1

An Introduction to High-Throughput Bioinformatics Data

KEITH A. BAGGERLY, KEVIN R. COOMBES, AND JEFFREY S. MORRIS

University of Texas M.D. Anderson Cancer Center

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 assays 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 Gene Chips

In looking at the structure of Affymetrix data, there are several in-depth resources [2, 3, 39] that 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 (e.g., 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 cannot 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 should not. 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 :    GCTAGTCGATGCTAGCTTACTAGTC
MM :    GCTAGTCGATGCAAGCTTACTAGTC
```

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 "probe sets"; a given gene would be represented on a U133A chip by a probe set containing 11 probe pairs, or 22 probes with distinct sequences. The probes within a probe

set 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 probe sets is not 1-to-1, so some genes are represented by several probe sets.

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 Gene Chips 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
[CEL] Contains feature quantifications
[CDF] Maps between features, probes, probe sets, 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 1.1A.

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 1.1B. 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 1.2A and B, we can see that features are square. The horizontal and vertical

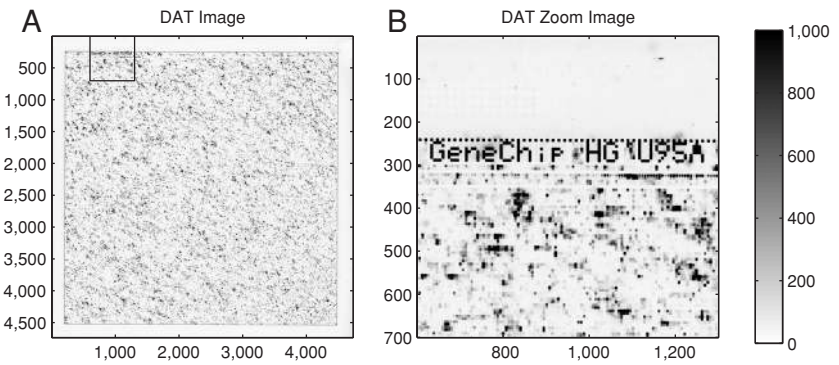

Fig. 1.1. An Affymetrix image (.DAT) file. (A) The entire image, 4,733 pixels on a side, containing 409,600 features. (B) A zoom on the upper left corner of the image. Controls are used in a checkerboard pattern to indicate the print region border, and to designate the chip type. This is a U95Av2 chip; on v2 chips the "A" is filled in.

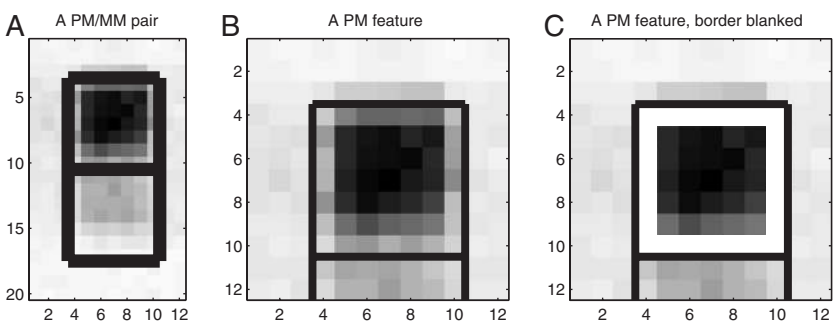

Fig. 1.2. Sets of Affymetrix image features. (A) A PM/MM pair. Note that the PM pixel readings are higher than the MM readings. (B) A zoom on the PM feature. (C) The PM feature after trimming the outer boundary. Only the remaining pixels are used in deriving a summary quantification (the 75th percentile).

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 μm on a side. The scanner used for this scan had a resolution of 3 μm/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.

The DAT file structure consists of a 512-byte header followed by the raw image data. The image shown above involved a 4733×4733 grid of pixels, so the total file size is $2 \times 4733^2 + 512 = 44{,}803{,}090$ bytes (45 MB). 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 1.2C. 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 brackets, 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 of a 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 12 MB from 45 MB 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 $2 \times 640^2 = 819{,}200$ 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 probe set to a single number.

In summarizing a probe set, 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 probe set, 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 probe set, 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 offset is 517) and in the 568th and 569th rows. The index values for these probes are $(567 \times 640) + 517 = 363397$ 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 probe set 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 probe set 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 data set, relating probe set values across chips.

Before we quantify individual probe sets, 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 does not tell us anything new about the underlying biology. In most microarray experiments, we are comparing samples of a single tissue type (e.g., diseased brain to normal brain), and in such cases we *assume* that "most genes do not 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 probe set. The PM and MM features for an example probe set are shown in Figure 1.3A 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 probe set 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 mean. All of the differences falling within this band are then averaged to produce the final AvDiff value. This is illustrated in Figure 1.3C. 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 signals for a probe set are concentrated in a very small number of probes, these may be omitted altogether if they fall outside