclidean Squared distance of the cluster's points to the cluster's centroid naturally gives the centroid as the arithmetic average.

For K-Medians Analysis the cluster centroids are the medians of the points in each cluster. Minimizing the City Block distance of the cluster's points to the cluster's centroid naturally gives the centroids as the median in this case.

## Gap Statistic Analysis

The Gap Statistic, proposed by Tibshirani *et al.* (2000), is a method for estimating the number of clusters in a set of data. It can use the output of any clustering algorithm, but in Acuity we restrict it to K-Means and K-Medians. The Gap Statistic compares the change in the within-cluster dispersion to that expected under a reference null distribution.

Tibshirani *et al.* offers two options for generating the reference distribution:

- Generate each reference feature uniformly over the range of the observed values for that feature.
- Generate the reference features from a uniform distribution over a box aligned with the principal components of the data.

The implementation of the Gap Statistic in Acuity uses (b) as the reference distribution, as this method takes into account the shape of the data distribution. Method (b) assumes that the sample data is column centered, so we have the requirement in our implementation that the data is column centered. After column centering a large number of data points are negative, and all of the binary K-Means and K-Medians metrics are incompatible with negative values. Therefore, you cannot use the Bray-Curtis, Jaccard, Simple Matching or Soergel metrics with the Gap Statistic. Apart from choice (b) for the reference distribution, the implementation in Acuity follows the computation as specified on page 6 of Tibshirani *et al.*

It is worth emphasizing that the optimal cluster size determined by the Gap Statistic is not the cluster with the largest Gap value. Rather, it is the smallest cluster whose Gap value is closer than one standard error to the Gap value of the next cluster. More formally, for a cluster size  $k$  and standard error  $s_k$  in the reference distribution, the optimal cluster size is:

$$\text{smallest } k \text{ such that } \text{Gap}(k) \geq \text{Gap}(k + 1) - s_{k+1}$$

The standard error  $s_k$  is displayed on Gap Statistic graphs in Acuity as error bars.

## Self-Organizing Maps Analysis

Self-Organizing Maps falls into the category of non-hierarchical cluster analysis, along with K-Means, K-Medians, Gene Shaving.

Self-Organizing Maps analysis is used to group substances with a similar pattern over arrays together. The inputs and outputs for a Self-Organizing Maps analysis are the same as for a K-Means or K-Medians cluster analysis.

Self-Organizing Maps are a simple amendment to K-Means analysis. The key addition to K-Means is that clusters are arranged on a two-dimensional grid, generated by two unobserved latent variables. Clusters are constrained to have a regular arrangement on that grid. The benefit provided by this is a simple representation of the similarity between clusters. Clusters showing similar profiles across substances occupy nearby slots on the grid. Clusters that are very dissimilar will occupy distant slots on the grid. This goes some way to dealing with the inherent ‘slop’ in cluster solutions, and represents the potential uncertainty in classifying a substance into one cluster or a similar cluster.

### Algorithm Details

The algorithm used for Self-Organizing Map analysis is based on the original algorithm developed by Kohonen (1990). The Self-Organizing Map algorithm is very similar to the basic K-Means algorithm: initialize  $n$  cluster centroids, assign elements to the closest cluster, update the cluster centroids, and iterate until the centroids are in stable locations. Self-Organizing Maps is a variant on

the K-Means algorithm in that the centroids are adjusted according to a weighting scheme after the update step. The centroid adjustment step is included at every iteration.

For Self-Organizing Maps the  $n$  centroids are initialized by randomly selecting  $n$  elements from the data set. If the clustering is on substances, then  $n$  substances are randomly chosen. Because the clusters are initialized randomly, if a Self-Organizing Maps analysis is run on the same dataset twice, the results will be slightly different. As with K-Means and K-Medians, this demonstrates the important fact that cluster membership is always somewhat arbitrary. When repeating a cluster analysis one looks for the substances that do not shift from one cluster to another.

If the number of clusters is assumed to be non-prime, then each cluster can be mapped to a location in a rectangular, two-dimensional latent (hidden) variable space. At each iteration of the algorithm, a cluster centroid is adjusted to be a weighted average of the neighboring centroids in latent variable space. The weights of the neighboring centroids are dependent on the number of elements in the respective cluster and the distance between the clusters in latent variable space. A ‘cooling schedule’ is employed so that the apparent distance between the centroids in latent variable space increases to infinity, so that the eventual influence of neighboring clusters tends to zero as the algorithm iterates.

The clustering results can be viewed on a plot of the clustering elements and groups in the latent variable space. This plot is important because the algorithm is constructed so that elements in neighboring clusters (in latent variable space) should be similar. For example, a data set containing two main classes of substances may be clustered (on substances) into 16 clusters using Self-Organizing Maps. Although the clustering results may suggest that the substances fall into many more than two groups, it may be found that the clusters occur in two general regions in latent variable space. For example, there may be 5 clusters containing substances in the top left of the latent variable space and 3 clusters containing substances in the bottom right of the latent variable space. This grouping pattern could only be discovered by plotting the clustering results in the latent variable space. Note that such a

grouping is invariant to a transposition in the two-dimensional latent variable space. That is, the regions would still appear distinct if the horizontal and vertical axes of the latent variable plot were reversed.

The Self-Organizing Maps algorithm in Acuity 4.0 uses a similar set of metrics as are available for hierarchical clustering. For more information on the metrics, see their description above in the Hierarchical Clustering section.

## Gene Shaving

Gene Shaving falls into the category of non-hierarchical cluster analysis, along with K-Means, K-Medians and Self-Organizing Maps.

Gene Shaving is a novel cluster analysis technique developed by Hastie *et al.* (2000), especially for expression analysis. Its aim is to identify groups of substances (genes) that have coherent expression and are optimal for various properties of the variation in their expression. The algorithm as implemented in Acuity is constrained to produce high-variance clusters, and high coherence between members of each cluster.

Gene Shaving differs from other cluster algorithms in a number of interesting ways:

- The clusters of substances are constructed to show large variation across the set of arrays. That is, they are likely to contain a strong differential expression signal.
- The clusters of substances are not exclusive. A substance may be allocated to more than one cluster.
- Cluster profiles are independent of each other.
- The sign of a substance's contribution to a cluster is potentially arbitrary. That is, a substance showing linear increase along the arrays is likely to be clustered with a substance showing linear decrease along the arrays. (You can also use the Absolute Pearson Correlation in Hierarchical Clustering to get both correlated and anti-correlated substances clustered together.)

Gene Shaving is relatively fast for a large number of substances, but the cost increases rapidly with the number of arrays. It is typically slower than K-Means. One constraint of Gene Shaving is that it cannot produce more clusters than there are arrays. In the extreme situation of only two arrays, at most two clusters are found.

Gene Shaving may on occasions reveal structure that is not apparent in more traditional cluster algorithms.

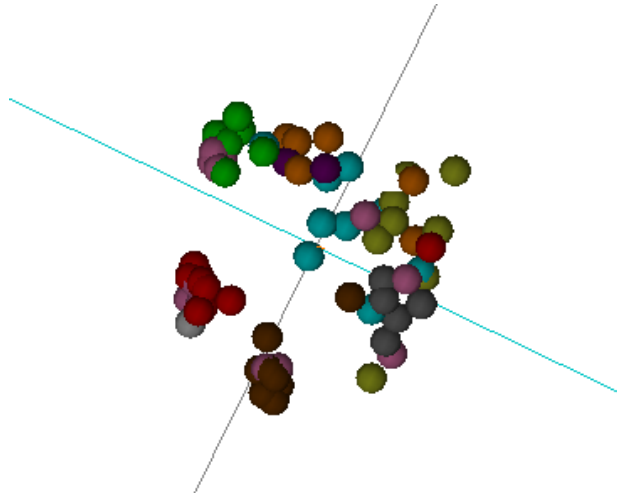
### **Algorithm Details**

See Hastie *et al.* (2000) for a complete description of the Gene Shaving algorithm.

## **Principal Components Analysis**

Principal Components analysis is used to produce a low dimensional summary of the data. It generates derived variables (the components) that are linear combinations of the results for different microarrays, and which have maximal variance (over substances) subject to an orthonormality constraint. Instead of looking at the expression profiles of 30,000 substances, we examine the response patterns on a handful of components.

Each component is a linear combination of the substances. We can examine two main quantities from Principal Components analysis: the component loadings which show the values of the derived components for each microarray, and the component scores, which show the coefficients of each substance on the components. Interpretation usually involves examination of both loadings and scores.



**Figure 1.** Principal Components Analysis of 11 cancer cells lines on 68 microarrays, from the Acuity PCA view.

The main advantage of Principal Components analysis over clustering methods is that it does not force us into a premature categorization of the data. Figure 1 shows a plot of microarrays derived from many different cancer cell lines. Some of the cancers, such as leukemia and melanoma, form into distinct clumps, while others are spread out and mixed in with a number of cell lines. This blurring of the distinctions between samples can be lost in a simple cluster analysis, but preserved in a scaling technique like Principal components.

### Algorithm Details

Principal Components Analysis is based on a standard decomposition in numerical analysis, and is described in books on multivariate analysis such as Mardia *et al.* (1979).

## Normalization

### Origins of variability

Comparing the data from different array experiments is a non-trivial task. Small variations in the many steps that produce an array image can make comparisons across arrays problematic. Variations can be due to differences in labeling efficiencies (dye and batch variations), chemical properties of different dyes, pin tips, slide batches, and scanner settings (*e.g.*, red/green channel settings, multiple scanners). Any of these variations can be corrected by normalization. No one normalization method will correct all types of variation. Choose a normalization method based on the known or expected sources of error, and the characteristics of the experiment. Validate the method empirically, for example by reversing the dye labeling to test a normalization method for channel balancing.

In the acquisition step of an experiment, one of the main contributors to variability between arrays is setting PMT values incorrectly, so that the total signal acquired in one channel is significantly different to the total signal acquired in the other. (We are assuming that when scanning a particular sample, the total signal in each channel is in fact the same.) In such a case ratio values may be biased towards one channel. To minimize this form of variability, you should perform preliminary scans to adjust the PMTs so that they are producing roughly the same response in both channels.

### Normalization Methods

In the analysis step of your experiment you can improve comparisons across many arrays by normalizing the data from each array. One method of normalization is based on the premise that most genes on the array will not be differentially expressed, and therefore the arithmetic mean of the ratios from every feature on a given array should be equal to 1. If the mean is not 1, a value is computed which represents the amount by which the ratio data should be scaled such that the mean value returns to 1. This value is the normalization factor.

Another method is to choose a subset of the features on an image as control features. All substances change expression levels under different conditions. Normalization control features should be selected based on their consistent



behavior in all experimental conditions used on your arrays, not on their historical use as “housekeeping genes” in other molecular biology techniques. For example, the control spots might be such that each is expected to have a ratio of 1. Hence the mean of the control features should be 1. Assuming that variations are uniform across the array, a single normalization factor can be calculated from these features and then applied to the whole array.

Both of these methods are linear and ratio-based, that is, they correct every feature on the array by the same multiplicative factor, and they correct intensities in order to balance ratio values.

The most common reason for normalizing microarray experiments is to correct for a scanner with an uncalibrated ratio channel. For a data distribution in which the average ratio value is different from 1.0, we can scale the intensity data in each channel with a linear transformation so that the ratio is equal to 1.0. Since PMT response is linear over a wide range of incident light, this type of data correction is equivalent to performing the experiment again with the PMTs calibrated. The linear transformation matches the instrument adjustments, and so we are justified in correcting the data.

A linear transformation to correct the balance between red and green across a whole slide is one method of normalization. There are a number of non-linear transformations that are also used to correct microarray data. A non-linear transformation corrects different spot intensities differently, so that, for example, low intensity features are shifted differently to high intensity features. These transformations are popular because if we do a scatter plot of red intensity versus green intensity, we often see the lower part of the scatter plot curving towards the red or the green, when we expect a straight line through the origin. A linear transformation shifts the distribution up or down without changing its shape; a non-linear transformation changes the shape of the distribution.

One of the more common non-linear normalization methods used on microarray data is lowess (locally weighted scatter plot smoothing). What imbalances in the data does Lowess normalization correct for? Commonly cited defects include the properties of the different dyes used (*e.g.*, different labeling efficiencies and

scanning properties) and experimental variability resulting, for example, from separate reverse transcription and labeling of the two samples.

Lowess normalization is somewhat problematic because the defects in experimental design or execution that are being corrected are not sufficiently well understood. There is no mathematical model of “the properties of the dyes” or their “labeling efficiencies” analogous to the mathematical model of the response of PMTs at different intensities. Therefore we recommend caution when using lowess normalization.

Whether or not you use lowess normalization will depend on your attitude towards the use of statistical techniques for data correction. Statistical techniques like lowess normalization can be used in one of two ways: to *diagnose* problems with experimental design and execution, or to *correct* those problems in software. If you are considering using lowess normalization, you need to ask yourself:

- Do I understand the physical basis of the defects that I am correcting?
- Could I perform this experiment with its systematic errors corrected and obtain the same results as I get from the lowess normalization of an experiment that has not had its systematic errors corrected?

If the answer to either of these questions is ‘No’, then it would be wiser to perfect your experimental technique to remove intensity-specific artifacts, than to modify your data without clearly understanding the reasons for the modification. If you do not understand the physical basis of what you are correcting, then you can have no more confidence in the corrected data than in the uncorrected data.

Having chosen a normalization method, it must be implemented in software.

### **Linear, Ratio Normalization**

Because ratios are not normally distributed, Acuity first takes the log of each ratio value when normalizing. The mean  $\bar{x}$  of a set of  $n$  ratios  $\{x\}$  is therefore:

$$\bar{x} = \exp\left(\frac{1}{n} \sum_{i=1}^n \ln x_i\right)$$

Ratio values less than 0.1 or greater than 10 can be excluded from the calculation as in GenePix Pro, as can features flagged *Bad*, *Absent* or *Not Found*. To do this, select the GenePix Pro Settings options in the Normalization Wizard.

Four-color normalization is done on a ratio-by-ratio basis. For example, if you choose to normalize so that the mean of the Ratio of Medians is set to 1.0, each Ratio of Medians data type that you have defined on the microarray is normalized independently.

To calculate the wavelength-specific normalization factors that are reported in the Normalization Viewer, the change to the ratio is distributed equally between the wavelengths, so one wavelength scales up by the square root of the ratio scale factor and the other scales down.

For example, suppose the mean of Ratio of Medians of 635/532 is 1.21. The square root of 1.21 is 1.1, so we set the scale factor for the 532 wavelength to 1.1, and the scale factor for the 635 wavelength to  $1/1.1 = 0.91$ . After applying these scale factors, the new mean of the Ratio of Medians is  $1.21 * 0.91 / 1.1 = 1.0$ .

Because normalization can scale up the data, it is possible for normalization to produce pixels with intensities greater than the hardware limit of 65535.

## Lowess Normalization

As described above, Lowess normalization is non-linear, so features with different intensities are normalized differently. The easiest way to see this effect is to look at a Lowess normalization in the Normalization Viewer with lowess curves displayed, such as in Figure 2.

The top pane shows the unnormalized data, and the bottom pane shows the normalized data. Both plots are M vs A (log ratio vs average intensity). The lines are print-tip lowess curves, *i.e.*, the lowess smoothing curves for the data from each block (the data from each block is smoothed separately, to account for variation among print tips). On the unnormalized data, the lowess curves show *how much the data will be normalized* by the specific lowess method chosen (*i.e.*, for the specific values of smoothing, centering and scaling, etc.) to produce the normalized

distribution in the bottom pane. That is, the lowess curves are generated from the lowess normalization options that you choose in the Normalization Wizard.

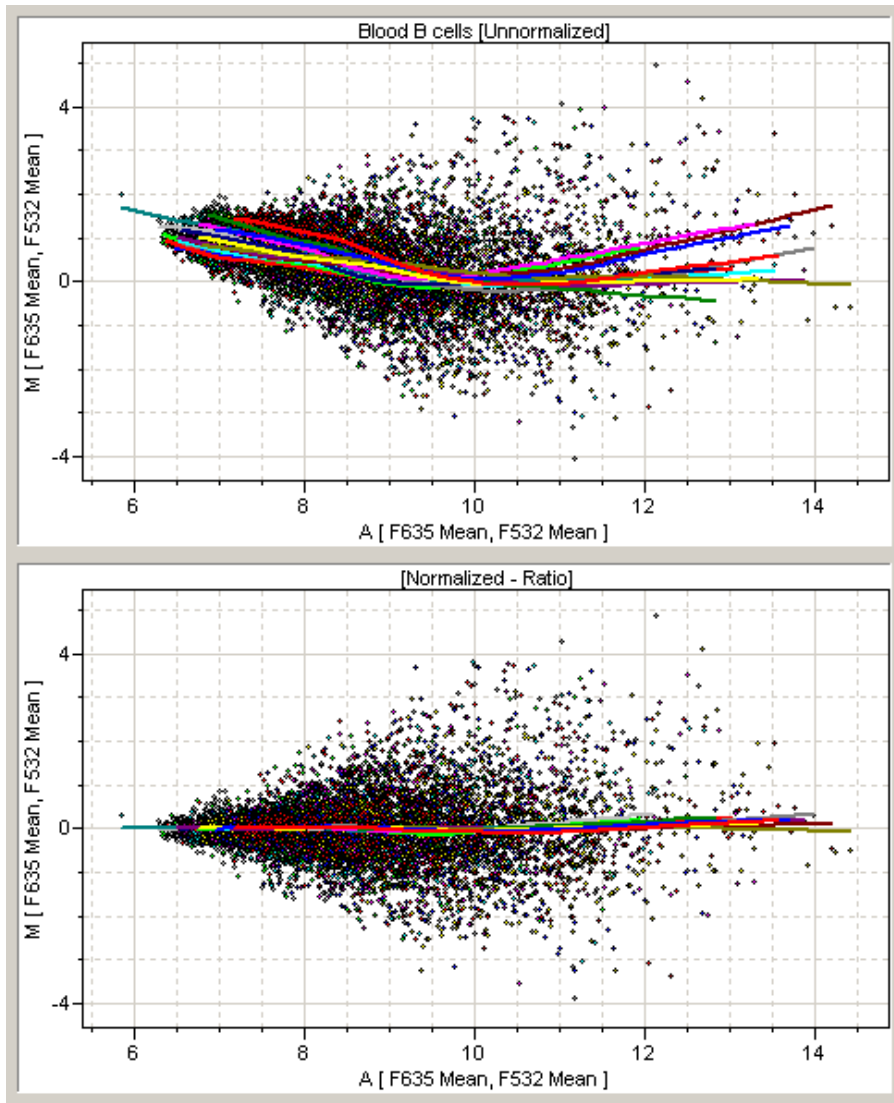
On the normalized data, the lowess curves show how much residual non-uniformity there is in the normalized data, again based on the options chosen in the Normalization Wizard.

In this particular normalization of an array with 16 blocks, you can see from the lowess curves that on all blocks low and high intensities are normalized much more strongly than mid-range intensities. You can also see from the smoothing curves in the bottom pane that there is very little non-uniformity left in the data after the smoothing is applied.

You might see residual non-uniformity if your original distribution is very strongly skewed due to, for example, a large number of control features on your array. If more than 10% of features are control spots that have ratio values very different to the rest of your data, you may have to adjust the number of iterations of the lowess smoother.

Lowess normalization is treated differently by the Acuity database than ratio normalization. When you do a linear ratio normalization, all data types for the microarray in the database that can be normalized are normalized. When you do a lowess normalization, because each data point on the array has to be individually changed, Acuity creates two new data types in the database, A and M. A is an average intensity constructed from the two intensity values that you select, while M is the lowess-normalized log ratio. To use lowess-normalized data in your downstream analyses, use the M data type.

Lowess normalization is described fully in Dudoit *et al.* (2002) and Yang *et al.* (2002).



**Figure 2.** Lowess normalization from the Acuity Normalization Viewer.

## Robust Multichip Analysis (RMA)

Robust Multichip Analysis is a method of taking Affymetrix probe-level signal intensities and performing the following operations:

- Background Correction
- Normalization
- Summarization

The techniques used are very different to those used in Affymetrix' software, so it is worth spending some time explaining them.

### Background Correction

The background correction algorithm operates on each array independently. Its main function is to determine an estimate of the background noise from the probe measurements, and to then subtract that noise from each probe value.

The background correction algorithm treats the measured probe values as a random variable  $S$ , which it then decomposes into a signal component  $X$ , and a noise component  $Y$ . Both  $X$  and  $Y$  are assumed to be independent random variables, with  $X$  being exponentially distributed and  $Y$  being normally distributed. Once we have determined the parameters of the  $X$  and  $Y$  distributions from the original data, we may estimate the true signal for each probe given the original measured value for each probe. In statistical terms, we compute the conditional expectation of  $X$  given  $S$ .

### Normalization

The two normalization algorithms provided as part of RMA in Acuity are Quantile Normalization, and Cyclic Lowess Normalization. Both attempt to normalize the chip data to a common baseline distribution of probe intensities.

## Cyclic Lowess Normalization

Cyclic Lowess Normalization is an extension of the standard method of normalizing two-color microarrays to the case where we have single channel data. For the case of two arrays, the Lowess process robustly fits a smooth trend function to the data, and then subtracts the trend from the original array data.

For more than two arrays the situation is slightly trickier—each pair of arrays has an associated trend function. Furthermore, each array is paired with every other array. The basis of the method is to repeatedly construct the trend curves for every pair of arrays, and to subtract each trend curve from all affected array pairs. Two or three cycles through every pair of arrays may be required for convergence.

Overall the Cyclic Lowess Normalization procedure robustly normalizes the array data while retaining significant quantitative information. The cost of this procedure is that it is demanding computationally. Lowess is a fairly expensive technique when applied to large data sets, and the cyclic part ensures it is applied some multiple of  $P^2$  times, where  $P$  is the number of arrays in the data set.

## Quantile Normalization

Quantile Normalization makes the probes on each array have the same distribution, by translating the cumulative distribution function of probes to a standard distribution function. The standard distribution is computed by ranking the probe values in each array, and then computing the average value for each rank. The original probe values are replaced by the average for the probe's rank.

For example, consider the case of several arrays, and consider further the minimum probe value on each array. The average of these minimum probe values is the estimate of the minimum value of the standard distribution. The minimum value in each array is then replaced by this average. The same procedure is done for the second-smallest value in each array, and so on, until all original values are replaced by their ranked equivalent value.

Quantile Normalization achieves statistical robustness by the use of the ranking process, and retains some of the original quantitative information via the averaging process. Since ranking is a relatively fast procedure (namely  $O(N\log_2 N)$  in the number of probes  $N$ ), the overall computational cost of Quantile Normalization is considerably lower than Cyclic Lowess Normalization.

## Summarization

The two Summarization algorithms provided in RMA in Acuity are Median Polish, and Robust Linear Model (RLM). Both algorithms estimate the expression value for each probe set by solving a statistical linear model for the probe set data. For a single probe set the model takes the form:

$$y_{i,j} = \mu + \alpha_i + \beta_j + \varepsilon_{i,j}$$

In this model  $i$  indexes the probe value (within the given probe set) and  $j$  indexes the Affymetrix chip. The  $y_{i,j}$  represents the logarithm of the probe value, the  $\alpha_i$  represents the probe effect and the  $\beta_j$  represents the chip effect. The  $\varepsilon_{i,j}$  represents statistical noise within the data, and finally  $\mu$  is the overall mean value of the probes across all of the chips and all of the probes in the given probe set. The summarized expression value of the probe set for chip  $j$  is simply  $\mu + \beta_j$ .

This model (or any equivalent alternative) may be solved by a variety of numerical methods. However, since outliers are common in Affymetrix data, it is best to use statistically robust methods that are relatively insensitive to outliers. Median polish and RLM are two examples of robust algorithms for solving this sort of linear model.

### Median Polish

Median polish is a very robust non-parametric algorithm for estimating the  $\mu$ ,  $\alpha_i$  and  $\beta_j$  terms using medians. It performs several sweeps over the data in which it progressively refines its estimates of the various terms. While fast, the non-parametric nature of the algorithm means that it potentially loses some information about the original statistical distribution.



## Robust Linear Model

Robust Linear Model (RLM) is an iteratively re-weighted least squares algorithm for solving the linear model. It achieves robustness by selectively down-weighting equations with large residuals. It is in a sense a compromise between a very robust non-parametric algorithm such as median polish, and a traditional non-robust least squares solution to the linear model. It potentially makes better use of the information within the data but at the expense of significantly greater computational effort.

## Algorithm Complexity

Large datasets can be fairly time consuming to run through RMA. In order to provide some guidance as to what to expect, we give the approximate order of complexity of each algorithm.

Algorithmic complexity is estimated in terms of the number of arithmetic operations the algorithm must perform; for example multiplication of two numbers constitutes a single operation. For some algorithms the exact behavior as a function of the number of arrays and number of probes is impossible to precisely characterize; the complexity quoted in the table below should be taken as a guide only.

In the table below the main variables are:

- P, the number of Affymetrix chips being processed as a batch.
- N, the number of Perfect Match probes on a single Affymetrix chip.

The “big-O” notation means that the dominant scaling factor in the complexity of the algorithm is of-the-order of the item in the parentheses. For example,  $O(P^2)$  implies that the number of operations performed by the algorithm scales quadratically in the number of arrays P.

Algorithm	Complexity
Background Correction	$O(PN \log_2 N)$
Quantile Normalization	$O(PN \log_2 N)$
Cyclic Lowess Normalization	$O(P^2 N \log_2 N)$
Median Polish Summarization	$O(PN \log_2 N)$
RLM Summarization	$O(P^2 N)$

## References

### General Multivariate Statistics

Hartigan, J.A. *Clustering algorithms*. New York: John Wiley & Sons, Inc., 1975.

Hartigan, J.A. & Wong, M.A. A K-means clustering algorithm: Algorithm AS 136. *Applied Statistics*, 28:126–130, 1979.

Hastie, T., Tibshirani, R., Eisen, M.B., Alizadeh, A., Levy, R., Staudt, L., Chan, W.C., Botstein, D., & Brown, P. ‘Gene shaving’ as a method for identifying distinct sets of genes with similar expression patterns. *Genome Biology* 1(2): research0003.1-0003.21, 2000.

Hastie, T., Tibshirani, R., Friedman, J. *The Elements of Statistical Learning: Data Mining Inference, and Prediction*. New York: Springer, 2001.

Linde, Y., Buzo, A. & Gray, R.M. An algorithm for vector quantizer design. *IEEE Transactions on Communications*, 28(1): 84–95, 1980.

Kohonen, T. The self-organising map. *Proceedings of the IEEE*. 78(9):1464–1480, 1990.

Mardia, K., Kent, J., and Bibby, J. *Multivariate Analysis*, Academic Press, 1979.

Tamayo, P., Slonim, D., Mesirov, J., Zhu, Q., Kitareewan, S., Dmitrovsky, E., Lander, E.S., Golub, T.R. Interpreting patterns of gene expression with self-organizing maps: Methods and application to hematopoietic differentiation. *PNAS* 96: 2907–2912, 1999.

### Normalization

Dudoit, S., Yang, Y.H., Callow, M.J., Speed, T.P. Statistical methods for identifying differentially expressed genes in replicated cDNA microarray experiment. *Statistica Sinica* 12:111–139, 2002.

Yang, Y.H., Dudoit, S., Luu, P., Lin, D.M., Peng, V., Ngai, J., Speed, T.P. Normalization for cDNA microarray data: a robust composite method addressing single and multiple slide systematic variation. *Nucleic Acids Research* 30(4): e15, 2002.

### **Robust Multichip Analysis (RMA)**

B. M. Bolstad, R. A. Irizarry, M. Astrand and T. P. Speed, “A Comparison of Normalization Methods for High Density Oligonucleotide Array Data Based on Variance and Bias”, *Bioinformatics*, 19, 185–193, 2003.

Irizarry, RA, Hobbs, B, Collin, F, Beazer-Barclay, YD, Antonellis, KJ, Scherf, U, Speed, TP, “Exploration, Normalization, and Summaries of High Density Oligonucleotide Array Probe Level Data”, *Biostatistics* Vol. 4, Number 2: 249–264, 2002.

Rafael. A. Irizarry, Benjamin M. Bolstad, Francois Collin, Leslie M. Cope, Bridget Hobbs and Terence P. Speed, “Summaries of Affymetrix GeneChip probe level data”, *Nucleic Acids Research* 31(4):e15, 2003.

### **Use of Algorithms for Gene Expression Analysis**

#### **Hierarchical Clustering (Rows Only)**

Iyer, V.R., Eisen, M.B., Ross, D.T., Schuler, G., Moore, T., Lee, J.C., Trent, J.M., Staudt, L.M., Hudson, J. Jr., Boguski, M.S., Lashkari, D., Shalon, D., Botstein, D., Brown, P.O. The transcriptional program in the response of human fibroblasts to serum. *Science* 283: 83–87, 1999.

Eisen, M.B., Spellman, P.T., Brown, P.O., Botstein, D. Cluster analysis and display of genome-wide expression patterns. *PNAS* 95: 14863–14868, 1998.

Spellman, P.T., Sherlock, G., Zhang, M.Q., Iyer, V.R., Anders, K., Eisen, M.B., Brown, P.O., Botstein, D., Futcher, B. Comprehensive identification of cell cycle-regulated genes of the yeast *Saccharomyces cerevisiae* by microarray hybridization. *Molecular Biology of the Cell* 9: 3273–3297, 1998.

#### **Hierarchical Clustering (Rows and Columns)**

Alizadeh, A.A., Eisen, M.B., Davis, R.E., Ma, C., Lossos, I.S., Rosenwald, A., Boldrick, J.C., Sabet, H., Tran, T., Yu, X., Powell, J.I., Yang, L., Marti, G.E., Moore, T., Hudson, J. Jr, Lu, L., Lewis, D.B., Tibshirani, R., Sherlock, G., Chan, W.C., Greiner, T.C., Weisenburger, D.D., Armitage, J.O., Warnke, R., Staudt, L.M., *et al.* Distinct types of diffuse large B-cell lymphoma identified by gene expression profiling. *Nature* 403: 503–511, 2000.

Perou, C.M., Sorlie, T., Eisen, M.B., van de Rijn, M., Jeffrey, S.S., Rees, C.A., Pollack, J.R., Ross, D.T., Johnsen, H., Akslén, L.A., Fluge, O., Pergamenschikov, A., Williams, C., Zhu, S.X., Lonning, P.E., Borresen-Dale, A.L., Brown, P.O., Botstein, D. Molecular portraits of human breast tumours. *Nature* 406: 747–752, 2000.

Ross, D.T., Scherf, U., Eisen, M.B., Perou, C.M., Rees, C., Spellman, P., Iyer, V., Jeffrey, S.S., van de Rijn, M., Waltham, M., Pergamenschikov, A., Lee, J.C., Lashkari, D., Shalon, D., Myers, T.G., Weinstein, J.N., Botstein, D., Brown, P.O. Systematic variation in gene expression patterns in human cancer cell lines. *Nature Genetics* 24: 227–235, 2000.

## **K-Means and K-Medians**

Aronow, B.J., Toyokawa, T., Canning, A., Haghghi, K., Delling, U., Kranias, E., Molkentin, J.D., Dorn, G.W. 2nd. Divergent transcriptional responses to independent genetic causes of cardiac hypertrophy. *Physiological Genomics* 6: 19–28, 2001.

Brar, A.K., Handwerger, S., Kessler, C.A., Aronow, B.J. Gene induction and categorical reprogramming during in vitro human endometrial fibroblast decidualization. *Physiological Genomics* 7: 135–148, 2001.

Soukas, A., Cohen, P., Socci, N.D., Friedman, J.M. Leptin-specific patterns of gene expression in white adipose tissue. *Genes & Development* 14: 963–980, 2000.

Tavazoie, S., Hughes, J.D., Campbell, M.J., Cho, R.J., Church, G.M. Systematic determination of genetic network architecture. *Nature Genetics* 22(3): 281–5, 1999.

## **Self-Organizing Maps**

Huang, Q., Liu, doN., Majewski, P., Schulte, leAC., Korn, J.M., Young, R.A., Lander, E.S., Hacohen, N. The plasticity of dendritic cell responses to pathogens and their components. *Science* 294: 870–875, 2001.

Saban, M.R., Hellmich, H., Nguyen, N.B., Winston, J., Hammond, T.G., Saban, R. Time course of LPS-induced gene expression in a mouse model of genitourinary inflammation. *Physiological Genomics* 5: 147–160, 2001.

Tamayo, P., Slonim, D., Mesirov, J., Zhu, Q., Kitareewan, S., Dmitrovsky, E., Lander, E.S., Golub, T.R. Interpreting patterns of gene expression with self-organizing maps: Methods and application to hematopoietic differentiation. *PNAS* 96: 2907–2912, 1999.

## **Principal Components Analysis**

Alter, O., Brown, P.O., Botstein, D. Singular value decomposition for genome-wide expression data processing and modeling. *PNAS* 97: 10101–6, 2000.

Hilsenbeck, S.G., Friedrichs, W.E., Schiff, R., O’Connell, P., Hansen, R.K., Osborne, C.K., Fuqua, S.A. Statistical analysis of array expression data as applied to the problem of tamoxifen resistance. *Journal of the National Cancer Institute* 91(5): 453–9, 1999.

Holter, N.S., Mitra, M., Maritan, A., Cieplak, M., Banavar, J.R., Fedoroff, N.V. Fundamental patterns underlying gene expression profiles: simplicity from complexity. *PNAS* 97(15):8409–14, 2000.

Raychaudhuri, S., Stuart, J.M., Altman, R.B. Principal components analysis to summarize microarray experiments: application to sporulation time series. *Pacific Symposium on Biocomputing*, 455–466, 2000.

Tamayo, P., Slonim, D., Mesirov, J., Zhu, Q., Kitareewan, S., Dmitrovsky, E., Lander, E.S., Golub, T.R. Interpreting patterns of gene expression with self-organizing maps: Methods and application to hematopoietic differentiation. *PNAS* 96: 2907–2912, 1999.



# Chapter 4

## Tutorial

### Introduction

This tutorial is in two main sections.

The first half of the tutorial is an extended tour of the Acuity interface, introducing the many different interface options in Acuity. It is worth going through this part of the tutorial at least once, so that you know just what is possible in the Acuity interface.

The second half of the tutorial, which begins with the “Performing Analyses” section, the guides you through a sample experiment similar to one of the first time series microarray experiments, DeRisi *et al.* (1997) “Exploring the Metabolic and Genetic Control of Gene Expression on a Genomic Scale” *Science* 278: 680–686. The data from this experiment is installed in the Acuity database as demonstration data. This part of the tutorial emphasizes the scientific aspects of using Acuity, and highlights the important data transformations and analyses with which you should be familiar.

This Tutorial should be read together with the following documents:

- The Acuity Online Help, which provides extensive documentation on the controls in every dialog box in Acuity.
- The Axon Guide to Microarray Analysis, which is on the Acuity installation CD, and which can be downloaded from the Axon web site.

There is more than one way to use Acuity. This tutorial is designed to introduce you to its major features and to explain their use. Equipped with this knowledge, you can be confident of exploring the program for yourself.

## Starting Acuity and Connecting To a Database

We assume that Acuity has been installed, together with a clean database. Please refer to Chapter 1 if you have not yet installed Acuity.

### Starting Acuity

When Acuity is installed, a shortcut is copied to your Windows desktop. To start Acuity, double-click this icon.

### Connecting To A Database

On starting Acuity, the *Welcome To Acuity* login dialog box is displayed. Use this dialog box to connect to a database before Acuity is opened.

To connect to a database:

- Configured databases are listed in the *Data Source* list.
- Select a database from the list, enter your user ID and password, and click *OK*.

On clicking *OK*, Acuity is opened, connected to your chosen database.



## Changing Your Password

To maintain the security of your data in Acuity you should change your password periodically. In particular, if Acuity is installed with a blank password you should change it immediately.

To change your password:

- Login to Acuity.
- Open the *Database / Users* dialog box.
- Select the user whose password needs to be changed, and click the *Properties* button.
- Select the *Change Password* button, and enter a new password.

## Forgotten Passwords

It is not uncommon for a user to forget the system administrator (“sa”) password. If this happens, you can change your system administrator password from the *Welcome to Acuity* login dialog box, so long as you are an administrator on the computer on which SQL Server is installed, and you are changing the password from that computer. To do this:

- Click the *Change Password* button in the *Welcome to Acuity* login dialog box.
- Type a new password in the *Change Password* dialog box.

## Importing Microarray Data

Our first task is to populate the empty database by importing microarray data.

Acuity can import any tab-delimited text file that has one row of column titles, and one column labeled ‘ID’ that contains the unique identifier of each substance. However, using the GPR file format has a number of advantages, such as enabling the import of GenePix Results JPG images.

To import GPR files:

- Select the *File / Import Microarrays* command.
- Navigate to a directory that contains GPR files. There are some sample GPR files on the Acuity installer CD in the Sample Data directory. You may like to import the GPR files in the Diauxic sub-directory, as we will use these later in the sample experiment.
- Select more than one file by holding down the <Ctrl> key when selecting.
- Click *Open*.
- The Import Microarrays dialog box is displayed.
- Select a folder on the Microarrays tab in the *Select location* pane. You can create new folders and rename them or existing folders with the *New Folder* and *Rename* buttons.
- If there are Results JPGs associated with your GPR files, they are imported automatically.
- Click *OK*.

Once imported, microarrays are listed on the Microarrays tab of the Project Tree on the left-hand side of the Acuity main window.

Note that you cannot add data to the root of the Microarrays folder.

## The Acuity Interface

The Acuity main application is made up of the Project Tree and data windows. You organize data in the Project Tree, and view data in data windows.

### Common Tasks

By default, the Common Tasks pane is docked on the left-hand side of the Acuity main window. It consists of a list of shortcuts to common tasks that you perform in

Acuity. Instead of having to find the tasks in the menus, they are organized by category. Simply click on the link, and the appropriate dialog box is opened for you.

Importantly, the tasks are organized in the order in which they should be performed. For example, data normalization must occur before any analyses are performed, and this order is reflected in the Common Tasks pane.

You can show and hide the Common Tasks pane with the *View / Common Tasks* command.

## Project Tree

The Project Tree is docked next to the Common Tasks pane on the left-hand side of the Acuity main window. It consists of three tabs, organizing three types of data in the database:

- The Microarrays tab lists all the microarrays (*e.g.*, GPR files) that have been imported to the Acuity database.
- The Datasets tab lists all datasets (sets of spots) and analysis results (*e.g.*, clusters) that you have created.
- The Quicklists tab lists all Quicklists (lists of substances) that you have created.

The Project Tree behaves like any other Windows tree. You can cut, copy, paste, drag and drop, rename and delete items in the familiar way.

To create a new folder in the tree:

- Right-click on the folder in which you want to create the new folder.
- Select *New Folder* from the right mouse menu.



To view and edit an item's properties:

- Select the item in the tree, for example a microarray.
- Choose *Properties* from the right mouse menu.





- The microarray's properties are displayed in the *Properties* dialog box and are editable.

### Microarrays Tab

Microarrays are represented by one of two different icons, to distinguish arrays with JPEGs from arrays without JPEGs:

- With JPEGs: 
- Without JPEGs: 

They can also be purple or orange, which denotes their normalization status:

- Unnormalized:  
- Ratio normalized:  

In addition to these icons, the Microarrays tab reports other information in columns next to the microarray names:

- The number of datasets in which the microarray is used.
- The number of features in the microarray.
- The normalizations that have been performed on the microarrays.
- The microarray parameters associated with the microarrays.

### Datasets Tab

A dataset is a set of spots. Typically, it consists of all the reliable spots from all the microarrays in a single experiment.

To create and open a dataset containing all the spots from a set of microarrays:

- On the *Microarrays* tab of the Project Tree, select the microarrays that you wish to analyze.
- From the right mouse menu, select *Create Dataset From Selection*.

Datasets can be as small or as large as you like: from several spots to all the spots in the database.

Datasets are the units on which most major Acuity analyses are performed (see below, “Performing Analyses”). Each time you do a cluster analysis, for example, the cluster result is listed in the Project Tree under the dataset from which it was created.

By default, the data type (*i.e.*, the GPR column displayed for each microarray) used in both microarrays and datasets is Log Ratio.

To change the data type of the current dataset:

- Select the *Configure / Current Data Type to Retrieve* command.
- Select a data type to view, and click *OK*.

Alternatively, the current data type is displayed in a list box at the top of the data window. You can select a new data type to retrieve from this list.

Creating a dataset from all the spots on a microarray, as described above, is not what we do when analyzing a real experiment. To do an actual data analysis, we use Acuity’s Query Wizard to extract just the reliable spots from our set of experimental arrays. This is described below, in the “Performing Analyses” section.

### **Quicklists Tab**

A quicklist is a set of substance and microarray names. It is a way of keeping lists of substances handy so that you can find them quickly in any microarray or dataset. That is, at any time the substances or microarrays in a quicklist can be selected in the Acuity interface.

There are two types of quicklists in Acuity:

- Global quicklists, which are organized in the Quicklists tab and which can be applied to any microarray or dataset.

- Dataset quicklists, which are saved with a dataset in the Datasets tab.

To create a global quicklist:

- Select some rows from the Table view of a microarray or dataset.
- Select the *Create Global Quicklist* command from the right mouse menu.
- Give the quicklist a name and click OK.

To highlight all the substances from a global quicklist in the current dataset:

- Open the dataset by selecting it on the *Datasets* tab and choosing *Open Selected* from the right mouse menu.
- Select the quicklist from the Quicklists tab and choose *Apply As Selection* from the right mouse menu.
- The substances in the quicklist are highlighted in the data window.

Quicklists are saved with colors associated with them, so that you can add a color to substances throughout the Acuity interface. This is particularly helpful when tracking substances from multiple quicklists.

To apply a quicklist color to substances in the current data window:

- Select a quicklist on the Quicklists tab.
- Select *Apply Colors* from the right mouse menu.

## Data Windows

When you open data from a microarray or a dataset it is displayed in a data window. Once you have opened data into a data window, you can open other windows displaying different views of the same data with the *Window / New Window* command, or you can open unrelated data in a new window with *File / Open Selected In New Window*.

If you use *Window / New Window*, the windows are linked, and so you can look at multiple views of the same dataset. For example, you might want to look at a

Self-Organizing Map and a Principal Components Analysis of the same dataset, and see how the clusters are plotted in the space of the principal components. Because the windows are linked, selecting a cluster also selects the same genes in the principal components analysis.

Data windows consist of two main panes: Table pane (top), which contains five tabs, and Views pane (bottom), which contains eight tabs.

### Table Pane

Although the top pane has five tabs—Data, Annotations, Web Links, Statistics and Advanced—you do have the option of viewing tabs side by side.

To do this, select *View / Split Substance Table* to split the top view into two. To continue splitting the view into more panes, keep selecting *Split Substance Table*. After splitting into four panes, selecting *Split Substance Table* returns the pane to its original configuration.

### ***Data Tab***

The Data tab in the Table displays a single data type from the current data source, *i.e.*, a single GPR column from each microarray.

**Important Note:** Replicates within microarrays, *i.e.*, spots with the same IDs, are automatically averaged in Acuity. To see individual feature values for replicate spots, use the Features tab in the bottom Views pane.

You can change the averaging method from the default mean by selecting a method from the list box at the top of the window.

The display of columns in the Data table is highly configurable:

- To hide data values and display colored cells only, use the *Data / Columns / AutoFit Color* command.
- To change the color scheme used in the Table, use the *Configure / Color Map* command.

- To remove all color from data cells in the Table, use the *Data / Color Map* command.
- To change column widths, use the *AutoFit* commands in the *Data / Columns* sub menu, or use the various *AutoFit* commands in the right mouse menu.
- To sort data values in a column, use the *Data / Sort Ascending* and *Data / Sort Descending* commands, or double-click on a column title.
- To group together discontinuously selected rows in the table, use the *Data / Group Selection* command.

Selections in the table are always linked with selections in the other panes, so selecting in one pane selects in all panes.

### ***Annotations Tab***

The Annotations tab displays substance annotations. Substance annotations are imported to Acuity from plain text files.

To import substance annotations:

- Select the *File / Import Other / Substance Annotations* command.
- Select an SDT file.

SDT files are tab-delimited text files that contain a row of column titles, a column of substance IDs, and other columns of annotations. There is a sample SDT file on the Acuity installer CD in the Sample Data \ Diauxic directory. You may like to import this file, as we will use it later in the sample experiment.

For more detail on the SDT file format, and where to obtain annotation data, see the “Import Substance Annotations” topic in the Online Help.

To show or hide substance property columns, use the *Configure / Columns / Substance Annotations* dialog box.



If you have gene ontology properties such as Component, Function and Process, you can create gene ontology quicklists from the substance properties pane using *Analysis / Quicklist and Coloring Operations / Create Quicklist From Substance Annotations*.

### ***Web Links Tab***

The Web Links tab displays URLs that link the genes in your dataset directly to online genomics databases. You can submit the various IDs and gene names that you have in the Annotations tab to online databases, and have the results of those queries displayed directly in the Report tab, or in an external web browser window.

To define a new web link:

- Open the *Configure / Web Links* dialog box.
- Click the *New* button.
- In the *Display Name* field, enter a name for the new web link (this can be anything).
- In the *URL* field, enter the URL for the web link. If you include a substance name or ID in square brackets anywhere in the URL (*e.g.*, at the beginning or the end of the URL), the name or ID is submitted directly to the web database. Many databases have a specific syntax for such automated queries; this is usually documented on the web site itself.

For the SGD database, for example, the URL with ID field is:

[http://genome-www4.stanford.edu/cgi-bin/SGD/locus.pl?locus=\[ID\]](http://genome-www4.stanford.edu/cgi-bin/SGD/locus.pl?locus=[ID])

You are not restricted to submitting the ID column; you can put any substance property column name in the square brackets and submit it to the web-based database. For example, if you have a column titled “GI” or “EC” numbers, you can use [GI] or [EC] in place of [ID], assuming the web-based database accepts those numbers.

Show, hide and re-order web links with substance properties, in the *Configure / Columns / Web Links* dialog box.

### ***Statistics Tab***

The Statistics tab displays basic statistics on columns in the current dataset. For example, if you have technical replicates (*i.e.*, replicate arrays) and you're interested in the standard deviation or coefficient of variation across those replicate arrays, then those statistics are displayed here.

As the Data tab displays already averaged replicates from within arrays, these statistics are slightly different to what you would get if you calculated statistics on all replicates on all arrays.

Statistics are calculated on selected columns, so to calculate statistics:

- Switch to the Data tab.
- Select the columns on which to calculate statistics by <Ctrl>+clicking on their titles in the Data tab.
- Switch to the Statistics tab.
- Press the <F5> button to calculate statistics.

To calculate statistics on replicate features within a single array, simply change the averaging method that is used on the Data pane. You can do this by selecting a method from the list box at the top of the Data pane, or by using the *Configure / Current Data Type to Retrieve* dialog box.

### ***Advanced Tab***

The Advanced tab displays various advanced statistical data types, such as p-values, principal components scores, and correlation coefficients, which have been calculated by Acuity statistical analyses. The use of the Advanced tab is discussed more below, under “Performing Analyses”.

## Views Pane

The Views pane contains the many graphical and other derived views of the data that are possible in Acuity.

### *Profiles Tab*

The Profiles tab graphs rows of data for selected substances against a chosen microarray parameter. For example, the Profiles tab is where you display time-course profiles of genes, or gene expression profiles across all samples from an experiment.

Because selections in any Acuity view are linked to all views, you can choose substances in the Data tab, for example, and then switch to the Profiles pane to see them graphed.

To select a microarray parameter apart from the current data type to graph on the X-axis:

- Select *Properties* from the right mouse menu on the Profiles tab.
- Choose a new parameter from the list in the *X-axis* group.

To zoom into any rectangular region of the graph:

- Select *Zoom Mode* from the *View* menu or the right mouse menu.
- Drag the region on the graph to be zoomed.

### *Plot Tab*

The Plot tab graphs any two columns of data from the top pane of a Data window. So for example you can:

- Plot data from any two arrays against each other, in order to see expression changes between them.
- Plot data from an array against an Annotation column, such as gene length or chromosome position.

- Plot data from an array against a Statistics or Advanced column, such as p-value (volcano plot).

To plot data:

- Switch to the Plot tab.
- Select the data to plot on the X-axis from the *X* list.
- Select the data to plot on the Y-axis from the *Y* list.
- Color the data by selecting a data type from the *Color by* list.

You can also draw two histograms on the Plot tab:

- Switch to the Plot tab.
- Select the data to histogram on the X-axis from the *X* list, and select the X Histogram button.
- Select the data to histogram on the Y-axis from the *X* list, and select the Y Histogram button.

### ***Visualizations Tab***

The Visualizations tab displays analysis results, such as:

- Dendrograms,
- Various non-hierarchical clusters and
- Principal components analyses.

See “Performing Analyses” below.

### ***Features Tab***

The Features tab is a “GPR viewer” for the column selected in the Table view. It functions very much like the GenePix Pro Image, Results and Scatter Plot tabs.

By default, the Features tab displays only a small number of GPR columns from the selected microarray.

To download more columns from the database:

- Select a microarray in the Table view of the top pane and switch to the Features tab.
- Use <F5> to retrieve data from the selected column.
- Select *Configure Columns* from the right mouse menu on the Features tab table.
- Select the GPR columns to display in the Features tab, and click *OK*.

To graph two data types against each other in the Features tab:

- Select all the substances to plot.
- Select the data to plot on the X-axis from the *X* list.
- Select the data to plot on the Y-axis from the *Y* list.
- Color the data by selecting a data type from the *Color by* list.

Because the Features tab lists individual spots (as opposed to spot averages for individual substances, as in the main Table view) if you select a substance in the Table view, all its replicates are selected in the Features tab.

The images displayed in the Features tab are the JPG images that are imported with GPR files. When you select rows in the Features tab table, they are automatically selected and zoomed on the image.

### ***Parameters View***

The Parameters tab displays all defined microarray parameters and their values. Like substance properties, microarray parameters can be imported from a tab-delimited text file.

To import microarray parameters:

- Select the *File / Import Other / Microarray Parameters* command.
- Select an MDT file. MDT files are tab-delimited text files that contain a row of column titles, a column of microarray names, and other columns of annotations. There is a sample MDT file on the Acuity installer CD in the Sample Data \ Diauxic directory. You may like to import this file, as we will use it later in the sample experiment.

You can edit microarray parameters manually: right-click on the tab and select *Properties* to open the *Microarray Properties* dialog box, where you can edit the parameters of the selected microarray.

To show or hide microarray parameter columns:

- Open the *Configure / Columns / Microarray Parameters* dialog box, or select *Configure Columns* from the right mouse menu on the Parameters tab.
- Select the microarray parameters to display, and click *OK*.

### ***Summary Tab***

The Summary tab reports a summary of the current data source and the substance selected in the data window. This can be useful for finding datasets and folders associated with microarrays.

### ***Report Tab***

You can script your own analyses of the current data source in the Report tab. Consult the Scripting Tutorial in the on-line Help for more information on Acuity scripting.

To open an example Acuity Report, click one of the hyperlinks on the default Report page, or use the *File / Open Report* command.

The Report tab is also where Acuity web links are displayed.

### ***Chromosome Tab***

The Chromosome tab draws genes on chromosomes for your chosen genome, and can also plot gene expression levels directly on the chromosomes.

In order to construct a map of a genome, you need to import a CDT file, a tab-delimited text file containing the chromosome coordinates for each gene in your genome of interest. See the Acuity online Help for more details on the CDT file format.

Only a small number of major genomes have been well enough annotated to support the Chromosome tab. Most of these are available from the UCSC genome browser download page:

<http://genome.ucsc.edu/cgi-bin/hgText>

UCSC supports the following genomes: Human, chimp, mouse, rat, chicken, *Fugu*, *Drosophila*, *C. elegans*, *C. Briggsae*, *S. cerevisiae*, SARS.

In addition, there are a number of scripts on the Acuity Report tab that generate CDT files for other organisms.

## **Performing Analyses**

In this part of the tutorial we will walk through the analysis of a typical microarray experiment. Not all steps outlined in the tutorial will be appropriate to every microarray experiment, but you should be able to learn enough to be able to understand the issues involved in microarray analysis.

We begin from the assumption that you have already imported GPR files. The Acuity installer CD contains two sets of sample GPR files in the Sample Data directory. The files in the Diauxic folder are from one of the first published time series microarray experiments, DeRisi *et al.* (1997) “Exploring the Metabolic and Genetic Control of Gene Expression on a Genomic Scale” *Science* 278: 680–686. The purpose of this

part of the tutorial is to reproduce some of the results in this study, which examines changes in yeast metabolism from fermentation to respiration. The data from these files is already in the Acuity demo database.

However, you may wish to import them again. To import the GPR files in the Diauxic folder:

- Select *File / Import Microarrays*.
- Navigate to the folder containing the Diauxic GPR files, and select them all.
- Click *Open*.
- In the *Select Destination* dialog box, click the *Create New Folder* icon to create a new folder in the tree, and then click *OK* to import the files.

## Normalization

The theory behind normalization is described in Chapter 3, so at this point we will discuss only the practical issues.

After importing your data into Acuity, the first thing you need to do is normalize it. If you have already created datasets on unnormalized microarrays, you need to delete the datasets before performing normalization. If you are working on the GPR files in the Microarrays tab of the demo database, you need to delete all the demo datasets from the Datasets tab.

To normalize a set of microarrays:

- Select the microarrays on the Microarrays tab of the Project Tree.
- Select *Normalization Wizard* from the right mouse menu.
- Select *Ratio-based* normalization (the default). By default, outlier features are not used to calculate normalization factors, but all features are normalized.
- Click the *Next* button.



- On the Summary page the normalization factors are displayed. These should be close to 1.0. If they are far from 1.0, for example greater than 1.3 or less than 0.7, then you should consider scanning your microarray again, as the PMT gain settings were not set very well.
- Click *Finish*.
- The normalization is performed, and the Normalization Viewer is opened. To see the effects of the normalization, you can switch to *Histogram* mode, and see how the histogram has been shifted to be centered on zero.

When your microarrays are ratio normalized, all relevant columns in the microarray are modified in the database. Therefore, whenever you analyze the data from that microarray, you are using the ratio-normalized data.

Let us also perform a Lowess normalization:

- Select the microarrays on the Microarrays tab of the Project Tree.
- Select *Normalization Wizard* from the right mouse menu.
- Select Lowess Slide Normalization.
- Click the *Finish* button.
- The normalization is performed, and the Normalization Viewer is opened showing before and after M versus A plots.

Unlike ratio-based normalization, when a Lowess normalization is performed a new data type (new column in the GPR file) is created in the database containing the Lowess normalized log ratio. By default, this is called “Lowess M Log Ratio”. We create a new data type because Lowess normalization is not reversible, and we do not want the original data in the database to be irreversibly affected.

To use the Lowess-normalized data for your analysis, you need to select Lowess M Log Ratio as your data type before beginning your analyses.

## Creating a Dataset

Datasets are the units of analysis in Acuity. Typically, a dataset consists of all the reliable data from a set of microarrays that together form an experiment.

There are two main reasons why we might create a dataset from only a subset of the available data, instead of from each feature from every microarray in an experiment:

- We remove unreliable data from the dataset. For example, we remove data points derived from slide defects such as smears.
- We remove uninteresting data from the dataset. For example, we may have control features used for normalization that are not needed for downstream analysis; or, we remove substances that do not show any interesting behavior in order to make the analysis task more tractable.

Let us concentrate on removing the unreliable data. This can be a treacherous task, due to the subjective nature of what counts as “good” data, the variability in data quality across microarrays, the lack of accepted standards for good data, and the problem of translating image-based defects into numerical conditions on array data types.

The easiest way to do this, and it is relatively easy, is to make a list of common feature and slide defects, and then translate them into numerical conditions on GenePix Pro and Acuity data types. Note that all these conditions should be applied to microarrays that have already been normalized.

We apply the quality control conditions and create a dataset in the Acuity Query Wizard:

- Select *Analysis / Create Dataset From / Query Wizard*.
- At the first step of the Query Wizard we need to select the microarrays from which we are going to create a dataset. If all the microarrays are in the same folder, then the easiest way to do this is to click the *Select From Folder* button, and select the microarrays there.

- We are constructing a query across a number of steps in this wizard, so we need to click the *Add To Query* button to add this first criterion to our query.
- Click *Next* to get to the next step of the wizard.
- The second step of the Query Wizard is where we apply quality control conditions on our spots. All GenePix Pro data types are listed in the Parameter column. Let's apply the following filters to our dataset. We include only the following spots:
  - Spots with only a small percentage of saturated pixels.
  - Spots that are not flagged bad, nor found or absent.
  - Spots with relatively uniform intensity and uniform background.
  - Spots that are detectable above background.

The first criterion requires us to construct two conditions:

F635 % Sat. < 3

F532 % Sat. < 3

The second criterion can be applied with this condition:

Flags >= 0

The third criterion can be applied with this condition:

Rgn R2(635/532) > 0.6

0.6 is a recommended threshold, but you could use 0.5 or 0.4 if too many features are failing, or 0.7 or higher if you want the filter to be more stringent.

The last criterion can be applied with the two signal-to-noise ratio data types:

SNR 635 > 3

SNR 532 > 3

However, because we want spots that have signal in at least one channel, we need to select both of these and then select the *Apply OR* button.

- Each of these conditions needs to be successively constructed using the lists at the top of this dialog, after which you click the *Add To List* button. When you have all four conditions in the *Combine Conditions* pane, select them and click *Add To Query*, after which your query should look like this:

```
(('F532 % Sat.' < 3) AND ('F635 % Sat.' < 3) AND ('Flags' >= 0) AND ('Rgn R2 (635/532)' >= 0.6) AND (('SNR 635' >= 3) OR ('SNR 532' >= 3)))
```

- Click *Next*.
- At this step of the Query Wizard we could filter further based on substance annotation; for example, we could select only the stress response genes. However, we are doing a global analysis, so we can leave this page blank and click *Next*.
- The *Evaluate* page of the Query Wizard reports the percentage of features that have matched our query. You will have more or fewer spots, depending on the quality of the arrays, and the thresholds that you chose in your query. If the percentage of features is acceptable, click *Finish*.
- Click *OK* in the dialog where you are asked to create or append.
- You need to give the dataset a name, and select a folder in the tree in which to save it, and then click *OK*.

The dataset is opened in a new window.

## Preparing a Dataset for Analysis

The Query Wizard performs only spot-specific filtering. Once we have a dataset, we may want to remove whole rows from it if they do not conform to further quality control criteria. We may also want to transform the data in other ways to prepare it for analysis. The commands to perform these operations are organized in the *Analysis* menu.

## Normalize to Column

The first transformation to consider is called *Normalize to Column* in Acuity. A common experimental design hybridizes the experimental sample with a pooled reference sample, and so the ratios that are measured on the microarray are ratios relative to the pooled reference. These are not biologically interesting. The biologically interesting ratios are ratios of sample to sample.

In a time course experiment where we have used a pooled reference  $r$ , the measured ratios on five microarrays are:

$$t_1/r \quad t_2/r \quad t_3/r \quad t_4/r \quad t_5/r$$

However, the biologically interesting ratios might be something like:

$$t_1/t_1 \quad t_2/t_1 \quad t_3/t_1 \quad t_4/t_1 \quad t_5/t_1$$

We transform the ratios from the first set to the second set using *Normalize to Column*, which basically performs a division (or subtraction for log ratio data):

- Select *Analysis / Normalize to Column*.
- Select the microarrays to normalize; typically, this will be all the microarrays in the dataset.
- Select the microarray to normalize to. In a time course experiment, typically this will be the time zero microarray.
- Click *OK*.

Notice that the dataset in your Dataset tree has acquired a sub-tree. This is because whenever we modify a dataset, the modification is recorded in the tree so that we have a record of how we have modified it. We can always remove dataset modifications by deleting their entries in the Datasets tree, and selecting *Data / Refresh Data* to retrieve the original data from the database.

## Combine Columns

You can use *Combine Columns* to average or otherwise combine technical replicates.

## Dye-swap Columns

You can use *Dye-Swap Columns* to create reciprocal ratios from dye-swap replicate arrays, so that they can be compared with arrays where the dyes are labeled conventionally.

## Sort Columns

By default, microarrays in a dataset are organized alphabetically. You can use *Sort Columns* to sort them into their experimental order, such as time order, or by sample type.

## Remove Selected Rows

So far we have been discussing dataset transformations.

Remove Selected Rows is an important command for performing dataset-level quality control: you can use it to remove whole rows that fail some row-specific property like:

- Fold change across arrays.
- Percentage of missing values.

Removing rows is a two-step process: you need to find the rows first, then you need to remove them. To find the rows, we use *Analysis / Find Specified Values*.

Let's remove rows that have fewer than 70% present values. These can exist in our dataset because spots failed the Query Wizard, or log ratios, for example, may be undefined because of negative ratios:

- Select *Analysis / Find Specified Values*.
- Check the *Present in at least* option, and enter 70 for the percentage.

- Check the NOT option at the bottom of the dialog.
- Click *OK*.

This query finds substances that are not present in at least 70% of microarrays.

To remove them, now select Analysis / Remove Selected Rows.

## Finding Differentially Expressed Genes

Once we have transformed and cleaned our dataset, we are ready to look for differentially expressed genes. There are many ways of identifying differentially expressed genes, so let's look at a few of them.

### By Fold Change

One way of quantifying differential expression in a dataset is to look at genes that have changed by a certain amount on a specified number of arrays. This is sometimes called a fold-change filter.

To find genes based on a fold-change filter:

- Select *Analysis / Find Specified Values*.
- Make sure that all options are unchecked.
- Assuming you are working with log ratio data, check the *Absolute value >=* option, and enter 2 in at least 2 microarrays.
- Click *OK*.

This query finds substances that have changed 4-fold (2-fold in log space) on at least 2 microarrays. You can see their profiles by switching to the Profiles tab.

To save these genes in a list with your dataset, right-click on the Data tab and select *Create Dataset Quicklist*. Your dataset acquires another tree, this time of quicklists, and the quicklist is saved there.

## By Statistical Significance

The fold-change method of finding differentially expressed gene is a rather blunt instrument. One adjusts the value of the fold change and the number of microarrays until one has a manageable number of genes.

A more objective way of quantifying differential expression in a dataset is to look at genes that have statistically significant differences in expression among groups of arrays. One statistical test we can do, for two groups, is a two-sample t-Test:

- Select *Advanced / One and Two-Sample Significance Tests*.
- Select the first test, *Student's t-Test, equal variances*.
- Click *OK* to move to the next step.
- In the next dialog box you have to specify the microarrays in each group. In the Diauxic demo data in the Acuity database, even though it is a time course, there are two groups of arrays: the first five and the last two. We can create two groups of microarrays by clicking the *Create from Microarrays* button. Select the microarrays in the first group, and create a group by moving them to the list on the right hand side. Do the second for the second group.
- Click *OK* to move to the next step.
- The next dialog reports the results of the t-test, as a sorted list of p-values. You can save all p-values to the Advanced tab by clicking the *Save* button.
- You can select all genes that pass a p-value threshold, say 0.001, by entering 0.001 in the *Select all substances* field and clicking select.
- Save these substances as a dataset quicklist by selecting *Create Dataset Quicklist* from the right mouse menu.
- Click *Close*.



You can view the expression profiles of the substances that you selected by switching to the Profiles tab.

If you have more than two groups in your experiments, for example from multiple treatments, you can use *Advanced / One-way ANOVA For Multiple Groups* to find the differentially expressed genes. It works the same as the t-Test, but on multiple groups.

One can look at the genes that are common between the fold-change filter and the t-Test:

- Select both quicklists that you created by holding down the <Shift> key while selecting them in the tree.
- Select *Create Intersection Quicklist* from the right mouse menu.

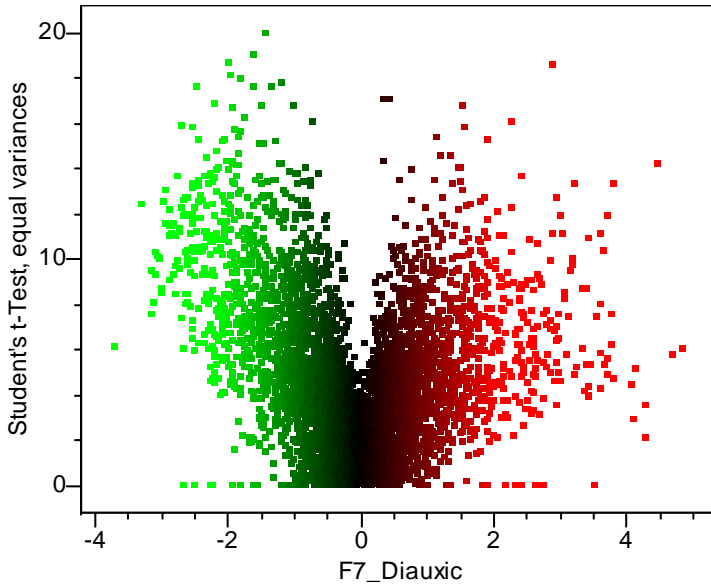
The intersection quicklist contains substances that are common to both quicklists. To see their profiles:

- Right-click on the intersection quicklist.
- Select *Apply as Selection*.
- Switch to the Profiles tab.

Another way of looking at the interaction between fold-change and statistical significance is to plot log ratio against p-value or  $-\log(p)$ . First, let's convert our p-values to  $-\log(p)$ :

- Assuming that you saved your p-values to the Advanced tab (see above), select *Advanced / Transform Advanced Columns*.
- Select *Student's t-Test* from the list of *Columns to transform*.
- In the X' field, select  $-(\log(x) / \log(2))$ .
- Click *OK*.

We use  $-\log(p)$  because the p-values are distributed over many orders of magnitude.



**Figure 3.** Volcano plot.

To plot log ratio against  $-\log(p)$ :

- Click on the Plot tab.
- From the X control at the top of the Plot tab, select *F7\_Diauxic*.
- From the Y control at the top of the Plot tab, select *Student's t-Test*.
- From the Color By control, select *F7\_Diauxic*.
- Select all genes in the dataset by clicking in the Data pane at the top and using  $\langle \text{Ctrl} \rangle + \langle \text{A} \rangle$ .

The resulting scatter plot is called a volcano plot, because the distribution often looks like a volcano erupting. The interesting thing about the volcano plot is that one can select features that have both large fold change, and are statistically significant: look at the genes that fall in the top left or top right parts of the distribution.

### By Principal Components Analysis

In principal components analysis entirely new variables (the components) are derived from the data, and substances are plotted in the space defined by these variables. The components can be thought of as corresponding to the dimensions in the data that account for the most variance.

To perform a principal components analysis:

- Ensure that you have a dataset open in the active data window.
- Select *Clustering / Principal Components Analysis*.
- Accept the defaults and click *OK*.

The *Cluster Progress* dialog box is displayed, reporting the progress of the task. When the progress reaches 100%, the clustering result is added to the Project Tree under the dataset that was clustered.

Double-click on the result in the Project Tree to display the result in the Visualizations tab. This is a three-dimensional scatter plot. You can rotate the axes by clicking the mouse, holding down the button and dragging.

Each of the axes can be thought of as representing an expression profile that explains variance in the dataset, where the first component explains the most variance. If you select Properties from the right mouse menu, the components, the amount of variance they explain and their profiles are displayed. Looking at the scatter plot, genes are plotted according to their similarity to the loadings profiles of the various components.

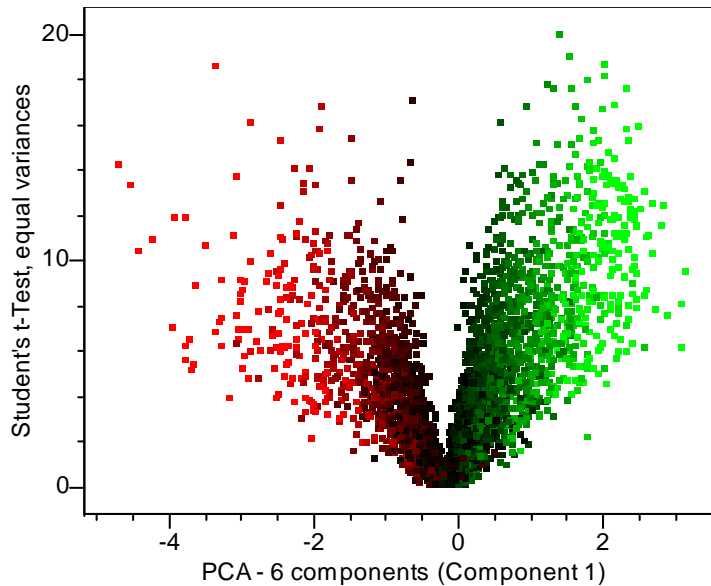
To select genes that score highly on the first principal component:

- Hold down the <Alt> key and drag a region around the points on the far right of the PCA scatter plot. They are selected in red.
- Switch to the Profiles tab, and you can see their profiles.
- If you do the same for points at the extreme left of the X axis, you see that they have the same profile, but reflected in the X axis.

To select these differentially expressed genes a little more rigorously:

- Right-click on the principal components analysis display and select *Component Scores*.
- Click the *Add* button to save them to the Advanced tab.
- On the Advanced tab you can sort them, and choose, for example, the top 20 and the bottom 20 genes in the list, and create a dataset quicklist from them.

As we did with the volcano plot above, you can plot principal component score on the X axis versus p-value on the Y axis, and obtain an even more informative volcano plot. Instead of looking at the log ratio on one microarray, one is looking at the first principal component, which represents the most variation in the dataset.



**Figure 4.** Volcano plot.

The advantage of Principal Components Analysis over a simple fold change filter, or even a t-Test, is that genes are organized both by variance and by profile:

- Genes far from the origin are changing more.
- Genes closer to the different axes are changing with different profiles.

So whereas a fold-change filter and a t-Test would group together genes with different profiles, Principal Components Analysis separates them. While this may sound like clustering, the difference between Principal Components Analysis and clustering is that in PCA each gene obtains a definite score on each component, so that genes are ordered with respect to each component.

We can now select all three quicklists that we have created, from the fold-change condition, t-Test and PCA, and create the intersection quicklist from these.

## Hierarchical Clustering

Hierarchical clustering, along other clustering methods, is a powerful way of looking at the global structure of a dataset.

To apply hierarchical clustering to a dataset:

- Ensure that you have a dataset open in the active data window.
- Select *Clustering / Hierarchical Clustering*.
- The Hierarchical Clustering dialog offers a number of similarity metrics and linkage methods. Accept the defaults.
- Click *OK*.

The *Cluster Progress* dialog box is displayed, reporting the progress of the task. When the progress reaches 100%, the clustering result is added to the Project Tree under the dataset that was clustered.

Double-click on the result in the Project Tree to display the result in the Visualizations tab.

### Using Dendrograms

The output of the hierarchical clustering algorithm is displayed in a visualization called a dendrogram. It is important to understand that the structure as displayed in the Visualizations tab is not a unique representation of the mathematical clustering operation. For this reason we are able to swap branches in the dendrogram.

To swap branches:

- Select a large branch with the mouse (so that one can see the obvious effects of the swap).
- From the right mouse menu select *Swap Branches*.

We are able to swap branches because swapping does not change the similarity of substances clustered under a node. Similarity is plotted along the bottom of the

dendrogram, and it is also reported in tooltips when you place the mouse over any part of the dendrogram.

We are interested in substances that are very similar, so we will want to zoom in to some small portion of the dendrogram.

To zoom a branch:

- Select a branch with a high degree of similarity by clicking it with the mouse.
- Select *Zoom Branches* from the right mouse menu, or use the  $\langle \text{Shift}+\text{Z} \rangle$  Hot Key.

Selecting the branch selected all the substances in the branch. These are also highlighted in the Table pane and in all View tabs in the bottom pane of the window.

To view all selected substances in the dendrogram together in the Table pane:

- Ensure that you have a branch selected on the dendrogram.
- Select *Data / Group Selection*, or use the  $\langle \text{G} \rangle$  Hot Key.

To see a graph of all selected substances:

- Ensure that you have a branch selected on the dendrogram.
- Switch to the *Graph* tab, and selected substances are graphed automatically.

No one hierarchical clustering method can tell you all there is to know about a dataset. Furthermore, the various similarity metrics and linkage methods introduce different assumptions to the process, so it is worth trying a number of methods just to see the results.

### ***Branch Swapping Dendrograms with PCA or SOMs***

As explained above, when doing a Principal Components Analysis each gene is given a score for each component. Acuity can use the resulting order to apply an optimal branch order to a dendrogram, thereby partially obviating the need to swap branches manually.

To do this, first perform a PCA by selecting *Clustering / Principal Components Analysis*.

Once it is finished, open the *Clustering / Hierarchical Clustering* dialog box. In the *Order substance branches by* field, the recently completed PCA is listed (along with any SOM that has been performed on the dataset). Select the PCA from the list, and the component to use, then click *OK* to start the hierarchical cluster.

Because Self-Organizing Maps (SOM) order their clusters on a 2-dimensional grid, one can use this ordering to swap branches. To use this feature, first perform a Self-Organizing Map analysis of the dataset. Typically, we perform a  $1 \times n$  or  $n \times 1$  SOM, as we are interested in ordering the tree in one dimension only.

### ***Color Map Only***

Another feature of hierarchical clustering is that you can create a completely unclustered color map of a whole dataset. This allows you to view the global structure of a dataset very quickly, for example to see which microarrays have the most missing values:

- Ensure that you have a dataset open in the active data window.
- Select *Clustering / Hierarchical Clustering*.
- Under Data to Process select *None (Color Map Only)*.
- Click *OK*.



This feature can be particularly useful when used together with the one described below.

### ***Match Expression***

If you want to find substances that have a similar expression profile to a selected substance in a dataset, you can perform a “quick cluster” of a data source using *Advanced / Match Expression on Mean*. This command sorts the data source by their similarity to the selected substance.

To use Match Expression:

- Select a substance (row) in the Table View in which you are interested.
- Select *Advanced / Match Expression on Mean*.
- The selected substance is now in the first row of the table, and the rest of the table is sorted from most correlated to the selected row, to least correlated to the selected row.
- You can save the correlation coefficients to the Advanced tab by clicking the *Save* button.

After doing a *Match Expression*, you can select the first 10 or 20 substances in the table and visualize them in a number of different ways:

- Switch to the Graph tab to see them graphed together. Select *Average Mode* from the right mouse menu to see an average trace of the selected substances.
- Select *Create Quicklist* from the right mouse menu to create a quicklist.
- Select *Create Dataset From Selection* from the right mouse menu to create a dataset of the spots.

You can also match the expression of a user-defined profile. To do this:

- Select *Advanced / Match Expression on User Profile*.
- In this dialog you can draw a profile in which you are interested. On clicking OK, the main Data table is sorted according to the correlation to that profile.

If visualizing the results with a Color Map Only dendrogram, select *Auto Sort Color Map* from the right mouse menu to sort the color map to the sorted order of the correlation coefficients.

### **Non-hierarchical Clustering**

Non-hierarchical clustering partitions substances into unrelated sets, so that membership of one set does not necessarily imply membership of any other set. K-Means, K-Medians and Self-Organizing Maps clusters are mutually exclusive, while the Gene Shaving algorithm allows substances to belong to more than one cluster.

K-Means, K-Medians and Self-Organizing Maps use essentially the same algorithm, except that K-Means and K-Medians produce an unordered list of clusters, while Self-Organizing Maps organizes the clusters on a 2-dimensional grid according to their relative similarity. It is therefore always more informative to use Self-Organizing Maps instead of K-Means or K-Medians.

To analyze a dataset using Self-Organizing Maps:

- Ensure that you have a dataset open in the active data window.
- Select *Clustering / Self-Organizing Maps*.
- Click *OK*.

The *Cluster Progress* dialog box is displayed, reporting the progress of the task. When the progress reaches 100%, the clustering result is added to the Project Tree under the dataset that was clustered.

Double-click on the result in the Project Tree to display the result in the Visualizations tab.

### ***Using Self-Organizing Maps***

The result of a Self-Organizing Maps cluster analysis is displayed in the Visualizations tab. Each cluster is represented as a thumbnail consisting of:

- A compressed color table containing the colored profiles of all the substances in the cluster.
- An average trace of the expression profiles of all the substances in the cluster.
- A title bar containing the number of substances in each cluster. The grayscale shade of the title bar represents the relative number of substances in each cluster, where white is the cluster with the most substances, and black is the cluster with the fewest substances.

The clusters are arranged on a grid according to their similarity to each other: similar clusters are close together, while dissimilar clusters are separated. Clusters diagonally opposite on the grid are essentially anti-correlated.

To show the color table only without the average profile, select *Color Map* from the right mouse menu.

To show the average profile only, select *Graph* from the right mouse menu.

To view the distribution of a Self-Organizing Maps cluster on a Hierarchical Cluster display:

- Select a cluster on the Self-Organizing Maps display.
- Double-click on a hierarchical cluster result in the Project Tree to open it.
- The substances from the Self-Organizing Maps cluster are highlighted in the hierarchical cluster.

To view a hierarchical cluster and Self-Organizing Maps together:

- Open a hierarchical cluster analysis result.
- Select *Window / New Window* to open a new window on the same data source.
- Double-click on a Self-Organizing Maps analysis result to open it into the new window.
- Select *Window / Tile Horizontal* to tile the windows.
- Expand the displays by clicking on each analysis result in turn and selecting *View / Expand*, or use the <Ctrl+E> Hot Key.

To find the cluster in which a substance is:

- Select the substance in the Table view.
- Its cluster is selected in the Visualizations tab.

## **Visualizing Principal Components Analysis and SOMs Together**

You will often find that the first principal component corresponds closely to a cluster produced by the Self-Organizing Maps algorithm.

To view Principal Components and Self-Organizing Maps together:

- Open a Principal Components Analysis result.
- Select *Window / New Window* to open a new window on the same data source.
- Double-click on a Self-Organizing Maps analysis result to open it into the new window.
- Select *Window / Tile Horizontal* to tile the windows.
- Expand the displays by clicking on each analysis result in turn and selecting *View / Expand*, or use the <Ctrl+E> Hot Key.

You should now have one analysis in one window, and a second analysis in the second window:

- Select a Self-Organizing Maps cluster, and the substances are highlighted in the principal components analysis display.
- Select a region in the principal components display, and the closest matching cluster is selected in the Self-Organizing Maps display.

You can use the principal components analysis display to investigate any cluster solution by selecting the cluster, and seeing the points that are selected in the principal components analysis display:

- Open a principal components analysis and a Self-Organizing Maps or other cluster analysis as described above.
- Select any cluster from a hierarchical or non-hierarchical method.
- The substances are selected in the principal components analysis display.
- If they form a tight set with no unselected substances in their midst, then the cluster forms a homogeneous group of substances.

If the cluster is mixed with unselected substances, then it may not sufficiently distinct from other clusters to make it an interesting group of substances. This may be because you have forced a non-hierarchical clustering algorithm to find too many clusters. Repeat the analysis with fewer clusters and check them again against the principal components.

Alternatively, select the cluster and switch to the Graph tab, where you can view all the expression profiles from the cluster together. This gives you another view of all the members of a cluster.

## **Web Links and Pathways**

We can view the gluconeogenesis pathway in Acuity by using a web link to connect to a pathway database, such as the Kyoto Encyclopedia of Genes and Genomes (KEGG).

GPM1 is one of the crucial gatekeeper genes in the Diauxic study. Right-click on the Gene column in the Substance Properties pane and select Sort Ascending to sort the substances alphabetically by gene name. Scroll down to GPM1, and click on its SGD hyperlink on the Web Links tab. The GPM1 page in the SGD database is opened in the Report tab.

You can also click on the hyperlink in the KEGG column of the Web Links tab to open the EC 5.4.2.1 page from the KEGG database. Scroll down here and click the In the Pathway field click the Map00010 link to open the gluconeogenesis pathway.

## Reproducing the Published Results

In Figure 5 on page 685, six main expression profiles and key gatekeeper genes are reported. We will try to find these profiles and hence discover these genes:

- To find the profiles, you need to examine your cluster solutions to see if any matches the profiles in Figure 5. You may need to perform more clusters, with various algorithms, before you find what you are looking for.
- The gene names in Figure 5 that are associated with the expression profiles are listed in the Gene column of the substance properties pane. Once you think you have found the right expression profiles, look in the Gene column to see if the genes are in it.

Finding the expression profiles, and identifying the genes, is not a simple step-by-step procedure. You may have to cluster your data using several algorithms, and with several different sets of options for each algorithm, before you find the profiles in Figure 5. This is exactly the way that microarray experiments are performed.

## Summary

This is the end of the tutorial. We hope that you enjoy using Acuity. Axon Instruments / Molecular Devices has put every effort into designing and constructing an application that will work with you to get your microarray informatics tasks done efficiently.

Remember to refer to the on-line Help and the rest of the Manual if you have any further questions about using Acuity. If you encounter a problem that you can't solve, don't hesitate to contact Technical Support.

## Feedback

Axon Instruments / Molecular Devices welcomes feedback on all its products. There is a Web page devoted exclusively to comments and suggestions on how to improve Acuity. If your computer is networked, select the *Help / Axon on the Web / Send Feedback* command to open a page on the Axon Instruments web site, from where you can send a message to Axon, or ask a question about Acuity.





# Technical Assistance

If you need help to resolve a problem, there are several ways to contact Axon Instruments / Molecular Devices:

## **World Wide Web**

[www.axon.com](http://www.axon.com)

## **Phone**

1 (800) 635-5577

## **Fax**

+1 (510) 675-6300

## **E-mail**

[axontech@axon.com](mailto:axontech@axon.com)

## **Questions?**

See Axon's Knowledge Base: <http://support.axon.com>



# Customer License Agreement

## **Customer License Agreement for Single User of Acuity 4.0**

This software is licensed by Axon Instruments / Molecular Devices Corp. (“MOLECULAR DEVICES”) to you for use on the terms set forth below. By opening the sealed software package, and / or by using the software, you agree to be bound by the terms of this agreement.

MOLECULAR DEVICES hereby agrees to grant you a non-exclusive license to use the enclosed MOLECULAR DEVICES software (the “SOFTWARE”) subject to the terms and restrictions set forth in this License Agreement.

## **Copyright**

The SOFTWARE and its documentation are owned by MOLECULAR DEVICES and are protected by United States copyright laws and international treaty provisions. This SOFTWARE may not be copied for resale or for bundling with other products without prior written permission from MOLECULAR DEVICES.

## **Restrictions on Use and Transfer**

You may not reverse engineer, decompile, disassemble, or create derivative works from the SOFTWARE.

## **Export of Software**

You agree not to export the SOFTWARE in violation of any United States statute or regulation.

## **Ownership of Software and Media (CD-ROM)**

You own the media (CD-ROM) on which the SOFTWARE is recorded, but MOLECULAR DEVICES owns the SOFTWARE and all copies of the SOFTWARE.

## **Product Improvements**

MOLECULAR DEVICES reserves the right to make corrections or improvements to the SOFTWARE and its documentation and to the related media at any time without notice, and with no responsibility to provide these changes to purchasers of earlier versions of such products.

## **Term**

This license is effective until terminated. You may terminate it by destroying the SOFTWARE and its documentation and all copies thereof. This License will also terminate if you fail to comply with any term or condition of this Agreement. You agree upon such termination to destroy all copies of the SOFTWARE and its documentation.

## **Limited Warranty and Disclaimer of Liability**

MOLECULAR DEVICES warrants that the media on which the SOFTWARE is recorded and the documentation provided with the SOFTWARE are free from defects in materials and workmanship under normal use. For 90 days from the date of receipt, MOLECULAR DEVICES will repair or replace without cost to you any defective products returned to the factory properly packaged with transportation charges prepaid. MOLECULAR DEVICES will pay for the return of the product to you, but if the return shipment is to a location outside the United States, you will be responsible for paying all duties and taxes.

Before returning defective products to the factory, you must contact MOLECULAR DEVICES to obtain a Service Request (SR) number and shipping instructions. Failure to do so will cause long delays and additional expense to you.

MOLECULAR DEVICES has no control over your use of the SOFTWARE. Therefore, MOLECULAR DEVICES does not, and cannot, warrant the results or performance that may be obtained by its use. The entire risk as to the results and performance of the SOFTWARE is assumed by you. Should the SOFTWARE or its documentation prove defective, you assume the entire cost of all necessary servicing, repair or correction. Neither MOLECULAR DEVICES nor anyone else who has been involved in the creation, production, or delivery of this SOFTWARE and its documentation shall be liable for any direct, indirect, consequential, or incidental damages arising out of the use or inability to use such products, even if MOLECULAR DEVICES has been advised of the possibility of such damages or claim.

This warranty is in lieu of all other warranties, expressed or implied. Some states do not allow the exclusion or limitation of implied warranties or liability for incidental or consequential damages, so the above limitations or exclusions may not apply to you.

## **U.S. Government Restricted Rights**

The SOFTWARE and its documentation are provided with RESTRICTED RIGHTS. Use, duplication or disclosure by the U.S. Government is subject to restrictions as set forth in subparagraph (c)(1)(ii) of The Rights in Technical Data and Computer Software clause at DFARS 252.227-7013, or subparagraphs (c)(1) and (2) of the Commercial Computer Software -Restricted Rights at 48 CFR 52.227-19, or clause 18-52.227-86(d) of the NASA Supplement to the FAR, as applicable. Manufacturer is Molecular Devices Corp., 1311 Orleans Drive, Sunnyvale, CA, 94089-1136, USA.

## **Governing Body**

This Agreement is governed by the laws of the State of California.

# Licensing Notice

Axon Instruments / Molecular Devices is not licensed under any patents owned by Oxford Gene Technology Limited (“OGT”), covering oligonucleotide arrays and methods of using them to analyze polynucleotides. The purchase of Axon Instruments / Molecular Devices products does not convey any license under any of OGT’s patent rights, including any right to make or use oligonucleotide arrays under OGT’s patents.

Customers may use Axon Instruments / Molecular Devices products to analyze oligonucleotide arrays according to OGT’s patented methods if those arrays have either been purchased from OGT’s licensed suppliers, or have been made by the customer under a license from OGT.

Please contact OGT to enquire about a license under OGT’s patents at [licensing@ogt.co.uk](mailto:licensing@ogt.co.uk) <<mailto:licensing@ogt.co.uk>>.

USE OF THIS INSTRUMENT WITH MICROARRAYS MAY REQUIRE A LICENSE FROM ONE OR MORE THIRD PARTIES THAT HAVE PATENTS RELEVANT TO SUCH USE. AXON INSTRUMENTS / MOLECULAR DEVICES DOES NOT SUGGEST OR PROMOTE THE USE OF THIS INSTRUMENT IN A MANNER THAT INFRINGES ON THE PATENT RIGHTS OF A THIRD PARTY. YOU ARE ENCOURAGED TO EVALUATE WHETHER A LICENSE IS REQUIRED FOR YOUR SPECIFIC APPLICATION OF THIS INSTRUMENT. COMPANIES THAT HAVE INTELLECTUAL PROPERTY RIGHTS IN THE POTENTIAL FIELD OF APPLICATION OF THIS INSTRUMENT INCLUDE WITHOUT LIMITATION, AFFYMETRIX, INC. (“AFFYMETRIX”), AGILENT, AND OXFORD GENE

TECHNOLOGY. THIS INSTRUMENT HAS NOT BEEN LICENSED OR APPROVED FOR DIAGNOSTIC APPLICATIONS.

THE USE OF THIS INSTRUMENT IN CONNECTION WITH MICROARRAYS MAY BE WITHIN THE SCOPE OF PATENTS HELD BY AFFYMETRIX. TO THE EXTENT THAT AFFYMETRIX PATENT RIGHTS ENCOMPASS THIS INSTRUMENT OR ITS USE, AFFYMETRIX HAS GRANTED A LIMITED PATENT LICENSE FOR RESEARCH USE ONLY AND NOT FOR USE IN DIAGNOSTIC PROCEDURES. SUCH LICENSE, IF APPLICABLE, IS LIMITED TO USE OF THIS INSTRUMENT WITH SPOTTED MICROARRAYS SEPARATELY LICENSED BY AFFYMETRIX. NO LICENSE IS CONVEYED, BY IMPLICATION, ESTOPPEL OR OTHERWISE, TO USE THIS INSTRUMENT WITH MICROARRAYS MADE USING *IN SITU* OR PHOTOLITHOGRAPHIC SYNTHESIS. NO OTHER LICENSE IS CONVEYED, BY IMPLICATION, ESTOPPEL OR OTHERWISE, UNDER ANY AFFYMETRIX PATENT OR OTHER INTELLECTUAL PROPERTY RIGHT.

This instrument is licensed by Affymetrix under the following patents: U.S. Patent Nos. 5,578,832; 5,631,734; 5,834,758; 5,936,324; 5,981,956; 6,025,601; 6,141,096; 6,171,793; 6,185,030; 6,201,639; 6,207,960; 6,218,803; 6,225,625; 6,252,236; 6,262,838; 6,335,824; 6,403,320; 6,403,957; 6,407,858; 6,472,671; 6,490,533; 6,545,264; 6,597,000; 6,643,015; and 6,650,411.