Differential Expression, Class Discovery and Class Prediction using S-PLUS and S+ArrayAnalyzer

Michael O'Connell
moconnell@insightful.com

ABSTRACT

Microarrays are a powerful experimental platform, allowing simultaneous studies of gene expression for thousands of genes under different experimental conditions. However there is much biological variability induced throughout the experimental process that can obscure the biological signals of interest. As such, the need for experimental design, replication and statistical rigor are now widely recognized. Statistical hypothesis testing has become the accepted differential expression analysis approach and many classification and prediction methods used in class discovery and class prediction now incorporate stochastic modeling components.

This paper provides a review of statistical analysis approaches to the analysis of data from microarray experiments. This includes discussion of experimental design, data management, pre-processing, differential expression, clustering and class prediction, reporting and annotation. The review is illustrated with the analysis of an experiment with 3 experimental conditions using the Affymetrix murine chip mgu74av2; and with descriptions of available functionality in the statistical analysis software S-PLUS and its associated module for microarray analysis, S+ArrayAnalyzer.

Keywords
Differential expression, class discovery, class prediction, S-PLUS, S+ArrayAnalyzer

.

INTRODUCTION

The development and refinement of high throughput microarray assays for RNA transcripts has stimulated the statistical and computational communities in many ways. High-throughput runs somewhat counter to high-precision, and there are many sources of variability in microarray data. Dealing with this variability in a systematic and statistically rigorous manner is crucial in providing biologically-valid inference. Sources of unwanted variability include microarray manufacturing, sample mRNA preparation, hybridization, scanning and signal extraction.

Microarray data have many inferential challenges, most obviously the number of genes/probes for which an assessment of (differential) expression must be made – with so many tests, the chance of false positives is high and must be managed. Indeed, the many inferential challenges of high throughput data from microarrays, mass spectrometry, nuclear magnetic resonance spectroscopy and two-dimensional gel electrophoresis have led to the recent development of many new statistics and computational methods. Key statistical areas are (a) identification of differentially expressed genes across experimental conditions; and (b) discovery and prediction of classes of experimental samples.

This paper provides a review the microarray data analysis pipeline with an emphasis on these areas, to highlight key statistical issues and to suggest appropriate methods and models for common analytic situations. A running example is provided based on data from a mouse immune response experiment as analyzed by Jain et al. [1] to identify differentially expressed genes in three populations of immune exposure: naïve (no exposure), 48 hour activated, and CD8+ T-cell clone D4 (longterm mild exposure). The example illustrates some of the analysis steps and is presented in the context of the statistical analysis software S-PLUS and its associated module for microarray analysis, S+ArrayAnalyzer.

THE MICROARRAY ANALYSIS PIPELINE

At a high-level, the microarray analysis pipeline follows the standard analysis pipeline viz. experimental design, data access, data preparation, modeling, reporting, deployment. Specific components and issues include:

1. Experimental Design
   - Sample size estimation
   - Assignment of experimental conditions to arrays – particularly important in 2-channel arrays, which are naturally incomplete block designs with blocks of size 2.
2. Data Access
   - Database and LIMS access; dealing with (sometimes proprietary/binary) specific file formats.
3. Pre-processing
   - Image analysis – registration, segmentation, estimation of signal and background;
   - Gene filtering e.g. removal of genes that show no expression at any experimental condition;
   - QA of chips and data within chips;
   - Background and non-specific binding adjustment;
   - Probe level analysis of arrays with more than one probe per transcript e.g. Affymetrix;
   - Normalization within and between arrays.
4. Analysis and modeling
   - Identification of genes that are differentially expressed across experimental conditions – methods for two-level and multi-level designs;
   - Clustering of samples and genes (class discovery);
   - Classification of samples (class prediction);
   - Validation and use of data from related experiments.
5. Reporting
   - Analysis of function for identified genes – e.g. assessment of gene ontology categories that are over- (and under-) represented in the differentially expressed genes identified
   - Annotation of tabular and graphical analysis summaries
6. Deployment
   - Making the analyses available to the biologist/scientist user community

EXPERIMENTAL DESIGN AND DATA ACCESS

Microarrays are used in many different experimental scenarios. Two broad inferential classes are (a) identification of differentially expressed genes across specific experimental conditions of interest, and (b) exploratory studies involving many experimental samples.

Examples of (a) include many functional genomics studies exploring cellular pathways and events under experimental conditions of interest e.g. comparisons of wild-type v knockouts, drug candidates and known potent/toxic agents, state/tissue comparisons (e.g. tumor v non-tumor). Experiments typically include two experimental conditions and sometimes extend to time course (progression) and/or multi-factorial conditions.

Examples of (b) include studies of disease taxonomy and genotype-phenotype relationships i.e. class discovery and class prediction studies. Applications include developing mRNA signatures for tumors (and resulting risk stratification, survival outcomes prediction and personalized diagnosis/treatment); profiling of chemotherapeutic (and potentially toxic) drugs; and obtaining insight into molecular mechanisms of disease and drugs.

A key experimental design question, particularly in the case of (a) is "how many replicate chips?" Pan et al. [2] provide some insight into this as well as S-PLUS code for estimating sample sizes. With larger experimental designs, e.g. factorial designs and time-course experiments, less replication is needed provided there is an adequate number of total chips in the experiment. Two-channel studies can be set up as incomplete block designs with blocks of size 2, and linear models including terms for the channel within chip effect are a natural approach [3]. Dye-swap designs, where treatments are run on different channels of the same chip and again with the channels reversed on another chip, help balance/ orthogonalize the dye/channel effect. Loop designs correspond to (balanced) incomplete block designs and are efficient in that sense. Reference designs are useful in that they allow additional runs to be added to the experiment, but they are inefficient [4] since they involve the use of the reference sample on half of the observations; also the treatments are confounded with dyes in this design. Saturated designs have potential for larger experiments but have not been explored for microarray applications.

Accessing and managing microarray data is an important convenience in facilitating data analysis. S+ArrayAnalyzer includes the Affymetrix API's, which makes S+ArrayAnalyzer fully compatible with all Affymetrix microarray systems, current and future, including the new GeneChip® Operating Software (GCOS) output and the entire installed base of Affymetrix instruments and scanners. Affymetrix data are read-in through point-and-click/browse-files or through a single file that specifies location of files for import (Figure 1). Such methods are similarly available for cDNA data.

Searching and importing data from databases is also readily accomplished. S-PLUS/S+ArrayAnalyzer includes native driver access to Oracle, SQL-Server and Sybase; and simple user-interfaces for searching databases are available. For example, the Affymetrix AADM schema is easily searched to assemble chips for analysis. Other databases e.g. BioDiscovery GeneSight, Iobion Gene Traffic, and Rosetta Resolver are also easily accessed. The simple access to data files and databases provides convenience in assembling data for analysis and avoids many otherwise manual data import steps and the management of additional and redundant file types. The format of microarray file types will likely change in the future, and by abstracting access through API's, seamless access for all future systems using S+ArrayAnalyzer is assured.

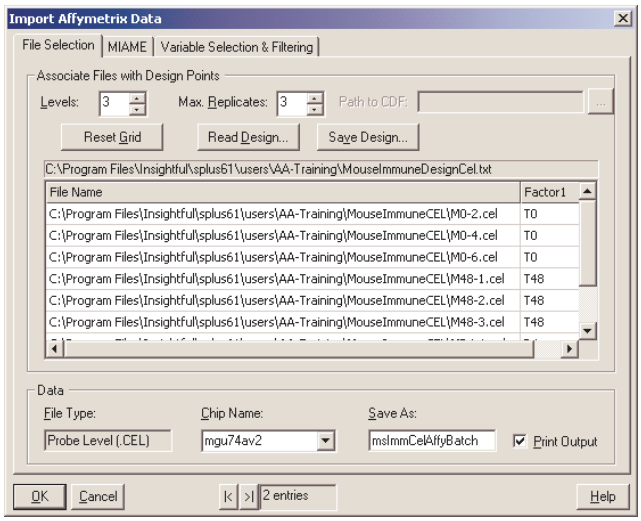

Figure 1. Reading in Affymetrix (binary) CEL data. Mouse immune response study: one-way design, 1 factor, 3 levels, 3 replicates. Files can be loaded individually by point-and-click; or as a group using the Read Design option in which case file locations are supplied as a text file.

PRE-PROCESSING

There are four main aspects of pre-processing microarray data viz. adjustment for background, adjustment for non-specific binding (e.g. mismatch in Affymetrix GeneChips), normalization, and probe-level summary for chips with multiple probes per transcript (Affymetrix GeneChips).

The goal of adjusting for background or non-specific binding is to obtain accurate estimates of signal intensities. Assuming X and B are unbiased and precise estimates of total and background intensity, S = X - B will be an accurate and precise estimate of signal. Note that variance of S is the variance of X + the variance of B. One problem that often arises is that estimates of B are not accurate or precise. For example, Yang et al. [5] suggest background estimates from many image analysis packages are unreliable. Also, the subtraction of mismatch (MM) intensities from perfect match (PM) intensities at the probe level in the MAS5 analysis of Affymetrix probe-level data [6] typically results in 30% of the PM-MM differences being less than zero. This implies that the MM intensities are estimated with more error than the PM intensities; and/or that when both the PM and MM intensity measure is low, that the errors are on par with the signal. Thus, while subtraction of background is desirable in the goal of producing accurate estimates of signal intensity, precision may be compromised in this operation, particularly when overall expression intensities are low. This may be overcome by using better estimates of background e.g. median background in two-channel chips and global MM estimates in Affymetrix chips [7].

Negative and very small estimates of background-adjusted signal intensities are nonetheless problematic and require careful handling in downstream analyses.

In two-channel arrays, the main pre-processing required is normalization within slides for balancing intensities between channels/dyes. The standard method emerging it to normalize as a function of expression intensity using a smooth function of intensity e.g. the `loess()` function in S-PLUS [8]. This approach may also be used to remove spatial effects of print-tips by fitting a separate `loess()` function for each print-tip.
In the Affymetrix system, each gene is represented by 11-20 PM and MM pairs of probes, each probing a different region of the mRNA transcript, typically within 600 base pairs of the 3' end. In the Affymetrix MAS5 software system, these 11-20 data pairs are combined into a single value per gene by adjusting for background, subtracting MM from PM within each (PM, MM) pair, normalizing chips within an experiment by a simple location adjustment that aligns within-chip means, and combining the PM-MM differences within probe pair sets using a Tukey biweight function that downweights PM-MM values according to their distance from the median(PM-MM) within the probe pair set.

Many researchers have developed alternatives to the MAS5 approach. Of note are the methods of Li and Wong [9], Irizarry et al. [10], Zhang et al. [11] and Wu et al. [7,12]. Li and Wong [9] provide a model-based expression index (MBEI) via estimating and removing a probe effect using a multiplicative model, and normalizing based on genes that don't vary much across chips within an experiment. Irizarry et al. [10] model PM intensity as a sum of exponential and Gaussian distributions for signal and background respectively, and use a quantile normalization [13] and a log-scale expression effect plus probe effect model that they fit robustly (median polish) to define the robust multi-array analysis (RMA) expression estimate for each gene. Zhang et al. [11] propose a stacking energy, positional-dependent-nearest-neighbor (PDNN) model for the RNA/DNA duplex. This includes terms for the sequence of nearest neighbors (adjacent two bases) and the position of these nucleotide pairs. Wu et al. [7] describe an algorithm similar to RMA, but incorporating the MM using a model based on GC content (GC-RMA). Wu et al. [12] propose a unified physical/stochastic model, incorporating background and non-specific binding, using physical aspects of the Zhang et al. [11] model and a non-specific binding model of Naef and Magnasco [14], in a stochastic framework. The performance of the physical models [11,12] in practical situations is unclear at this point.

RMA, MBEI and MAS5 models are all implemented in S+ArrayAnalyzer (Figure 2). The success of the probe-level analysis and normalization can be assessed from diagnostic plots e.g. the MvA plot (Figure 3), and boxplots showing distribution summaries for each chip.

MAS5 provides an accurate summary, incorporating background and MM corrections. The drawback is that MM subtraction is done within each probe pair and variability in MM is high, resulting in approximately 30% of such subtractions being negative. This is ameliorated somewhat by the Tukey biweight algorithm used to combine the differences; and the resulting expression measure is accurate, albeit somewhat variable for low expression values. Conversely, RMA provides precise summary expression measures, with the drawback that such precision is obtained at the expense of some accuracy, particularly for low expression values. Also, the somewhat aggressive quantile normalization method should be applied with care, since it may wash out differential expression when applied across all experimental conditions, especially if a large number of genes vary across experimental conditions. The S+ArrayAnalyzer GUI allows RMA with and without normalization; normalization can be done within experimental conditions, and results of probe-level analyses can be simply merged. MBEI is a nice model conceptually but suffers from the same bias issues as RMA and requires many chips per experimental condition for estimation of the model-based index.

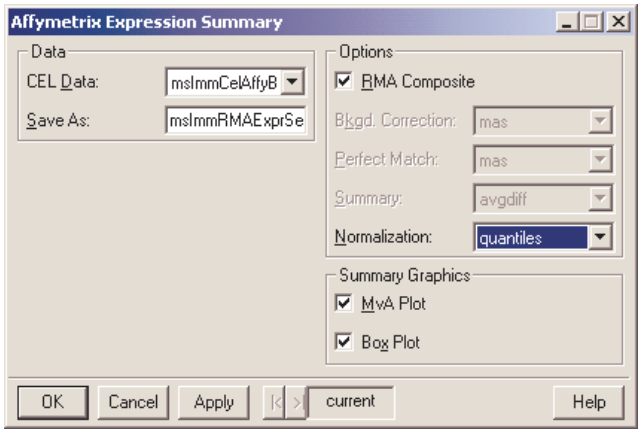

Figure 2. Probe level analysis of Affymetrix data; RMA Composite was chosen in this analysis.

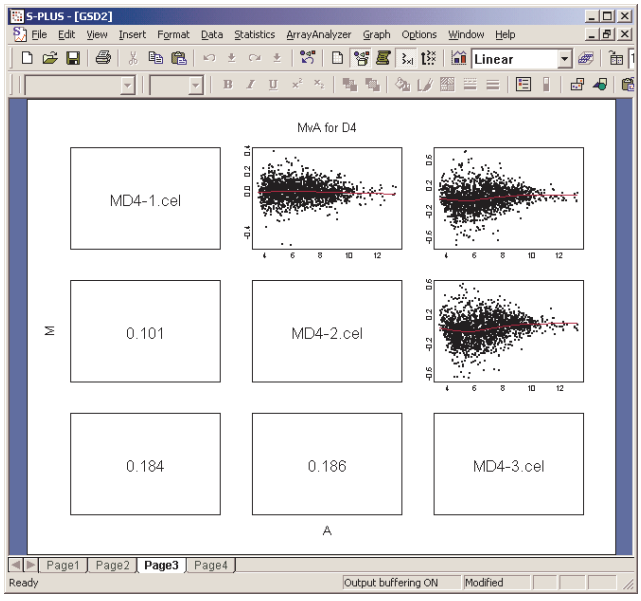

Figure 3. MvA plot showing results of RMA analysis for the MD4 experimental condition. Note y-axis scale is small, -0.6 to 0.6, and RMA has resulted in tight agreement between replicates.

The physical/stochastic model combinations hold promise in obtaining estimates of expression for individual genes that are

both accurate and precise. Once verified in practical situations, these models will be included in S+ArrayAnalyzer.

DIFFERENTIAL EXPRESSION

A key goal of microarray experiments is to identify genes that are differentially expressed while keeping the probability of false discoveries acceptably low. From a statistical perspective, the first part of this involves minimizing false negatives or maximizing power of statistical test, and the second part minimizing false positives.

Microarray data are often assessed as fold-changes between experimental conditions. While this scale has interpretive value, inference based solely on fold-change is misleading because error variability for each gene is heterogeneous under different biological conditions and intensity ranges.

With two experimental conditions, differential expression tests within genes are an example of the most basic statistical test – the two-sample comparison. Standard statistical approaches include the t-test and the Wilcoxon test. These approaches require a substantive number of replicates, since variances are much harder to estimate than means. Depending on the nature of the experiment, 6 or more replicates per experimental condition may be needed for a reliable estimate of within-gene error to be constructed [2]. When many replicates are available, permutation versions of the null distribution can be calculated by shuffling the experimental condition labels and calculating the test statistic for each permutation. Note that the significance cutoffs obtained from a permutation distribution are asymmetric and this may be biologically meaningful in that there is no reason to think that the cutoffs for up- and down-regulation would be symmetric.

The usual t-statistic is a signal-to-noise ratio:

$$t_g = (X_{g1} - X_{g2}) / se(X_{g1} - X_{g2})$$

where X_{gi}, i = 1; 2, is the mean intensity (log2 scale) of the i-th experimental condition, and se( ) denotes a pooled standard error function within gene g and between conditions 1 and 2. We conclude differential expression if the observed t_g for a given gene is greater than expected for the appropriate t-distribution or permutation distribution. A summary of multiple hypothesis testing methods in the context of microarray data analysis is given in [15].

Estimation of the pooled standard error is a key issue. While a large number of replicates is desirable from a statistical perspective, microarrays can be expensive, and target RNA sample availability is often limited. This results in some experiments being performed with a limited number of replicates. In this case, within-gene estimates of variability do not provide a reliable hypothesis testing framework. For example, a gene may have very similar differential expression values in duplicate experiments by chance alone. This can lead to inflated signal-to-noise ratios for genes with low but similar expression values.

As such, a number of statistical hypothesis tests have been developed that provide more reliable estimate of the standard error for comparisons of expression between experimental conditions by borrowing strength from among the genes. These can be broadly grouped into two classes: (a) variance function and transformation methods, and (b) error fudge factors and empirical Bayes methods.

Variance function methods typically model the variance of expression for a gene within an experimental condition as a function of the mean expression for that gene [16-19]. Of note are the methods of Durbin et al. [16] and Huber et al. [17] who propose a generalized log transformation and corresponding two parameter variance function; and Jain et al. [1] who propose a non-parametric variance function, estimated by pooling variance estimates within bins of mean intensity, and smoothing the variance function using the `loess()` function in S-PLUS.

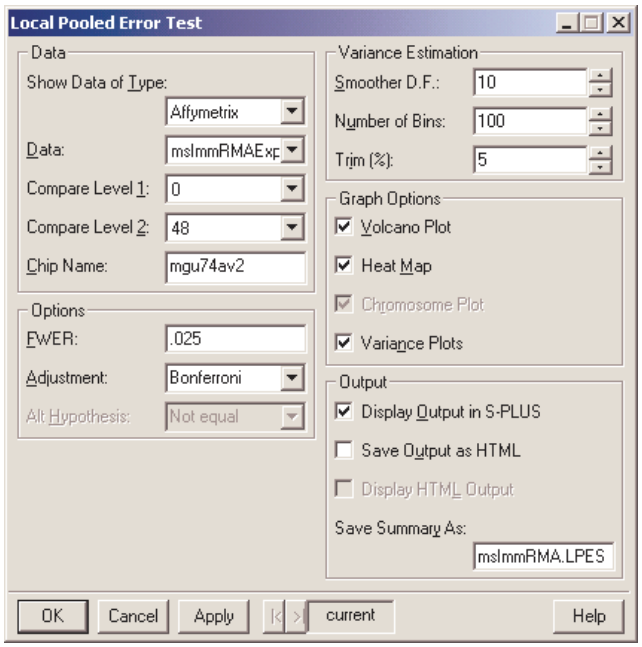

Figure 4. Differential expression using the LPE test (Jain et al. [1]) with Bonferroni FWER control.

Empirical Bayes methods include those of Efron et al. [19], Baldi and Long [20], Newton et al. [21] and Lönnstedt and Speed [22]. Lönnstedt and Speed [22] shrink the within-gene variance estimate towards an estimate including more genes, and construct signal-to-noise ratios using the shrunken variance. This is similar to the method of Tusher et al. [23] who include a fudge factor in the denominator of the signal-to-noise ratio.

For more than two experimental conditions, ANOVA and mixed effect models can be used effectively [24-26]. In the case of cDNA data, Wolfinger at al [25] suggest fitting 2 models: a normalization model fit to all the data, followed by a gene expression model fit to each gene separately.

$$log_2(y_{ijg}) = \mu + A_i + T_j + (AT)_{ij} + E_{ijg} \quad (1)$$

$$R_{ijkg} = \mu_g + T_{jg} + S_{kg} + A_{ijg} + E_{ijkg} \quad (2)$$

Model (1) is the normalization model, where i, j and g are indices for arrays, treatments and genes respectively. Arrays are considered as a random effect and an additional (random) effect for arrays within dyes is sometimes needed.

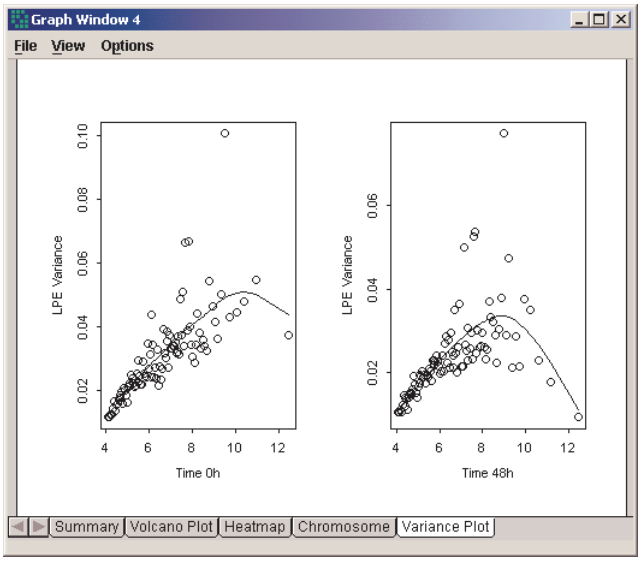

Figure 5. Estimated LPE variance functions for time 0 and 48 hours on RMA-summarized expression data.

The term (*AT*) models the (random) channel effect. Model (2) is the gene expression model, fit to the residuals, *R*, from model (1). The term *S* models the (random) spot effect. Kerr et al [24] fit similar models and assume all effects are fixed.

For Affymetrix data, Chu et al. [27] formulate a similar pair of models fit to the probe level data:

$$log_2(PM_{ijg}) = \mu + T_j + A_{ij} + E_{ijg} \quad (1)$$

$$R_{ijkg} = \mu_g + T_{ig} + P_{kg} + A_{ijg} + E_{ijkg} \quad (2)$$

Model (1) is the normalization model, where i, j and g are indices for denotes arrays, treatments and genes respectively. Model (2) is the gene expression model, fit to the residuals, *R*, from model (1). The term *P* models the (random) probe effect.

The above mixed models provide a flexible modeling framework; for example in time course experiments, the treatment effect T can be parameterized as contrasts versus baseline or as polynomials. One criticism of these models is that the normalization models are inadequate in some situations. An alternative approach is to normalize first e.g. using `loess()` for cDNA or quantile normalization for Affymetrix data, and to fit the gene expression models (2) above to the normalized data.

S+ArrayAnalyzer includes several methods for two sample comparisons and multi-sample comparisons including various flavors of t-tests and Wilcoxon tests with both distribution and permutation-based null distributions. The LPE method [1] is particularly suited to experiments with low replication. Additional methods for borrowing strength across genes are planned for the next release [16, 22]. The ANOVA and mixed effects models are readily fit in S-PLUS using `lme()` and examples of the models outlined above are provided with S+ArrayAnalyzer. A new fast ANOVA method has been recently developed; this can fit ANOVA models to many chips (e.g. 30-100+) and 12,000 genes in a few seconds.

No matter what test statistic is used, multiple comparisons are an important consideration given the number of genes and tests. Dudoit et al. [27] provide an overview in the context of microarray experiments. In a single test, one controls the type I error and chooses a rejection region that maximizes power (1 – type II error) while controlling type I error. In multiple tests, one can control the family wise error rate FWER=Pr(V>0), the false discovery rate FDR=E(V/R | R>0)*Pr(R>0) or the positive false discovery rate pFDR= E(V/R | R>0) where V=number of false positives and R=total number of genes declared significantly differentially expressed.

The simplest method to control the FWER is the Bonferroni method, in which a FWER is chosen and the p-value for a single test is multiplied by the number of tests and compared to the chosen FWER. In this scenario if the FWER is α, the g individual tests (one for each gene) have type I error α/g. The Bonferroni procedure is conservative and several step-down procedures (e.g. Westfall and Young [30]) have been proposed whereby the Bonferroni adjustment is used for the most extreme value of the test statistic/p-value for an individual test, and this adjustment is stepped down as the p-value becomes less extreme [28-30].

In microarray experiments the number of tests is often very large e.g. >10,000 and FWER control may be too strict. Control of FDR is a viable alternative. FDR was introduced by Benjamini and Hochberg [31] and has recently been extended to pFDR by Storey [32] who also introduced the notion of q-values as an error measure applying to observed statistics with respect to the pFDR, in the same way as the p-value provides this context with respect to the type I error and the adjusted p-value with respect to the FWER. Storey [32] and Storey and Tibshirani [33] both fix the rejection region and estimate the FDR. Reiner et al. [34] describe resampling-based methods for controlling FDR.

S+ArrayAnalyzer includes several methods for FWER and FDR control for all its multiple testing procedures. These include Bonferroni and various step-down FWER control procedures [26-28], as well as the FDR control procedures of Benjamini and Hochberg [29] and Benjamini and Yeketueli [35]. The LPE test includes a resampling-based method for controlling FDR [33, 34].

Results from differential expression analysis are typically presented as a gene list. This is managed as a dataframe in S+ArrayAnalyzer, sorted by adjusted p-values and including columns for means of experimental conditions as well as fold changes and raw p-values for each contrast between experimental conditions. The genelist dataframe is indexed by gene name and includes indices for associated metadata and for access to raw chip data.

The differential expression analysis is graphically presented as a volcano plot (Figure 6). This combines the adjusted p-value (y-axis) with this fold change (x-axis) and thus provides both statistical and biological perspectives.

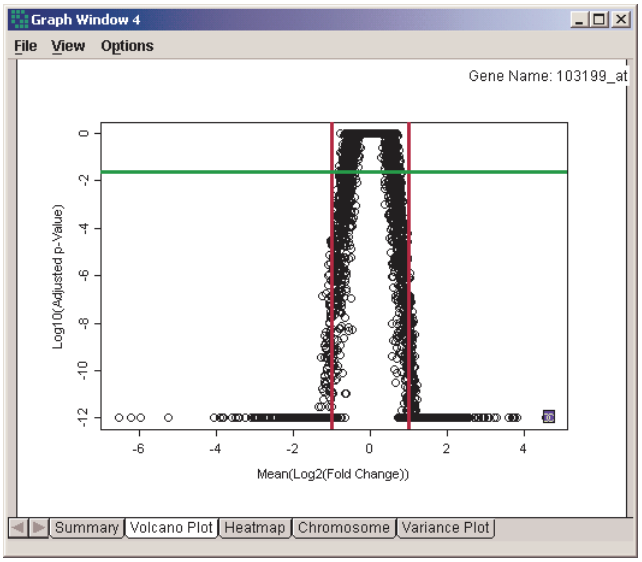

Figure 6. Volcano plot for the 1 df contrast time 0h vs. time 48h (LPE analysis of RMA-summarized expression data).

CLASS DISCOVERY AND CLASS PREDICTION

Cluster analysis has been a standard approach to microarray data since the beginnings of microarray technology and is the basis of most class discovery efforts. Hierarchical cluster analysis and resulting dendrograms represent distances between samples' expression profiles in a visually appealing manner and handle the many gene measurements on microarrays in a simple and concise manner. Hierarchical cluster analysis applied to genes, similarly summarizes and presents the many genes in a visually appealing way. Partitioning cluster analysis helps identify candidate subgroups within collection of samples. As such, both hierarchical and partitioning clustering have been widely used, particularly in the area of oncogenomics, in the identification and characterization of cancer subtypes.

When clinical data are available e.g. survival times, supervised analyses are possible and have been used more recently in class prediction studies. However, even when phenotype data are available, unsupervised clustering methods are often used to identify classes of samples/genes and to then relate these classes to the phenotype data.

Class discovery and class prediction studies have much potential in providing a molecular basis for tumor classification and more personalized treatments based on such classification. Current methods for classifying tumors rely on symptoms i.e. morphological and clinical variables, and patients with the same diagnosis often have very different treatment responses e.g. survival times. Expression intensities from microarrays may be used to characterize molecular variations among tumors. This can provide a finer and more reliable classification, and facilitate identification of marker genes that distinguish sub-classes. This more detailed classification can improve the understanding and prediction of cancer survival and targeting of treatments (e.g. drug, chemotherapy, surgery) to molecular subtypes, thus personalizing the cancer treatment.

Key early work in this area was done by Alizadeh et al. [36], Perou et al. [37] and Khan et al. [38]. Alizadeh et al. [36] identified subtypes of diffuse large B-cell lymphoma (DLBCL) using hierarchical clustering methods and related the subtypes to survival data. Perou et al. [37] did a similar study on breast cancer samples. Khan et al. [38]. studied small round blue cell tumors (SRBCTs) and used a neural network to predict four subtypes of SRBCTs. Tibshirani et al. [39] develop a nearest shrunken centroid method and applied this successfully to the Khan et al. [38] data. This simpler method also provides an intuitive list of genes that are used in predicting the classes. Similar, more recent studies involving pediatric acute lymphoblastic leukemia, BLBCL and breast cancer are presented in [40-42].

Another active area for class prediction methods is in the profiling of chemotherapeutic drugs. This involves the typing of new novel agents and development of pathway-targeted drugs where the microarray experimental data provides insight into the mode of drug action. Several systematic profiling studies have been done and there is much current proprietary research underway within pharmaceutical companies. A review is provided by O'Neill et al. [43] and includes work by Dan et al. [44], profiling/typing in cell lines: 55 drugs, 39 cancer lines, and Zembutsu et al. [45], profiling/typing in xenografts: 85 human cancer xenografts.

While many early cluster analyses of expression data were applied to all genes and all samples, this is not advisable from a statistical perspective – hierarchical clustering always finds structure, even with random noise. We recommend filtering the genes prior to clustering e.g. by including only significantly differentially expressed genes, genes with a minimum fold-change etc. Also, one may cluster other values besides the raw expression values; for example in time course experiments one may cluster the time contrast coefficients or t-statistics e.g. 1 df contrasts with baseline or polynomial time effects.

S+ArrayAnalyzer and S-PLUS include many methods for class discovery and class prediction. Cluster analysis methods include the library of algorithms described in Kaufman and Rousseeuw [46]. The partitioning methods include K-means – `kmeans()`, partitioning around medoids – `pam()`, the model-based methods – `Mclust()` and `EMclust()`, and a fuzzy clustering method in which probability of membership of each class is estimated – `fanny()`. A method for large datasets, `clara()`, is also included, which is based on `pam()`. The hierarchical methods include agglomerative methods (which start from individual points and successively merge clusters until one cluster representing the entire dataset remains) and divisive methods (which consider the whole dataset and split it until each object is separate). The available agglomerative methods are `agnes()` and `hclust()`. The available divisive methods are `diana()` and `mona()`. The `Mclust()` and `EMclust()` methods assume that data are generated from an underlying mixture of Gaussian distributions; and provide an estimate of the number of clusters using the Bayes information criterion (BIC) for each model considered. The `Mclust()` and `EMclust()` methods are available in the `Mclust` library (Fraley and Raftery [47]) from http://www.stat.washington.edu/fraley/mclust/. The `pam()` method also allows automatic estimation of the number of clusters using the average silhouette width for each model considered. Self organizing maps [48] are available in the S-PLUS `class` library.

Class prediction methods available in S-PLUS include all of the standard statistical modeling and supervised learning methods e.g. regression, discriminant functions, trees (CART and recursive partitioning), neural nets, generalized additive models, support vector machines. Many of these methods are available in the S-PLUS libraries `MASS` and `class` as described in Venables and Ripley [49].

Graphical summaries of hierarchical cluster analysis of genes and samples are provided in Figure 7; and from partitioning cluster analysis of the genes in Figures 8-10.

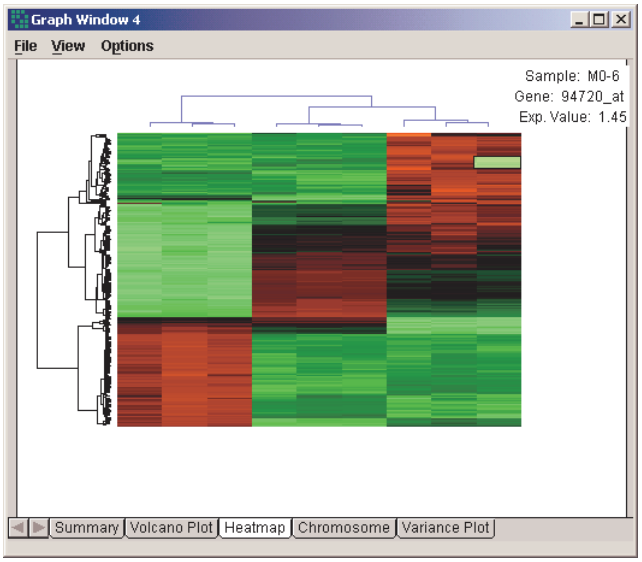

Figure 7. Heatmap and dendrogram from hierarchical cluster analysis of significant genes from LPE analysis. Samples appear from left to right as MD4, M48, M0 in groups of the 3 replicates.

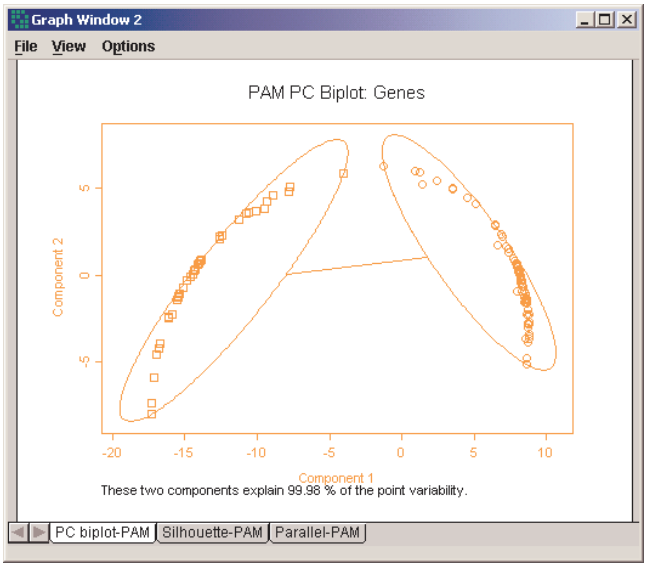

Figure 8. Partitioning cluster analysis (pam) of significant genes from 2 d.f. F-test in one-way ANOVA analysis. Principal component biplot shows clear separation into 2 major classes.

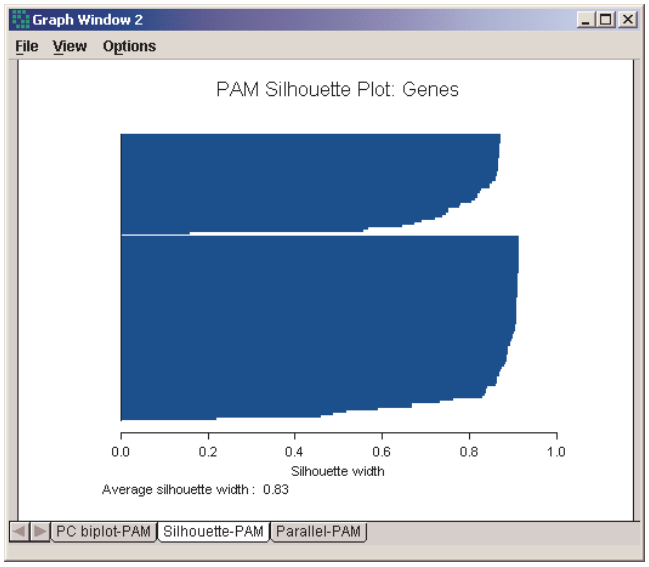

Figure 9. Partitioning cluster analysis (pam) of significant genes from 2 d.f. F-test in one-way ANOVA analysis. Silhouette plot shows excellent 2-class cluster analysis fit.

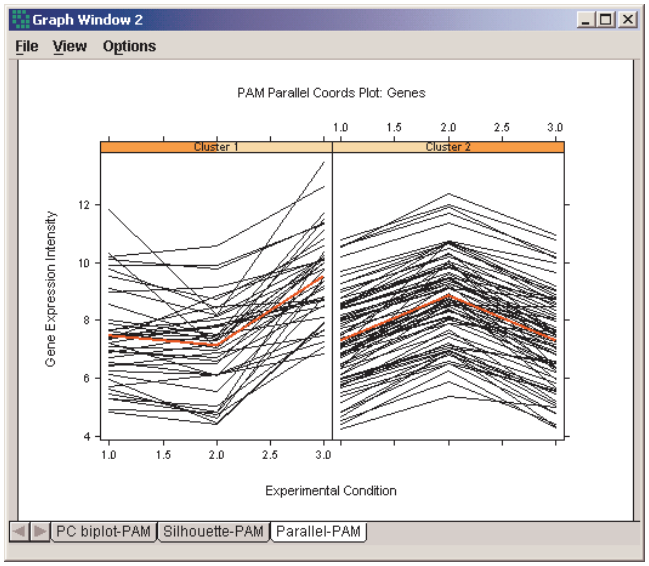

Figure 10. Partitioning cluster analysis (pam) of significant genes from 2 d.f. F-test in one-way ANOVA analysis. Parallel coordinates plot shows expression intensity patterns for the 2 major classes.

ANNOTATION

There is a great deal of annotation metadata available for any given gene. Examples include LocusLink, Unigene, chromosome number, chromosomal location (cytoband or bp), KEGG pathway information and Gene Ontology (GO) categorizations. A microarray dataset typically includes a set of known identifiers corresponding to the probes/probesets used. These identifiers are typically unique for any manufacturer or spotted array system;

and can be simply linked to identifiers for the metadata sources, so that each probe/probeset on a microarray chip has readily available identifiers for looking up metadata information in the various annotation databases.

Many online databases (Unigene, LocusLink, GO/Amigo) support querying on the URL. S+ArrayAnalyzer includes S-PLUS functions that push database identifiers for probes/probesets to online databases and open a browser window with the gene annotation information displayed. Other sites e.g. Affymetrix GO browser, require a list of Affymetrix ID's to be uploaded. In this case S+ArrayAnalyzer has an S-PLUS function that writes the relevant IDs to a file that can be uploaded to the Affymetrix GO browser. S+ArrayAnalyzer includes libraries with annotation identifiers for most common Affymetrix chips e.g. HGU95*, HGU133*, HU6800, MGU74*, MOE430*, RGU34*, RAE230*. Figure 11 shows results from uploading an identified gene list to the Affymetrix GO browser. The uploaded list comprised 11 genes from the time0 vs. time48 RMA/LPE analysis, filtered with fold change > 3 and adjusted p-value < 0.05.

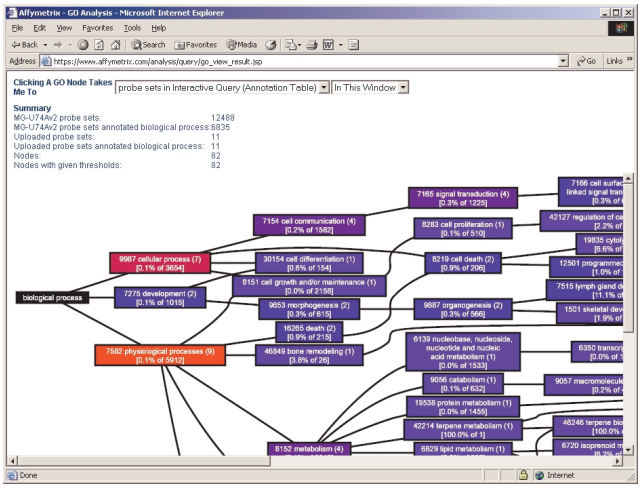

Figure 11. Annotation of 11 genes (fold change > 3, adjusted p-value < 0.05) using Affymetrix GO browser. S+ArrayAnalyzer writes out file that is uploaded to the Affymetrix GO browser website.

There are now several websites and web-based applications that merge annotation information from a variety of sources and provide annotation services for an uploaded gene list. Of note is the Onto-Express family of applications as described by Draghici [50] at: http://vortex.cs.wayne.edu/Projects.html.

Figure 12 shows results from uploading an identified gene list to the LocusLink website. The uploaded list again comprised the same 11 genes, filtered with fold change > 3 and adjusted p-value < 0.05. S+ArrayAnalyzer includes functions that query the LocusLink and Unigene websites on the URL with a user-defined, filtered gene list. S+ArrayAnalyzer also includes functions for importing PubMed abstracts into S-PLUS for further analysis.

The annotation functions referred to above are also simple to launch from the S+ArrayAnalyzer command line, given a set of gene identifiers `gnames`:

```
> #LocusLink
llnames <- as.numeric(mgu74aLOCUSID[gnames])
locuslinkByID(llnames)
> #Unigene
accids <- unlist(mgu74aACCNUM[gnames])
genbank(accids, disp="browser")
> #Pubmed
pmedids < mgu74aPMID[gnames[1]]
pubmed(pmedids, disp="browser")
> #GO
genelist.GOids <- mgu74aGO[gnames]
browsego(genelist.GOids)
```

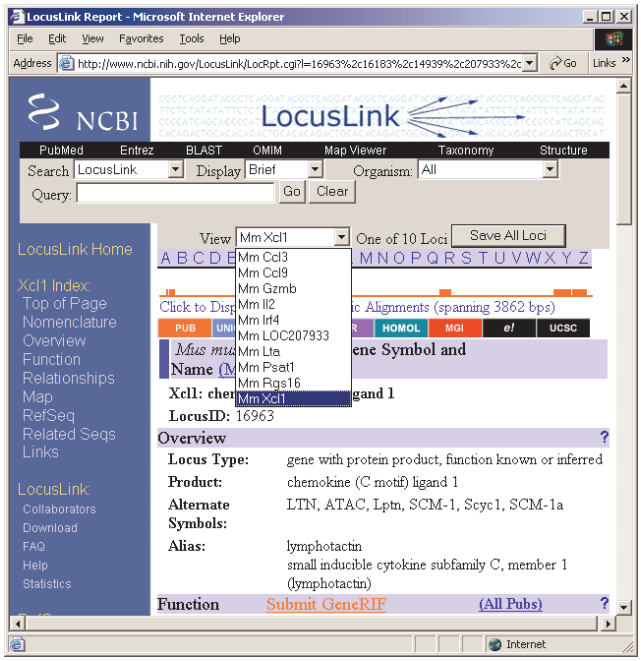

Figure 12. Annotation of 11 genes (fold change > 3, adjusted p-value < 0.05) using LocusLink website. S+ArrayAnalyzer includes functions that query LocusLink, Unigene and other annotation web databases, on the URL with a user-defined filtered gene list.

S+ArrayAnalyzer makes it easy to send lists of genes resulting from significance testing (e.g. LPE, ANOVA) and/or cluster analysis to the various annotation web sites referred to above.

SOFTWARE AND DEPLOYMENT

S+ArrayAnalyzer is an add-on module to S-PLUS and can be run by a single user as part of S-PLUS for Windows or by multiple users through a web user interface. S+ArrayAnalyzer includes much of the Biconductor functionality (www.bioconductor.org), as well as methods developed at Insightful. S+ArrayAnalyzer includes a user interface with point and click workflow functionality for data management, pre-processing, differential expression and clustering as well as tabular and graphical reporting and annotation. There is, of course, much functionality that is available through S-PLUS scripting that is not exposed in the S+ArrayAnalyzer user interface. S+ArrayAnalyzer includes several such worked examples in documentation and scripts; and through an extensive help system.

S+ArrayAnalyzer is simply deployed through a web user interface running on Solaris, AIX, Linux and Windows. The web user interface is similar to the S-PLUS for Windows interface and includes a wizard-style walk through of user-defined options. Both the desktop and web-server interfaces are very simply customized using S-PLUS functions on the desktop, and javascript and related tools on the web implementation. An optimal configuration for S+ArrayAnalyzer is a situation with 1-5 power users using S-PLUS and S+ArrayAnalyzer on the desktop, and managing deployment of the web-based S+ArrayAnalyzer to a community of scientists.

ACKNOWLEDGMENTS

Thanks to the great work going on in the Bioconductor community and to the Bioconductor core team. Thanks also to Nitin Jain, Jayant Thatte, Thomas Braciale, Klaus Ley and Jae K. Lee from University of Virginia for the work on the mouse immune response study that is referenced in this paper; and to Nitin Jain and Jae K. Lee for work on the LPEtest library. Thanks also to Stephen Kaluzny, Bob Treder, Peter McKinnis, Lou Bajuk, Bill Dunlap and the rest of the Insightful ArrayAnalyzer team.

REFERENCES

[1] Jain, N., Thatte, J., Braciale, T., Ley, K., O'Connell, M. and Lee, J.K. (2003). Local-pooled-error test for identifying differentially expressed genes with a small number of replicated microarrays. *Bioinformatics* 19: 1945-1951.

[2] Pan, W., Lin J. and Le, C. (2002). How many replicates of arrays are required to detect gene expression changes in microarray experiments? A mixture model approach. *Genome Biology*, 3,5: research0022.1- research0022.

[3] Kerr, K. and Churchill, G. (2001). Experimental design for gene expression microarrays, *Biostatistics*, 2:183-201.

[4] Dobbin, K. and Simon, R. (2002). Comparison of microarray designs for class comparison and class discovery. *Bioinformatics* 18:1462-1469.

[5] Yang, Y. H., Buckley, M. J., Dudoit, S. and Speed, T. P. (2002) Comparison of methods for image analysis on cDNA microarray data. *Journal of Computational and Graphical Statistics* 11: 108-136.

[6] Affymetrix Inc. Statistical algorithms description document. (2002). www.affymetrix.com/support/technical/whitepapers/sadd_whitepaper.pdf

[7] Wu, Z., Irizarry, R. A., Gentleman, R., Murillo, F. M. and Spencer, F. (2003). A model based background adjustment for oligonucleotide expression arrays. Technical report, Johns Hopkins University, Dept. of Biostatistics Working Papers. (www.bepress.com/jhubiostat/paper1)

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

[9] Li, C. and Wong, W. (2001). Model-based analysis of oligonucleotide arrays: Expression index computation and outlier detection. *Proceedings of the National Academy of Science USA* 98: 31–36.

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

[11] Zhang, L., Wang, L., Ravindranathan, A., and Miles, M. (2002). A new algorithm for analysis of oligonucleotide arrays: application to expression profiling in mouse brain regions. *Journal of Molecular Biology* 317: 227–235.

[12] Wu, Z., LeBlanc, R. and Irizarry, R. A., Stochastic Models Based on Molecular Hybridization Theory for Short Oligonucleotide Microarrays Technical report, Johns Hopkins University, Dept. of Biostatistics Working Papers. (www.bepress.com/jhubiostat/paper4)

[13] Bolstad, B., A., I. R., Astrand, M., and Speed, T. (2003). A comparison of normalization methods for high density oligonucleotide array data based on bias and variance. *Bioinformatics* 19,2: 185-193.

[14] Naef, F. and Magnasco, M. O. (2003). Solving the riddle of the bright mismatches: labeling and effective binding in oligonucleotide arrays. *Physical Review E* 68, 011906.

[15] Dudoit, S., Yang, Y. H., Callow, M. J. and Speed, T. P. (2002). Statistical methods for identifying differentially expressed genes in replicated cDNA microarray experiments, *Statistica Sinica*, 12, 1: 111-139.

[16] Durbin, B. Hardin, J., Hawkins, D. and Rocke, D. (2002). A Variance-Stabilizing Transformation for Gene-Expression Microarray Data, *Bioinformatics* 18, Number Supplemental 1, pp S105-S110.

[17] Huber, W., von Heydebreck, A., Sultmann, H., Poustka, A., and Vingron, M. (2002). Variance stabilization applied to microarray data calibration and to the quantification of differential expression. *Bioinformatics* 1: 1:9.

[18] Lin, Y., Nadler, S. T., Attie, A. D., and Yandell, B. S. (2003) Adaptive gene picking with microarray data: detecting important low abundance signals. in *The Analysis of Gene Expression Data: Methods and Software*, edited by G Parmigiani, ES Garrett, RA Irizarry, SL Zeger. Springer-Verlag, ch. 12, Springer-Verlag.

[19] Efron B., Tibshirani R., Storey J.D., Tusher V. (2001). Empirical Bayes analysis of a microarray experiment. *Journal of the American Statistical Association* 96: 1151-1160.

[20] Baldi, P. and Long, A. D. (2001) A Bayesian framework for the analysis of microarray expression data: regularized t-test and statistical inferences of gene changes, *Bioinformatics* 17: 509-51.

[21] Newton, M., Kendziorski, C., Richmond, C., and Blattner, F. (2001). On differential variability of expression ratios: improving statistical inference about gene expression changes from microarray data. *Journal of Computational Biology* 8: 37-52.

[22] Lonnstedt, I. and Speed, T. (2002). Replicated microarray data. *Statistica Sinica* 12: 31-46.

[23] Tusher V, Tibshirani R, Chu G (2001). Significance analysis of microarrays applied to the ionizing radiation response. *Proc Natl Acad Sci* USA 98: 5116-5121.

[24] Kerr, M.K., Martin, M. and Churchill, G.A. (2000). Analysis of variance for gene expression microarray data, *Journal of Computational Biology* 7: 819.

[25] Wolfinger, R., Gibson, G.,Wolfinger, E., Bennett, L., Hamadeh, H., Bushel, P., Afshari, C., and Paules, R. (2001). Assessing gene significance from cDNA microarray expression data via mixed models. *Journal of Computational Biology* 8,6: 625–637.

[26] Chu, T., Weir, B. and Wolfinger, R. (2002). A systematic statistical linear modeling approach to oligonucleotide array experiments. *Mathematical Biosciences* 176: 35–51.

[27] Dudoit, S., Yang, Y., Callow, M., and Speed, T. (2002). Statistical methods for identifying differentially expressed genes in replicated cDNA microarray experiments. *Statistica Sinica* 12: 111-139.

[28] Holm, S. (1979). A simple sequentially rejective multiple test procedure. *Scandinavian Journal of Statistics.* 6: 65-70.

[29] Hochberg, Y. (1988). A sharper Bonferroni procedure for multiple tests of significance. *Biometrika* 75: 800-802.

[30] Westfall, P. H. and Young, S. S. *Resampling-based multiple testing: Examples and methods for p-value adjustment*. John Wiley & Sons, 1993.

[31] Benjamini Y, Hochberg Y (1995). Controlling the false discovery rate: A practical and powerful approach to multiple testing. *Journal of the Royal Statistical Society, Series B, Methodological* 57: 289-300.

[32] Storey J. D. (2002). A direct approach to false discovery rates. *Journal of the Royal Statistical Society*, Series B 64: 479-498.

[33] Storey J. and Tibshirani, R. (2003). SAM Thresholding and False Discovery Rates for Detecting Differential Gene Expression in DNA Microarrays, in *The Analysis of Gene Expression Data: Methods and Software*, edited by G Parmigiani, ES Garrett, RA Irizarry, SL Zeger. Springer-Verlag, ch. 12. Springer-Verlag.

[34] Reiner, A., Yekutieli, D. and Benjamini, Y. (2003). Identifying differentially expressed genes using false discovery rate controlling procedures. *Bioinformatics* 19: 368-375.

[35] Benjamini, Y., Yekutieli, D. (2001). The control of the false discovery rate in multiple hypothesis testing under dependency. *Annals of Statistics* 29,4: 1165-1188.

[36] Alizadeh AA, Eisen MB, Davis RE, Ma C, Lossos IS, Rosenwald A, Boldrick JC, Sabet H, Tran T, Yu X, Powell JI, Yang L, Marti GE, Moore T, Jr JH, Lu L, Lewis DB, Tibshirani R, Sherlock G, Chan WC, Greiner TC, Weisenburger DD, Armitage JO, Warnke R, Levy R, Wilson W, Grever MR, Byrd JC, Botstein D, Brown PO, Staudt LM (2000). Distinct types of diffuse large B-cell lymphoma identified by gene expression profiling. *Nature* 403: 503-511.

[37] Perou, C.M., Sorlie, T., Eisen, M.B., van de Rijn, M., Jeffrey, S.S., Rees, C.A., Pollack, J.R., Ross, D.T. et al. (2000). Molecular portraits of human breast tumors. *Nature* 406: 747-752.

[38] Khan, J., Wei, J., Ringner, M., Saal, L., Ladanyi, M.,Westermann, F., Berthold, F., Schwab, M., Antonescu, C., Peterson, C. (2001) *Nat. Med.* **7,** 673–679. 1799–1810.

[39] Tibshirani R, Hastie T, Narasimhan B, Chu G (2002). Diagnosis of multiple cancer types by shrunken centroids of gene expression.Proc. *Natl. Acad. Sci.* USA 99: 6567-6572.

[40] Yeoh, E.J., Ross, M.E., Shurtleff, S.A., Williams, W.K., Patel, D., Mahfouz, R., Behm, F.G., Raimondi, S.C. et al. (2002). Classification, subtype discovery and prediction of outcome in pediatric acute lymphoblastic leukemia by gene expression profiling. *Cancer Cell* 1: 133-143.

[41] Rosenwald, A., Wright, G., Chan, W.C., Connors, J.M., Campo, E., Fisher, R.I., Gascoyne, R.D., Muller-Hermelink, H.K. et al. (2002). The use of molecular profiling to predict survival after chemotherapy for diffuse large B-cell lymphoma. *N Engl. J Medicine* 346: 1937-1947

[42] van't Veer, L.J., Dai, H., van de Vijver, M.J., He, Y.D., Hart, A.A., Mao, M., Peterse, H.L., van der Kooy, K. (2002). Gene expression profiling predicts clinical outcome of breast cancer. *Nature* 415: 530-536.

[43] O'Neill, G.M., Catchpole, D.R. and Golemis, E.A. (2003). From correlation to causality: microarrays, cancer and cancer treatment. *Biotechniques* 34: S64-S71.

[44] Dan, S., Tsunoda, T., Kitahara, O., Yanagawa, R., Zembutsu, H., Katagiri, T., Yamazaki, K., Nakamura, Y. and Yamori, T. (2002). An integrated database of chemosensitivity to 55 anticancer drugs and gene expression profiles of 39 huan cancer cell lines. *Cancer Research* 62: 1139-1147.

[45] Zembutsu, H., Ohnishi, Y., Tsunoda, T., Furukawa, Y., Katagiri, T., Ueyama, Y., Tamaoki, N., Nomura, T. et al. (2002). Genome wide cDNA microarray screening to correlate gene expression profiles with sensitivity of 85 human cancer xenografts to anticancer drugs. *Cancer Research* 62: 518-527.

[46] Kaufmann L, Rousseeuw PJ (1990). *Finding Groups in Data: An Introduction to Cluster Analysis*. Wiley, NY.

[47] Fraley C. and Raftery A. E. (2002). MCLUST: Software for Model-Based Clustering, Discriminant Analysis and Density Estimation. Technical Report no. 415, Department of Statistics, University of Washington.

[48] Kohonen T (1995). *Self Organizing Maps*. Springer, NY.

[49] Venables, W.N. and Ripley, B.D. (2002). *Modern Applied Statistics with S*. Springer, NY.

[50] Draghici, S. (2003). *Data Analysis Tools for DNA Microarrays.* Chapman and Hall, London.

ABOUT THE AUTHOR:

Michael O'Connell is Director, Biopharmaceutical Solutions at Insightful Corp. and is product manager for S+ArrayAnalyzer and other Life Sciences applications. He earned a Ph.D. in statistics from North Carolina State University.