Journal of Data Science **5**(2007), 151-182

# Exploring Gene Expression Data, Using Plots

Dianne Cook, Heike Hofmann, Eun-Kyung Lee,
Hao Yang, Basil Nikolau, Eve Wurtele
*Iowa State University*

*Abstract*: This paper describes how to explore gene expression data using a combination of graphical and numerical methods. We start from the general methodology for multivariate data visualization, describing heatmaps, parallel coordinate plots and scatterplots. We propose new methods for gene expression data analysis using direct manipulation graphics. With linked scatterplots and parallel coordinate plots we explore gene expression data differently than many common practices. To check replicates in relation to treatments we introduce a new type of plot called a "replicate line" plot. There is a worked example, that focuses on an experimental study containing two two-level factors, genotype and cofactor presence, with two replicates.

*Key words:* Alpha-blending, data visualization, dendrogram, direct manipulation graphics, gene expression microarray, heatmap, interactive graphics, linked plots, multivariate data, parallel coordinate plots.

## 1. Introduction

Gene expression analysis is a relatively new application for statisticians. It has arisen from new technology that uses image analysis on microarray chips to measure individual gene expression for whole (or almost whole) genomes. There is a huge expectation that it will lead to a better understanding of life, and thus it has fueled intense activity by many researchers in diverse disciplines - biologists, computer scientists, bioinformaticists - to harness this new technology. As with any new field of research the dizzying pace of development has created some confusion, forward advances and backward disappointments, and new but often immaturely developed methods. There is now a growing maturity to the field and a growing understanding of the primary purposes of gene expression studies, partly due to the increased involvement of statisticians. But it is still far from a fully-fledged discipline, and new issues continue to arise with the technology, analytical methods, software, and related data and information.

This paper is about the surprises that may arise by plotting gene expression data, and how plots can reveal inadequacies in models and suggest ways to

improve the models. When analyzing locally collected data, using conventional ANOVA methods, we found that the results didn't match findings that were obvious from data plots. We describe why this happened, and why it happens generally for similar types of data analyses, and the implications for ANOVA, $p$-values, false discovery rates and filtering of gene expression data.

It will be helpful to start with a review of the state of graphical methods for multivariate analysis, with an emphasis on gene expression data. There are some commonly used graphics for gene expression data which are fraught with problems, and there are some less commonly used methods that are especially helpful. This paper discusses why this is, and ways to improve them. It is also describes direct manipulation and linked plots that are extremely useful yet somewhat under-utilized.

The paper is organized into two parts. This first part is a review of multivariate graphical methods for gene expression data. The second part places the abstract methods in the context of new data from an experiment on Arabidopsis plants measured using Affymetrix GeneChips, and the analysis, and insights about modeling, and the role of interactive graphics. The last section is a discussion.

### 1.1 Gene expression data and the questions it yields

Gene expression data is multivariate data. The basic form of gene expression data is a matrix of numerical values, where $n$ denotes the number of genes being analyzed and $p$ is the number of chips used for the experiment:

$$\mathbf{X} = \begin{bmatrix} x_{11} & x_{12} & \dots & x_{1p} \\ x_{21} & x_{22} & \dots & x_{2p} \\ \vdots & \vdots & & \vdots \\ x_{n1} & x_{n2} & \dots & x_{np} \end{bmatrix}_{n \text{ genes} \times p \text{ chips}} \tag{1.1}$$

The language used can vary. Sometimes measurements taken using a single chip are called an experiment, but it is increasingly common to see the measurements on several chips referred to as an experiment. This makes sense because it is common to conduct a designed experiment where several chips are used, and these chips may be treatments, controls or replications. This is the language we will use in this paper: *chip* refers to the measurements taken on genes of a single chip, and *experiment* is for the full set of chip measurements.

There are different types of problems tackled with gene expression data. For some it is convenient to work from the transposed matrix, where chips correspond to the rows and genes correspond to the columns. This type of analysis

is conducted when the chips correspond to some known groups, such as cancer type, and the task is to determine which genes are expressing most differently between the different classes, or to build a classifier for cancer type based on gene expression. Dudoit, Fridyland and Speed (2002) is a starting point for a discussion on this type of analysis. We don't discuss this type of analysis here. We will concentrate on data in the described by (1.1).

It is also important to know that each element of the gene expression matrix is calculated from a varying number of values measured on a single chip, because on each chip there are numerous spots corresponding to each gene. For example, the data used in this paper comes from an Affymetrix GeneChip. It is considered to be a high-density oligonucleotide array, with each gene represented by a probe set. The probe set has either 16 or 20 spots on each chip, and these values are combined to give a single expression value described in the matrix above. This information can be plotted together with the expression values in the analysis to better understand the uncertainty related to the expression value, although the focus of the next section is on finding patterns in the $n \times p$ gene expression data matrix.

It is important to emphasize the difference between the analysis of gene expression data from many other multivariate data analysis tasks. In gene expression data it is important to find a small number of genes that are behaving differently to others in an understandable way. It is expected that most genes are expressing as normal, and only a few are expressing differently in response to a treatment. The expression values for each treatment are collected using a single microarray chip, which is read by an image reader to give numerical values for expression. These image values are calibrated across the chips, based on the assumption that the expression of most genes on all the chips is similar. The values on each chip are logged because the distributions are skewed, and there is some evidence to suggest that intensity is exponentially distributed. Software such as BioConductor (2004)[1] provides tools for normalizing microarray data and making plots to check the results of the normalization. When the calibration and normalization is complete, the task is to find the outliers, the most differently expressed genes, relative to the expression of most genes. Thus the task involves outlier detection.

It is also a multiple comparisons problem. From the perspective of a traditional statistical analysis we are merely dealing with a situation which could be solved by a $t$-test for comparing means. The drawback is that we have to do a test for every single gene! For example, the newer Affymetrix Arabidopsis ATH1 Genome Array chips have about 22000 genes. In calculating 22000 test statistics,

[1]BioConductor (2004). Open source software for bioinformatics
http://www.bioconductor.org

according to a normal distribution we would expect to see at least 1100 genes with a significant $p$-value on a conventional level of 5%. That is simply by chance 1100 genes will emerge as interesting. This is not a practical approach to finding the few truly interesting genes to study in greater depth. These issues will be discussed later in the paper.

This is the question of interest. Which genes are expressing differently in response to the experimental treatments? Difference is measured relative to the distribution of all the genes, and relative to consistent values in the replications. We need plots of the multivariate distributions to assess this.

### 1.2 Common plots for gene expression data

The techniques developed for visualizing multivariate data for the most part work well with gene expression data also. Surprisingly, though, the most commonly used plots in the gene expression literature are astonishingly bad. Plots of gene expression data are used to:

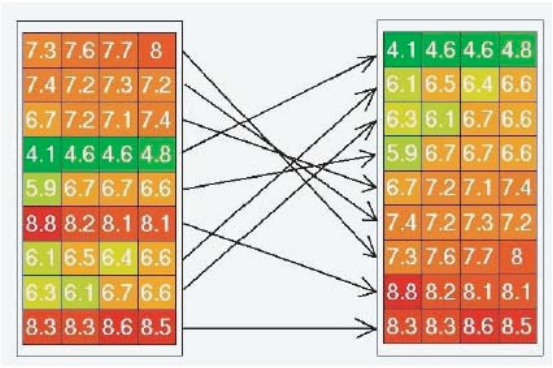
1. Overview the distribution of values in the data, to check the pre-processing, and to assess patterns visible in subsets of genes relative to all the genes.
2. Focus on the few genes which are expressing differently, in response to some treatment, or through some unexpected mechanism.

#### Heatmap

Many analysts display the data as a heatmap, which is a particularly awkward way to view multivariate data. The name for this display varies in the literature. It is sometimes called an image map. The term heatmap arises from the red to green color scale rather than the raw information shown in the plot, so it is a misnomer, because alternative color scales can be used. We would prefer to call this type of display a colored matrix plot, because it is simply the cells of the matrix colored according to the cell value, which, for gene expression data, is a measure of the level to which that gene is expressed. The genes or oligonucleotide sequences correspond to the rows of the matrix and the chips correspond to the columns. A colored matrix display will represent the matrix of values as a grid, number of rows equal to the number of genes being analyzed, and the number of columns equal to the number of chips. The boxes of the grid are colored according to the numerical value in the corresponding matrix cell. For example, here is a small simulated data set. A colored matrix representation of this data is shown in the left-hand side plot of Table 1. What can we see? One row is all green, meaning that this gene has all low values; two rows are all red (dark grey) meaning these genes have all high values, and the rest are combinations of orange, red and light green (light grey).

Table 1: Data displayed as colored matrix plot: (Left) Table of data. (Middle) Displayed in the same order as the matrix. (Right) Hand-reordered where similar genes are grouped together, providing a gradual increase of expression value from lowest to highest over the rows. The ability to perceive patterns in this data depends on the spatial proximity of similar numerical values, and thus the re-ordering the rows, and columns of the data.

| Genes ⇓ | Chips ⇒ | | | |
|---|---|---|---|---|
| | 7.3 | 7.6 | 7.7 | 8.0 |
| | 7.4 | 7.2 | 7.3 | 7.2 |
| | 6.7 | 7.2 | 7.1 | 7.4 |
| | 4.1 | 4.6 | 4.6 | 4.8 |
| | 5.9 | 6.7 | 6.7 | 6.6 |
| | 8.8 | 8.2 | 8.1 | 8.1 |
| | 6.1 | 6.5 | 6.4 | 6.6 |
| | 6.3 | 6.1 | 6.7 | 6.6 |
| | 8.3 | 8.3 | 8.6 | 8.5 |




A big challenge for interpreting patterns in a colored matrix is that the rows and columns need to be re-ordered in a meaningful way. We can only perceive structure or patterns in the plot if the rows, or columns, that are similar to each other are near each other in the matrix. Re-ordering is a little like playing a Rubik's cube game! In the literature often a dendrogram is drawn to the side of a colored matrix, and this is because the rows, and/or columns, have been re-organized using hierarchical clustering. This is an automatic approach to organizing the rows and columns.

Clustering as a way to re-organize a heatmap is *confounded* with clustering as an analysis technique for gene expression data. There are two different purposes to clustering. The first purpose is that a colored matrix needs to have organized rows and columns in order to perceive patterns. The second purpose is that cluster analysis is a method used on gene expression data to find groups of genes that are expressing similarly, because it is expected that genes work together to control metabolic function.

Figure 1 displays a simulated data set, with 77 rows and 4 columns, constructed to emulate examples seen in the literature. A heatmap is used to explain how the genes cluster. Different re-organization of the rows changes the perception of clusters. In the left plot the rows are in random order. Nothing can be learned from the random order! In the middle plot the rows are manually organized by the authors, and on the right the rows and columns are both sorted using average linkage hierarchical clustering. Both of these displays suggest that there are two clusters in this data: one with high expression on chips 1,2, and

low expression on chips 3,4, and conversely for the other cluster.

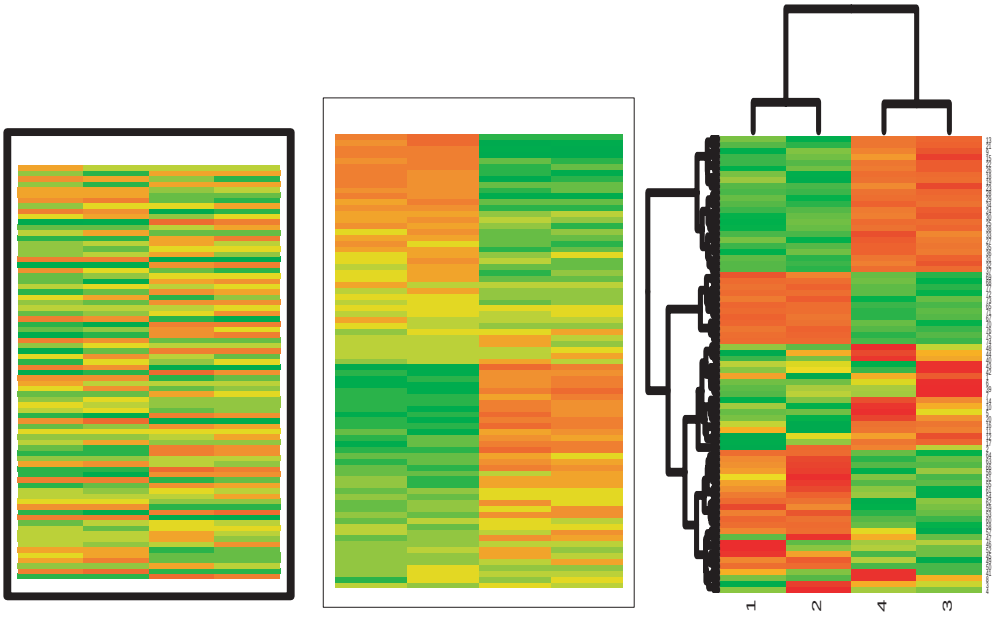



Figure 1: This data was constructed to emulate common examples of gene expression data. (Left) Rows are in random order, which is basically an uninterpretable mess. (Middle) The rows are sorted by the authors. Doesn't it look like there are two very distinct clusters in the data? One with high expression on chips 1,2, and low expression on chips 3,4, and conversely for the other cluster. (Right) Average linkage hierarchical clustering of rows and columns. Again, it looks like there are roughly two clusters, although the organization is not as neat as the left plot. To discover the real story about this data take a look at Figure 2.

### Scatterplots

Take a look at Figure 2. This is the same simulated data shown as a heatmap in Figure 1 now displayed as a scatterplot matrix. Surprise! There are not two clusters. The data is one huge cluster of inseparable data. This may come as a shock to some readers, but it shouldn't. The values on the chips are massaged by normalization so that the distribution of values is approximately the same from chip to chip, and on each chip the values are unimodal and slightly right-skewed. Most cluster algorithms are designed to operate on data where there are gaps between distinct clumps of points. As one unseparated clump, the data can be partitioned in many different, and equally interpretable, ways (Tarpey, Li and Flury 1995)

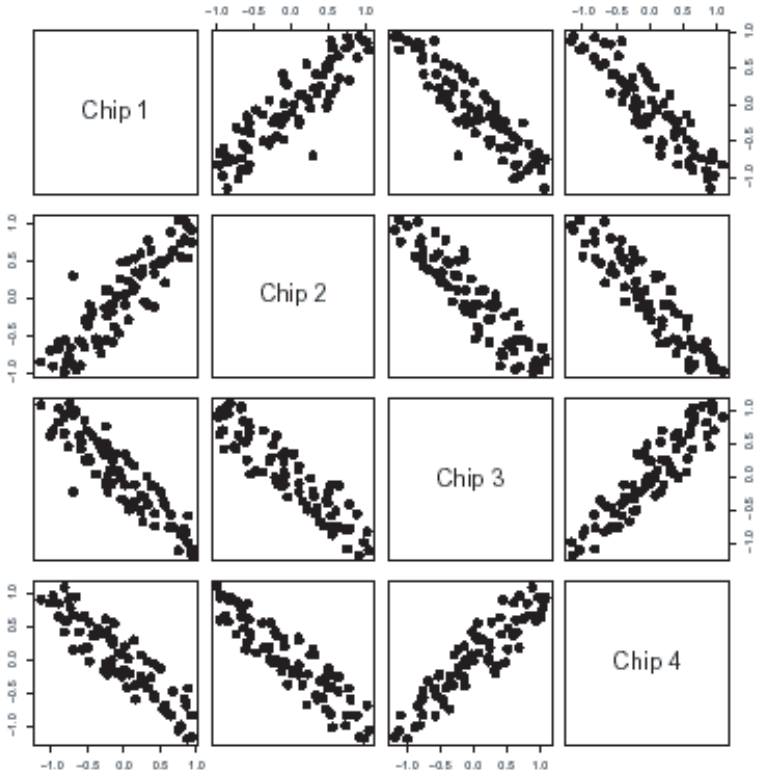



Figure 2: The simulated data from Figure 1 displayed as a scatterplot matrix: Can you see the clusters now? Do you also notice the outlier? This is exactly the type of gene that we want to detect - it has a lower expression on chip 1 than chip 2. This means that the gene is behaving differently between the treatments applied to the different chip. Can you see this outlier in the colored matrix?

*Do you also notice the outlier in the scatterplot matrix?* This corresponds to a gene which has low expression on chip 1 relative to chip 2. If *chip 1 is a control and chip 2 is a treatment, this gene is responding to the treatment.* This is exactly what we want to learn from gene expression analysis. We would like to find genes that have the most difference in expression due to a treatment or genotype compared to it's expression on the other chips, which corresponds to points in the scatterplot that are on the extremes of the scatter cloud far away from the $x = y$ line. Can you detect the outlying gene in the heatmap? Probably not. It is difficult to detect outliers from the sea of color in a heatmap. It is also difficult to read the strength of correlation between columns, and recognize geometric shapes, all of which are readily visible in the Euclidean geometry-based Cartesian coordinates of the scatterplot.

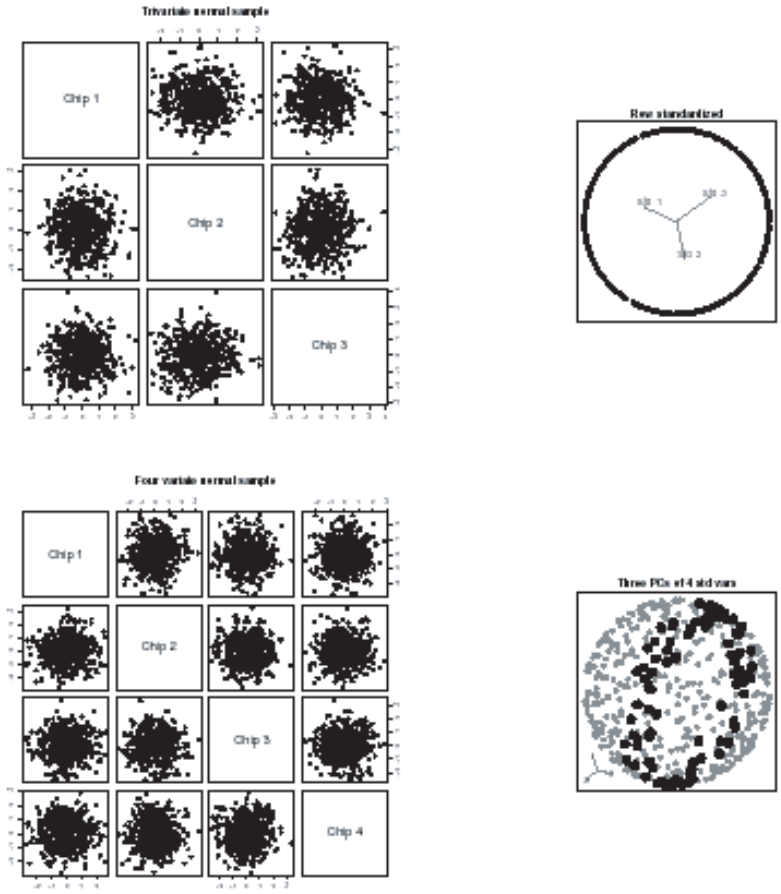



Figure 3: Demonstration of the effect of row standardizing data. (Top) A sample from a trivariate standard normal distribution: (left) raw data as a scatterplot matrix, and (right) tour projection of the row standardized showing the data lies on a circle. (Bottom) A sample for a four variable standard normal: (left) Raw data as a scatterplot matrix, and (right) tour projection of the principal components of the standardized data. The highlighted points (solid circles) show a slice through the sphere.

### Aside: A note on cluster analysis

With the goal being to find groups of genes that have relative patterns, such as, higher expression on chip 1 relative to it is expression on chips 2, 3 and 4, it usual that clustering is done using a distance metric based on the correlation between rows of measurements. The problem with clustering a single clump of points, however, does not disappear. Using correlation distance does not magically create separations or gaps in the distribution of the data. Correlation

distance might be understood intuitively as angular distance: think of a pivot in the center of the data and spin an ice cream cone around through the data. Genes within the ice cream cone are considered to have a similar pattern of expression. Correlation distance is equivalent to euclidean distance if the rows of the data are standardized individually, $(x_{i1} \ \ldots \ x_{ip})/\sqrt{\sum_{j=1}^{p} x_{ij}^2}, \quad i = 1, \ldots, n$ (Hastie, Tibshirani and Friedman 2001). It is usual to see this type of transformation prior to using model-based clustering (Falrey and Raftery 2002). Geometrically this operation transforms the data from a clump to points on the surface of a $(p-1)$-dimensional sphere, and then points near each other on the sphere will be clustered together by Euclidean distance. Figure 3 illustrates this effect. On the left side are the plots of the raw data and at right are plots of the row standardized data, for 3- and 4-dimensional data. The 3-D data reduces to points on a 2-D circle. The 4-D data reduced to points on a 3-D sphere. The plots on the right are projections from a tour (Cook, Buja, Cabrera and Hurley 1995) of the row-standardized data.

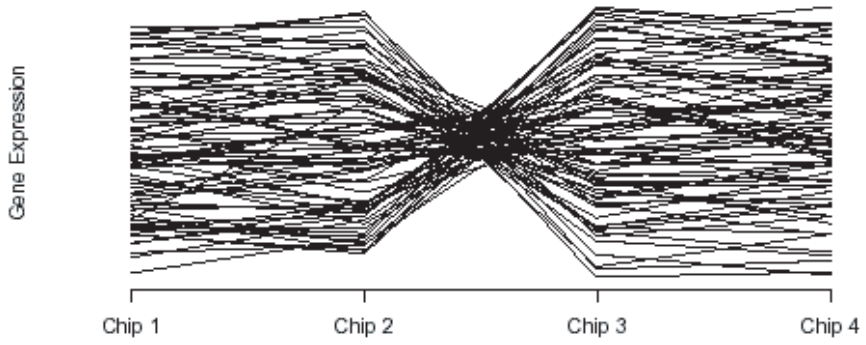



Figure 4: The simulated data from Figure 1 displayed as parallel coordinates.

## Parallel coordinates

The parallel coordinate plot (Inselberg 1985, Wegman 1990) is useful for examining relative patterns, such as those sought in gene expression data analysis. The axes are laid out in parallel rather than the orthogonal axes of scatterplots. These types of plots are commonly used to display gene expression data. Figure 4 displays the simulated data in parallel coordinates. What can we see from this plot? The pattern is a bow tie. The twist, between chips 2 and 3, means that

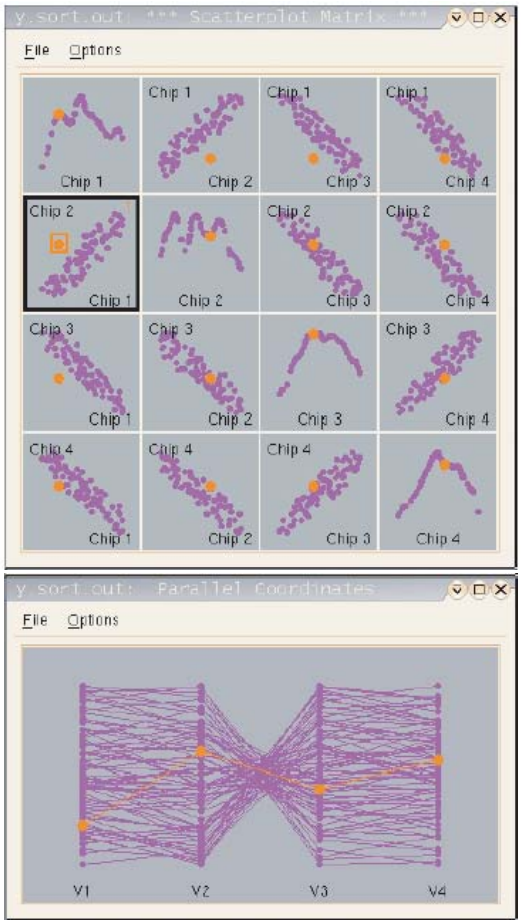



Figure 5: The simulated data from Figure 1 with linked scatterplot matrix and parallel coordinates.

the values of these chips are negatively correlated. The flat profiles between chips 1 and 2, and again between chips 3 and 4, mean that the values are positively correlated. Thus we get an overall picture that there is no strong clustering in this data. Yet, it is not a very clear picture because there are a lot of lines drawn here. Even though there are only 77 genes, which is tiny by microarray standards, the overplotting obscures important profiles such as the outlier. Commonly alpha-blending is used to alleviate the problem.

### 1.3 Direct manipulation and linking between plots

*Interaction on plots is critical!* Especially important for gene expression analysis is the ability to probe and link plots. Exploring data depends on the ability to interact with a plot and link the changes to elements of other simultaneously

visible displays (Beckel and Cleveland 1987, Newton 1978, McDonald 1982) .

With direct manipulation and linking the multivariate plots become very useful. Figure 5 shows the scatterplot matrix of the data from Figure 1 linked to parallel coordinates, generated with GGobi (Swayne, Lang, Buja and Cook 2003). The outlier that is clearly visible in the scatterplot of chip 1 vs chip 2 is highlighted (orange or medium grey). This profile can now be seen in the profile plot, and it is obviously an outlier, on chip 1 relatively to it is expression on the other three chips. This is the simplest type of linking: the elements of all plots corresponding to the same row of the data matrix are all highlighted. This type of linking is commonly available in graphics software.

A more sophisticated linking, available in GGobi, uses a categorical variable to link plot elements. This works like a key indexing of metadata. In gene expression data the unique identifier for the gene, such as AffyID, can be used to link plots of different information about the gene. For example, plots of the gene expression values (1 point), can be linked to plots of the PM values (16 or 20 points), and spatial location plots of the chip structure (16 or 20 points).

### 1.4 Summary of plots for gene expression data

- **Heatmaps** provide an overview of the data. It is difficult to grasp the distribution of values, because of a lack of geometric interpretability, difficulty in laying it out, and the mapping of numerical value to color. However, there is an appeal to heatmaps, and their use is very wide-spread. These plots can be enhanced to make them more useful. This is our wish list:

  - Different color mappings change the perception of patterns, and in exploratory analysis we want to see many different patterns to learn about different aspects of the data. We would like to add controls to allow the analyst to rapidly change the mapping of numerical value to color, and also the color scale. It would also be useful to switch from using color to glyphs (symbols) with size of glyph representing the the numerical value.
  - Methods to re-organize the rows and columns. Cluster analysis is one approach. For small subsets of data, manual controls like a large Rubik's cube, would be helpful. There is a package for R (lhaka and Gentleman 1996) available on `gclus`, written by Catherine Hurley, which has some alternative approaches for rearranging the results of a dendrogram. Orca (Sutherland, Rossini, Lumley, Lewin-Koh, Dickerson, Cox and Cook 2000). has a few manual controls for rearranging rows and columns in a colored matrix.

  - The plot is not scalable, because at some point one reaches the maximum available plotting space. There need to be ways to reduce the size of a plot, such as spatial smoothing, and pan and zoom and slice controls for focusing on smaller subsets.
  - The heatmap needs to be probeable, to allow queries such as, "Which gene and which chip is it that has such a high expression value?"

- **Scatterplots** are useful for pairwise comparisons, finding which genes are disproportionately expressed, and in tours they provide an overview of the multivariate distribution of expression values. Overplotting can be a problem for a large number of genes, but this can be alleviated using a density representation of the overplotted points. When there are many chips it is not possible to organize all pairs of plots into a scatterplot matrix. Linking between scatterplots is common and simple to implement.

- **Parallel coordinates** can be used to overview data, but only when there are very few cases. Generally the large number of genes means that it is not possible to see the distribution of values using a parallel coordinate plot, although density plots based on alpha blending, such as those available in ExplorN[2], Mondrian[3] and Cassatt[4], can help. In the literature, and gene expression software packages, we usually see only subsets of the data plotted using parallel coordinates.

The next section contains the most interesting part of this paper. It describes how these methods are used together to analyze a data set collected by the authors. EDA has advanced by breakthroughs in direct manipulation graphics. Linking information between plots enables an analyst to discover relationships in data that cannot be uncovered by any other method. This next section is a concrete example of how to do EDA on a difficult data problem.

## 2. Example

This section describes an analysis of locally collected gene expression data using direct manipulation graphics. The approach provides several new methods for gene expression analysis, and has helped us understand why different analyses produce dramatically different results.

[2]http://www.science.gmu.edu/∼rmoustaf/explorn.html
[3]`http://www.rosuda.org`
[4]`http://www.rosuda.org`

### 2.1 Data description

The type of data that this paper addresses is a $2 \times 2$ factorial design as follows:

| | Cofactor added | |
|---|---|---|
| | no | yes |
| Mutant | *M1,M2* | MT1, MT2 |
| Wildtype | W1, W2 | WT1, WT2 |

Normally, the mutant organism is defective in the ability to synthesize an essential cofactor, which is provided by the treatment. The experiment used 8 Affymetrix GeneChip Arabidopsis Genome Array chips. Each has 8297 oligonucleotide sequences (or genes). Each chip was measured on the material pooled from a tray of plants. Replicates were biological and conducted six months apart.

### 2.2 Expected results

The primary interest is in detecting the genes that are expressed differently on the first cell relative to the other three cells: mutant without cofactor against the rest. That is, we are interested in answers to:

> "Which genes are differentially expressed when the cofactor is not added, with special interest in the mutant genotype?"

Difference is measured by how the individual gene varies in the replicate, and also in comparison to how all the genes vary in expression value. We would hope to see

- small differences in expression values in the replications,
- small differences between expression values in wildtype with and without the cofactor added, and
- little difference between expression values in the wildtype and mutant with the cofactor added.
- the results to be robust to different normalizations, and different statistical tests.

### 2.3 Data Preparation

Our data was recorded from the older Affymetrix GeneChip Arabidopsis Genome Array. The `CEL` files from Affymetrix contain the image scores for each gene as a matrix of 16 or 20 pairs of numbers:

| | | | |
|---|---|---|---|
| $PM_1$ | $PM_2$ | ... | $PM_{16}$ |
| $MM_1$ | $MM_2$ | ... | $MM_{16}$ |

PM refers to perfect match, and MM means mismatch. Each PM and MM consist of 25 base pairs. PM is the correct gene string and MM contains a modification of the middle base pair. The MM string has been included to quantify the background noise during the image analysis, but now there is some uncertainty whether it is useful for this purpose because the MM strings may actually be responsive gene sequences.

Based on the assumption that most genes are behaving as usual we would expect that the distributions of gene expression to be the same from chip to chip. Thus the measurements need to be to calibrated across chips, resulting in the same distribution of values on each chip. We use the probe level quantile method as suggested in Bolstad, Irizarry, Astrand and Speed (2003). This method uses the 8-dimensional quantile-quantile plot and projects all the points of this plot onto the diagonal $d = (1/\sqrt{8}, \cdots, 1/\sqrt{8})$. These projected data are used for the next steps, normalization and summary.

We use the Robust Multichip Average (RMA) (Irizarry, Hobbs, Collin, Beazer-Barclay, Antonellis, Scherf and Speed 2003) to normalize and calculate the summary values for gene expression data. For each gene,

$$\log_2(\mathrm{PM}^*{}_{lm}) = \tau_l + \lambda_m + \eta_{lm}, \quad l = 1, \cdots, 8; m = 1, \cdots, M \tag{2.1}$$

where $\mathrm{PM}^*$ is a PM value after the quantiles step, $\tau_l$ represents the log scale expression level for chip $l$, $\lambda_m$ represents the $m$th probe effect and $\eta_{lm}$ represents an independent identically distributed error term with mean 0 and variance ${\sigma^2}_{\mathrm{PM}}$. After estimating this model using median polish, the estimate of $\tau_l$ gives the expression measures for the chip $l$.

There is still considerable discussion about standardization and normalization of gene expression data, and while it is not the focus of this paper, the methods described in this paper can be used to understand and assess the algorithms. Graphics in BioConductor (2004) help evaluate the results of normalization. Henderson (2004) describes dynamic graphics for gene expression data with a focus on checking data quality and normalization of the data.

A common approach to analyze gene expression data is to build an ANOVA model for each gene. For example, each model looks like:

$$Y_{ijk} = \mu + \alpha_i + \beta_j + \gamma_{ij} + \varepsilon_{ijk}, \quad i = 1, 2; j = 1, 2; k = 1, 2 \tag{2.2}$$

where $Y_{ijk}$ is the expression value ($\hat{\tau}_l$), $\mu$ is the overall mean, $\alpha_i$ is the effect due to genotype, $\beta_j$ is the effect due to cofactor added, $\gamma_{ij}$ is the interaction between genotype and cofactor added, and $\varepsilon_{ijk}$ is the error with mean 0 and variance ${\sigma^2}_{\text{EXP}}$. The model estimates are: the overall mean $\hat{\mu} = \bar{Y}_{...}$, effect due to genotype $\hat{\alpha}_i = \bar{Y}_{i..} - \bar{Y}_{...}, i = 1, 2$, effect due to cofactor added $\hat{\beta}_j = \bar{Y}_{.j.} - \bar{Y}_{...}, j = 1, 2$, and the interaction effect $\hat{\gamma}_{ij} = \bar{Y}_{ij.} - \bar{Y}_{i..} - \bar{Y}_{.j.} + \bar{Y}_{...}, i = 1, 2; j = 1, 2$.

The reason to fit an ANOVA model is due to the experimental design. It is a $2 \times 2$ factorial design. Although, we expect that only one cell of the four cells to be different from the other cells, and we could calculate t-tests for this, the ANOVA model better fits the design, and allows us to detect unexpected patterns of gene expression. The results of a t-test are embodied in the ANOVA results: all three tests would be significant. The result is an ANOVA which tests the significance of each factor in the design, and the interaction of the factors. To examine the ANOVA results for all the genes, we examine the summary values from the ANOVA table: mean square values, $F$-statistics and $p$-values.

To explore the data, we construct three different data sets. The first data set has all the treatments with replicates (M, MT, W, and WT). The replicates are connected to each other using a line segment. We use this to compare the variation in treatment relative to the variation in replicate. The length of the line between treatments corresponds to the variation in replicate.

The second data set has all the chips (M1, M2, MT1, MT2, W1, W2, WT1, and WT2), the averages of each treatment (M.av, MT.av, W.av, and WT.av), the results from ANOVA (MS.Geno, MS.Treatment, MS.Interaction, PP.Geno, PP.Treatment, and PP.Interaction) and PM summaries (PM.variation, MS.error, and reliability). The third data set contains the PM values (PM.M1, $\cdots$, PM.WT2). These three data sets are will be linked interactively using the categorical variable corresponding to the AffyID. Thus a plot of pairs of treatments in data set 1 can be linked to profiles of the expression data in data set 2, and the PM values in data set 3.

Data 1: Replicates

| GeneID | M | MT | W | WT | AffyID |
|---|---|---|---|---|---|
| 1 | 6.00 | 6.26 | 6.36 | 6.56 | "11986_at" |
| ⋮ | | | | | |
| 8297 | 4.74 | 4.60 | 4.85 | 4.67 | "AFFX-YEL024w/RIP1_at" |
| 1 | 6.14 | 6.20 | 6.33 | 6.10 | "11986_at" |
| ⋮ | | | | | |
| 8297 | 4.53 | 4.93 | 4.57 | 4.50 | "AFFX-YEL024w/RIP1_at" |

Data 2: Statistics

| GeneID | M | | MT | | W | | WT | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | R1 | R2 | R1 | R2 | R1 | R2 | R1 | R2 | Statistics | AffyID |
| 1 | 6.00 | 6.14 | 6.26 | 6.33 | 6.36 | 6.33 | 6.56 | 6.10 | | "11986_at" |
| ⋮ | | | | | | | | | | |
| 8297 | 4.74 | 4.53 | 4.60 | 4.93 | 4.85 | 4.57 | 4.67 | 4.50 | | * |

*"#AFFX-YEL024w/RIP1_at#"

Data 3: PM

| GeneID | M | | MT | | W | | WT | | |
|---|---|---|---|---|---|---|---|---|---|
| | R1 | R2 | R1 | R2 | R1 | R2 | R1 | R2 | AffyID |
| 1 | 7.27 | 7.35 | 7.37 | 7.51 | 7.45 | 7.78 | 7.55 | 7.37 | "11986_at" |
| ⋮ | | | | | | | | | |
| 1 | 7.62 | 7.81 | 7.96 | 7.77 | 7.87 | 8.09 | 7.79 | 7.59 | "11986_at" |
| ⋮ | | | | | | | | | |
| 8297 | 7.34 | 7.28 | 7.47 | 7.60 | 7.67 | 7.53 | 7.66 | 7.30 | "AFFX-YEL024w/RIP1_at" |
| ⋮ | | | | | | | | | |
| 8297 | 7.33 | 7.06 | 7.08 | 7.21 | 7.15 | 7.03 | 7.04 | 6.86 | "AFFX-YEL024w/RIP1_at" |

### 2.4 Analysis: Checking data quality

#### Examining the replicates

We expect that the replicates are similar to each other. The expression for a gene on the first replicate is very similar to it's expression on the second replicate, so that the values should fall close to the $x = y$ line. The replicates were measured on different plants six months apart, so we do expect some variation, but for the significance of the treatments to be assessed we need to see replicability in the experiment.

The replicates are plotted against each other (Figure 6) in addition to guide lines corresponding to fold change 1, 2, and 3. We are most concerned about genes that have large differences in the replicates, that is, the points having larger fold

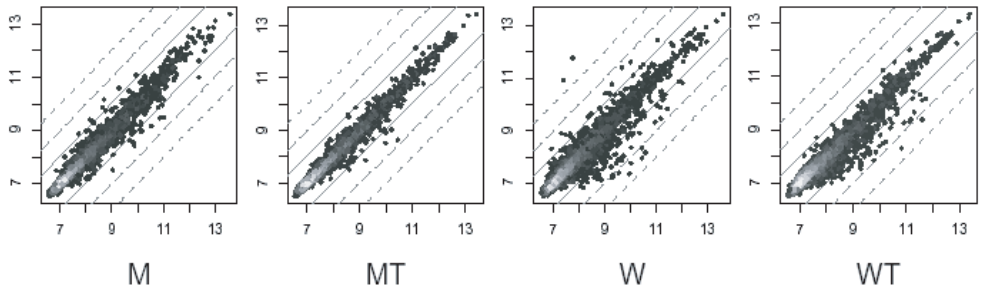



Figure 6: Density plots of the replicates of each treatments. (White is higher density, more points overplotted here.) The dashed lines represent 1, 2, 3 degrees of fold change difference between the replicates. The replicate measurements should look very similar, with the ideal plot having all points lying on the $x = y$ line. Points with large difference in the replicates, fold change greate than 2, say, are cause for concern.

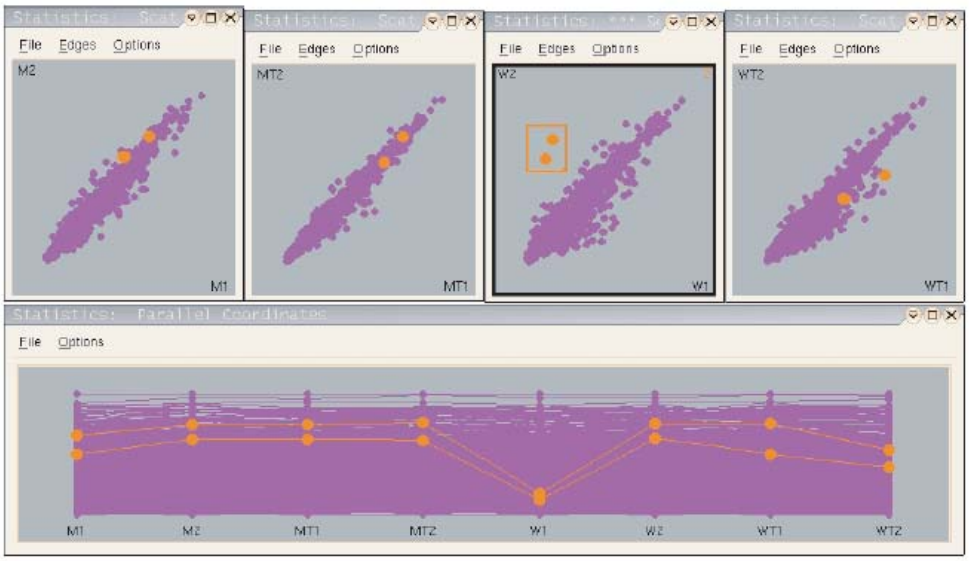



Figure 7: Plots of the replicates of each of the four treatments, linked by brushing. A profile plot of the 8 treatment/replicate expressions is also linked. The two genes that had big differences in the replicates on WT are highlighted (orange or medium grey). The values for these genes on W1 appear to be too low.

change. These genes are the ones that have large variability in expression values from chip to chip. These may be volatile genes, or there may have been errors in the expression reading on one chip, such as a hair or scratch.

There are two genes in the WT cell that have replicates with dramatically different values between replicates. These genes are identified as unreliable observations.

Both wildtype plots have a strange pattern, a bulge at the lower right, a pregnant belly, of genes that have lower expression on replicate 2 than on replicate 1. When we examined the AffyID's and TAIR annotations for these genes there were some labelled "heat shock", which led to some suspicion about a systematic error such as overheating of first replicate. But this turned out not to have happened. There is a better explanation which will become obvious in the next couple of sections.

**Examining replicates using linking**

We expect to see no patterns here: genes that are extreme in the plot for one treatment will not be extreme in the plots of the other treatments. Any pattern of extreme values across all treatments would suggest that this is a gene with volatile expression. Linking between plots is used to detect patterns across treatments. The four replicates plots are visible, and points in one plot are brushed and observed in the other plots. Figure 7 illustrates the process, focusing on the two extreme replicates for W. A profile plot of the eight treatment/replicate expressions is also linked. The two genes that had big differences in the replicates on W are highlighted. The values for these genes on wildtype replicate 1 are much lower than the other values. There was clearly a problem with W replicate 1 for these two genes. The replicate expression values on M and MT are not extreme, but they are extreme on WT. These two genes are clearly problematic.

All the other genes in the extremes of these replicate plots are examined similarly. There were no strong patterns here: genes that had a big difference on replicates in one treatment mostly had little difference in other treatments. That is, the data looks good, there are no wildly volatile genes, and the replicates have similar expression values.

**Examining treatments in relation to replicates**

We would expect that the pairwise plots of replicates to be less varied than the plots of different treatments. Why? Because all the genes should have similar expression from replicate to replicate, but not necessarily between treatments. We expect that some genes to have noticeably more variation in expression on one treatment than on another. To explore these patterns in the data we plot the chips pairwise in a scatterplot matrix.

The left-side scatterplot matrix in Figure 8 shows the the pairwise plots of the four mutant genotype chips. We plot four chips at a time because eight chips would give too many plots to easily compare. Breaking the chips into two groups based on genotype is a convenient way to group the chips. The least varied scatterclouds should be the plots of the replicates, that is, the top left and bottom right plots. And indeed they are, as expected.

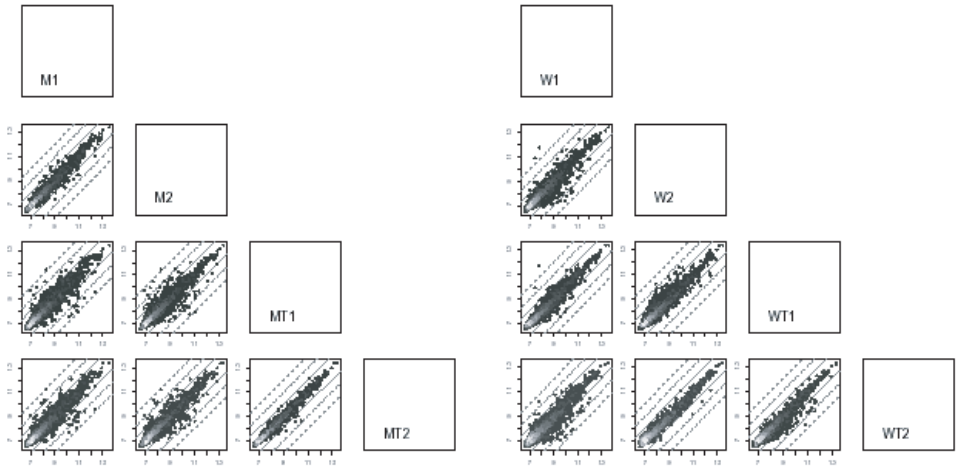



Figure 8: Scatterplot matrix of the (left) mutant and (right) replicates.

There's something strange in the right-side scatterplot matrix in Figure 8, in the pairwise plots of the wildtype treatments. The least varied scatterclouds should be the plots of the replicates, that is, the top left and bottom right plots. But, surprise, they are not.

Notice what happens if the labels are changed. Interchange the W2 with WT1 (Figure 9 left). The problem disappears. Now the scatterclouds of the replicates are the least varied. Also notice that the other strange replicate effect – the pregnant belly of genes with higher expression on replicate 1 than on replicate 2 – disappears. We suspect that there has been a labelling error in this data.

It is interesting to examine the heatmap of this data (Figure 9, right plot). The heatmap supports the suspicion: chips WT1 and W1 are grouped together, and chips WT2 and W2 are grouped together. But, you cannot determine why from this plot. Clustering is a blind technique: we cannot see why they have been grouped like they have. To see why they have been grouped together a scatterplot is needed. It should also be noted that the heatmap took 70 minutes to draw.

The data was checked, and re-checked, different normalization methods were used and normalizations repeated. We could not dismiss the suspicion. Thinking biologically we may explain it as variation in wildtype plants. We would expect wildtype plants to be more varied than the mutant, but Arabidopsis is a model plant, that has been cultivated and studied for a long time, so that even the wildtype plants could be considered to be relatively homogeneous compared to the natural diversity in the wild. With these thoughts we proceeded to analyze the data using the switched labels. This data is otherwise very beautiful, consistent,

gene expression data. The main point here is not so much that there is an error in the data but that without the scatterplot we might not have detected a problem. Plots are essential for checking data quality.

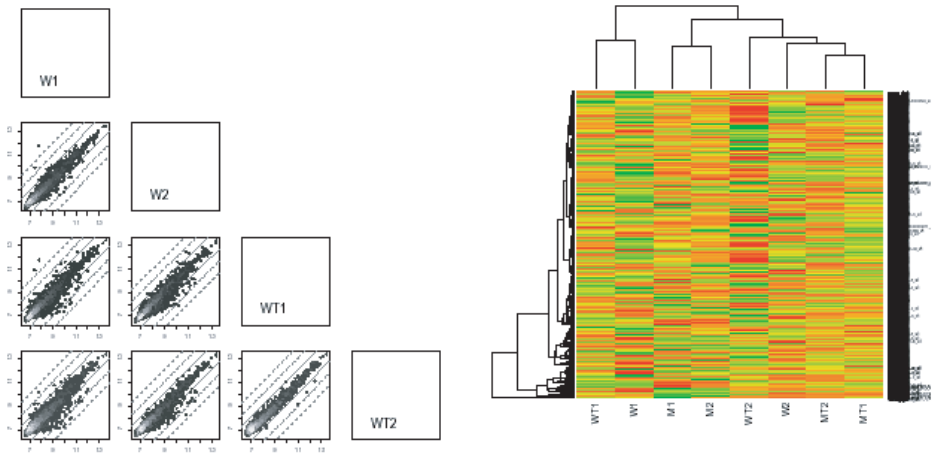



Figure 9: (Left) Scatterplot matrix of the "corrected" wildtype replicates. (Right) Heatmap of the original order, also suggests that a switch happened because WT1 and W1 are grouped together.

**Examining the treatment averages**

Each pair of replicates is averaged giving a gene expression value for each treatment. We would expect more variation in expressions for the mutant average in relation to the other three treatments to be the greatest. Why? Because the mutant genotype has reduced ability to produce the essential nutrient, and there is no nutrient added to the soil. These are examined using a scatterplot matrix (Figure 10). There are several interesting patterns:

- As we expected, the most variation is seen in the first column of plots, where M.av is paired with the other three treatments.
- Now here's a surprise. Look at the plot of MT.av vs WT.av. The expression values vary very little between these treatments. That is, when the cofactor is added to the soil the genes of both genotypes have very similar expression values.
- There is some difference in variation pattern of the three controls, MT.av, W.av, WT.av.

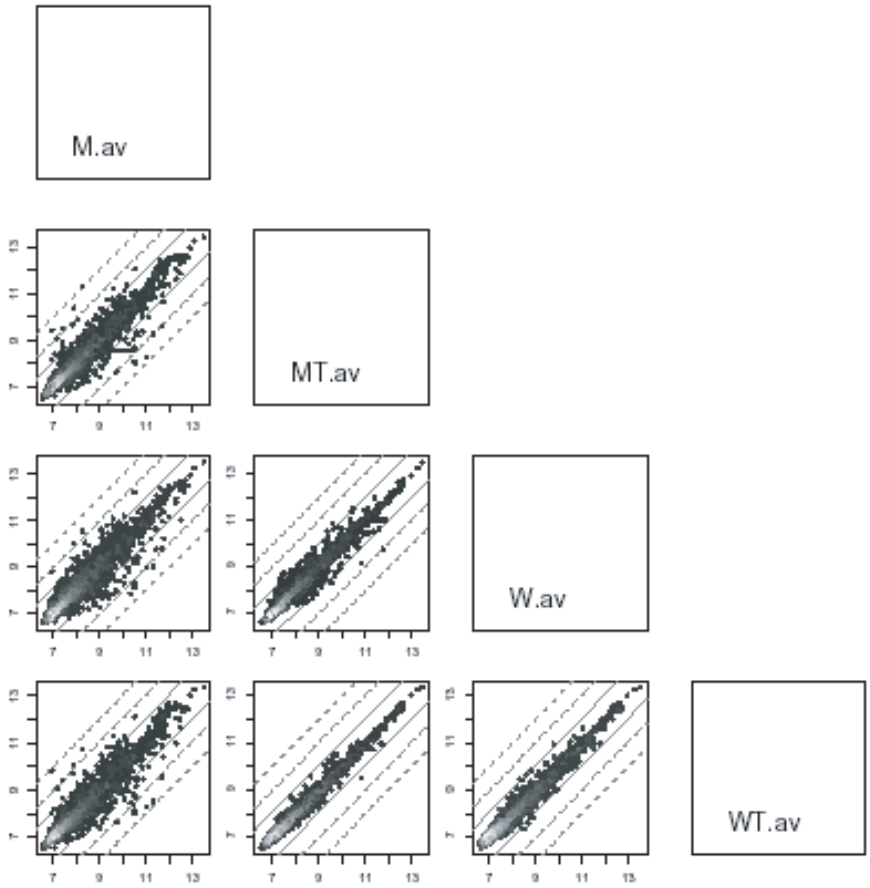



Figure 10: Plots of the treatments against each other. The treatment pair where the genes behave the most similarly are the two genotypes with cofactor added (MT, WT). The mutant genotype without cofactor has more difference in expression value compared to all other treatments (first column of plots, first row of plots). There is some difference in expression values between MT and W, and W and WT.

## 2.5 Analysis: Finding the interesting genes

### Linking treatment plots

We would expect the genes that are expressing most differently between M vs MT, and M vs W, and M vs WT, are not so different on the three control comparisons. To check this we will use linked brushing on the pairwise plots of the treatments. We do this in a way that we can see the replicates along with the treatments - both replicates are included in the treatment plots and the two points for each gene are connected by a line. These are a new type of plot, that

we call a *replicate line (RL) plot*. We are now looking for genes that have a big difference between treatments (far from $x = y$), and that have little difference in replicates, small line length. Figure 11 shows the plots.

The genes that are in the extremes in the plots of M vs the three controls, but not so extreme in the plots of the controls, are very interesting. Brushing and identifying these genes yields a list of genes in Table 2.

**ANOVA statistics**

We would expect that the genes we extracted above will also appear on the list of genes with the lowest $p$-values in ANOVA. The 10 genes with the lowest $p$-values (across all three tests) are listed in Table 2. Only one of the genes (`13212_s_at`) identified in the data plots appears! Figure 12 shows the profiles of the five of the top ten of each list. Which line of plots would you pick as having more interesting genes? The genes that we plucked out from the data plots are not ranked in the lowest $p$-values. Gene `15160_s_at` has the $1092^{th}$ lowest $p$-value. It is definitely not going to easily float to the top of the interesting chart using ANOVA statistics. Only `13212_s_at`, `15122_at`, and `16150_at` have $p$-values in the lowest 100.

Table 2: Table of top 10 interesting genes from inspecting data plots, and from ANOVA $p$-values.

| Data plots | Rank of $p$-value | ANOVA | Rank |
|---|---|---|---|
| 15160_s_at | 1092 | 19348_at | 1 |
| 15189_s_at | 367 | 18038_i_at | 2 |
| 16016_at | 61 | 13212_s_at | 3 |
| 12748_f_at | 218 | 19991_at | 4 |
| 15122_at | 26 | 12313_g_at | 5 |
| 16054_s_at | 152 | 14825_at | 6 |
| 16053_i_at | 38 | 14634_s_at | 7 |
| 20491_at | 186 | 18255_at | 8 |
| 13212_s_at | 3 | 19991_at | 9 |
| 16150_at | 145 | 12312_at | 10 |

Why does this disagreement happen? Figure 13 shows what ANOVA "sees". Notice the scale on the vertical axis. This is a multiple comparisons problem. Because each gene is modeled individually the scale of the difference in expression is relative to the individual scale, not the global expression scale.

A common fix for multiple testing is to reduce the significance level, using Bonferroni for example. Multiple comparisons creates the problem that along with the "truly" significant genes, true positives, there will be false positives.

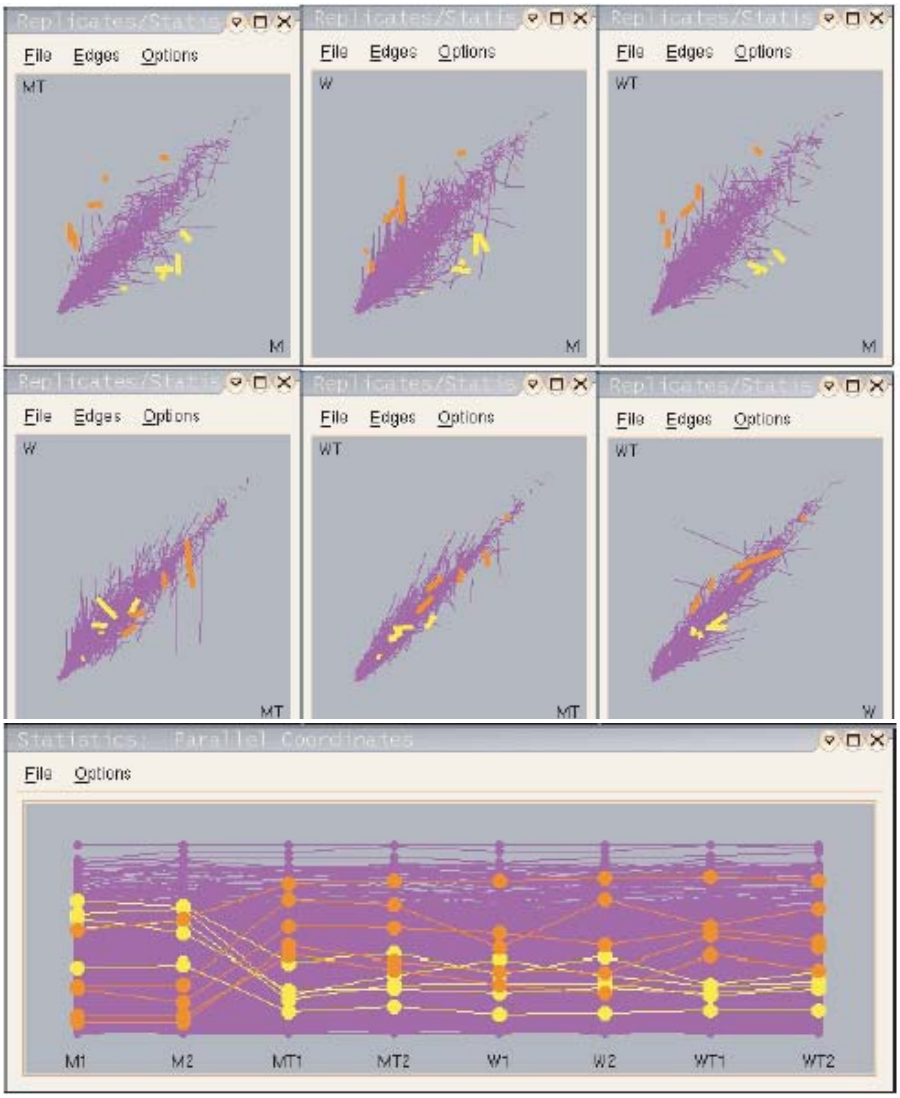



Figure 11: Searching for interesting genes: genes that have large MS treatment value and small $p$-value are considered interesting. Here several interesting genes are highlighted. These genes have higher or lower expression on M than the controls, but not unusual expression amongst the controls.

Simnonl, Korll, McShane, Radnlacller, Wright and Zhao (2003) (p.94) say "In order to compensate for the multiple comparisons, statistical significance at the $p < 0.001$ level should be required, or some other multiple testing procedure implemented for controlling the number or percentage of false discoveries." False discovery rate (Storey and Tibshiranl 2003)[5] addresses this by quantifying the

[5]Storey, J. and Tibshiranl, R. (2003). Statistical significance for genome-wide experiments.

proportion of genes that are likely to be false positives. What would be the effect of this? Instead of taking the genes with the lowest, say 100, $p$-values, only the genes with 30 lowest $p$-values would be reported as interesting. The result is to discard even more interesting genes. These approaches preserve the order of the $p$-values, simply shifting the cutoff for interpreting significant difference. These methods don't advise which genes amongst the significant ones are true positives, and they ignore those genes below the cutoff, those that are now considered to be not significant, but these genes may indeed be important. Multiple comparisons is also confounded with numerical error: some calculations are made on very small numbers. False discovery rates and Bonferroni-adjusted $p$-values side-step the issue of numerical inaccuracy. In fact they can lead to concentrating attention on genes with small overall expression, and low reliability, in favor of genes that have more interesting expression patterns.

Other approaches consider different ways to calculate the MSE for each test. The most extreme approach is to pool the MSE based over all genes - the mean differences for individual genes are compared with the variance of all genes. But using the pooled MSE is considered to be too conservative. There is a continued debate amongst biologists and analysts that the magnitude of expression may not match the biological impact of expression: some genes may express very little but have a huge biological impact. Biological significance is confounded with statistical significance, and numerical accuracy. Instead of pooling the MSE, attenuated statistics add a constant to each individual MSE (Tusher, Tibshirani and Chu 2001).

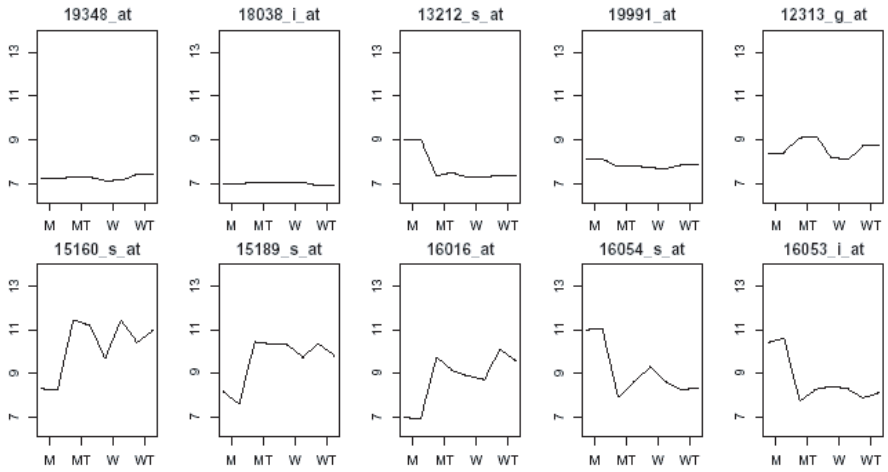



Figure 12: Profiles of top genes found interesting by (top) ANOVA $p$-values, (bottom) plotting the data. Which row do you consider has more interesting genes?

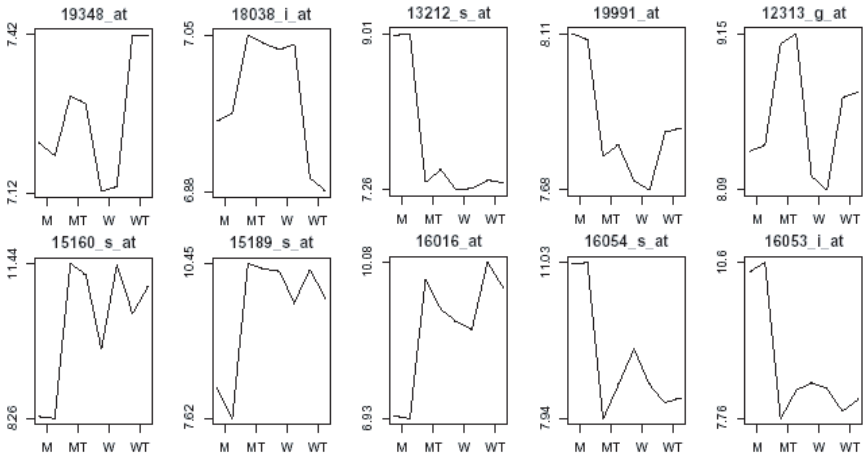



Figure 13: The profiles as ANOVA sees it. Each gene is modeled on it is own scale.

An approach to remove small numbers from the analysis is discard genes with small overall variance (Simon *et al.* 2003). But how small is "small"? It requires that a threshold value is set ahead of the analysis. Genes with expression below the threshold are considered to have variation due to measurement error rather than biological variability. Many arbitrary approaches to determining appropriate thresholds have been proposed, some based on calculating fold change using ratios of expression values. These small variation genes are important to include while normalizing gene expression data, but exacerbate the multiple comparisons when searching for interesting genes.

Is the problem due to using ANOVA instead of paired $t$-tests? Perhaps ANOVA tests are lacking in power? Remember that the main question to answer compares the M cell with three control cells (MT,W,WT). To directly test this hypothesis we would compute a paired sample $t$-test for each gene, and pick off the top of the list of lowest $p$-values to return the most interesting genes. Try it. There are the same multiple comparison problems as ANOVA, and the remedies are similar.

### Our approach

By plotting the data it is obvious immediately that some genes are interesting, and they should not be missed by any analysis, but they are missed by ANOVA analysis. But in defense of modeling, relying only on the data plots will result in missing many interesting genes, too. With 8297 line segments there is a lot of overplotting, and some of the most interesting genes will have the smallest line segments. Thus we approached the problem by combining modeling with data

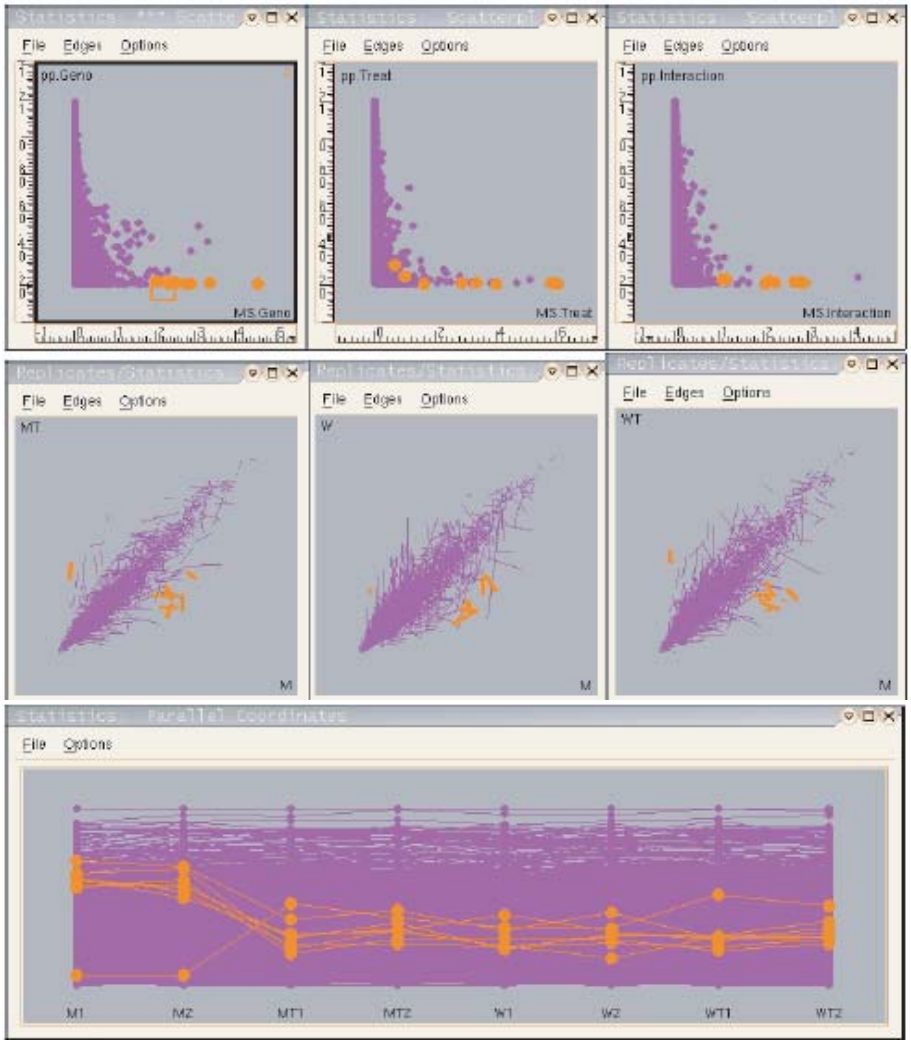



Figure 14: Selecting interesting genes on the basis of large MS value and small $p$-value.

plots: we modeled the genes and added the statistics from the models to the list of variables to plot.

The aim of our approach is to favor the selection of genes that have large variance overall with the variance primarily due to treatments, selecting more of these genes and less of the genes with small overall variance. Thus we plot the MS values against the $p$-values for each of the three tests, and brush along large MS values and small $p$-values, linking the plot to profile plots of all the chips. Figure 14 illustrates this. The genes that have large MS values along with small $p$-values are the one that are obviously interesting from the data plots. The genes are selected in this manner, brushing in from the right of the MS/$p$-value

plots, and observing the profile for the gene. To the bottom right of the MS/$p$-value scattercloud the gene expression patterns are clearly interesting, but as the brush gets closer to the bottom left of the scattercloud there will be more of a mix of genes, some with interesting expression patterns, some with less interesting patterns.

### Generalization and extensionsmedskip

Now before the gentle reader responds naively that this is obvious. Attenuated statistics fix the problem. Think for a moment. Does it? Try it. Put your favorite analytical method here. Look at the $p$-values obtained using attentuated statistics. Plot the profiles of the genes that emerge as interesting. You might be surprised. The particular use of ANOVA in our approach is not important. Studying plots of the data, along with the model diagnostics, allows us to check that the model is doing as expected and that the results are sensible. What is obvious about ANOVA modeling to us, now, is due to making plots when we were surprised by the disparity in the gene lists. Results from modeling should be consistent with is learned from plotting the data.

Similarly, put your favorite normalization method here. The relative merits of different normalization methods for gene expression data is still debated. We used the PM values alone for our normalization, which is arguably incorrect. Ideally the results obtained for different normalizations are consistent. That is, there should be some robustness to normalization method. A gene that emerges as interesting using one normalization should also be high on the list using another method. With linked plots of differently normalized data this is easily checked. The data can be structured so that several matrices containing differently normalized data can be inspected together.

The problem of multiple comparisons is confounded with the reliability of the expression values. Each expression score arises by combining the values of the probe set, either 16 or 20 values in our data. Affymetrix reports an absolute call for each gene based on comparing the PM and MM values. If the PM values differ little from the MM values then the gene is reported as an absent call, there is no expression (Amaratunga and Cabrera 2004). What is missing from the ANOVA analysis is slightly different: The MSE does not incorporate the variation in PM values.

### Aside: A note on MSE

The normalization process introduces errors. The variation in PM values, that are "averaged" to get the expression values, is not incorporated into the ANOVA model.

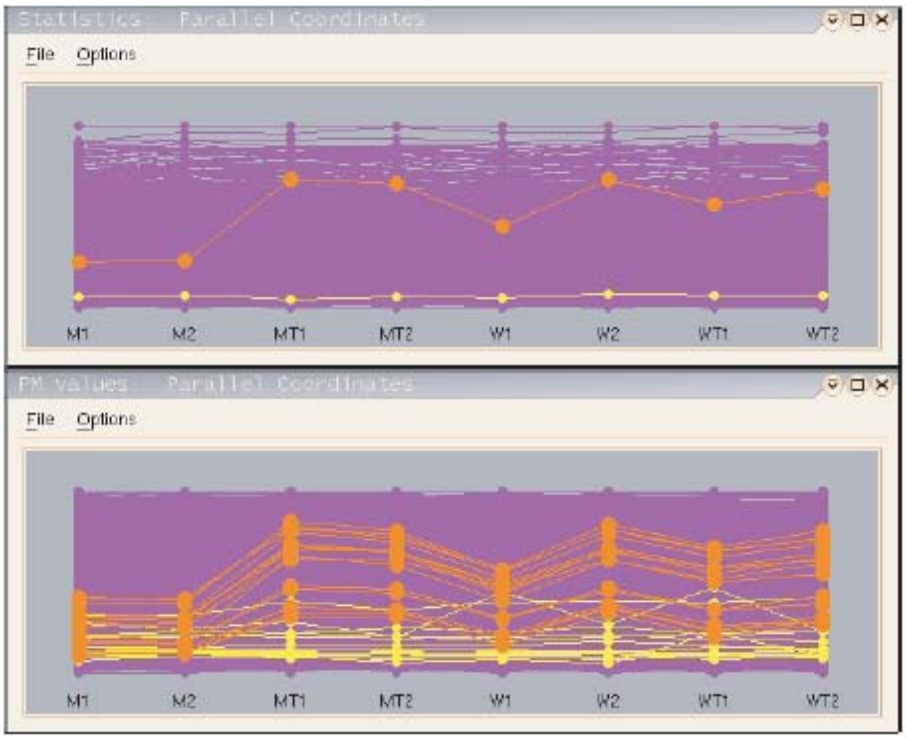



Figure 15: Focusing on two genes: `19991_at` (yellow or light grey), `15160_s_at` (orange or medium grey). The top plot is the profile of the expression values, the bottom of the PM values for the genes. The PM values for `19991_at` are quite varied, even though the MSE is very small. It has a very small reliability value.

There are many genes with very small differences in the expression values over all treatments: the mean square error is close to zero. Because of that these genes have large $F$-statistics, and hence small $p$-values.

For a guideline to decide how small a difference in expression values is too small, we consider the probe level data - PM values. The ANOVA model (equation 2.1) uses the expression level data and ignores the lower level data - probe level. If we combine the equations (2.1) and (2.2), this model is similar to a hierarchical linear model (HLM) (Byrk and Ralldenbush 1992). Even though it is not the exact HLM we can borrow some of the ideas to compare the ANOVA MSE with variation in PM. First, we will use PM.variation and MS.error as the estimates of $\sigma^2_{\text{PM}}$ and $\sigma^2_{\text{EXP}}$, respectively where

$$PM.variation = \frac{1}{8(M-1)} \sum_{l=1}^{8} \sum_{m=1}^{M} (\log(\text{PM}^*_{lm}) - \hat{\tau}_l)^2, \tag{2.3}$$

with $M = 16$ or 20, depends on gene; and

$$MS.error = \frac{1}{4}\sum_{i=1}^{2}\sum_{j=1}^{2}\sum_{k=1}^{2}(y_{ijk} - \hat{\mu} - \hat{\alpha}_i - \hat{\beta}_j - \hat{\gamma}_{ij})^2. \tag{2.4}$$

Then the average of the experiment reliabilities can be calculated as follows:

$$Reliability = \frac{\sigma^2_{\text{EXP}}}{\sigma^2_{\text{EXP}} + \sigma^2_{\text{PM}}/M} = \frac{MS.error}{MS.error + PM.variation/M} \tag{2.5}$$

This reliability indicates the reliability of the expression values. Even though a difference in expression values is too small, we can keep the ANOVA results for genes that have high reliability.

The reliability for gene `19991_at` is very low, 0.004, compared to the reliability for gene `15160_s_at`, 0.77. Figure 15 shows the PM values of the two genes.

## 3. Summary and Discussion

This paper has presented approaches for exploring gene expression data using lots of plots. The major points made are:

- We can learn a lot by plotting the data and diagnostics from models. Plots enable the analyst to check the quality of the data, assess the models, and compare results from different methods.
- Exploratory data analysis methods using lots of plots are useful for gene expression data, and the plots we use are highly interactive. It is quite different EDA today than when it arose decades ago.
- Heatmaps are less useful for *exploring* gene expression data than scatterplots and parallel coordinate plots. Stare at the heatmap of our data in Figure 9, and tell us how many of the observations that we made about this data, are detectable using this plot.

The methods described in this paper generalize quite broadly. When there are more than two replicates these can be examined in a pairwise manner using scatterplot matrices. Multiple replicates can be connected by line segments when comparing treatments.

When we believe that we have finished with a gene expression analysis, what do we do? Hand our list to the experimenters and move on to a different project? The important part of the work is still to be done. The list represents a hypothesis about which genes are involved in a biological activity. The gene expression

analysis is not a confirmatory study. Either more conventional genetic studies need to be conducted, or more evidence needs to be collected and weighed to assess the genes' importance. Conventional genetic studies are time-consuming, and may take several years to complete. Gathering associated information can be faster, using quick searches of the literature, and public databases. Gene expression analysis, text-based searches, comparative expression studies are hypothesis generators, that can be used to design confirmatory studies, but they are primarily exploratory tools. It is the conventional genetic studies which can confirm or reject a gene's importance.

## Appendix

The analysis was done using R [6], with the BioConductor package [7], GGobi [8] and MANET [9]. The Rggobi package, available from the ggobi web site, is used to integrate numerical calculations with interactive graphics. This is a beautiful package for integrating modeling with exploratory data analysis.

The relevant source code for R, and data sets in ggobi's `xml` format, available on the web[10].

## Acknowledgements

This research was conducted with the support of a grant from the National Science Foundation Arabidopsis 2010 DBI-0209809, and United States Department of Agriculture National Resource Inventory grant 2002-0358.

## References

Amaratunga, D. and Cabrera, J. (2004). *Exploration and Analysis of DNA Microarray Data and Protein Array Data*. Wiley-Interscience.

Beckel, R. A. and Cleveland, W. S. (1987). Brushing scatter plots. In *Dynamic Graphics for Statistics* (Edited by W. S. Cleveland and M. E. McGill), 201-224. Wadsworth.

Bolstad, B., Irizarry, R., Astrand, M. and Speed, T. (2003). A comparison of normalization methods for high density oligonucletide array data. *Bioinformatics* **19**, 185-193.

Byrk, A. S. and Ralldenbush, S. W. (1992). *Hierarchical Linear Models: Applications and Data Analysis*. SAGE.

[6] `http://www.R-project.org`
[7] `http://www.bioconductor.org`
[8] `http://www.ggobi.org`
[9] `http://www1.math.uni-augsburg.de/Manet/`
[10] `http://www.public.iastate.edu/~dicook/papers/Microarray`

Cook, D., BuJa, A., Cabrera, J. and Hurley, C. (1995). Grand tour and projection pursuit. *J. of Computational and Graphic Statistics* **4**, 155-172.

Dudoit, S. Fridyland, J. and Speed, T. (2002). Comparison of discrimination methods for the classification of tumors using gene expression data. *J. Amer. Statist. Asso.* **97**, 77-87.

Falrey, C. and Raftery, A. E. (2002). Model-based clusting, discriminant analysis, density estimation. *J. Amer. Statist. Asso.* **97**, 611-631.

Hastie, T., Tibshirani, R. and Friedman, J. (2001). *The Elements of Statistical Learning*. Springer.

Henderson, H. (2004). *Dynamic Graphics for Microarray Data Before Normalization* Invermay, Statistics and Bioinformatics Group, AgResearch, pp. 147-158.

lhaka, R. and Gentleman, R. (1996). R: A Language for data analysis and graphics. *Journal of Computational and Graphical Statistics* **5**, 299-314.

Inselberg, A. (1985). The plane with parallel coordinates. *The Visual Computer* **1**, 69-91.

Irizarry, R., Hobbs, B., Collin, F., Beazer-Barclay, Y., Antonellis, K., Scherf, U. and Speed, T. (2003). Exploration, normalization, and summaries of high-density oligonucleofcide array probe level data. *Biostatistics* **4**, 249-264.

McDonald, J. A. (1982). Interactive graphics for data analysis. Technical Report Orion II, Statistics Department, Stanford University.

Newton, C. (1978). Graphica: From alpha to omega in data analysis. In *Graphical Representation of Multivariate Data* (Edited by P. C. C. Wang), 59-92. Academic Press.

Simnonl, R. M., Korll, E. L., McShane, L. M., Radnlacller, M. D, Wright, G. W. and Zhao, Y. (2003). *Design and Analysis of Microarray Investigations*. Springer.

Sutherland, P., Rossini, A., Lumley, T., Lewin-Koh, N., Dickerson, J., Cox, Z. and Cook, D. (2000). Orca: A visualization toolkit for high-dimensional data. *Journal of Computational and Graphical Statistics* **9**, 509-529.

Swayne, D. F., Lang, D. T., Buja, A. and Cook, D. (2003). Ggobi: Evolving from xgobi into an extensible framework for interactive data visualization. *Journal of Computational Statistics and Data Analysis* **43**, 423-444.

Tarpey, T., Li, L. and Flury, B. (1995). Principal points and self-consistent points of elliptical distributions. *The Annals of Statistics* **23**, 103-112.

Tusher, V. G., Tibshirani, R. and Chu, G. (2001). Significance analysis of microarrays applied to the ionizing radiation response. *Proceedings of the National Academy of Science* **98**, 5116-5121.

Wegman, E. (1990). Hyperdimensional data analysis using parallel coordinates. *Journal of American Statistics Association* **85**, 664-675.

Received April 1, 2005; accepted February 17, 2006.

Dianne Cook
Department of Statistics
Iowa State University
Ames, IA 50011, USA
dicook@iastate.edu

Heike Hofmann
Department of Statistics
Iowa State University
Ames, IA 50011, USA
hofmznn@iastate.edu

Eun-Kyung Lee
Department of Statistics
Iowa State University
Ames, IA 50011, USA
lee.eunk@gmail.com

Hao Yang
Department of Statistics
Iowa State University
Ames, IA 50011, USA

Basil Nikolau
Department of Biochemistry, Biophysics and Molecular Biology
Iowa State University
Ames, IA 50011, USA
dimmas@iasteat.edu

Eve Wurtele
Department of Genetics, Development and Cell Biology,
Iowa State University
Ames, IA 50011, USA
mash@iastate.edu