Statistics for Biology and Health

Robert Gentleman
Vincent J. Carey
Wolfgang Huber
Rafael A. Irizarry
Sandrine Dudoit

Editors

Bioinformatics and Computational Biology Solutions Using R and Bioconductor

With 128 Illustrations

Springer

Editors
Robert Gentleman
Program in Computational Biology
Division of Public Health Sciences
Fred Hutchinson Cancer Research Center
1100 Fairview Ave. N, M2-B876
PO Box 19024
Seattle, Washington 98109-1024 USA
rgentlem@fhcrc.org

Vincent J. Carey
Channing Laboratory
Brigham and Women's Hospital
Harvard Medical School
181 Longwood Ave Boston MA 02115 USA
stvjc@channing.harvard.edu

Wolfgang Huber
European Bioinformatics Institute
European Molecular Biology Laboratory
Cambridge, CB10 1SD UK
huber@ebi.ac.uk

Rafael A. Irizarry
Department of Biostatistics
Johns Hopkins Bloomberg School of Public Health
615 North Wolfe Street
Baltimore, MD 21205 USA
rafa@jhu.edu

Sandrine Dudoit
Division of Biostatistics
School of Public Health
University of California, Berkeley
140 Earl Warren Hall, #7360
Berkeley, CA 94720-7360 USA
sandrine@stat.berkeley.edu

Series Editors
Wing Wong
Department of Statistics
Stanford University
Stanford, CA 94305
USA

M. Gail
National Cancer Institute
Rockville, MD 20892
USA

K. Krickeberg
Le Châtelet
F-63270 Manglieu
France

A. Tsiatis
Department of Statistics
North Carolina State University
Raleigh, NC 27695
USA

J. Samet
Department of Epidemiology
School of Public Health
Johns Hopkins University
615 Wolfe Street
Baltimore, MD 21205
USA

Library of Congress Control Number: 2005923843

ISBN-10: 0-387-25146-4 Printed on acid-free paper.
ISBN-13: 978-0387-25146-2

Printed in China. (EVB)

9 8 7 6 5 4 3 2 1

springeronline.com

Preface

During the past few years, there have been enormous advances in genomics and molecular biology, which carry the promise of understanding the functioning of whole genomes in a systematic manner. The challenge of interpreting the vast amounts of data from microarrays and other high throughput technologies has led to the development of new tools in the fields of computational biology and bioinformatics, and opened exciting new connections to areas such as chemometrics, exploratory data analysis, statistics, machine learning, and graph theory.

The Bioconductor project is an open source and open development software project for the analysis and comprehension of genomic data. It is rooted in the open source statistical computing environment R. This book's coverage is broad and ranges across most of the key capabilities of the Bioconductor project. Thanks to the hard work and dedication of many developers, a responsive and enthusiastic user community has formed. Although this book is self-contained with respect to the data processing and data analytic tasks covered, readers of this book are advised to acquaint themselves with other aspects of the project by touring the project web site `www.bioconductor.org`.

This book represents an innovative approach to publishing about scientific software. We made a commitment at the outset to have a fully *computable book*. Tables, figures, and other outputs are dynamically generated directly from the experimental data. Through the companion web site, `www.bioconductor.org/mogr`, readers have full access to the source code and necessary supporting libraries and hence will be able to see how every plot and statistic was computed. They will be able to reproduce those calculations on their own computers and should be able to extend most of those computations to address their own needs.

Acknowledgments

This book, like so many projects in bioinformatics and computational biology, is a large collaborative effort. The editors would like to thank the chapter authors for their dedication and their efforts in producing widely used software, and also in producing well-written descriptions of how to use that software.

We would like to thank the developers of R, without whom there would be no Bioconductor project. Many of these developers have provided additional help and engaged in discussions about software development and design. We would like to thank the many Bioconductor developers and users who have helped us to find bugs, think differently about problems, and whose enthusiasm has made the long hours somewhat more bearable.

We would also like to thank Dorit Arlt, Michael Boutros, Sabina Chiaretti, James MacDonald, Meher Majety, Annemarie Poustka, Jerome

Ritz, Mamatha Sauermann, Holger Sültmann, Stefan Wiemann, and Seth Falcon, who have contributed in many different ways to the production of this monograph. Much of the preliminary work on the MLInterfaces package, described in Chapter 16, was carried out by Jess Mar, Department of Biostatistics, Harvard School of Public Health. Ms Mar's efforts were supported in part by a grant from Insightful Corporation.

The Bioconductor project is supported by grant 1R33 HG002708 from the NIH as well as by institutional funds at both the Dana Farber Cancer Institute and the Fred Hutchinson Cancer Research Center. W.H. received project-related funding from the German Ministry for Education and Research through National Genome Research Network (NGFN) grant FKZ 01GR0450.

Seattle — *Robert Gentleman*
Boston — *Vincent Carey*
Cambridge (UK) — *Wolfgang Huber*
Baltimore — *Rafael Irizarry*
Berkeley — *Sandrine Dudoit*
February 2005

J. Gentry, Center for Cancer Research, Massachusetts General Hospital, Boston, MA, USA

F. Hahne, Division of Molecular Genome Analysis, German Cancer Research Center, Heidelberg, FRG

L. Harris, Department of Cancer Biology, Dana Farber Cancer Institute, Boston, MA, USA

T. Hothorn, Institut für Medizininformatik, Biometrie und Epidemiologie, Friedrich-Alexander-Universität Erlangen-Nürnberg, FRG

W. Huber, European Molecular Biology Laboratory, European Bioinformatics Institute, Cambridge, UK

J. Ibrahim, Department of Biostatistics, University of North Carolina, Chapel Hill, NC, USA

J. D. Iglehart, Department of Cancer Biology, Dana Farber Cancer Institute, Boston, MA, USA

R. A. Irizarry, Department of Biostatistics, Johns Hopkins Bloomberg School of Public Health, Baltimore, MD, USA

X. Li, Department of Biostatistics and Computational Biology, Dana Farber Cancer Institute, Boston, MA, USA

X. Lu, Department of Biostatistics, Harvard School of Public Health, Boston, MA, USA

A. Miron, Department of Cancer Biology, Dana Farber Cancer Institute, Boston, MA, USA

A. C. Paquet, Department of Biostatistics, University of California, San Francisco, CA, USA

K. S. Pollard, Center for Biomolecular Science and Engineering, University of California, Santa Cruz, USA

D. Scholtens, Department of Preventive Medicine, Northwestern University, Chicago, IL, USA

Q. Shi, Department of Cancer Biology, Dana Farber Cancer Institute, Boston, MA, USA

Contents

List of Contributors

B. M. Bolstad, Department of Statistics, University of California, Berkeley, CA, USA

J. Brettschneider, Department of Statistics, University of California, Berkeley, CA, USA

P. Buhlmann, Swiss Federal Institute of Technology, Zürich, CH

V. J. Carey, Channing Laboratory, Brigham and Women's Hospital, Harvard Medical School, Boston, MA, USA

F. Collin, Department of Statistics, University of California, Berkeley, CA, USA

L. Cope, Division of Oncology Biostatistics, The Sidney Kimmel Comprehensive Cancer Center, Johns Hopkins Medical School, Baltimore, MD, USA

M. Dettling, Division of Oncology and Biostatistics, The Sidney Kimmel Comprehensive Cancer Center, Johns Hopkins Medical School, Baltimore, MD, USA

B. Ding, Medical Affairs Biostatistics, Amgen Inc., Thousand Oaks, CA, USA

S. Dudoit, Department of Biostatistics, University of California, Berkeley, CA, USA

L. Gautier, Independent investigator, Copenhagen, DK

R. Gentleman, Program in Computational Biology, Fred Hutchinson Cancer Research Center, Seattle, WA, USA

K. Simpson, The Walter and Eliza Hall Institute of Medical Research, Melbourne, Australia

C. A. Smith, Department of Molecular Biology, The Scripps Research Institute, La Jolla, CA, USA

G. K. Smyth, The Walter and Eliza Hall Institute of Medical Research, Melbourne, Australia

T. P. Speed, Department of Statistics, University of California, Berkeley, CA, USA

D. Temple Lang, Department of Statistics, University of California, Davis, CA, USA

M. J. van der Laan, Department of Biostatistics, University of California, Berkeley, CA, USA

A. von Heydebreck, Global Technologies, Merck KGaA, Darmstadt, FRG

Z. Wu, Department of Biostatistics, Johns Hopkins Bloomberg School of Public Health, Baltimore, MD, USA

Y. H. Yang, Department of Biostatistics, University of California, San Francisco, CA, USA

J. Zhang, Department of Medical Oncology, Dana-Farber Cancer Institute, Boston, MA, USA

Part I

Preprocessing data from genomic experiments

1

Preprocessing Overview

W. Huber, R. A. Irizarry, and R. Gentleman

Abstract

In this chapter, we give a brief overview of the tasks of microarray data preprocessing. There are a variety of microarray technology platforms in use, and each of them requires specific considerations. These will be described in detail by other chapters in this part of the book. This overview chapter describes relevant data structures, and provides with some broadly applicable theoretical background.

1.1 Introduction

Microarray technology takes advantage of hybridization properties of nucleic acid and uses complementary molecules attached to a solid surface, referred to as *probes*, to measure the quantity of specific nucleic acid transcripts of interest that are present in a sample, referred to as the *target*. The molecules in the target are labeled, and a specialized scanner is used to measure the amount of hybridized target at each probe, which is reported as an intensity. Various manufacturers provide a large assortment of different platforms. Most manufacturers, realizing the effects of optical noise and non-specific binding, include features in their arrays to directly measure these effects. The raw or *probe-level* data are the intensities read for each of these components. In practice, various sources of variation need to be accounted for, and these data are heavily manipulated before one obtains the genomic-level measurements that most biologists and clinicians use in their research. This procedure is commonly referred to as *preprocessing*.

The different platforms can be divided into two main classes that are differentiated by the type of data they produce. The *high-density oligonucleotide array* platforms produce one set of probe-level data per microarray with some probes designed to measure specific binding and others to measure non-specific binding. The two-color spotted platforms produce two sets

of probe-level data per microarray (the red and green channels), and local background noise levels are measured from areas in the glass slide not containing probe.

Despite the differences among the different platforms, there are some tasks that are common to all microarray technology. These tasks are described in Section 1.2. The data structures needed to effectively preprocess microarray data are described in Section 1.3. In Section 1.4 we present statistical background that serves as a mathematical framework for developing preprocessing methodology. Detailed description of the preprocessing tasks for this platforms are described in Chapters 2 and 3. The specifics for the two-color spotted platforms are described in Chapter 4. Chapters 5 and 6 describe preprocessing methodology for related technologies where similar principles apply.

1.2 Tasks

Preprocessing can be divided into 6 tasks: image analysis, data import, background adjustment, normalization, summarization, and quality assessment. *Image analysis* permits us to convert the pixel intensities in the scanned images into probe-level data. Flexible data import methods are needed because data come in different formats and are often scattered across a number of files or database tables from which they need to be extracted and organized. *Background adjustment* is essential because part of the measured probe intensities are due to non-specific hybridization and the noise in the optical detection system. Observed intensities need to be adjusted to give accurate measurements of specific hybridization. Without proper *normalization*, it is impossible to compare measurements from different array hybridizations due to many obscuring sources of variation. These include different efficiencies of reverse transcription, labeling, or hybridization reactions, physical problems with the arrays, reagent batch effects, and laboratory conditions. In some platforms, *summarization* is needed because transcripts are represented by multiple probes. For each gene, the background adjusted and normalized intensities need to be summarized into one quantity that estimates an amount proportional to the amount of RNA transcript. *Quality assessment* is an important procedure that detects divergent measurements beyond the acceptable level of random fluctuations. These data are usually flagged and not used, or down weighted, in subsequent statistical analyses.

The complex nature of microarray data and data formats makes it necessary to have flexible and efficient statistical methodology and software. This part of the book describes what Bioconductor has to offer in this capacity. In the rest of this section, we describe prerequisites necessary to perform these tasks and two general approaches to preprocessing.

1.2.1 Prerequisites

A number of important steps are involved in the generation of the raw data. The experimental design includes the choice and collection of samples (tissue biopsies or cell lines exposed to different treatments); the choice of probes and array platform; the choice of controls, RNA extraction, amplification, labeling, and hybridization procedures; the allocation of replicates; and the scheduling of the experiments. The experimental design must take into account technical, logistic, and financial boundary conditions. Its quality determines to a large extent the utility of the data. A fundamental guideline is the avoidance of *confounding* between different biological factors of interest or between a biological factor of interest and a technical factor that is anticipated to affect the measurements. The experiment then has to be carried out, which requires great skill and expertise.

In the *image analysis* step, we extract probe intensities out of the scanned images containing pixel-level data. The arrays are scanned by the detector at a high spatial resolution to produce a digitized image in which each probe is represented by dozens of pixels. To obtain a single overall intensity value for each probe, the associated pixels need to be identified (segmentation) and their intensities summarized (quantification). In addition to the overall probe intensity, further auxiliary quantities may be calculated, such as an estimate of apparent unspecific "local background" intensity, or spot quality measures. Various software packages offer different segmentation and quantification methods. They differ in their robustness against irregularities and in the amount of human interaction that they require. The different platforms present different problems which implies that the types of image analysis algorithms used are quite different. Currently, Bioconductor does not offer image processing software. Thus, the user will need alternative software to process the image pixel-level data. However, import functions that are compatible with most of the existing image analysis products are available. For an evaluation of image analysis methods for two-color spotted arrays see, for example, the study of Yang et al. (2002a). Details on image analysis methodology for high-density oligonucleotide arrays were described by Schadt et al. (2001).

1.2.2 Stepwise and integrated approaches

The *stepwise* approach to microarray data preprocessing starts with probe-level data as input, performs the tasks sequentially and produces an *expression matrix* as output. In this matrix, rows correspond to gene transcripts, columns to conditions, and each element represents the abundance or relative abundance of a transcript. Subsequent biological analyses work off the expression matrix and generally do not consider the statistical manipulations performed on the probe-level data. The preprocessing task are divided into a set of sequential instructions: for example, subtract

the background, then normalize the intensities, then summarize replicate probes, then summarize replicate arrays. The modularity of this approach allows us to structure the analysis work-flow. Software, data structures, and methodology can be easily re-used. For example, the same machine learning algorithm can be applied to an expression matrix irrespective of whether the data were obtained on high-density oligonucleotide chips or two-color spotted arrays. A potential disadvantage of the stepwise approach is that each step is independently optimized without considering the effect of previous or subsequent steps. This could lead to sub-optimal bottom-line results.

In contrast, *integrated* approaches solve specific problems by carrying out the analysis in one unified estimation procedure. This approach has the potential of using the available data more efficiently. For example, rather than calculating an expression matrix, one might fit an ANOVA-type linear model to the probe-level data, which includes both technical covariates, such as dye and sample effects, and biological covariates, such as treatment effects (Kerr et al., 2000). In the affyPLM package, the weighting and summarization of the multiple probes per transcript on Affymetrix chips is integrated with the detection of differential expression. Another example is the vsn method (Huber et al., 2002), which integrates background subtraction and normalization in a non-linear model.

Stepwise approaches are often presented as modular data processing pipelines; integrated approaches are motivated by statistical models with parameters representing quantities of interest. In practice, data analysts will often choose to use a combination of both approaches. For example, a researcher may start with the stepwise approach and do a first round of high-level analyses that motivates an integrated approach that is applied to obtain final results. Bioconductor software allows users to explore, adapt, and combine stepwise and integrated methods.

1.3 Data structures

1.3.1 Data sources

The basic data types that we deal with in microarray data preprocessing are probe and background intensities, probe annotations, array layout, and sample annotations. Typically, they come in the form of rectangular tables, stored either in flat files or in a database server. The probe intensities are the result of image processing. The format in which they are reported varies between different vendors of image processing software. Examples are discussed in Sections 2 and 4.

The probe annotations are usually provided by the organization that selected the probes for the array. This may be a commercial vendor, another laboratory, or the experimenters themselves. For high-density oligonucleotide arrays, the primary annotation is the sequence. In addition, there

may be a database identifier of the gene transcript that the probe is intended to match and possibly the exact location. Often, the probe sequences are derived from cDNA sequence clusterings such as Unigene (Pontius et al., 2003). For spotted cDNA arrays, the primary probe identifier is often a clone ID in a nucleotide sequence database. The largest public nucleotide sequence databases are EMBL in Europe, DDBJ in Japan, and Genbank in the United States. Through a system of cross-mirroring, their contents are essentially equivalent. These databases contain full or partial sequences of a large number of expressed sequences. Their clone identifiers can be mapped to genomic databases such as Entrez Gene, H-inv, or Ensembl. Further annotations of the genes that are represented by the probes are provided by various genomic database, for example genomic locus, disease associations, participation in biological processes, molecular function, cellular localization. This will be discussed in Part II of the book.

The array layout is provided by the organization that produced the array. As a minimum, the layout specifies the physical position of each probe on the array. In principle, this can be done through its x- and y-coordinates. For spotted arrays, it is customary to specify probe coordinates through three coordinates: *block*, *row*, and *column*, where the *block* coordinate addresses a particular sub-sector of the array, and the *row* and *column* coordinates address the probe within that sub-sector. Details are discussed in Sections 2 and 4.

The sample annotations describe the labeled cDNA that has been hybridized to the array. This includes technical information on processing protocols (e.g., isolation, amplification, labeling, hybridization) as well as the biologically more interesting covariates such as treatment conditions and harvesting time points for cell lines or histopathological and clinical data for tissue biopsies and the individuals that the biopsies originated from. A table containing this information can sometimes be obtained from the laboratory information management system (LIMS) of the lab that performed the experiments. Sometimes, it is produced *ad hoc* with office spreadsheet software.

1.3.2 Facilities in R and Bioconductor

Specific data structures and functions for the import and processing of data from different experimental platforms are provided in specialized packages. We will see a number of examples in the subsequent sections. A more general-purpose data structure to represents the data from a microarray experiment is provided by the class *exprSet* in the package Biobase.

The design of the *exprSet* class supports the stepwise approach to microarray preprocessing, as discussed in Section 1.2. This class represents a self-documenting data structure, with data separated into logically distinct but substantively interdependent components. Our primary motivation was to link together the large expression arrays with the phenotypic data in such

a way that it would be easy to further process the data. Ensuring correct alignment of data when subsets are taken or when resampling schemes are used should be left to well-designed computer code and generally should not be done by hand.

The general premise is that there is an array, or a set of arrays, that are of interest. The *exprSet* structure imposes an order on the sample-specific expression measures in the set, provides convenient access to probe and sample identifier codes, allows coordinated management of standard errors of expression, and couples to this expression information sample- and experiment-level information, following the MIAME standard (Brazma et al., 2001). This data structure is straightforwardly employed with data from single-channel experiments, for ratio quantities derived from double-channel experiments, and for protein mass-spectrometry data. It can be extended, using formal inheritance infrastructure, to accommodate other output formats. One advantage to the use of `exprSet`s is demonstrated in Chapter 16 where we describe the use of a uniform calling sequence for many machine learning algorithms (package MLInterfaces). This greatly simplifies individual users' interactions and will simplify the design and construction of graphical user interfaces. Establishment of a standardized calling paradigm is most simply accomplished when there are structural standards for the inputs. Both users and developers will profit from closer acquaintance with the `exprSet` structure, especially those who are contemplating complex downstream workflows.

1.4 Statistical background

The purpose of this section is to provide a general statistical framework for the following components of preprocessing: background adjustment, normalization, summarization, and quality assessment. More specific issues relating to the individual technological platforms will be discussed in Chapters 2–4.

With a microarray experiment, we aim to make statements about the abundances of specific molecules in a set of biological samples. However, the quantities that we measure are the fluorescence intensities of the different elements of the array. The measurement process consists of a cascade of biochemical reactions and an optical detection system with a laser scanner or a CCD camera. Biochemical reactions and detection are performed in parallel, allowing up to a million measurements on one array. Subtle variations between arrays, the reagents used, and the environmental conditions lead to slightly different measurements even for the same sample.

The effects of these variations may be grouped in two classes: *systematic effects*, which affect a large number of measurements (for example, the measurements for all probes on one array; or the measurements from one probe

across several arrays) simultaneously. Such effects can be estimated and approximately removed. Other kinds of effects are completely random, with no well-understood pattern. These effects are commonly called *stochastic components* or *noise*.

Stochastic models are useful for preprocessing because they permit us to find *optimal* estimates of the systematic effects. We are interested in estimates that are precise and accurate. However, given the noise structure of the data, we sometimes have to sacrifice accuracy for better precision and *vice versa*. An appropriate stochastic model will aid in understanding the accuracy-precision, or bias-variance, trade-off.

Stochastic models are also useful for construction of inferential statements about experimental results. Consider an experiment in which we want to compare gene expression in the colons of mice that were treated with a substance and mice that were not. If we have many measurements from two populations being compared, we can, for example, perform a Wilcoxon test to obtain a p-value for each transcript of interest. But often it is not possible, too expensive, or unethical, to obtain so many replicate measurements for all genes and for all conditions of interest. Often, it is also not necessary. Models that provide good approximations of reality can add power to our statistical results.

Quality assessment is yet another example of the usefulness of stochastic models: if the distribution of a new set of data greatly deviates from the model, this may direct our attention to quality issues with these data. Chapter 3 demonstrates an example of the use of models for quality assessment.

1.4.1 An error model

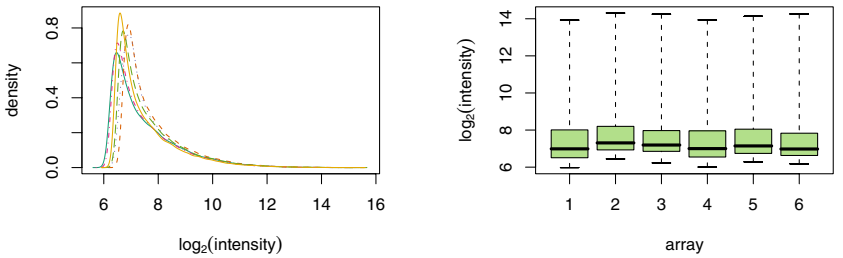
A generic model for the value of the intensity y of a single probe on a microarray is given by

$$Y = B + \alpha S \tag{1.1}$$

where B is a random quantity due to *background noise*, usually composed of optical effects and non-specific binding, α is a gain factor, and S is the amount of measured specific binding. The signal S is considered a random variable as well and accounts for measurement error and probe effects. The measurement error is typically assumed to be multiplicative so we write:

$$\log(S) = \theta + \phi + \varepsilon. \tag{1.2}$$

Here θ represents the logarithm of the true abundance, ϕ is a probe-specific effect, and ε accounts for measurement error. This is the *additive-multiplicative error model* for microarray data, which was first proposed by Rocke and Durbin (2001) and in a closely related form by Ideker et al. (2000).



a) b)

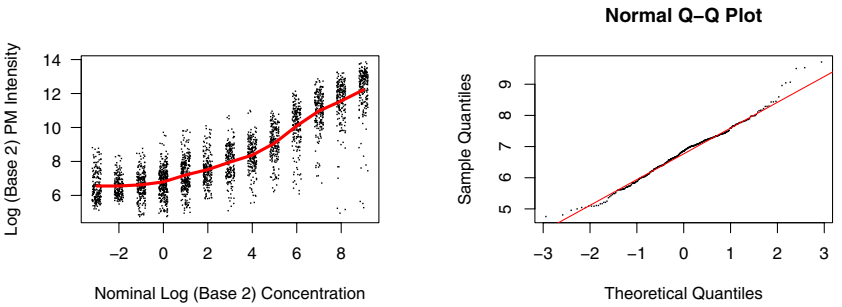
Figure 1.1. a) Density estimates of data from six replicate Affymetrix arrays. The x-axis is on a logarithmic scale (base 2). b) Box-plots.

Different arrays will have different distributions of B and different values of α, resulting in quite different distributions of the values of Y even if S is the same. To see this, let us look at the empirical distribution of six replicate Affymetrix arrays.

```
> library("affy")
> library("SpikeInSubset")
> data("spikein95")
> hist(spikein95)
> boxplot(spikein95)
```

The resulting plots are shown in Figure 1.1.

Part of the task of preprocessing is to eliminate the effect of background noise. Notice in Figure 1.1 that the smallest values attained are around 64, with slight differences between the arrays. We know that many of the probes are not supposed to be hybridizing to anything (as not all genes are expressed), so many measurements should indeed be 0. A bottom line effect of not appropriately removing background noise is that estimates of differential expression are biased. Specifically, the ratios are attenuated toward 1. This can be seen using the Affymetrix spike-in experiment, where genes were spiked in at known concentrations. Figure 1.2a shows the observed concentrations versus nominal concentrations of the spiked-in genes. Measurements with smaller nominal concentrations appear to be affected by attenuation bias. To see this, notice that the curve has a slope of about 1 for high nominal concentrations but gets flat as the nominal concentration gets closer to 0. This is consistent with the additive background noise model (1.1). Mathematically, it is easy to see that if s_1/s_2 is the true ratio and b_1 and b_2 are approximately equal positive numbers, then $(s_1 + b_1)/(s_2 + b_2)$ is closer to 1 than the true ratio, and the more so the smaller the absolute values of the s_i are compared to the b_i.



a) b)

Figure 1.2. a) Plot of observed against nominal concentrations. Both axes are on the logarithmic scale (base 2). The curve represents the average value of all probes at each nominal concentration. Nominal concentrations are measured in picomol. b) Normal quantile-quantile plot of the logarithmic (base 2) intensities for all probes with the same nominal concentration of 1 picomol.

Figure 1.2b shows a normal quantile-quantile plot of logarithmic intensities of probes for genes with the same nominal concentration. Note that these appear to roughly follow a normal distribution. Figure 1.2 supports the multiplicative error assumption of model 1.1.

1.4.2 The variance-bias trade-off

A typical problem with many preprocessing algorithms is that much precision is sacrificed to remove background effects and improve accuracy. Model (1.1) can be used to show that subtracting unbiased estimates of background effects leads to exaggerated variance for genes with small values of a. In fact, background estimates that are often used in practice, such as the "local background values" from many image analysis programs for two-color spotted arrays and the mismatch (MM) value from Affymetrix arrays, tend to be *over*-estimates, which makes the problem even worse.

Various researchers have used models similar to Equation (1.1) to develop preprocessing algorithms that improve both accuracy and precision in a balanced way. Some of these methods propose variance stabilizing transformations (Durbin et al., 2002; Huber et al., 2002, 2004), others use estimation procedures that improve mean squared error (Irizarry et al., 2003b). Some examples will be provided in Chapters 2 and 4.

1.4.3 Sensitivity and specificity of probes

The probes on a microarray are intended to measure the abundance of the particular transcript that they are assigned to. However, probes may

differ in terms of their sensitivity and specificity. This fact is represented by the existence of ϕ in model (1.2). Here, sensitivity means that a probe's fluorescence signal indeed responds to changes in the transcript abundance; specificity, that it does not respond to other transcripts or other types of perturbations.

Probes may lack sensitivity. Some probes initially identified with a gene do not actually hybridize to any of its products. Some probes will have been developed from information that has been superseded. In some cases, the probe may correspond to a different gene or it may in fact not represent any gene. There is also the possibility of human error (Halgren et al., 2001; Knight, 2001).

A potential problem especially with short oligonucleotide technology is that the probes may not be specific, that is, in addition to matching the intended transcript, they may also match some other gene(s). In this case, we expect the observed intensity to be a composite from all matching transcripts. Note that here we are limited by the current state of knowledge of the human transcriptome. As our knowledge improves, the information about sensitivity of probes should also improve.

1.5 Conclusion

Various academic groups have demonstrated that the use of modern statistical methodology can substantially improve accuracy and precision of bottom-line results, relative to *ad hoc* procedures introduced by designers and manufacturers of the technology. In the following chapters, we provide some details of how Bioconductor tools can be used to do this, not only in microarray platforms, but also in other related technologies.

2

Preprocessing High-density Oligonucleotide Arrays

B. M. Bolstad, R. A. Irizarry, L. Gautier, and Z. Wu

Abstract

High-density oligonucleotide expression arrays are a widely used microarray platform. Affymetrix GeneChip arrays dominate this market. An important distinction between the GeneChip and other technologies is that on GeneChips, multiple short probes are used to measure gene expression levels. This makes preprocessing particularly important when using this platform. This chapter begins by describing how to import probe-level data into the system and how these data can be examined using the facilities of the *AffyBatch* class. Then we will describe background adjustment, normalization, and summarization methods. Functionality for GeneChip probe-level data is provided by the affy, affyPLM, affycomp, gcrma, and affypdnn packages. All these tools are useful for preprocessing probe-level data stored in an *AffyBatch* object into expression-level data stored in an *exprSet* object. Because there are many competing methods for this preprocessing step, it is useful to have a way to assess the differences. In Bioconductor, this can be carried out using the affycomp package, which we discuss briefly.

2.1 Introduction

The most popular microarray application is measuring genome-wide expression levels. High-density oligonucleotide expression arrays are a commonly used technology for this purpose. Affymetrix GeneChip arrays dominate this market. In this platform, the choice of preprocessing method can have enormous influence on the quality of the ultimate results. Many preprocessing methods have been proposed for high-density oligonucleotide array data. In this chapter, we discuss methodology and Bioconductor tools