March, 2006

An Introduction to High-Throughput Bioinformatics Data

Keith A. Baggerly
Kevin R. Coombes
Jeffrey S. Morris

1

An Introduction to High-Throughput Bioinformatics Data

Keith A. Baggerly,

Kevin R. Coombes,

Jeffrey S. Morris,

Department of Biostatistics and Applied Mathematics
University of Texas M.D. Anderson Cancer Center
1515 Holcombe Blvd, Unit 447, Houston, TX 77030
Email: {kabagg, kcoombes, jefmorris}@mdanderson.org

Abstract

High throughput biological assays supply thousands of measurements per sample, and the sheer amount of related data increases the need for better models to enhance inference. Such models, however, are more effective if they take into account the idiosyncracies associated with the specific methods of measurement: where the numbers come from. We illustrate this point by describing three different measurement platforms: microarrays, serial analysis of gene expression (SAGE), and proteomic mass spectrometry.

1.1 Introduction

In our view, high-throughput biological experiments involve three phases: experimental design, measurement and preprocessing, and postprocessing. These phases are otherwise known as: deciding what you want to measure, getting the right numbers and assembling them in a matrix, and mining the matrix for information. Of these, it is primarily the middle step that is unique to the particular measurement technology employed, and it is there that we shall focus our attention. This is not meant to imply that the other steps are less important! It is still a truism that the best analysis may not be able to save you if your experimental design is poor.

We simply wish to emphasize that each type of data has its own quirks

associated with the methods of measurement, and understanding these quirks allows us to craft ever more sophisticated probability models to improve our analyses. These probability models should ideally also let us exploit information across measurements made in parallel, and across samples. Crafting these models leads to the development of brand-new statistical methods, many of which are discussed in this volume.

In this chapter, we address the importance of measurement-specific methodology by discussing several approaches in detail. We cannot be all-inclusive, so we shall focus on three. First, we discuss microarrays, which are perhaps the most common high throughput assay in use today. The common variants of Affymetrix gene chips and spotted cDNA arrays are discussed separately. Second, we discuss serial analysis of gene expression (SAGE). As with microarrays, SAGE makes measurements at the mRNA level, and thus provides a picture of the expression profile of a set of cells, but the mechanics are different and the data may give us a different way of looking at the biology. Third, we discuss the use of mass spectrometry for profiling the proteomic complement of a set of cells.

Our goal in this chapter is not to provide detailed analysis methods, but rather to place the numbers we work with in context.

1.2 Microarrays

Microarrays let us measure expression levels for thousands of genes in a single sample all at once. Such high throughput assays allow us to ask novel biological questions, and require new methods for data analysis.

In thinking about the biological context of a microarray, we start with our underlying genomic structure [4]. Your genome consists of pairs of DNA molecules (chromosomes) held together by complementary nucleotide base pairs (in total, about 3×10^9 base pairs). The structure of DNA provides an explanation for heredity, by copying individual strands and maintaining complementarity.

All of your cells contain the same genetic information, but your skin cells are different from liver cells or kidney cells or brain cells. These differences come about because different genes are expressed at high levels in different tissues. So, how are genes "expressed"?

The "central dogma" of molecular biology asserts that "DNA makes RNA makes protein". In order to direct actions within the cell, parts of the DNA will uncoil and partially decouple to expose the piece of the single strand of DNA on which a given gene resides. Within the nucleus, a

complementary copy of the gene sequence (not the entire chromosome) is assembled out of RNA. This process of RNA synthesis is called transcription: copying the message. The initial DNA sequence containing a gene may also contain bits of sequence that will not be used – one feature of gene structure is that genes can have both "coding" regions (exons) and "noncoding" regions (introns). After the initial RNA copy of the gene is made, processing within the nucleus removes the introns and "splices" the remaining pieces together into the final messenger RNA (mRNA) that will be sent out to the rest of the cell. Once the mRNA leaves the nucleus, the external machinery (ribosomes) will read the code and assemble proteins out of corresponding sequences of amino acids. This process of assembling proteins from mRNA is called translation: mapping from one type of sequence (nucleotides) to another (amino acids). The proteins then fold into 3d-configurations that in large part drive their final function. If different genes are copied into RNA (expressed) in different cells, different proteins will be produced and different types of cells will emerge. Microarrays measure mRNA expression.

In thinking about the informational content of these various stages for understanding cellular function, we need to know different things. For DNA, we need to know sequence. For mRNA, we need both sequence and abundance; many copies can be made of a single gene. Gene expression typically refers to the number of mRNA copies of that gene. For protein, we need sequence, abundance, and shape (the 3d-configuration).

If we could count the number of mRNA molecules from each gene in a single cell at a particular time, we could assemble a barchart linking each gene with its expression level. But how do we make these measurements? As suggested, we exploit complementarity: sequences of DNA or RNA containing complementary base pairs have a natural tendency to bind together:

```
...AAAAAGCTAGTCGATGCTAG...
...TTTTTCGATCAGCTACGATC...
```

If we know the mRNA sequence (which we typically do these days, since we can look it up in a database), we can build a probe for it using the complementary sequence. By printing the probe at a specific spot on the array, the probe location tells us the identity of the gene being measured.

There are two common variants of microarrays:

- Oligonucleotide (oligo) arrays, where short subsequences of the gene

are deposited on a silicon wafer using photolithography (primarily Affymetrix).
- Full length (entire gene) arrays, where probes are spotted onto a glass slide using a robotic arrayer. These generally involve two samples run at the same time with different labels.

1.2.1 Affymetrix GeneChips

In looking at the structure of Affymetrix data, there are several in depth resources [2, 3, 39] which serve as major sources for what follows, including the company's web site, `www.affymetrix.com`.

In general, genes will be hundreds or thousands of bases in length, and the probes are shorter by an order of magnitude. This is driven in part by the manufacturing process, as the cost of synthesis increases with the number of bases deposited. Thus, choosing probes to print requires finding sequences that will be unique to the gene of interest (for specific binding) while still being short enough to be affordable. The final length decided on was 25 bases, and all Affymetrix probes are this length. It is important to note that different probes for the same gene have different binding affinities, and these affinities are unknown *a priori*. Thus, it's difficult to tell whether "gene A beats gene B", as opposed to "there's more gene A here than there". Microarrays only produce relative measurements of gene expression.

Given that the affinities are unknown, we can guard against problems with any specific probe by using several different probes for each gene. The optimal number of probes is not clear. Subsequent generations of Affymetrix chips have used 20 (eg HuGeneFL, aka Hu6800), 16 (U95 series), and 11 (U133 series) probes. There are some further difficulties with choosing probes:

- some genes are short, so multiple subsequences will overlap.
- genes have an orientation, and RNA degradation begins preferentially at one end (3' bias).
- the gene may not be what we think it is, as our databases are still evolving.
- probes can "cross-hybridize", binding the wrong targets.

Overlapping, we can live with. Orientation can be addressed by choosing the probes to be more tightly concentrated at one end. Database evolution we simply can't do anything about. Cross-hybridization, however, we may be able to address more explicitly.

Affymetrix tries to control for cross-hybridization by pairing probes that should work with probes that shouldn't. These are known as the Perfect Match (PM) and Mismatch (MM) probes, and constitute "probe pairs". The PM probe is perfectly complementary to the sequence of interest. The MM probe is the same as the PM probe for all bases except the middle one (position 13), where the PM base is replaced by its Watson-Crick complement.

PM: GCTAGTCGATGC**T**AGCTTACTAGTC
MM: GCTAGTCGATGC**A**AGCTTACTAGTC

Ideally, the MM value can be used as a rough assessment of the amount of cross-hybridization associated with a given PM probe.

Affymetrix groups probe pairs associated with a given gene into "probesets"; a given gene would be represented on a U133A chip by a probeset containing 11 probe pairs, or 22 probes with distinct sequences. The probes within a probeset are ordered according to the position of the specific PM sequence within the gene itself. We have described the ideal case above, but in practice the correspondence between genes and probesets is not 1-to-1, so some genes are represented by several probesets.

Having printed the probes, we now need to attach the target mRNA in such a way that we can measure the amounts bound. When we extract mRNA from a sample of cells, we do not measure this mRNA directly. Rather, we make copies. Copies are produced of the complementary sequence out of RNA (cRNA). Some of the nucleotides used to assemble these copies have been modified to incorporate a small molecule called biotin. Biotin has a strong affinity for another molecule called streptavadin; their binding affinity is the strongest known noncovalent biological interaction. After the biotin-labeled cRNA molecules are hybridized to the array, they are stained with a conjugate of streptavadin and phycoerythrin; phycoerythrin is one of the brightest available fluorescent dyes. The final complex of printed probe, biotinylated target, and streptavadin-phycoerythrin indirect label is then scanned, producing an image file. For our purposes, this image constitutes bedrock: *The image is the data.*

All Affymetrix GeneChips are scanned in an Affymetrix scanner, and the initial quantification of features is performed using Affymetrix software. The software involves numerous files. The file types are:

EXP Contains basic information about the experiment.
DAT Contains the raw image.

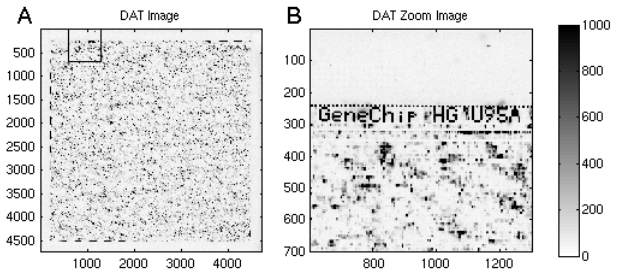

Fig. 1.1. An Affymetrix image (.DAT) file. (A) The entire image, 4733 pixels on a side, containing 409,600 features. (B) A zoom on the upper left corner of the image. Controls are used in a checkerboard pattern to indicate the print region border, and to designate the chip type. This is a U95Av2 chip; on v2 chips the "A" is filled in.

CEL Contains feature quantifications.

CDF Maps between features, probes, probesets, and genes.

CHP Contains gene expression levels, as assessed by the Affy software.

Most frequently, we start with a DAT file, derive a CEL file, and then make extensive use of the CEL and CDF files. We make no further use of the EXP and CHP files here.

To illustrate the procedure, we begin by looking at the contents of a DAT file from a U95Av2 chip (the raw image), shown in Figure 1.1 A. The array has 409,600 probes (features) arranged in a 640×640 grid. There is actually some structure that can be seen by eye, as we can see if we zoom in on the upper left corner: Figure 1.1 B. The pixelated features have been combined with positive controls to spell out the chip type – this helps ensure that the image is correctly oriented. We note the border lattice of alternating dark and bright QC probes, making image alignment and feature detection easier.

If we zoom in further on a single PM/MM pair or feature, shown in Figure 1.2 A and B, we can see that features are square. The horizontal and vertical alignment with the edges of the image is pretty good, but feature boundaries can be rather blurry.

Each feature on this chip is approximately 20 microns on a side. The scanner used for this scan had a resolution of 3 microns/pixel, so the feature is about 7 pixels on a side (more recent scanners have higher resolution). In general, Affymetrix features are far smaller than the round spots in the images of other types of microarrays.

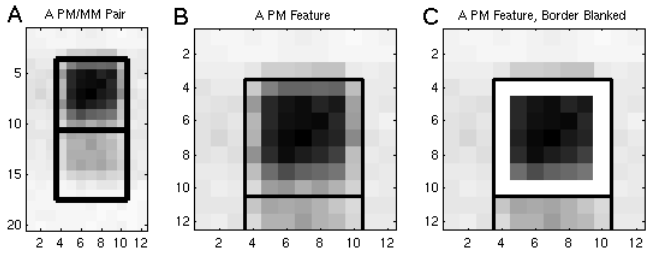

Fig. 1.2. Sets of Affymetrix image features. (A) A PM/MM pair. Note that the PM pixel readings are higher than the MM readings. (B) A zoom on the PM feature. (C) The PM feature after trimming the outer boundary. Only the remaining pixels are used in deriving a summary quantification (the 75th percentile).

The DAT file structure consists of a 512 byte header followed by the raw image data. The image shown above involved a 4733 by 4733 grid of pixels, so the total file size is $2*4733^2+512 = 44803090$ bytes (45M). This is big. File size is a nontrivial issue with Affy data; earlier versions of the software could only work with a limited number of chips (say 30). Given this size, our first processing step is to produce a single quantification for each feature, keeping in mind that the edges are blurry and that the features may not be perfectly uniform in intensity.

The CEL file contains the feature quantifications, achieved as follows. First, the four corners of the entire feature grid (here 640×640) are located within the DAT file, and a bilinear mapping is used to determine the pixel boundaries for individual features. Given the pixels for a single feature, the outermost boundary pixels are trimmed off, as shown in Figure 1.2 C. Finally, the 75th percentile of the remaining pixel values is stored as the feature summary. Trimming is understandable, as this accounts for blurred edges in a moderately robust way. Similarly, using a quantile makes sense, but the choice of the 75th percentile as opposed to the median is arbitrary.

When Affymetrix data is posted to the web, CEL files are far more often supplied than DAT files. Over time, there have been various versions of the CEL format. Through version 3 of the CEL file format, this was a plain text file. In version 4, the format changed to binary to permit more compact storage of the data. Affymetrix provides a free tool to convert between the file formats.

In the plain text version, sections are demarcated by headers in brack-

ets, as in the example below. The header tells us which DAT file it came from, the feature geometry (e.g., 640×640), the pixel locations of the grid corners in the DAT file, and the quantification algorithm used. This is followed by the actual measurements, consisting of the X and Y feature locations (integers from 0 to 639 here), the mean (actually the 75th percentile) and standard deviation (this, conversely, *is* the standard deviation), and the number of pixels in the feature used for quantification after trimming the border. An example CEL file header is given below.

```
[CEL]
Version=3
[HEADER]
Cols=640
Rows=640
TotalX=640
TotalY=640
OffsetX=0
OffsetY=0
GridCornerUL=219 235
GridCornerUR=4484 253
GridCornerLR=4469 4518
GridCornerLL=205 4501
Axis-invertX=0
AxisInvertY=0
swapXY=0
DatHeader=[0..19412]  U95Av2_CDDO_12_14_01: CLS=4733
RWS=4733 XIN=3  YIN=3  VE=17 2.0 12/14/01 12:23:30
HG_U95Av2.1sq  6 Algorithm=Percentile
AlgorithmParameters=Percentile:75;CellMargin:2;
OutlierHigh:1.500;OutlierLow:1.004
[INTENSITY]
NumberCells=409600
CellHeader=X Y MEAN STDV NPIXELS
  0   0 133.0 16.6  25
  1   0 8150.0 1301.3  20
```

A version 3 CEL file reduces the space required to about 12M from 45M for a DAT file, but we could do better. The X and Y fields are not necessary, as these can be inferred from position within the CEL file. Keeping 1 decimal place of accuracy for the "mean" and standard deviation doubles the storage space required (moving from a 16-bit integer to a float in each case) and supplies only marginally more information. Finally, most people do not use the STDV and NPIXELS fields. Keeping only the mean values and storing them as 16-bit integers, storage can be reduced to $2 * 640^2 = 819200$ bytes. This type of compression is becoming more important as the image files get even bigger.

The above description covered Affymetrix version 3.0 files. In version

4.0, in binary format, each row is stored as a MEAN-STDV-NPIXEL or float-float-short triplet, which cuts space, but not enough. Most recently, Affymetrix has introduced a CCEL (compact CEL) format, which just stores the integer mean values as discussed above.

The above problem, going from the image to the feature quantification, is a major part of the discussion for quantification of other types of arrays because there, we get only one spot per gene. For Affymetrix data, the company's quantification has become the de facto standard. It may not be perfect, but it is reasonable. The real challenge with Affymetrix data lies in reducing the many measurements of a probeset to a single number.

In summarizing a probeset, we first need to know where its component probes are physically located on the chip. With any set of microarray experiments, one of the major challenges is keeping track of how the feature quantifications map back to information about genes, probes, and probe sets. The CDF file specifies what probes are in each probeset, and where the probes are. There is one CDF file for each type of GeneChip. The header is partially informative, as shown in the example below.

```
[CDF]
Version=GC3.0
[Chip]
Name=HG_U95Av2
Rows=640
Cols=640
NumberOfUnits=12625
MaxUnit=102119
NumQCUnits=13
ChipReference=

[Unit250_Block1]
Name=31457_at
BlockNumber=1
NumAtoms=16
NumCells=32
StartPosition=0
StopPosition=15
CellHeader=X Y        PROBE   FEAT       QUAL       EXPOS
            POS      CBASE   PBASE      TBASE      ATOM   INDEX
            CODONIND CODON   REGIONTYPE REGION
Cell1=517   568      N       control    31457_at   0
            13       A       A          A          0      364037
            -1       -1      99
Cell2=517   567      N       control    31457_at   0
            13       A       T          A          0      363397
            -1       -1      99
Cell3=78    343      N       control    31457_at   1
            13       T       A          T          1      219598
```

For this probeset, 31457_at, there are 16 "atoms" corresponding to probe pairs (this is the standard number for this vintage chip), and 32 "cells" corresponding to individual probes or features. The first probe pair (index 0), with the PM sequence closest to one end of the gene, is located on the chip in the 518th column (the X offest is 517) and in the 568th and 569th rows. The index values for these probes are (567 * 640) + 517 = 363397 and 364037. The feature in Cell 2 is the PM probe, as (a) it has a smaller Y index value, and (b) the probe base (PBASE) in the central base position (POS) 13 is a T, which is complementary to the corresponding target base (TBASE). The remaining values in a given row are less important. The CDF files do not contain the actual probe sequences, but all CDF files and probe sequences are now downloadable from `www.affymetrix.com`.

On early Affymetrix chips, all probes in a probeset were plotted next to each other. This was soon realized to be imperfect, as any artifact on a chip could corrupt the measurements for an entire gene. On more recent chips, probes within a probeset are spatially scattered, though PM/MM pairs are always together (the PM probe is always closer to the edge on which the chip id is spelled out).

Given quantifications for individual chips, we turn next to quantifying a dataset, relating probeset values across chips.

Before we quantify individual probesets, however, we need to address the problem of **normalization**: is the image data roughly comparable in intensity across chips? Adding twice as much sample may make the resultant image brighter, but it doesn't tell us anything new about the underlying biology. In most microarray experiments, we are comparing samples of a single tissue type (eg diseased brain to normal brain), and in such cases we *assume* that "most genes don't change". Typically, we enforce this by matching quantiles of the feature intensity distributions. Given that the chips have been normalized, we still need to find a way of summarizing the intensities in a probeset. The PM and MM features for an example probeset are shown in Figure 1.3 A and B.

The earliest widely applied method was supplied by Affymetrix in version 4 of their Microarray Analysis Suite package, and is commonly referred to as MAS 4.0 ("Mass 4") or AvDiff [2]. AvDiff works with the set of PM−MM differences in a probeset one array at a time. These differences are sorted in magnitude, the minimum and maximum values are excluded, and the mean and standard deviation of the remaining differences are computed. Using this mean and standard deviation, an "acceptance band" for the differences is defined as ±3 s.d. about the

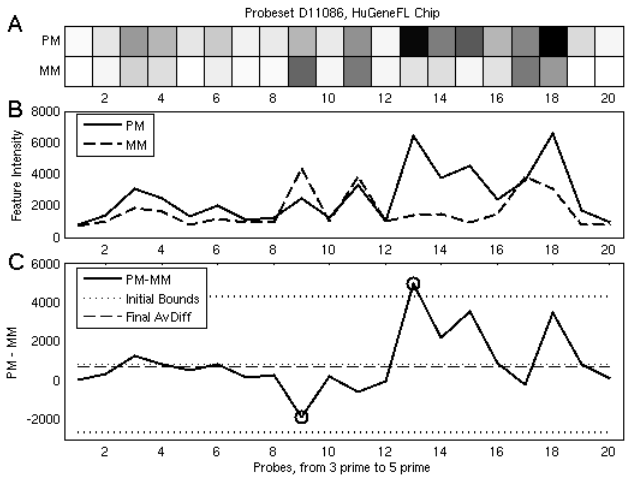

Fig. 1.3. A single probeset from a Hu6800 chip, containing 20 PM/MM pairs. (A) A heatmap of the feature intensities extracted from the CEL file. (B) Plots of the PM (solid) and MM (dashed) values shown in A. Feature values are not uniform across the probeset, and MM values occasionally exceed PM. (C) A plot of the PM−MM differences, showing the computation of AvDiff. The extreme values (circled) are initially excluded, and the mean and ±3 s.d. bounds (dotted) are imposed. All points within this band are then averaged to produce AvDiff (dashed).

mean. All of the differences falling within this band are then averaged to produce the final AvDiff value. This is illustrated in Figure 1.3 C. In the case illustrated here, the minimum value was excluded at the first step, but fell into the acceptance band and was thus included in the final average, moving the value down slightly.

AvDiff does have some nice features. It combines measurements across probes, trying to exploit redundancy, and it attempts to insert some robustness. However, there are some questionable aspects. AvDiff weights the contributions from all probes equally, even though some may not bind well. It works on the PM−MM differences in an additive fashion, but some of the effects may be multiplicative in nature. It can give negative values, which are hard to interpret. In some cases, where all of the signal for a probeset is concentrated in a very small number of probes, these may be omitted altogether if they fall outside the band. All of these drawbacks, in our view, can be tied to the fact that AvDiff works one chip at a time, and does not "learn" with the addition of more chips.

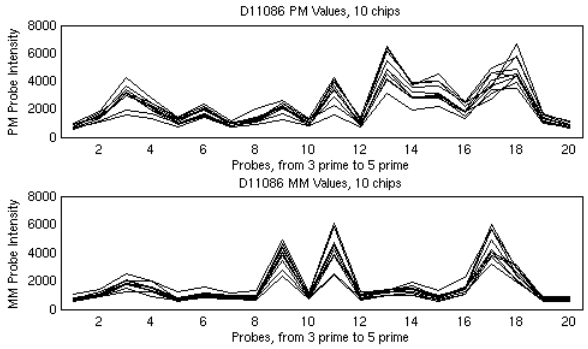

Fig. 1.4. Plots of PM and MM intensities for the same probeset on 10 different chips. The overall profile shapes are fairly consistent across chips, with changes in gene expression linked to amplitude. Modelling the shapes can improve inferences about expression levels.

Learning from multiple chips requires an underlying model with parameters that can be estimated. In 2001, Cheng Li and Wing Wong introduced a new method of summarizing probeset intensities as "model-based expression indices", or MBEI [35, 36, 59]. At the crux of their argument was a very simple observation – the relative expression values of probes within a probeset were very stable across multiple arrays.

Looking at the PM and MM profiles for the same probeset in 10 chips from a single experiment, as shown in Figure 1.4, we can see that the overall shape of the profile is fairly consistent. It is the amplitude of this profile which changes, and which contains the summary information about the level of gene expression.

In order to exploit this stability, Li and Wong fit a model for each probeset: for sample i, and probe pair j, they posit that

$$\begin{aligned} PM_{ij} &= \nu_j + \theta_i\alpha_j + \theta_i\phi_j + \epsilon, \\ MM_{ij} &= \nu_j + \theta_i\alpha_j + \epsilon, \end{aligned}$$

where ν_i and $\theta_i\alpha_j$ are intended to capture nonspecific binding, and ϵ is Gaussian noise. Focusing on the PM$-$MM differences, this model condenses to one with two sets of unknowns: θ_i and ϕ_j. The ϕ_j terms correspond to the individual probe affinities, and give the shape of the profile. The θ_i values give the amplitudes.

The MBEI approach caught on fairly quickly, in part because the numerical approach made sense, but also due to the fact that it was imbed-

ded in the freeware "DNA Chip Analyzer" (dChip) package, available at `www.dchip.org`. This package has a very friendly user interface, and addressed many of the most common questions (which genes are different? how should I cluster them?) in a straightforward fashion. Further, by encoding the contents of CEL and CDF files in a binary format using analogs of the data structures outlined above, the program could handle lots of chips at once, and it could handle them quickly.

Using a model has several benefits. By using multiple chips, it can keep all of the probes; there is no tossing of the most informative ones. By checking the residuals from the model, it is possible to identify outliers due to artifacts. Using the hypothesized error model, confidence bands for the fold change can be computed. Probe profiles can be computed in one experiment and used in another.

The downside of most models is that they require several chips in order to estimate the underlying model parameters. It is not a good idea to trust the fits too much if they are based on just one or two chips; 10 or more is better. However, we're not convinced that it is a bad thing to require a larger minimum number of chips for drawing inferences.

The dChip model captures effects that are multiplicative, and inherits the other good features of a model. However, the probability model is too simplistic, as larger intensity probes typically also have larger variances.

In the wake of dChip, several other quantification methods have been suggested, with many (but not all) using model-based approaches. A partial list includes MAS 5.0, RMA, and PDNN.

The next algorithm from Affymetrix, MAS 5.0 [3], still produces quantifications one chip at a time, but replaces the MM values with a rather intricate change threshold (CT) to avoid negative values. The differences are then combined using a robust measure:

$$\text{Tukey Biweight}(\log(\text{PM}_\text{j} - \text{CT}_\text{j})).$$

The robust multichip analysis (RMA) method of Irizarry et al. [27, 28] also uses a model for fitting the data, but the model differs from dChip's in some key ways. First, the authors elected to ignore the MM values, contending that any gains in accuracy were more than offset by losses in precision, in a classic bias-variance tradeoff. Second, since the MM values were not on hand, "background" levels were estimated from the distribution of PM probe intensities and subtracted off in such a way as to avoid negative values [13]. Third, the model introduced stochastic errors on the log scale as opposed to the raw intensity scale. The final

model is of the form

$$log(PM_{ij} - BG) = \mu_i + \alpha_j + \epsilon_{ij}.$$

The above approaches use the probe intensities, but there is addtional biological structure that can be exploited. In particular, Affymetrix now makes the actual probe sequences available, though it did not when it first started selling chips. Using the sequences, it is possible to build models describing the default binding efficiencies for individual probes, and to decouple this from binding due to gene abundance. This approach was first exploited in the Position-Dependent Nearest Neighbor (PDNN) approach introduced by Zhang et al. [76]. The RMA method has since been extended to incorporate sequence information in its modeling, giving GCRMA [73].

Given the proliferation of models, we need some means of deciding which ones are "better". In order to make such assessments, we need to have some data sets for which "truth" can be known *a priori*, and some set of defined metrics that measure proximity to truth. The most widely used truth-known data set is a Latin Square experiment supplied by Affymetrix, in which 14 genes were spiked into a common mixture according to a twofold dilution series, which was then cyclicly permuted so that each gene was assessed at each dilution level. In this case, only the spiked in genes should be changing in expression, and the amount of change is potentially known. In order to quantify truth, Cope et al. [20] introduced a suite of metrics for putting each method through its paces on the canonical datasets. The results for many different methods have been assembled and posted at

`http://affycomp.biostat.jhsph.edu/`,

and new submissions are welcome.

In addition to dChip, there are now several software packages available for analyzing Affymetrix data, but the most widely used in the statistical community are probably those implemented in R and freely available from Bioconductor. R packages exist for implementing all of the approaches discussed here, and most methods are sufficiently modular that different background correction, normalization, and quantification methods can be juggled to suit. The book by Gentleman et al. [25] provides an excellent introduction to this resource. Not all of the methods available are equally fast, however, so for the analysis of large datasets dChip and "justRMA" or "justGCRMA" in R are the ones that we would suggest.

The models for Affymetrix data are now reasonably good, but dozens

of questions remain. Combining results of Affymetrix experiments across different labs and different chip types is still difficult, and integrating these results with those from glass arrays is still harder. Eventual combination of results at the RNA level with those from the DNA and protein levels is tantalizing.

1.2.2 Spotted cDNA Arrays

We now shift from Affymetrix oligonucleotide arrays to spotted cDNA arrays. Here, a good set of overview ariticles (from 1999) is available as a special supplement to *Nature Genetics*, "The Chipping Forecast" [47]; see also [61, 60]. While the biological questions of interest are similar, the probes used are quite different. On most cDNA arrays, the probes used correspond to full-length copies of the gene of interest (sans introns), though there has been recent interest in long-oligo arrays that use probes that are 60 or 70 bases in length (60-mers or 70-mers). Typically, each gene will be represented by one probe, not a set. The other major distinction is that two samples, not one, are typically hybridized to each array. The samples are prepared using different incorporated dyes, mixed, and the mixture is then hybridized to the array.

The method of dye incorporation is different for spotted arrays than for Affymetrix gene chips. On a gene chip (as noted), the fluorescent dye is applied after hybridization has taken place (indirect labeling), but this strategy does not work if multiple samples need to be labeled with different dyes. Rather, when copies of mRNA are made for spotted arrays, they are made of cDNA, and some of the bases used in the assembly of these copies have had molecules of fluorescent dye attached. Thus, the dye is incorporated into the copies before hybridization (direct labeling). These labeled copies are then hybridized to the array, binding molecules of dye in specific positions.

The most commonly reported gene summary is the log ratio of two intensity measurements, corresponding to the two dyes with which the two types of cells being compared have been respectively tagged. The most commonly used dyes are Cy5, red, and Cy3, green. Thus, the single number quoted is derived from the two intensity values. The intensity values are also derived quantities; they are derived from images. Again, for our purposes these images represent bedrock. Images are our raw data.

These images are scans of slides with lots of dots on them, each dot corresponding to the location of a DNA probe to which labeled cDNA

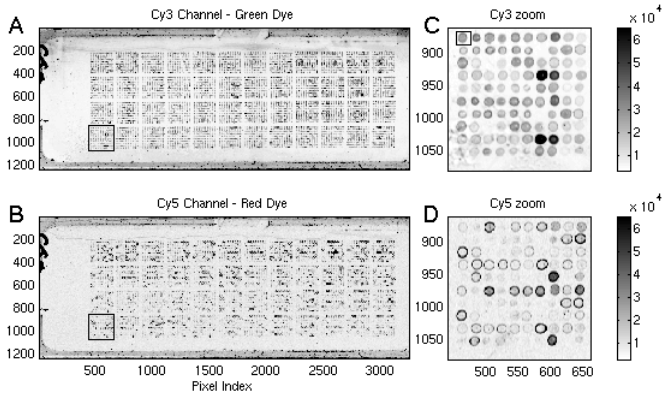

Fig. 1.5. Cy3 and Cy5 image scans from a spotted cDNA microarray. (A) The full Cy3 image. (B) The full Cy5 image. In both A and B, the patch structure (one per print-tip on the arrayer) is apparent. (C) A zoom on a Cy3 patch. (D) A zoom on the corresponding Cy5 patch. The top half of each patch is replicated in the bottom half, and this structure is visible. Imperfections in both the spotting and the image can also be seen, most clearly in the zooms.

derived from the cells of interest has been bound. In some early experiments from MD Anderson, there were approximately 4800 dots on a slide, arranged in a 4 by 12 grid of patches, with each patch containing a 10 by 10 grid of dots. When the images of the slide were produced, we got 3248 by 1248 arrays of grayscale pixel values. The scans from one such slide are shown in Figure 1.5 A and B. The patch structure is quite apparent. This structure is linked to the method of depositing the probes. In printing, a robotic arrayer takes an array of print tips (similar to needles), dips them in wells of the DNA to be printed, moves the coated print tips over to the slide, taps lightly to transfer probes, and takes the print tips over to a wash solution before repeating the process. The arrayer we used had a 4 by 12 array of print tips; each visible patch has been applied by a single print tip.

Returning to consideration of the images, each pixel is a 16-bit intensity measurement, so values range from 0 to 65535. There is no color inherently associated with these images, which is why we have presented them in grayscale; other colormaps are externally applied to enhance contrast. Each image is about 8M in size, which is large enough to make manipulation and transmission somewhat unwieldy at times. As more genes are spotted on the arrays, and the scanner resolutions are improved so that smaller objects can be seen, these images will increase

in size. It should be noted that the 16-bit nature of the images can make things difficult to work with in ways not having to do with file size. Some image viewing software assumes that the values are 8-bit, ranging from 0 to 255, and consequently either fails to show the large image or shows it as full white (all values set to 255). The values can be converted to 8-bit fairly simply, as 8-bit = floor(16-bit/256), but we lose gradation information. As the dynamic range of these images is quite large, this loss can be be damaging for the purposes of analysis.

To make things more concrete in getting down to the actual spot level, we focus on a single 10 by 10 patch, marked in the bottom left of the large images. The corresponding regions from the two image files are shown in Figure 1.5 C and D. These arrays were printed with replicate spottings of the same genes: within each patch, the top half of the patch is replicated in the bottom half. This replicate structure is visible — the brightest Cy3 spots are in rows 4 and 9 of column 7 of the patch, a replicate pair — giving us some confidence in the assay.

A few other things are immediately apparent. First, the "dots" are not really "dot-like" in most cases. Rather, there are rings of high intensity about lower-level centers. This is true across both channels, indicating that the ring pattern matches the amount of cDNA on the slide. The most likely explanation is that surface tension on the drop as it dries may cause clumping at the edges. In any event, how does morphology affect our measurements? Second, the dots are not of equal size. This may make it difficult for an automatic procedure to find the appropriate placement of a dot-shaped target ring. Third, there is some mottling in the lower left corner (most visible in the Green channel). How does this affect our assessment of how intense the dots in that region are?

Before considering these questions further, let's take a closer look at a single spot, highlighted in Figure 1.5 C. An expanded view of this spot is shown in Figure 1.6. The ring shape is visible, indicating uneven hybridization. Further, the side view shows that readings outside the spot are not at zero intensity, indicating the need for some type of background subtraction so that we have moderately good estimates of where zero should be.

All of these issues point out the need for good image quantification algorithms for summarizing the spots. Some more detailed descriptions of algorithms for image segmentation, background estimation, and spot summaries are given in Yang et al. [74]. There are several software packages (mostly commercial) now available for quantifying array images.

Given the metrics, however, a more basic question is why two sam-

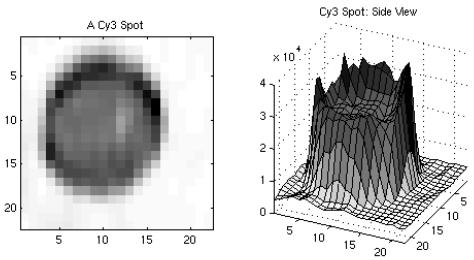

Fig. 1.6. Zoom on a single Cy3 spot. The ring shape is visible, indicating uneven hybridization. Further, the side view shows that readings outside the spot are not at zero intensity.

ples are used per array as opposed to one. The main reason is to guard against artifacts. Some spots are bigger than others, and thus bind more material. The slide can be tilted while hybridization is proceeding, resulting in more binding at one edge than another. Ideally, such artifacts will affect both channels similarly, and taking ratios will cancel them out. If there are replicate spots printed on the arrays, the importance of ratios can be checked by plotting the variance of the replicate log intensities as a function of the mean, first for each individual channel and then for the ratios. The variability of the ratios is typically less (often much so).

While the use of two samples does protect against some large-scale biases, it can also introduce new ones. The dyes used have different physical shapes, and thus can have different binding efficiencies for given genes. In recognition of this fact, many studies use one of two approaches for comparing two groups of samples. The first approach involves direct comparison of a sample of type A with a sample of type B on the same array. In this case, "dye swaps" are used so that the A samples are labeled with Cy3 on some arrays and with Cy5 on others, so that dye biases can be factored out. The second approach is to use the same dye to label the samples from both groups of interest, and to contrast these with some common reference material labeled with the other dye. Some of the design issues raised by this natural paired blocking strucure are discussed in [33, 63].

Even with these balancing features, normalization remains an issue, both within and across arrays. Again, most methods make the simplifying assumption that most genes don't change. Given this assumption,

a common means of correction is to plot the difference in channel log intensities as a function of the average log intensity, and to fit a loess curve to the dot cloud. These plots were introduced by Bland and Altman [12], but are more commonly referred to as "MA" plots in the microarray context [22]. Subtracting the loess curve ideally normalizes expression values within the array. A further extension of this approach is to apply a separate loess fit for the spots associated with each print tip. This makes stronger assumptions about which groups of genes are not expected to change, but smooths things more evenly. While we have seen cases where print-tip loess has produced more stable values (and better agreement between replicate spots), in many of these cases we are correcting for spatial trends that are visible on the array images, as opposed to discrepancies that are ascribable to the pins. Print-tip loess works in part because it is a surrogate for spatial position. Once the individual arrays have been normalized, quantile normalization can be used to match log ratio values across arrays [75].

Given the spot quantifications, and knowledge of what samples are bound on which arrays, there are freeware tools available for most basic analyses. Again, the book by Gentleman et al. [25] provides a nice survey of the suite of R tools available with Bioconductor.

One last concern with glass arrays relative to Affymetrix chips is simply that the number of different array configurations and gene spotting patterns is legion. This means that annotation and gene information must be checked carefully keeping the gene to spot mappings clear. It also means that comparisons across different array platforms may yield different measures of the "same" gene if different cDNAs are used.

1.3 SAGE

Microarrays work by exploiting hybridization to assess amounts of dye aggregating to specific probes printed on the arrays. There are, however, some potential downsides to microarrays. First, a microarray is a closed system, in the sense that you will only be able to measure an mRNA if you have printed a probe for it. Unexpected transcripts will not be seen. Second, the quantitative nature of the data is somewhat questionable, as dye response is a nonlinear phenomenon. Third, differences in protocols or preparations have made comparison of array results across labs difficult.

We would like to have some mechanism for more directly counting all of the mRNA transcripts of a given type. Failing that, if we could take

a random sample of all of the mRNA transcripts available and count those, then this would still provide an unbiased and quantitative profile of mRNA expression. This idea of sampling and counting underlies the serial analysis of gene expression (SAGE) technique. Some case studies are given in [68, 69, 67, 77, 51, 50, 52, 64, 57, 56].

As before, we still need to know both sequence (identity) and abundance to characterize the expression profile. With microarrays, the unknown sequence of the transcript is inferred from the known sequence of the printed probe. With SAGE, a part of the transcript itself is sequenced. Restricting attention to only a part of the transcript is deliberate. While sequencing the entire transcript would identify it unambiguously, sequencing is time-consuming and costly enough that the expense would be prohibitive. We want to sequence just enough of the transcript to identify it, and then move on. The question now becomes one of how to biologically extract an identifying subsequence.

An identifying subsequence need not be long. Current estimates of the number of genes in the human genome are around 25,000 to 30,000. While alternative splicing of the exons within the gene may allow the same gene to produce several distinct transcripts, the total number of distinct transcripts is unlikely to be more than a few hundred thousand. Considering the 4 letter DNA alphabet, there are $4^{10} = 1{,}048{,}576$ distinct 10-letter "words", suggesting that a 10 base pair (bp) subsequence may be enough for unique identification. This rough calculation implicitly assumes that the 10 bp are in a specific location; it is considerably harder to find unique subsequences if these are allowed to occur anywhere within the gene. We are going to first specify position, and then extract sequence. This process is rather intricate. The steps are illustrated in Figure 1.7, and discussed in detail below.

We begin by harvesting the mRNA from a biological sample. The mRNA is single-stranded and has a poly-A tail (Figure 1.7 A). The mRNA is difficult to work with, as it is prone to degradation, but DNA is more stable. We would thus like to map the mRNA to cDNA. To get to DNA, we introduce a biotin-labeled dT primer (Figure 1.7 B) and use reverse transcriptase to synthesize more stable double-stranded complementary DNA (cDNA; Figure 1.7 C). Like the initial mRNA, there is something special about one end (the biotin label), and we can use this to "anchor" the cDNAs.

We anchor the cDNAs by binding the biotin to streptavidin-coated beads. To focus on specific sites within the sequences, we introduce a restriction enzyme, known in the SAGE context as the "anchoring en-

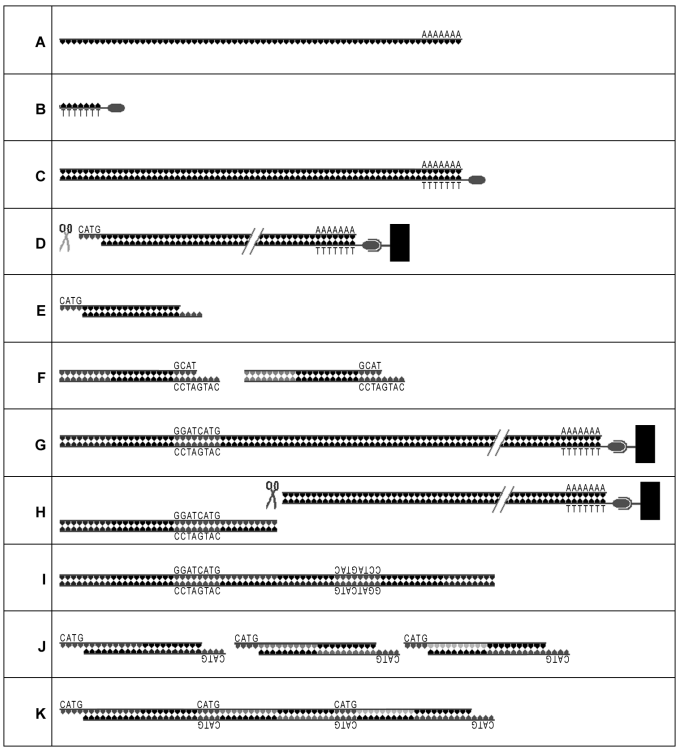

Fig. 1.7. Steps in the preparation of a SAGE library. (A) Extract mRNA. (B) Add a biotin-labeled primer. (C) Synthesize cDNA. (D) Cleave with an anchoring enzyme (AE). (E) Discard loose segments. (F) Split cDNA into two pools, and introduce a linker for each. (G) Ligate linker to bound cDNA fragments. (H) Cleave the product with a tagging enzyme, and discard the bound parts. In addition to the linker, the piece remaining contains a 10-base "tag" that can be used to identify the initial mRNA. (I) Ligate the fragments, and use PCR starting from the primers attached to the linkers to amplify. (J) Cleave with the AE again, and discard the pieces bound to linker. The remaining fragments contain pairs of tags, or "ditags", bracketed by the motif recognized by the AE. (K) Ligate the ditags and sequence the product.

zyme" (AE), which will cut the cDNA whenever a specific DNA "motif" occurs. We will only measure genes that contain at least one occurrence of the motif, so we want the motif to be fairly common; this in turn implies that the motif should be fairly short. Conversely, we don't want the motif to be too short, or it will reduce the number of distinct subsequences available afterwards. The most commonly used such enzyme is NlaIII, which searches for the motif "CATG". When this enzyme cleaves

the cDNA, it produces an "overhang" (an unmatched single strand) at the cleavage site (Figure 1.7 D). Cleaving produces a number of substrands, most of which are "loose" — unconnected to the streptavidin bead (Figure 1.7 E). These loose fragments are washed away before the next step. At this point, we have zoomed in on a particular site on each cDNA: the occurrence of the AE motif closest to the bead (the mRNA poly-A tail).

As noted, cleaving typically produces an "overhang". We can use this overhang to bind new "linker sequences" at the end. As it turns out, we're going to bind *two distinct linkers* (Figure 1.7 F). The two distinct linkers will be exploited in a PCR amplification step described below. So, we divide the material into two pools, and add the two linkers. The linkers are different only at one end; at the other they have an overhang (to match the bound sequence) and another short motif, which will guide yet another enzyme. Within each pool, the linker sequences will bind to the bound cDNAs due to base pairing – and the sequences are ligated (Figure 1.7 G).

Next, we introduce a "type IIS" restriction enzyme (called the "tagging enzyme"; TE) which looks for the motif we introduced with the linker sequence. Type IIS restriction endonucleases cleave not at the motif itself, but rather a specific number of base pairs (say 20) away from it. Unlike the motif for the anchoring enzyme, the motif for the tagging enzyme is asymmetric, so there is a direction for placing the cut site. This cut is "blunt", producing no overhang (Figure 1.7 H).

At this point, the loose double strands in a pool have, in order: linker, the TE motif, the AE motif, and *the 10 bp from the cDNA next to the anchoring enzyme motif closest to the poly-A tail.* This 10bp subsequence is the "tag" that we shall use to identify the parent gene.

To focus on the tags, we now remove the beaded ends, leaving just the loose double strands. We then combine the two resultant pools, so that we have loose strands with two different linkers. We then induce ligation amongst the strands (Figure 1.7 I).

The sequence geometry is now

Linker A – TE AE (motifs) – ditag – AE TE – Linker B

where the central region, or "ditag" contains the identifying information for two distinct transcripts. Ideally, this ditag is bounded by linker A on one side, and linker B on the other. However, since the ligation is not targeted, it is possible to get linker A (or B) on both sides.

We now have a pool of DNA, but not necessarily a large amount.

Tag	Count	Tag	Count	Tag	Count
CCCATCGTCC	1286	CCTGTAATCC	448	TGATTTCACT	358
CCTCCAGCTA	715	TTCATACACC	400	ACCCTTGGCC	344
CTAAGACTTC	559	ACATTGGGTG	377	ATTTGAGAAG	320
GCCCAGGTCA	519	GTGAAACCCC	359	GTGACCACGG	294
CACCTAATTG	469	CCACTGCACT	359	...	...

Table 1.1. Part of a SAGE library.

Since it's easier to work with large amounts of DNA, we amplify what we have using PCR. PCR requires primers *at both ends* of the target amplification sequence, and we can choose primers to match the two distinct linkers (this is why we divided things into two pools). Thus, the resultant products will be overwhelmingly of the form shown above, with linker A at one end and linker B at the other.

An amplified ditag with linkers has a fairly well-defined mass, so filtering of unwanted amplification products can be achieved using a gel. At this point, the information containing part of the data has been compressed (the length of the linker is less than the length of the gene, on average), but the linkers and enzyme motifs are still extraneous and we would prefer not to sequence them. Fortunately, if we reintroduce the AE, the linkers and the TE motifs will be cleaved off from the ditags. To isolate the ditags (Figure 1.7 J), we use another gel to select for the appropriate target mass.

After the above selection and cleaving, the information content of a short piece of DNA (ditag plus overhangs) is quite high, but short reads are inefficient with respect to sequencing. Thus, we ligate the ditags together (Figure 1.7 K). We then sequence the concatenated product. A typical sequencing read involves 500bp, or about 20 ditags and motifs. The AE motif actually provides a useful bit of "punctuation" for quality control purposes.

Within the read, we locate a bracketing pair of motifs and extract the ditag (this can be between 20 and 26 bp). The 10 bp closest to the left end give one tag, and the 10 bp closest to the right end are reversed and complemented to give the other. The tabulated results from a set of reads comprise a SAGE "library". Part of a typical SAGE library is shown in Table 1.1.

Given the data, what questions can we ask? The most common goal is (as with microarray experiments) to find genes that show expression

levels that vary with phenotype. There are, however, complexities associated with the methods of measurement.

The first question is whether we see all the data. There are some sequences that we should not see. If we see the AE motif within a tag, we know that that is an artifact and should be excluded. In many cases, sequences corresponding to mitochondrial DNA will also be excluded. If there are multiple occurrences of a given ditag, typically only one is recorded, to preclude biases associated with PCR amplification. If there are genes that do not contain an occurrence of the cleavage site, these will not be seen. Similarly, if a cleavage site is too close to the poly-A tail, the true identity may be obscured. Conversely, if the RNA is of poor quality, sequence degradation can remove the cleavage site altogether.

There are other issues related to whether the tags we do see are "correct". Mappings of tags to genes are not always unique; the math suggesting that 10 bp should be "enough" relies on independence assumptions that likely do not hold. At present, our genomic information is still a draft, so annotations are not fixed. At the processing level, there are sequencing errors. Published rates are about 0.7% per bp, so to a first approximation 7% of the tags will be so affected [66]. This can produce small "shadow" counts for tags that are "similar" to abundant tags. This renders estimation with rare counts difficult, and somewhat limits the dynamic range.

Interim fixes have been suggested for some of the above problems, but there is still room for improvement. "Long SAGE" [58], where the tags are 14 bp or more in length, has been introduced to address the issue of identification ambiguity. Many of the issues with identification could also potentially be resolved by using multiple restriction enzymes to produce "coupled libraries", but in practical terms this rarely happens (see, however, [72]). Errors in sequencing can be addressed by deconvolution, pulling shadows back to their source, given definitions of a local neighborhood in the sequencing space [18]. Alternatively, information about the tag quality could be acquired at the time of sequencing and used to suggest the most likely fixes; sequences can produce quality "phred" scores associated with each base read [23, 11].

Once the table of counts has been "finalized", there is still the question of choosing a good test statistic for assessing differential expression. Many statistics have been proposed, most focusing on comparing one library with another and dealing primarily with the Poisson sampling variability associated with extracting a count [77, 40, 5, 30, 17, 34, 44, 42, 55]. Some papers have looked at more than two groups [77, 52, 57, 46],

and some analogs of ANOVA have been suggested [26, 65]. However, each library supplies a vector of proportions for an individual. Even under ideal conditions, estimates of the true level of a proportion in a group of individuals are subject to two sources of error: binomial variation associated with the count nature of the data, and variation in proportions between individuals within a group [6, 7]. Better methods for combining these proportions to estimate contrasts are still under development.

At present, SAGE is not as widely used as microarrays, due primarily to the higher costs of assembling libraries. However, these costs are also linked to the costs of sequencing, and the approach may become more viable as sequencing gets easier. The sequencing and counting approach, however, still has many open questions associated with it. Given estimated rates of sequencing errors, what is the realistic dynamic range of this approach? Given this dynamic range, how big does a library need to be to catch the measurable changes stably? Given the relative sizes of the between and within library variance components, should we assemble more small libraries or a small number of big ones? Massively parallel signature sequencing (MPSS) [16] enables the assembly of huge libraries, but the costs are still high. If we compare SAGE and microarray results, how should we measure agreement?

There are some software packages available for analyzing SAGE data, and some large repositories of SAGE data. We recommend SAGE Genie [14] as a source of data for further exploration.

1.4 Mass Spectrometry

Microarrays and SAGE let us measure the relative abundance levels of thousands of mRNA transcripts all at once, giving us some picture of the dynamic activity within the cell. However, much of the action is happening at the protein level, and we'd really like to have the equivalent of a microarray for proteins as well. Some progress has been made on this front, but there are several limitations here.

- the number of distinct proteins is larger than the number of genes.
- many proteins undergo post-translational modifications (eg, phosphorylation), and it is the amount of modification that can affect things.

Thus, it can be hard to get abundance and identity at the same time. However, we can make substantial progress if we relax one of these constraints, getting only partial identification. One tool for getting such

information, letting us measure hundreds of proteins at once, is mass spectrometry. (More extensive descriptions are given in [38, 62].)

Mass spectrometry works by taking a sample and sequentially adding a charge to the substances to be measured (ionizing proteins, protein fragments, or peptides), using electromagnetic manipulation to separate the ionized peptides on the basis of their mass to charge (m/z) ratios, and using a detector to count the abundance of ions with a given m/z ratio. Plotting abundance as a function of m/z gives a mass spectrum. There are many variants of mass spectrometry, corresponding to different modular configurations of ionization, separation, and detection tools (not all combinations are possible), with much greater emphasis on the methods of ionization and separation than detection.

Mass spectrometry has been around for a long time; it was first introduced by J.J.Thomson around 1900, but it is only in recent decades that it has generated great excitement as a tool for exploring the proteome. This delay was due to limitations of the first few ionization methods available; charges were attached or broken off with sufficient force that larger molecules (including proteins) were torn apart into much smaller chunks. The late 1980s saw the introduction of two "soft" ionization methods, matrix-assisted laser desorption and ionization (MALDI) [31, 32] and electrospray ionization (ESI) [24] that allowed measurements to extend to the tens and hundreds of kiloDaltons (kDa, 1 Da = the mass of a hydrogen atom).

Recently, mass spectra have begun to be explored for their potential diagnostic utility — can peaks in the spectra serve as biomarkers of the early stages of diseases such as cancer? See, for example, [1, 37, 48, 49, 53, 71, 78]. While similar questions have been asked with respect to microarrays, a key difference has been that many explorations with mass spectra have focused on spectra obtainable from readily available biological fluids such as blood, urine, or saliva. In this context, the most common mass spectrometry methods used have been variants of MALDI coupled with a time-of-flight (TOF) ion separator (MALDI-TOF). This is the only method that we discuss in detail here.

In MALDI-TOF, the sample of interest (e.g., serum) is combined with one of several matrix compounds, and this mixture is applied to a stainless steel plate. As the mixture dries, the matrix forms a crystal structure holding the proteins in place. Many samples are typically spotted on the same plate; one MALDI plate we have (square, and about 7cm on a side) has 100 deposition sites indicated. After the samples have been spotted, the plate is inserted into a receiving chamber connected

to the main measurement instrument. The chamber is then pumped out to near vacuum conditions. A robotic arm is used to position the plate so that the spot of interest is in a desired target area, and a laser is then fired at the spot. Most of the laser energy goes into breaking the crystal structure of the matrix apart, and less to shaking the peptides apart. The physics of exactly how this works is not well understood. As a result of the matrix fragmentation, many peptides break free into the gas phase. Most matrix compounds are slightly acidic, and thus are willing to donate spare protons to nearby molecules during fragmentation — the peptides going into gas phase are ionized by capturing a small number of protons (typically 1–3 in the data we have seen). In the receiving chamber, a strong electric field propels the ions towards a flight tube. This electric field is typically set up by raising the potential of the plate itself (to V) before the laser is fired; the flight tube entrance is at zero. The flight tube itself is field-free, so the ions drift with the velocity imparted by the electric field until they reach a detector at the far end of the tube. The detector attempts to record the number of ions hitting it as a function of time of flight, assembling an initial form of the spectrum. Typically, several (∼100) laser shots are made and the resulting spectra are summed to produce the final spectrum examined. To first order, the ions all cover the same potential difference and thus the kinetic energy imparted is proportional to the number of unbalanced charges, z (spare protons), the ion is carrying. The flight tube itself is typically much longer than the region over which the potential difference exists, and so the time spent in the acceleration region is typically discounted and the ion is treated as moving at a fixed velocity down the flight tube. Equating expressions for kinetic energy, we get

$$\frac{1}{2}mv^2 = z * V.$$

As the velocity is fixed in the drift tube, $v = L/t$ where L is the length of the tube and t is the time of flight. Substituting and rearranging the above equation, we get

$$m/z = t^2 * (2V/L^2) = kt^2,$$

showing how the m/z ratio can be inferred from the time of flight.

MALDI spectra are commonly supplied as comma-separated value (csv) files with two columns, containing the m/z value and spectrum intensity for each digitizer sample. Ignoring the m/z values, the rows give intensities that are equally spaced in time. An example MALDI

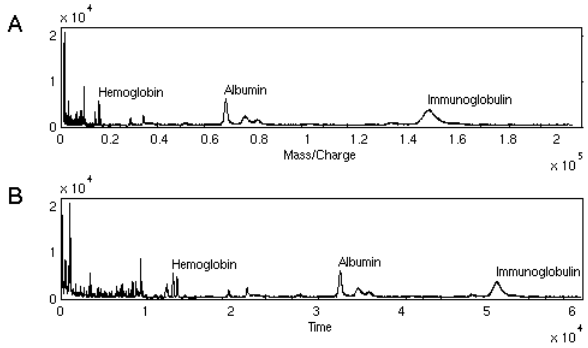

Fig. 1.8. Two views of the same MALDI-TOF spectrum. (A) Intensity plotted as a function of m/z, which is the standard display option. (B) Intensity plotted against time-of-flight, which is directly recorded by the instrument; m/z is a derived quantity. There are two natural scales on which to look at this data, as the time to m/z mapping is not linear.

spectrum is shown in Figure 1.8. The same spectrum is plotted against m/z in the panel A (the most common display option), and against file row in the panel B. This dual presentation is to emphasize that there is more than one natural scale on which to examine this type of data. This spectrum was derived from a serum sample, and peaks at 66 kDa and 150 kDa correspond to albumin and immunoglobulin, known serum proteins. Most of the interest in the biomarker papers published to date has been focused at somewhat lower m/z values; the identities of many of the peaks seen here are not known, and we want to find some that are present in patients with disease and not present in those without, or vice-versa.

It is important to realize that not all of the peptides present in the sample will be seen in a spectrum. Different types of matrix can cause different groups of peptides to ionize more readily, so choosing a specific matrix amounts to choosing a subset of the peptides to be examined. It is common to further subset the peptides by "fractionating" the samples in a variety of ways; some separation axes include pH (acidity) and hydrophobicity (greasiness). Fractionation yields two clear benefits. First, it can allow for more precise identification of a peptide of interest. Several peptides may share a common (or very similar) m/z value, and thus be "aliased" if the entire sample is used. Fractionation introduces a second axis of separation for dealiasing. Second, it can remove (or split off) some of the most abundant peptides. This is an issue because our

present instruments have a limited dynamic range, so if an abundant peptide is present at a level of 100, a trace peptide present at a level less than 1 will simply not be seen. The dynamic range of protein expression is thought to cover 9 or so orders of magnitude, which means that truly scarce peptides will be difficult to detect even with extensive fractionation [21]. The downside of fractionation is that it requires more time, effort, and amount of starting material. One variant of MALDI, known as surface-enhanced laser desorption and ionization (SELDI), works by depositing the sample/matrix mixture on a chemically precoated surface, where different surface coatings allow us to bind different subsets of peptides with high efficiency. SELDI has been commercialized by the company Ciphergen, which sells chips with different coatings preapplied, so some fractionation is done for you. Ciphergen also sells their own instruments and software, but there has been some experimentation with reading Ciphergen chips with other instruments.

Having introduced the structure of the data, we now turn to processing issues: Given a set of spectra, what do we have to do to it before analyzing an expression matrix? A partial list of important steps includes

- Spectral calibration,
- Correcting for matrix noise,
- Spectral denoising,
- Baseline estimation and subtraction,
- Peak detection and quantification,
- Normalization,
- Looking for common patterns and modifications (harmonics),

and we will address each in turn.

Earlier, we derived the relationship $m/z = kt^2$. In theory, physical parameters such as the potential difference, tube length, and digitizer rate of the detector are known and a value for k can be derived. In practice, the same peak may drift slightly over time due to changes in the instrument. One common way of addressing this problem is to run a "calibration sample" consisting of only a small number of proteins whose identities are known *a priori*, producing a spectrum with a small number of clearly defined peaks, as illustrated in Figure 1.9. The masses of the peptides are known, the flight times are empirically observed, and a set of (mass, time) pairs is used to fit a quadratic model of the form

$$m/z = at^2 + bt + c$$

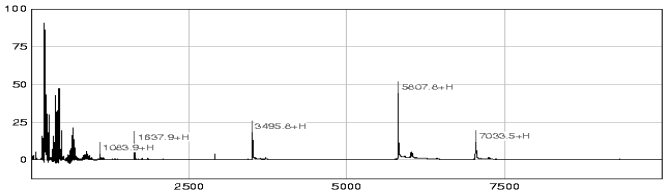

Fig. 1.9. A SELDI calibration spectrum. The sample was comprised of a small number of known peptides, and the associated peaks are clearly seen. The known masses and the observed times-of-flight are then used to fit a quadratic calibration equation.

by least squares. The model parameters found are then assumed to hold for several samples. These parameters can change over time, so it is often useful to check that some of the biggest peaks seen "line up" across samples [29].

Matrix noise is a problem unique to MALDI. When a sample is blasted with a laser, many things break free, not just the peptides of interest. This other, unwanted stuff is colloquially referred to as "matrix noise", and it is predominantly present at the very low m/z end of the spectrum. Matrix noise can often saturate the detector, and detectors do not immediately recover after saturation. This effect is quite unstable [41]. Empirically, this has largely been addressed by excluding values below some chosen m/z cutoff. Exactly where this cutoff should go is not clear, and it can be affected by other machine settings such as the laser intensity. Higher intensity settings can blast loose heavier ions, allowing higher m/z regions to be explored, but these same settings kick up more noise and distort a larger low m/z region with noise. Conversely, low m/z regions can be probed with lower laser settings.

Mathematically, we tend to think of spectra as being comprised of 3 pieces – the signals we want to extract, which are present as peaks, a smooth underlying baseline, and some high frequency noise. In short,

$$Y_i(t) = k_i S_i(t) + B_i(t) + \epsilon_{it},$$

where $Y_i(j)$ is the intensity of spectrum i at time index t, k_i is a normalization factor, S_i is the protein signal of interest (a set of peaks), B_i is baseline, and $\epsilon \sim N(0, \sigma^2(t))$. We would like to remove the noise, subtract the baseline, estimate the peaks, and scale the spectra. There is a natural order to these steps, and performing them out of sequence (or omitting some) can make the downstream analysis more difficult.

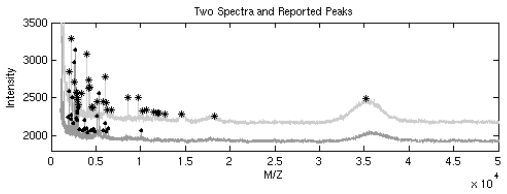

Fig. 1.10. Two raw MALDI spectra, with the peaks and intensities automatically flagged by software superimposed. There are differences in baseline and scaling visible in the raw spectra. These differences should be corrected for, but this was not done for the peaks found. Baseline cannot be estimated from the peaks alone.

Many mass spectrometry instruments are sold with associated software that will perform peak detection and quantification automatically, but these may not address all of the steps. For one dataset we examined, we were supplied with both raw spectra and associated lists of peak locations and intensities. Two spectra from this set are plotted as curves in Figure 1.10, with the peaks supplied plotted as asterisks. The two spectra obviously have different baseline levels, still have additive white noise present, and may involve different normalization factors. However, the peak lists supplied use the intensities from the peaks before adjusting for baseline or normalization, and baseline cannot be reliably estimated from the peaks alone. We also note that one of the larger peaks, near m/z 36000, is missed in one spectrum because it wasn't "sharp enough". Matrix noise is present at the very lowest m/z values, where the spectra jump out of view [10].

One problem with both denoising and peak detection is simply that peaks can have different shapes in different parts of the m/z range; higher m/z peaks are broader. Some factors that can contribute to this broadening are: uncertainty in the initial velocity of the peptide, isotopic spread, and the nonlinearity of the clock tick to m/z mapping. The last of these was mentioned earlier, so we expand only on the former two. When peptides are blasted loose from the matrix crystal, all peptides of the same type do not break out with the same initial velocity. Rather, there is a velocity distribution, causing the peak to be spread out. This spread becomes more pronouced the longer the peptide drifts down the tube, and is thus bigger at higher m/z values. For higher m/z peptides, the definition of "mass" can actually be somewhat ambiguous. Carbon, for example, exists 99% as ^{12}C, and 1% as ^{13}C. If a peptide contains 100 carbon atoms, the mass contribution from these atoms will be roughly

1200 plus a small integer; this integer will have a Poisson distribution with mean 100*1% = 1. Similar effects are associated with other elements. The overall isotopic spread widens as mass increases, so that it is common to refer to both the monoisotopic mass (assuming all carbons have mass 12) and the average mass (incorporating the isotopic effects). It is possible to devise an approximate isotopic spread for a peptide given either mass estimate, using the general abundances of carbon and other elements in the population of amino acids. This can be used to sharpen the peaks through deconvolution.

There are a number of denoising filters that exist for spectra (eg, Savitzky-Golay), but we admit a preference for wavelet-based methods which adapt naturally to the multi-scale nature of the data. Here, we map to the wavelet domain, zero out the small coefficients (hard thresholding), and map back before looking for peaks [19].

Once the spectra have been smoothed, we attempt to estimate baseline. At present, we do not use very sophisticated algorithms for this purpose, generally sticking with a local minimum fit so that negative intensities will not be produced by subtraction. Again, the "local" neighborhood used needs to be altered as m/z increases. Even with basic algorithms, the effects can be rather dramatic. In Figure 1.11, we show spectra derived from 20 pH fractions for a single patient both before and after denoising and baseline subtraction (panels A and B, respectively). In this case, baseline subtraction causes the more dramatic effect, giving all of the base levels the same hue. As an aside, we note that this display also points out that fractionation is an imperfect procedure, and that signal from the same peptide can be found in several adjacent fractions.

After subtracting baseline from smoothed spectra, we still need to identify peaks and get summary values for them. A first pass approach can use a simple maximum finder. We could attempt to use peak areas instead, but we do not pursue this here. We note, however, that locating the peaks can be aided by considering a set of spectra rather than a single spectrum. Assuming the spectra have been roughly aligned, we have found it useful to average spectra within a group and perform peak detection on the average spectrum [45]. Averaging may even be useful before doing wavelet denoising, as small peaks can be reinforced as the noise level drops, and they can be retained. Values for individual spectra can be extracted as local maxima in small windows about the central peak location. The width of this window can be linked to the nominal precision of the instrument. For a low-resolution instrument, the uncertainty can be on the order of 0.1% of the nominal m/z; higher-

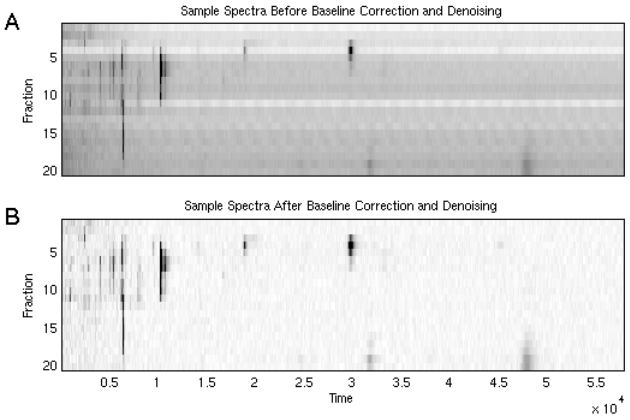

Fig. 1.11. Spectra derived from 20 pH fractions of the serum from a single patient. (A) Raw spectra. There are clear differences in baseline, seen as different shadings for the rows. There is also some unwanted noise, visible as periodic ripples in the spectra. (B) The same spectra after correcting for baseline and denoising. Peaks stand out more clearly against a flat "surface". In both cases, peaks can extend across neighboring fractions, as the separation process is imperfect.

resolution instruments will attain mass accuracies expressed in parts per million (ppm).

Before comparing peak intensities across spectra, we need to normalize the spectra to make them comparable. One common method is to use the total ion current, or summed intensities for the entire spectrum. This is done after excluding the matrix noise region and subtracting baseline. This step is where we feel there is the most room for improvement, as there may be local scaling factors that are more appropriate than a single factor throughout. Even if a single scaling is to be used, it may be better to identify a small number of key peaks that appear to be relatively stable and to target the median log ratio for the set of peaks.

Having identified some peaks as being of potential interest, it also makes sense to look at other peaks that may be related (as assessed by correlation) or that should be related. The idea of "should be related" is different for mass spectrometry data than for microarray data in that there is a natural ordering to the peaks in a spectrum. In Figure 1.12, we show zooms on two distinct regions of averaged spectra from a higher-resolution (Qstar) instrument. The patterns of peaks look the same, though the m/z range in the bottom panel is half that of the top panel,

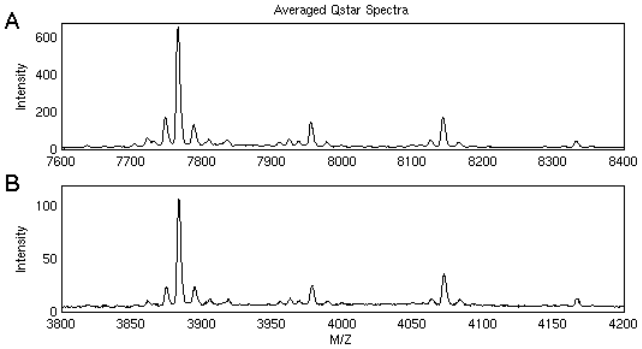

Fig. 1.12. Two regions derived from the average of several high-resolution Qstar spectra. (A) The m/z range from 7600 to 8400. (B) The m/z range from 3800 to 4200; values exactly half those in A. The peak patterns in the two panels are perfectly aligned, as we are seeing the same peptides. In A, the peptides are singly charged ($z = 1$), and in B they are doubly charged ($z = 2$). Other regularities (offsets of 189 in A) are due to further identifiable phenomena (matrix adducts).

and the intensities are dramatically reduced. In this case, the parallel structure is due to the fact that the two panels are showing singly and doubly charged versions of the same peptides; finding the appropriate harmonic patterns on the m/z scale can tell us both the charge state of the peptide (and thus its mass) and provide some reassurance that we have identified it correctly. (With higher resolution data, the charge state can also be inferred from the spacing between isotopic peaks, which should be 1Da apart.) Looking at the top panel, we can also see that there are groups of peaks offset from each other by 189 Da. This offset mass matches that of a single molecule of the the matrix used here: α-cyano-4-hydroxycinnamic acid. These peaks are referred to as matrix adducts. Similarly, there are smaller peaks close to the biggest one, with the largest ones 18 Da below the main peak and 22 Da above. These correspond to loss of a water molecule or replacing an ionized hydrogen atom with one of sodium, respectively. Viewing the ensemble, we can see that almost all of the peaks visible here are differently modified forms of the same major peptide.

Graphically, we have found it useful to construct heat maps of the spectral regions surrounding peaks identified as potentially useful markers in a few different ways. First, in a very localized region (say 20 Da on either side) simply to check that the peak is reasonably clear. Second,

in a larger window going out to either side by 250 Da or so, which is wide enough to capture most matrix adducts and common modifications such as phosphorlyation (a mass offset of 80 Da). Third, by checking heatmaps at half and twice the nominal m/z value to check the charge state.

Finally, a note of caution. The use of mass spectrometry data for biomarker discovery is more recent than the use of microarrays, and there are a number of external factors that can introduce unwanted biases. Some of these are discussed in Baggerly et al. [8, 9] and Villanueva et al. [70]. These tools are incredibly sensitive, which they need to be if they are to pick up new biomarkers. This very sensitivity, however, means that they will also pick up changes in experimental conditions quite well. In terms of keeping track of and reporting on your data, we recommend Ransohoff [54] for a discussion of some of the issues, and McShane et al. [43] for a more specific set of guidelines.

1.5 Finding Data

Simply discussing the features of various types of data is no substitute for diving in and working with raw data. If possible, we recommend visiting labs as the data is being collected or trying to collect some yourself. (Our colleagues have been willing to work with us on test cases.) Even without that, raw data of the types discussed are readily available on the web.

Lots of microarray data has been on the web for a while, and much more has been posted since the advent of the mimimum information about a microarray experiment (MIAME) standards [15]. Several major journals now require that the raw data be made available at the time of publication. For Affymetrix data, the first place to go is simply the company's web site, `www.affymetrix.com`. Sample data sets for several different chip types are available, as are all of the CDF files, probe sequences, and the latest annotation for what the probes on the chips actually correspond to. Registration is required, but free. For cDNA microarray data (and Affymetrix data), we also recommend the Gene Expression Omnibus (GEO) maintained by the NCBI, at

`http://www.ncbi.nlm.nih.gov/geo`.

For SAGE data, we recommend SAGE Genie [14], maintained as part of the Cancer Genome Anatomy Project (CGAP) at

`http://cgap.nci.nih.gov/SAGE`.

The data repositories for mass spectrometry data are not yet as ex-

tensive, but several proteomics journals are getting set to require raw data in a fashion akin to MIAME, so we hope this will change shortly. In the meantime, there are a few sites that have data of various types. The best known is probably the Clinical Proteomics program jointly run by the NCI and FDA [48]. The databank is currently located at

`http://home.ccr.cancer.gov/ncifdaproteomics/`

and has various SELDI and Qstar datasets. Questions have been raised about the quality of some of this data, and we strongly recommend reading Baggerly et al. [8] for a more detailed discussion of some of the issues involved. There is some SELDI data available from MD Anderson, at

`http://bioinformatics.mdanderson.org`

together with Matlab scripts for processing and analysis.

Acknowledgements

This work was partially supported by NCI grant CA-107304.

Bibliography

[1] B.-L. Adam, Y. Qu, J. W. Davis, et al., *Serum protein fingerprinting coupled with a pattern-matching algorithm distinguishes prostate cancer from benign prostate hyperplasia and healthy men*, Cancer Res, 62 (2002), pp. 3609–3614.

[2] Affymetrix, *Affymetrix Microarray Suite Users Guide, Version 4.0*, Affymetrix, 1999.

[3] ———, *Affymetrix Microarray Suite Users Guide, Version 5.0*, Affymetrix, 2001.

[4] B. Alberts, A. Johnson, J. Lewis, et al., *Molecular Biology of the Cell (4e)*, Garland Publishing, 2002.

[5] S. Audic and J.-M. Claverie, *The significance of digital gene expression profiles*, Genome Res, 7 (1997), pp. 986–995.

[6] K. A. Baggerly, L. Deng, J. S. Morris, et al., *Differential expression in SAGE: Accounting for normal between-library variation*, Bioinformatics, 19(12) (2003), pp. 1477–1483.

[7] ———, *Overdispersed logistic regression for SAGE: Modelling multiple groups and covariates*, BMC Bioinformatics, 5 (2004), p. 144.

[8] K. A. Baggerly, J. S. Morris, and K. R. Coombes, *Reproducibility of SELDI-TOF protein patterns in serum: Comparing datasets from different experiments*, Bioinformatics, 20(5) (2004), pp. 777–785.

[9] K. A. Baggerly, J. S. Morris, S. R. Edmonson, et al., *Signal in noise: Evaluating reported reproducibility of serum proteomic tests for ovarian cancer*, J Natl Cancer Inst, 97(4) (2005), pp. 307–309.

[10] K. A. Baggerly, J. S. Morris, J. Wang, et al., *A comprehensive approach to the analysis of MALDI-TOF proteomics spectra from serum samples*, Proteomics, 3(9) (2003), pp. 1667–1672.

[11] T. Beissbarth, L. Hyde, G. K. Smyth, et al., *Statistical modeling of sequencing errors in SAGE libraries*, Bioinformatics, 20 (2004), pp. i31–i39.

[12] J. M. Bland and D. G. Altman, *Statistical method for assessing agreement between two methods of clinical measurement*, The Lancet, i (1986), pp. 307–310.

[13] B. M. Bolstad, R. A. Irizarry, M. Åstrand, et al., *A comparison of normalization methods for high density oligonucleotide array data based on variance and bias*, Bioinformatics, 19 (2003), pp. 185–193.

[14] K. Boon, E. C. Osorio, S. F. Greenhut, et al., *An anatomy of normal and malignant gene expression*, Proc Natl Acad Sci USA, 99(17) (2002), pp. 11287–11292.

[15] A. Brazma, P. Hingamp, J. Quackenbush, et al., *Minimum information about a microarray experiment (MIAME): Toward standards for microarray data*, Nature Genetics, 29 (2001), pp. 365–371.

[16] S. Brenner, M. Johnson, J. Bridgham, et al., *Gene expression analysis by massively parallel signature sequencing (MPSS) on microbead arrays*, Nature Biotechnology, 18(6) (2000), pp. 630–634.

[17] H. Chen, M. Centola, S. F. Altschul, et al., *Characterization of gene expression in resting and activated mast cells*, J Exp Med, 188(9) (1998), pp. 1657–1668.

[18] J. Colinge and G. Feger, *Detecting the impact of sequencing errors on SAGE data*, Bioinformatics, 17(9) (2001), pp. 840–842.

[19] K. R. Coombes, S. Tsavachidis, J. S. Morris, et al., *Improved peak detection and quantification of mass spectrometry data acquired from surface-enhanced laser desorption and ionization by denoising spectra with the undecimated discrete wavelet transform*, Proteomics, 5(16) (2005), pp. 4107–4117.

[20] L. M. Cope, R. A. Irizarry, H. A. Jaffee, et al., *A benchmark for Affymetrix genechip expression measures*, Bioinformatics, 20 (2004), pp. 323–331.

[21] E. P. Diamandis, *Analysis of serum proteomic patterns for early cancer diagnosis: Drawing attention to potential problems*, J Natl Cancer Inst, 96 (2004), pp. 353–356.

[22] S. Dudoit, Y. H. Yang, M. J. Callow, et al., *Statistical methods for identifying differentially expressed genes in replicated cDNA microarray experiments*, Statistica Sinica, 12(1) (2002), pp. 111–139.

[23] B. Ewing, L. Hillier, M. C. Wendl, et al., *Base-calling of automated sequencer traces using Phred. I. Accuracy assessment*, Genome Res, 8 (1998), pp. 175–185.

[24] J. B. Fenn, M. Mann, C. K. Meng, et al., *Electrospray ionization for mass spectrometry of large biomolecules*, Science, 246 (1989), pp. 64–71.

[25] R. Gentleman, V. J. Carey, W. Huber, et al., eds., *Bioinformatics and Computational Biology Solutions Using R and Bioconductor*, Springer-Verlag, 2005.

[26] L. D. Greller and F. L. Tobin, *Detecting selective expression of genes and proteins*, Genome Res, 9 (1999), pp. 282–296.

[27] R. A. Irizarry, B. M. Bolstad, F. Collin, et al., *Summaries of Affymetrix genechip probe level data*, Nucleic Acids Res, 31 (2003), p. e15.

[28] R. A. Irizarry, B. Hobbs, F. Collin, et al., *Exploration, normalization, and summaries of high density oligonucleotide array probe level*

data, Biostatistics, 4 (2003), pp. 249–264.

[29] N. Jeffries, *Algorithms for alignment of mass spectrometry proteomic data*, Bioinformatics, 21 (2005), pp. 3066–3073.

[30] A. J. Kal, A. J. van Zonneveld, V. Benes, et al., *Dynamics of gene expression revealed by comparison of serial analysis of gene expression transcript profiles from yeast grown on two different carbon sources*, Mol Biol Cell, 10 (1999), pp. 1859–1872.

[31] M. Karas, D. Bachmann, U. Bahr, et al., *Matrix-assisted ultraviolet laser desorption of non-volatile compounds*, Intl J Mass Spectrometry Ion Processes, 78 (1987), pp. 53–68.

[32] M. Karas and F. Hillenkamp, *Laser desorption ionization of proteins with molecular masses exceeding 10,000 Daltons*, Anal Chem, 60(20) (1988), pp. 2299–2301.

[33] M. K. Kerr, M. Martin, and G. A. Churchill, *Analysis of variance for gene expression microarray data*, J Comp Biol, 7 (2000), pp. 819–837.

[34] A. Lal, A. E. Lash, S. F. Altschul, et al., *A public database for gene expression in human cancers*, Cancer Res, 59 (1999), pp. 5403–5407.

[35] C. Li and W. H. Wong, *Model-based analysis of oligonucleotide arrays: Expression index computation and outlier detection*, Proc Natl Acad Sci USA, 98 (2001), pp. 31–36.

[36] ———, *Model-based analysis of oligonucleotide arrays: Model validation, design issues and standard error application*, Genome Biol, 2(8) (2001), p. RESEARCH0032.

[37] J. Li, Z. Zhang, J. Rosenzweig, et al., *Proteomics and bioinformatics approaches for identification of serum biomarkers to detect breast cancer*, Clin Chem, 48(8) (2002), pp. 1296–1304.

[38] D. Liebler, *Introduction to Proteomics: Tools for the New Biology*, Humana Press, 2001.

[39] R. J. Lipshutz, S. P. Fodor, T. R. Gingeras, et al., *High density synthetic oligonucleotide arrays*, Nature Genetics, 21 (1999), pp. S20–S24.

[40] S. L. Madden, E. A. Galella, J. Zhu, et al., *SAGE transcript profiles for p53-dependent growth regulation*, Oncogene, 15 (1997), pp. 1079–1085.

[41] D. I. Malyarenko, W. E. Cooke, B.-L. Adam, et al., *Enhancement of sensitivity and resolution of surface-enhanced laser desorption/ionization time-of-flight mass spectrometric records for serum peptides using time-series analysis techniques*, Clin Chem, 51 (2005), pp. 65–74.

[42] M. Z. Man, X. Wang, and Y. Wang, *POWER_SAGE: comparing statistical tests for SAGE experiments*, Bioinformatics, 16(11) (2000), pp. 953–959.

[43] L. M. McShane, D. G. Altman, W. Sauerbrei, et al., *Reporting recommendations for tumor marker prognostic studies (REMARK)*, J Natl Cancer Inst, 97 (2005), pp. 1180–1184.

[44] E. M. C. Michiels, E. Oussoren, M. van Groenigen, et al., *Genes differentially expressed in medulloblastoma and fetal brain*, Physiol Genomics, 1 (1999), pp. 83–91.

[45] J. S. Morris, K. R. Coombes, J. Koomen, et al., *Feature extraction and quantification for mass spectrometry data in biomedical applications using the mean spectrum*, Bioinformatics, 21(9) (2005), pp. 1764–1775.

[46] M. Nacht, T. Dracheva, Y. Gao, et al., *Molecular characteristics of non-small cell lung cancer*, Proc Natl Acad Sci USA, 98(26) (2001),

pp. 15203–15208.
[47] Nature, *The chipping forecast*, Nature Genetics Supplement, 21 (1999).
[48] E. F. Petricoin, III, A. M. Ardekani, B. A. Hitt, et al., *Use of proteomic patterns in serum to identify ovarian cancer*, The Lancet, 359 (2002), pp. 572–577.
[49] E. F. Petricoin, III, D. K. Ornstein, C. P. Paweletz, et al., *Serum proteomic patterns for detection of prostate cancer*, J Natl Cancer Inst, 94(20) (2002), pp. 1576–1578.
[50] K. Polyak and G. J. Riggins, *Gene discovery using the serial analysis of gene expression technique: Implications for cancer research*, J Clin Oncology, 19(11) (2001), pp. 2948–2958.
[51] K. Polyak, Y. Xia, J. L. Zweler, et al., *A model for p53-induced apoptosis*, Nature, 389 (1997), pp. 300–305.
[52] D. A. Porter, I. E. Krop, S. Nasser, et al., *A SAGE (serial analysis of gene expression) view of breast tumor progression*, Cancer Res, 61 (2001), pp. 5697–5702.
[53] A. J. Rai, Z. Zhang, J. Rosenzweig, et al., *Proteomic approaches to tumor marker discovery: Identification of biomarkers for ovarian cancer*, Arch Path Lab Med, 126 (2002), pp. 1518–1526.
[54] D. F. Ransohoff, *Bias as a threat to the validity of cancer molecular-marker research*, Nature Reviews Cancer, 5(2) (2005), pp. 142–149.
[55] J. M. Ruijter, A. H. C. van Kampen, and F. Baas, *Statistical evaluation of SAGE libraries: Consequences for experimental design*, Physiol Genomics, 11 (2002), pp. 37–44.
[56] B. Ryu, J. Jones, N. J. Blades, et al., *Relationships and differentially expressed genes among pancreatic cancers examined by large-scale serial analysis of gene expression*, Cancer Res, 62 (2002), pp. 819–826.
[57] B. Ryu, J. Jones, M. A. Hollingsworth, et al., *Invasion-specific genes in malignancy: Serial analysis of gene expression comparisons of primary and passaged cancers*, Cancer Res, 61 (2001), pp. 1833–1838.
[58] S. Saha, A. B. Sparks, C. Rago, et al., *Using the transcriptome to annotate the genome*, Nature Biotechnology, 20 (2002), pp. 508–512.
[59] E. E. Schadt, C. Li, B. Ellis, et al., *Feature extraction and normalization algorithms for high-density oligonucleotide gene expression array data*, J Cell Bioc, 37 (2001), pp. S120–S125.
[60] M. Schena, ed., *Microarray Biochip Technology*, BioTechniques Books, 2000.
[61] M. Schena, D. Shalon, R. W. Davis, et al., *Quantitative monitoring of gene expression patterns with a complementary DNA microarray*, Science, 270 (1995), pp. 467–470.
[62] G. Siuzdak, *The Expanding Role of Mass Specrometry in Biotechnology*, MCC Press, 2003.
[63] T. Speed, ed., *Statistical Analysis of Gene Expression Microarray Data*, Chapman and Hall/CRC Press, 2003.
[64] B. St. Croix, C. Rago, V. Velculescu, et al., *Genes expressed in human tumor epithelium*, Science, 289 (2000), pp. 1197–1202.
[65] D. J. Stekel, Y. Git, and F. Falciani, *The comparison of gene expression from multiple cDNA libraries*, Genome Res, 10 (2000), pp. 2055–2061.
[66] J. Stollberg, J. Urschitz, Z. Urban, et al., *A quantitative evaluation of SAGE*, Genome Res, 10 (2000), pp. 1241–1248.

[67] V. E. Velculescu, S. L. Madden, L. Zhang, et al., *Analysis of human transcriptomes*, Nature Genetics, 23 (1999), pp. 387–388.
[68] V. E. Velculescu, L. Zhang, B. Vogelstein, et al., *Serial analysis of gene expression*, Science, 270 (1995), pp. 484–487.
[69] V. E. Velculescu, L. Zhang, W. Zhou, et al., *Characterization of the yeast transcriptome*, Cell, 88 (1997), pp. 243–251.
[70] J. Villanueva, J. Philip, C. A. Chaparro, et al., *Correcting common errors in identifying cancer-specific serum peptide signatures*, J Prot Res, 4 (2005), pp. 1060–1072.
[71] A. Vlahou, P. F. Schellhammer, S. Mendrinos, et al., *Development of a novel proteomic approach for the detection of transitional cell carcinoma of the bladder in urine*, Am J Path, 154 (2001), pp. 1491–1502.
[72] C.-L. Wei, P. Ng, K. P. Chiu, et al., *5' long serial analysis of gene expression (longSAGE) and 3' longSAGE for transcriptome characterization and genome annotation*, Proc Natl Acad Sci USA, 101(32) (2004), pp. 11701–11706.
[73] Z. Wu, R. A. Irizarry, R. Gentleman, et al., *A model based background adjustment for oligonucleotide expression arrays*, J Am Statist Assoc, 99 (2004), pp. 909–917.
[74] Y. H. Yang, M. J. Buckley, S. Dudoit, et al., *Comparison of methods for image analysis on cDNA microarray data*, J Comp Graph Statist, 11 (2002), pp. 108–136.
[75] Y. H. Yang, S. Dudoit, P. Luu, et al., *Normalization for cDNA microarray data: A robust composite method addressing single and multiple slide systematic variation*, Nucleic Acids Res, 30 (2002), p. e15.
[76] L. Zhang, M. F. Miles, and K. D. Aldape, *A model for interactions on short oligonucleotide microarrays: Implications for probe design and data analysis*, Nature Biotechnology, 21(7) (2004), pp. 818–821.
[77] L. Zhang, W. Zhou, V. E. Velculescu, et al., *Gene expression profiles in normal and cancer cells*, Science, 276 (1997), pp. 1268–1272.
[78] Z. Zhang, R. C. Bast, Jr., Y. Yu, et al., *Three biomarkers identified from serum proteomic analysis for the detection of early stage ovarian cancer*, Cancer Res, 64 (2004), pp. 5882–5890.