Visualizing Genomic Data

Book chapter for Handbook of Computational Statistics, edited by W. Härdle

Florian Hahne, Wolfgang Huber, Robert Gentleman

May 15, 2006

Contents

1 Introduction **2**

2 Data **3**

3 Methodology **4**
3.1 Plotting distributions 4
3.2 Color 6
3.3 Two-dimensional layouts of data 8

4 Visualization of experimental data **9**
4.1 Spatial layout 9
4.1.1 Plate plots of microtitre plates 9
4.1.2 Affymetrix probe set intensities 10
4.2 Distribution summaries 13
4.3 Scatterplots and 2-dimensional density plots 16
4.4 Clustering 16
4.5 Heatmaps 20
4.6 Diagnostics 21

5 Plotting in genomic coordinates **23**
5.1 Along-chromosome plots of high-density tiling array data . . 25
5.2 Genome browsers 28

6 Graphs **28**
6.1 The different graph layout engines in graphviz 30
6.2 Bipartite graphs 31

7 Acknowledgements 34

1 Introduction

The advent of experimental techniques capable of probing biomolecules and cells at high levels of resolution has led to a rapid change in the methods used for experimental molecular biology [8]. The range of applications is dramatic, from basic biology to the study of human diseases. Since these techniques produce massive amounts of data, there has been a corresponding need to develop new statistical methods for modeling and interpreting the observed data.

In this chapter we discuss visualization methods that can be applied to various sorts of data. This is not the place to define and describe all of the relevant biology, and interested readers are referred to texts such as [1, 29] for more complete details. We also note that the examples are intended to convey principles of visualization and are not in any way complete, or definitive descriptions of how to process genomic data; these topics are covered elsewhere [14, 28], for example.

There are many good books on visualization [9, 10, 32, 33] that can be consulted for ideas. Visualization does remain an art, yet there are underlying principles that should be adhered to. Visualization is more than simply producing a plot, or some other graphic. It involves conscious decisions about what message should be conveyed by a particular plot and the choice of methodologies that are likely to convey that information easily and accurately to the user. While we give specific examples, we will also try to point out where general principles apply and to indicate some reasonable extensions that could be made. The use of color is important, and appropriate choice of color schemes is essential.

Our examples and code are all written in R and make use of many different R and Bioconductor packages. Good references for more extensive discussion and examples are [14, 26]. There is an accompanying R package named *CompStatViz* that contains the data and supplementary R code that produces most of the figures, tables and statistics reported in this chapter. It can be obtained from the Bioconductor web site `http://www.bioconductor.org`.

2 Data

We use a number of different data sets in the examples given in this chapter and here provide brief descriptions of the different data sets.

CLL The chronic lymphocytic leukemia (CLL) microarray data were generated in Dr. J. Ritz's lab at the Dana Farber Cancer Institute, by Dr. S. Chiaretti. The data consist of 24 Affymetrix HGU95Av2 arrays on blood samples from patients with CLL. While there are a number of phenotypic characteristics of interest we have restricted our attention to one which is a characteristic of the disease and patients are classified as either stable or progressive. Data on this phenotype is available for 23 of the 24 arrays.

The CLL data are processed using *gcrma* [36] from the *affy* package to compute expression estimates for each probeset. We carried out two separate reductions of the genes. In one we selected those genes where the expression level on the measured scale was larger than 100 in at least 25% of the samples, the IQR was larger than 0.5 on the log scale, and the median expression, across samples, was larger than 300 on the normal scale. Such genes should be useful for prediction, but were purposely selected without regard to phenotype. This subset was used only for the second multidimensional scaling plot.

The second filtering was designed to select genes that are the best differentiators of stable disease versus progressive disease. To that end, in addition to the first two filters described above we also used a t-test to compare the two groups and selected those genes that had an unadjusted p-value less than 0.005. We were left with 81 genes. We will make use of this subset in most of the examples below, and will refer to it as the CLL disease progress subset.

Cytometry data The cytometry data used for some of the examples were generated in a cell-based screen probing for activating and inhibiting effects of over-expression of unknown genes conducted at the German Cancer Research Center Heidelberg by Dr. Dorit Arlt [2]. They comprise four replicate 96 well microtiter plates containing cells transfected with YFP-fused expression constructs. Cell proliferation was monitored though an antibody-based BrdU assay and data was subsequently processed using the package *prada* to obtain measures of inhibition and activation.

High density tiling array data *Saccharomyces cerevisiae* was grown in rich culture and RNA was isolated. Then labeled cDNA was produced and

hybridized to high-density tiling arrays designed by Dr. Lars Steinmetz from the European Molecular Biology Laboratory (EMBL) and manufactured by Affymetrix Corporation. The arrays cover both strands of the whole yeast genome with 25-mer oligonuleotide probes, whose start positions tile in intervals of 8 bases.The measurements on unsynchronized cells were taken by Dr. Lior David at the Stanford Genome Center (Figure 16), those on synchronized cells by Dr. Sandra Clauder-Münster at the EMBL (Figure 17).

3 Methodology

Exploratory data analysis is an omnibus name for a diverse set of computational and visualization methods that are employed to investigate data in order to discover interesting patterns, regularities or irregularities. The goal is not to fit models, to make estimates, or to test hypotheses, but rather to gain insight into the structure of the data without imposing conditions upon them. EDA often relies on graphical methods and many of the examples presented here arise from the application of these principles.

Among the basic graphical principles proposed by [9] and [10] is the importance of comparisons on a common scale. Cleveland also notes that there are problems with the visual comparison of curves, since the eye is drawn to areas where the curves are close, but it is difficult to determine either the vertical separation or the horizontal separation with any precision. However, we are often more interested in accurate estimation of either the horizontal or the vertical distance between the curves.

In other settings, especially those related to linear modeling, graphical methods have made good use of both models and residuals. It is often of substantial interest to visualize the impact of model assumptions and to also study deviations from those assumptions. Residuals can be computed by comparing the observed data with the predictions made using the hypothesized model. To date there has been little use of residuals in the analysis of genomic data, even when studying highly designed microarray experiments where they are likely to be of substantial benefit.

3.1 Plotting distributions

Often one wants to compare the distributions of different populations, or subsets, of the experimental data. For microarray experiments it is common practice to compare the distributions of probe values for different arrays while for flow cytometry data it is often of interest to compare distributions

of forward scatter, or side scatter between the wells in a plate. Three commonly used visualization methods for comparing distributions are boxplots, plots of the empirical distribution function and plots of density estimates. Since they are all relatively easy to produce, it seems prudent to examine all three for data that are to be analyzed. These methods are illustrated in Section 4.2.

Boxplots are useful if the data have a roughly unimodal and symmetric density. If that is not the case, then many important features of the data will not be observed by comparing boxplots. Boxplots show how well the medians align, whether spread, as measured by the IQR, is consistent across the samples and the extent to which outliers exist, again across samples. The comparison is on a common scale; for horizontal boxplots, the y-axis and hence adhere to the principles proposed by Cleveland.

Plots of the empirical distribution function can be illuminating with respect to shifts in the location of the distributions (some forms of this are detected by boxplots, but not all) and to other anomalies that effect the tails of the distribution. It is straightforward, if potentially tedious, to locate the median and other quantiles. Direct pairwise, or groupwise comparisons are less straightforward. A trained eye can detect multimodality and other anomalies, but for these defects a density plot will often yield a more direct answer. Despite these defects, and the fact that they do not adhere to Cleveland's recommendation their use is widespread and they are often valuable showing estimated distributions that cross and other unusual patterns that are not easily seen by the other two methods.

Density estimates are most revealing for the shape of the distribution. Shape can be important, especially if the density plot reveals a marked lack of symmetry or strong evidence of multimodality since both of these can effect the sensitivity and specificity of statistical routines applied to the data. As mentioned above, neither of these two problems is easily detected using the other plots of the distributions. Not detecting these problems can mislead the analyst into thinking that they have carried out an appropriate analysis when, in fact, they have not.

When plotting distributions and densities it is generally worth considering whether a transformation of the data might yield a better scale on which to carry out the comparisons. Often the data are first visualized on the scale in which they were obtained, but in many cases any monotone transformation, such as the logarithm or a power transformation, is equally plausible. Such transformations can be applied without loss of information, but potentially with a strong effect on the shape and moments of distributions. This can lead to quite different perceptions of the data. An example

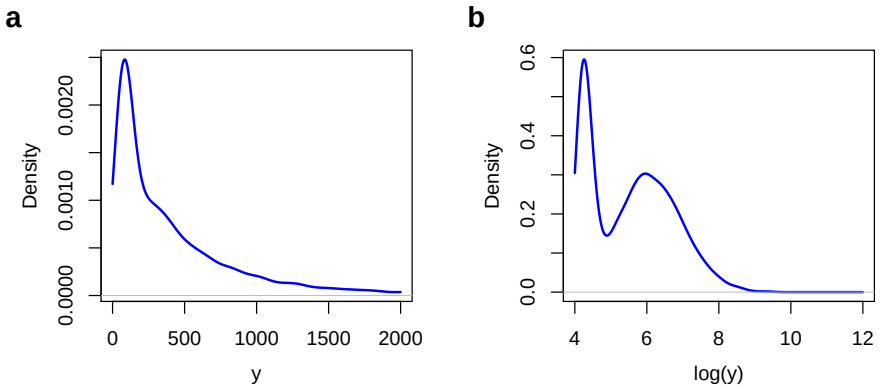

Figure 1: The impact of non-linear transformations on the shape of a distribution. The two panels show kernel density estimates of a random sample from a mixture of two log-normal distributions (a) and of the logarithmically transformed values (b).

is shown in Figure 1. If one does not know the underlying process and only sees the sample, the appropriate choice of scale may not be obvious and a prudent strategy is to examine different transformation scales.

3.2 Color

The use of color is an essential ingredient of many visualization methods. While the book containing this chapter is printed without color we provide on-line complements through the Bioconductor web site with color versions of the graphics.

We encourage the appropriate use of color as it can greatly help to make complex plots more readily interpretable. Unfortunately color can also be used to obfuscate the message. Red-green color schemes, so popular in bioinformatics, should be avoided since an appreciable proportion of the audience is red-green color blind and hence unable to read the plot. Lighter colors tend to make areas look larger than darker colors, thus colors of equal luminance should be chosen for graphics with large filled areas or where perception of area is important [17].

Following the results of Brewer, the *RColorBrewer* package offers three different types of color palettes; sequential, diverging and qualitative. These palettes have variations that provide between 3 and 9 colors and then the *colorRampPalette* function can be used to provide any number of intermedi-

ate color values, given a set of input colors. Sequential palettes are suitable for ordered values, that progress from low to high values. Diverging palettes put equal emphasis on mid-range values and extreme values. And qualitative palettes do not imply magnitude differences and hence are suitable for encoding unordered data such as race, or category.

There are also, often advantages to using well known palettes, such as those for topographic color schemes, or terrain color schemes, as readers will be familiar with them, and in some cases, their familiar encodings can help readers to comprehend the plots.

There are many uses for color in visualization. Color is good at showing connectivity or group membership. In many graphics, points from the same group will be colored the same. Color can also be used to highlight particular observations or samples. Most points or graphical objects are plotted in one color, and those to which particular attention should be devoted are plotted in another, distinct color. The heatmap in Figure 12 is a combination of a two-dimensional image (where the rows and columns identify objects) with a third continuous variable for each specific row-column pair. The third variable is encoded using color so these displays are often called *false color displays*. A color scheme capable of describing values on a common, ordered scale should be used. Groupings, of the variables that constitute either the rows or the columns of the image plot can be encoded using a colored bar.

While the space of light spectra is infinite-dimensional, the space of color perception in humans is three-dimensional [34]. There are different ways of parameterizing this space. Maybe the best known among computer programmers is the RGB coordinate system, which uses three values in $[0, 1]$. These coordinates reflect the design of current color displays, which use light sources in red, green, and blue primary colors, and they are hardware-dependent. There are many examples of visually unpleasant color schemes derived from extreme points in the RGB unit cube (Figure 18). The R function *hcl* uses three coordinates *hue* H, an angle in $[0, 360]$, *chroma* C, and *lightness* L as a value in $[0, 100]$. Allowable values for C depend on certain constraints, but generally are between 0 and 255. HCL is designed for area fills. By keeping chroma and luminescence coordinates constant and only varying hue, it is easy to produce color palettes that are harmonious, avoiding irradiation illusions that make light colored areas look bigger than dark ones (Figure 2). Our attention also tends to get drawn to loud colors and fixing the value of chroma makes the colors equally attractive to our eyes.

There are a number of ways of choosing colors from a color wheel. *Triads* are three colors chosen equally spaced around the color wheel; for example, $H = 0, 120, 240$ gives red, green, and blue. *Tetrads* are four equally spaced

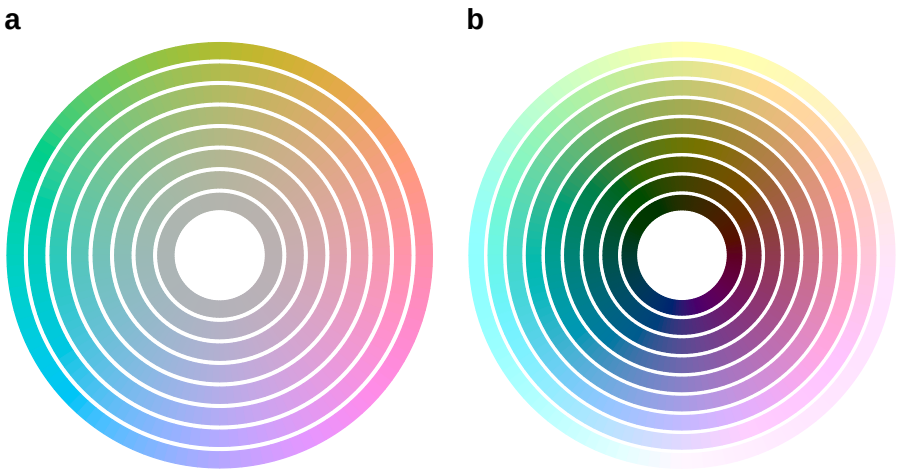

Figure 2: Circles in HCL colorspace. *a*: circles in HCL space at constant $L = 75$, with the angular coordinate H varying from 0 to 360 and the radial coordinate $C = 0, 10, \ldots, 60$. *b*: constant $C = 50$, and $L = 10, 20, \ldots, 90$.

colors around the color wheel, and some graphic artists describe the effect as "dynamic". *Warm colors* are a set of equally spaced colors close to yellow, *cool colors* a set of equally spaced colors close to blue. *Analogous color* sets contain colors from a small segment of the color wheel, for example, yellow, orange and red, or green, cyan and blue. *Complementary colors* are colors diametrically opposite each other on the color wheel. A tetrad is two pairs of complementaries. *Split complementaries* are three colors consisting of a pair of complementaries, with one partner split equally to each side, for example, $H = 60, 240 - 30, 240 + 30$. This is useful to emphasize the difference between a pair of similar categories and a third different one. A more thorough discussion is provided in the references [17, 25].

3.3 Two-dimensional layouts of data

When inspecting data from different genomic experiments, two-dimensional structures are often of great interest to us. For microarrays, we want to inspect the intensities in their original layout to look for systematic defects that could result from printing or hybridizing samples to the arrays, such as Figure 3 for Affymetrix arrays and Figure 4 for cDNA arrays.

For either 96 well or 384 well microtitre plates, we again have substantial interest in viewing the data in its original position. Many plates have

reagents handled using robotics with fluidics and integrated pipetting where handling problems often result in spatial patterns such as stripes or gradients in the two-dimensional view of the data. We have also noticed that one may encounter edge-effects, where the wells in the outside edge of the plate may behave differently (potentially drying out more rapidly, or heating or cooling more rapidly etc.).

Heatmaps have not been much used in statistics prior to their widespread use for viewing microarray data. A heatmap is a two dimensional false-color image of the data where the user has the option of rearranging either the columns or the rows (independently) so that similar rows (or similar columns) are adjacent. The reader is looking for rectangular regions of relatively constant intensity. These indicate a subset of the samples and the variables where there is relative homogeneity of signal which is distinct from that of other samples for the same genes.

In most cases hierarchical clustering is used to perform the rearrangement of rows and columns, and subsequently the resultant dendrograms are plotted on the sides. But there is no reason to prefer the reordering via hierarchical clustering over other methods for reorganizing the rows and columns such as those proposed in [19] which are implemented in the *gclus* package available from CRAN. Alternatively, good results have been reported by simply ordering each axis, independently, according to the values in the first principle component.

4 Visualization of experimental data

4.1 Spatial layout

In Figure 3 we see the false color image of an Affymetrix chip. In this case, there appear to be some minor problems since there are small cloudy patches in the lower right and left corners of the image. The images in Figure 4 show typical irregularities in the spatial distribution of the probe intensities from several two-color spotted cDNA microarrays. Again, the choice of appropriate scale is important particularly when using false color images.

4.1.1 Plate plots of microtitre plates

The top panel of Figure 5 shows a plate plot intended to resemble the geometrical structure of a 96 well microtitre plate that can be used to display arbitrary quantities of interest for individual wells. Most applications in cell

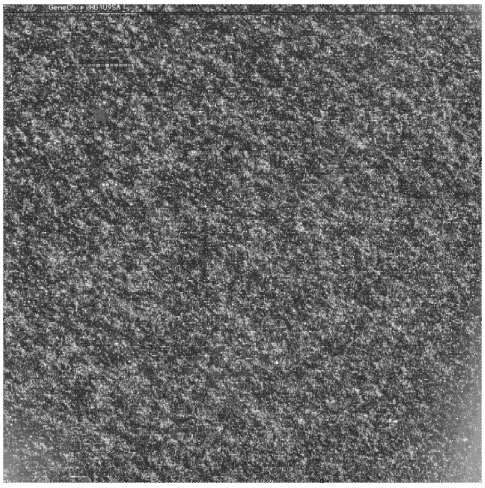

Figure 3: Grey-scale false color image of the probe fluorescence intensities measured from an Affymetrix chip. The dynamic range of the data is about 10^3, and a logarithmic transformation has been used in the mapping to the color scheme.

biology use microtitre plates, either for cell culture or during measurement. In this example we plot the number of cells per well as it was measured by flow cytometry. The consistently low number of cells around the edges of the plate indicates a handling problem during cultivation. For multifactorial data the wells on the plot can be subdivided into segments as shown in Figure 5 b. Each of the four segments represents one replicate experiment and the false colors indicate activation (red) and inhibition (blue) of cell proliferation. Regrettably, it is not possible to show this in black and white and readers are referred to the on-line complements for a colored version of this plot. While such data could just as easily be represented in a square data image we have found that the representation of wells as circles provides most biologist collaborators with an instant frame of reference and they are readily able to understand and interpret the data.

4.1.2 Affymetrix probe set intensities

On an Affymetrix array multiple probes, referred to as a probe set, are used to interrogate a particular mRNA. The number of probes in a probe set depends on the microarray being used. A rather interesting observation is the fact that there is a great deal of variation in these probes, given that they

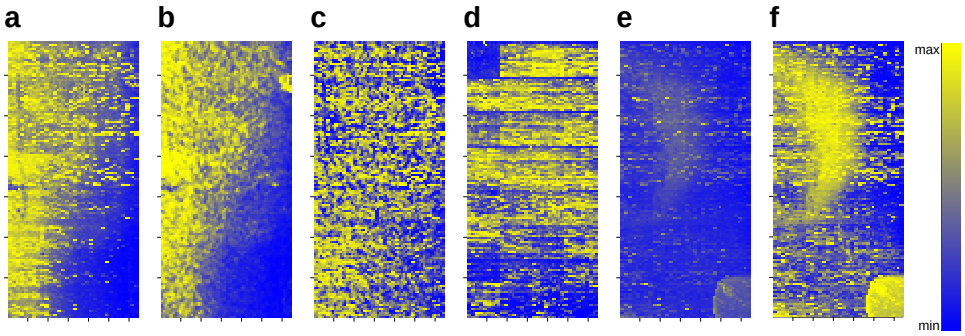

Figure 4: False color representations of the spatial intensity distributions of three different 64 × 136 spot cDNA microarrays from one experimental series [30]. The color scale is shown in the panel on the right. *a*: probe intensities in the red color channel, *b*: the corresponding local background estimates, *c*: the result of subtracting *b* from *a*. In *a* and *b*, there is an artifactual intensity gradient, which is mostly removed in *c*. For visualization, the color scale was chosen in each image to be proportional to the ranks of the intensities. *d*: for a second array, probe intensities in the green color channel. There is a rectangular region of low intensity in the top left corner, corresponding to one print pin. Apparently, there was a sporadic failure of the pin for this particular array. Panels *e* and *f* show the probe intensities in the green color channel from a third array. The color scale was chosen proportional to the logarithms of intensities in *e* and proportional to the ranks in *f*. Here, the latter provides better contrast. The bright blob in the lower right corner appears to be the result of being touched by a finger. Interestingly, it appears only in the green color channel, while the half moon shaped region in the upper left appears both in the green and red channels (data from red color channel not shown).

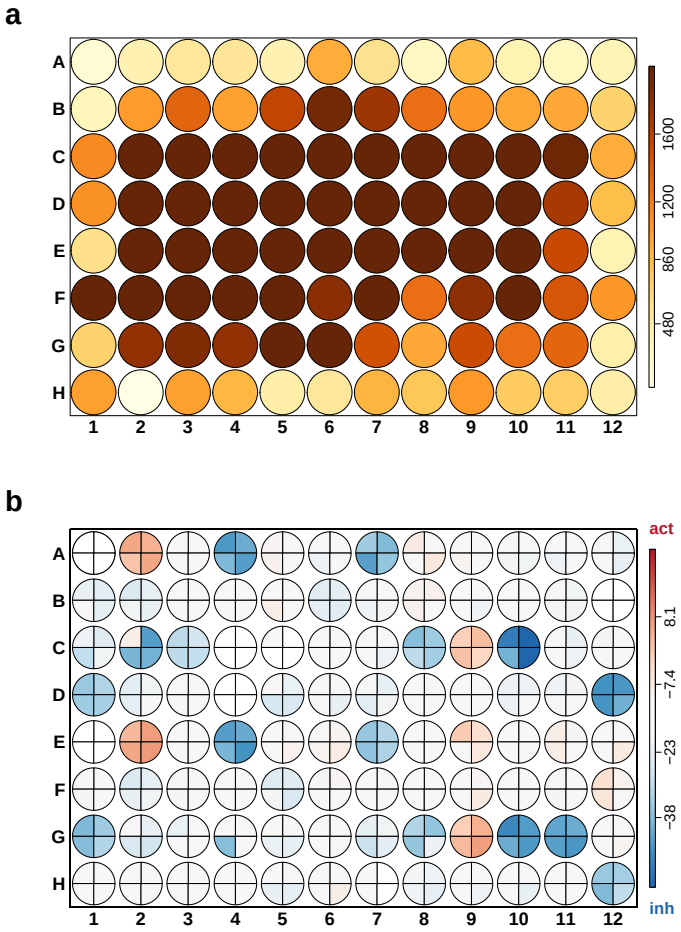

Figure 5: *a*: plate plot showing number of cells for each well of a 96 well microtitre plate. Cell numbers at the edges of the plate seem to be consistently low, indicating a handling problem. *b*: plate plot of analysis results from four replicate experiments probing for activation (red) and inhibition (blue) of cell proliferation (see on-line complements for a colored version). Reproducibility between experiments seems to be high.

are meant to be measuring the same quantity – the abundance of the target mRNA. In part the variation is due to the difference in GC content as will be demonstrated in Figure 7, but there are also additional sequence features, which are currently not well understood, that significantly contribute to the effect. Figure 6 a and b provides two plots of these data, one with the probes represented on the x-axis and the other with arrays represented on the x-axis. Since the data are essentially three dimensional it is easy to produce a false color image map, Figure 6 c.

In these various plots we can see how the values from the different probes vary in a correlated manner across arrays, however with different baselines. This observation was the starting point for the model-based analysis of Affymetrix probe set intensities that led to a dramatic improvement in the sensitivity of the technology [20, 24].

4.2 Distribution summaries

For a high-throughput experiment that measures a certain property across a large set of reagents, we expect the result to depend on the biological properties of the reagents, but not on the particular instance of experimental equipment in which the reagents were manipulated. For example, a cDNA microarray may be produced from a library of thousands of cDNA molecules, each specific for a certain gene, which are produced and stored in 384 well microtitre trays and deposited on the microarray using a print head with 16 pins. We expect that there should not be any reason for the distribution of resulting spot intensities to be different for the different print pins, or for the different trays. Hence, examining both boxplots and empirical distribution functions of spot intensities by print pin and tray is a commonly employed procedure for the quality assessment of spotted cDNA arrays. Industrially manufactured arrays such as those made by Affymetrix, are usually not subject to these sorts of problems, but other effects can be of interest. One example is the dependency of the measured intensities on the GC-content of the microarray probes, that is, the fraction of cytosines and guanines among the nucleotides that make up each 25-mer probe on the array (Figure 7). Cytosin and guanin are able to form three hydrogen bonds while adenine (A) and thymine (T) only form two. This leads to more stable hybridization bindings, and subsequently to higher intensities measured on the array, regardless of the abundance of target molecules.

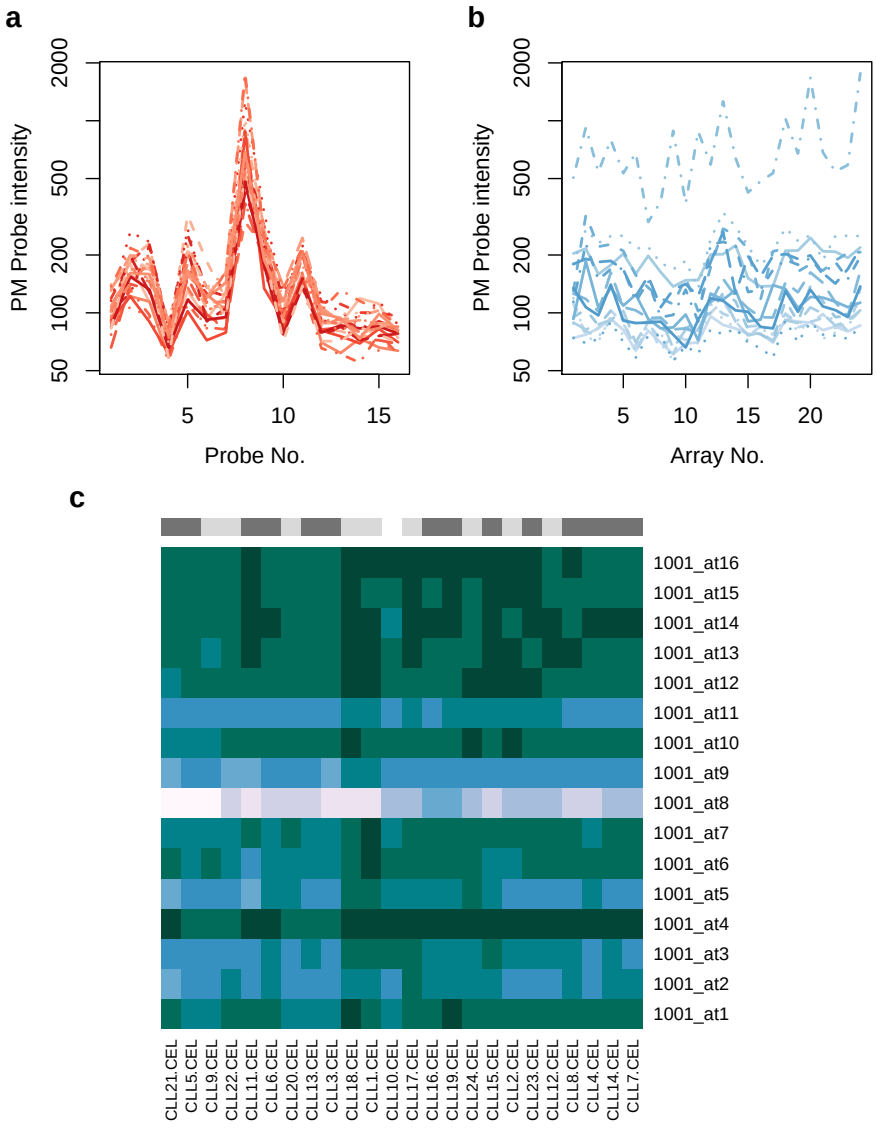

Figure 6: Plotted along the y–axis are probe intensities from the CLL data for the probe set `1001_at`. The probe set contains 16 probes and is specific for the receptor tyrosine kinase TIE. a: plotted along the x–axis are the 16 probes, and each line corresponds to one of the 24 microarrays in the dataset. b: plotted along the x–axis are the arrays, each line corresponds to a different probe. c: false color image map of the intensities. Rows: the 16 probes in the dataset. Columns: the 24 microarrays. An additional categorical covariate for the microarrays, the disease type, is indicated by the grey-scale color bar on top.

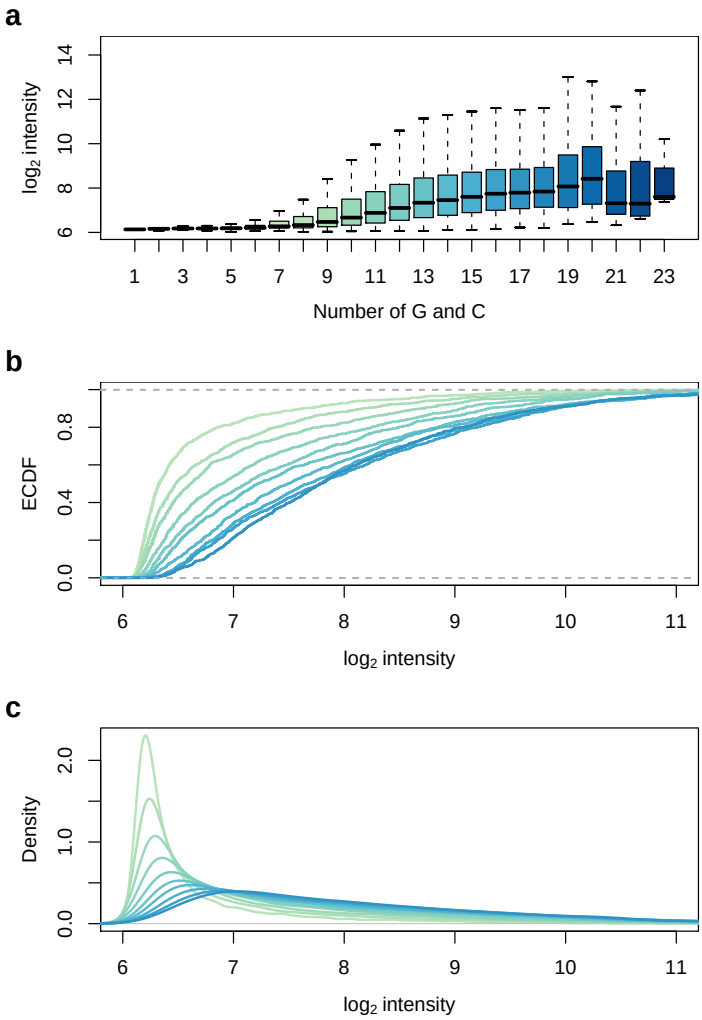

Figure 7: Distributions of the $\log_2$–intensities from the CLL dataset (see Section 2) grouped by the number of cytosines (C) and guanines (G) among the 25 nucleotides in each probe. Box plots (a), empirical cumulative distribution functions (ECDF, b) and kernel density estimates (c). Box and line colors in the three panels correspond to the same groups. The top plot shows all groups from 1 to 23, the middle and lower plots show the ten most populated groups from 8 to 17. Cytosine and guanine are able to form three hydrogen bonds, while adenine (A) and thymine (T) only form two, hence oligonucleotides with a higher proportion of C and G can form more stable hybridization bindings. The plots show how this results in higher intensities measured on the array, regardless of the abundance of target molecules.

4.3 Scatterplots and 2-dimensional density plots

Scatterplots are a powerful tool for the visualization of data from two variables given the number of observations is relatively small. For scatterplots of high observation density it is often difficult to get a good impression of the distribution underlying the data because the mass of points forms a featureless dark area (Figure 8 a). This problem can be addressed by dividing the data into a two-dimensional histogram of hexagonal bins [7] and plotting these bins using false-color coding (Figure 8 b). An alternative approach is provided by plotting the local densities either as false-color images (Figure 8 c) or each observation individually coloring the points with respect to their density (Figure 8d).

4.4 Clustering

Clustering, sometimes also referred to as unsupervised machine learning, is a widely used method for finding groups in data. For a clustering problem the setting is generally that there is some number of objects, and for each of those objects a set of variables have been measured. The notion of clustering is to group together objects that are similar to each other, for the variables that were measured (or the subset of those variables that are deemed to be important). To carry out such an operation the user must carefully select a distance measure that will be used to combine the different variables. In many cases the variables were measured on different scales, possibly in different units, and this needs to be considered. For many genomic data sets many more variables are measured than can easily, or sensibly, be used. Some objects group on one set of variables, while other samples group on other sets of variables and hence, one might say that the problem is itself ill-posed.

Perhaps the most widely used clustering algorithm is agglomerative hierarchical clustering. This algorithm is easy to implement, and somewhat easy to interpret, but the resulting dendrogram is less easy to deal with. It is essential to note that given any data set, regardless of the existence of real clusters in the data one can perform hierarchical clustering and often the resulting dendrogram will appear to indicate that there are groups in the data. Dendrograms are not visualization methods. Visualization is the process of revealing structure in data, hierarchical clustering (and the resulting dendrogram) impose structure and hence the result is often of little practical use.

The silhouette plot was proposed in [22] and is easily applied to the

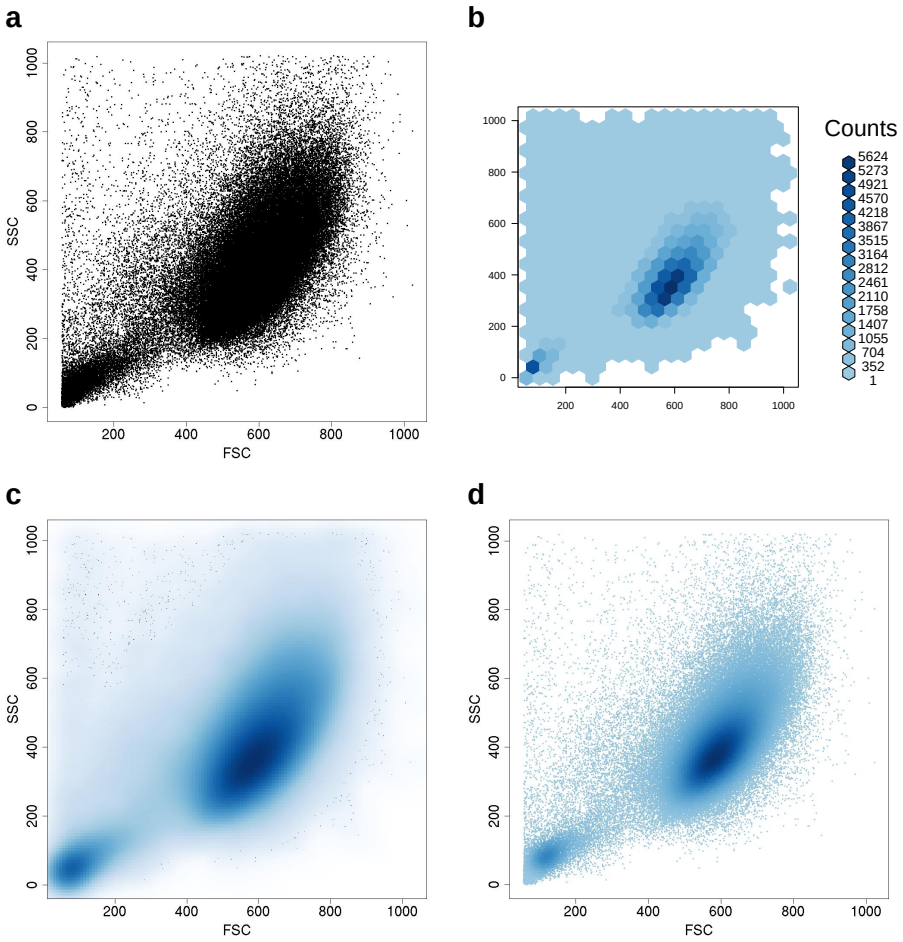

Figure 8: Four different visualizations of a scatterplot of flow cytometry data (forward light scatter versus side light scatter). *a*: the usual scatterplot. Because of the large number of points, it is a rather featureless black blot. *b*: the result of a hexagon binning procedure. The color code at each hexagon represents the number of data points that it contains. *c*: color representation of smooth point densities calculated from the data using a kernel density estimator. In the sparse regions of the density, the plot is augmented by black dots that represent individual data points. In the denser regions, these are omitted. *d*: usual scatterplot with points colored according to the local density.

output of any clustering algorithm. For each observation a new quantity is computed. That quantity is related to the relative difference between the average distance between the observation and all other members of the cluster it is assigned to, and the average distance between the observation and the members of another cluster (for each observation the *other cluster* is the cluster with the smallest average distance that the observation was not assigned to). This quantity is then displayed using horizontal bars, grouped by cluster membership, as seen in Figure 9. For any observation the bar is positive if the point is closer to the cluster it was assigned to and the bar is negative if the observation is closer to some other cluster. Note that this ensures that comparisons between observations are carried out on a common scale, in accordance with good visualization principles.

Using the CLL data and the genes selected to discern between stable and progressive disease the samples were clustered using the *pam* function in the *cluster* package. In Figure 9 we present a silhouette plot of the clustering information. In this plot, each sample is represented by a horizontal bar. The length (and direction) of the bar is related to the average distance from the sample and the other members of the group it is in, versus the smallest average distance from the sample, to all members of another cluster. Thus, bars that are large (close to 1) indicate observations that are well clustered, while bars that are negative (eg. CLL8) indicate observations that may be in the wrong cluster.

There is interest in clustering itself and methods that are referred to as *biclustering*, a procedure originally proposed in [16] for simultaneously clustering samples and variables to find interesting subgroups. Biclustering turns out to be a difficult problem to solve in general; some heuristics for addressing it have been discussed in Section 3.3.

Two dimensional projections of data are useful since they can be easily communicated on paper or screen, and we find it much easier to interpret than higher dimensional representations. Often used tools are principal component reductions and multi-dimensional scaling (MDS). For an in depth discussion of MDS and its many variations we recommend consulting [11].

Here we compare the two dimensional views that can be obtained by applying classical MDS to the CLL data. In Figure 10 two different two dimensional MDS reductions are shown. They are remarkably different. The left panel clearly indicates two groups, but this is most likely due to the fact that it is based on genes that were selected to emphasize the difference between the two disease groups. The right panel, was constructed using genes that showed high variability across arrays rather than those that can separate the two disease groups. In the right panel we do not see evidence for

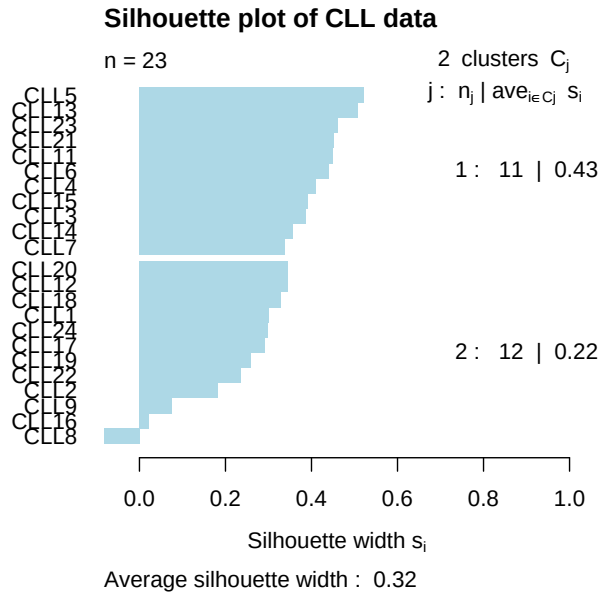

Figure 9: A silhouette plot based on clustering the CLL samples into two groups based on genes selected to differentiate those with stable disease from those with progressive disease.

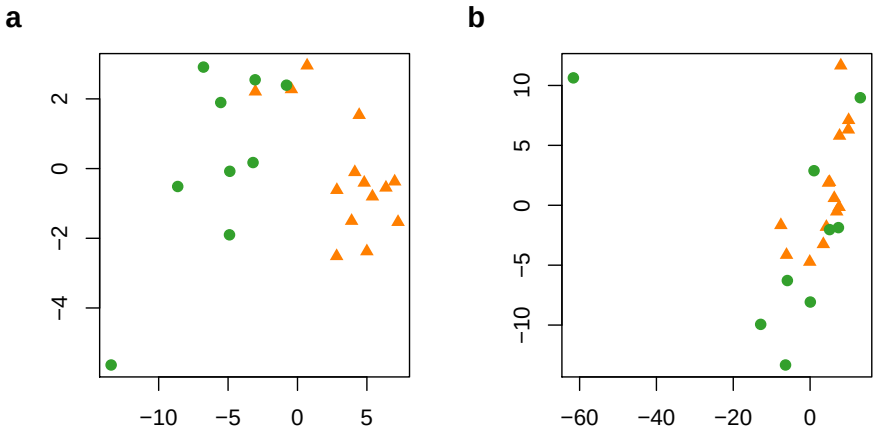

Figure 10: Two different MDS views of the arrays in the CLL dataset. *a*: feature selection that prefers genes that separate the two disease groups. *b*: features selected on the basis of their variability.

two groups, but rather that there is one peculiar microarray and some effort should be expended to identify that array and the causes of the behavior shown in that panel.

Unfortunately, dimension reduction methods such as MDS and principal components are often misused. It is always possible to compute and visualize the first two, or three components but these need not be meaningful in any way. There are two common problems that can arise. One being that the first two components do not describe the bulk of the variability in the data and that in reality one needs more components to provide a reasonable description of the data. It is prudent and important to compute, and report, the proportion of the variability explained by the components that are visually displayed. A second major problem is that the manner in which the data were collected, or assembled, and processed has a profound effect on the variability in the data and hence on visualizations of it. As we demonstrated in Figure 10 the left panel indicates two groups, while the right panel does not.

We also emphasize that some effort should be expended to determine the number of dimensions needed to explain the data. In Figure 11 we compare the MDS goodness of fit statistics for different numbers of dimensions, for both those genes selected to be to differentially expressed between the disease groups and those genes selected because they show large overall variation across samples. We see that low-dimensional embedding generally provide a somewhat better fit for the disease selected genes, but there is no clear break, in either graph, between two and three dimensions. Thus, there is little evidence that a two dimensional projection is particularly adequate for either data set.

4.5 Heatmaps

In Figure 12 we see a heatmap showing the expression estimates of genes selected according to the p-value in a t-test comparing those with stable disease to those with progressive disease. The top bar contains one small block for each sample, those samples that correspond to the stable disease phenotype have a dark colored block, those with progressive disease a light colored block. It is not surprising that the two groups are well separated (all the dark blocks are to the right), since that is how genes were selected.

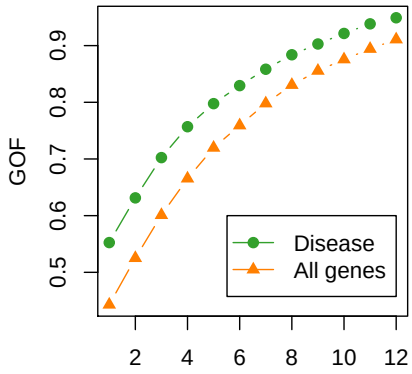

Figure 11: Goodness of fit statistic from MDS, for different numbers of components, comparing the MDS fit for genes selected for disease status to the GOF for those selected for variability across samples.

4.6 Diagnostics

The RNA degradation plot, Figure 13, indicates that there is in fact one unusual array, CLL1. These plots show a relationship between probe position and intensity. In general, the relationship is non-constant, and increasing in the 5' to 3' direction. The shape of the curve is generally not of interest, but rather whether there are different curves for different samples, since if this occurs it is likely that the computed expression values will not be directly comparable. In this case, one might be tempted to think that perhaps the handling of the RNA for the first array was potentially different from the handling for other arrays and hence that the exclusion of this array from subsequent analyses might be a good approach.

The Bioconductor package *affyPLM* provides a number of more detailed computational methods for performing diagnostics on Affymetrix microarrays. It fits a model to the dataset and provides methods for plotting false color images of the fitted parameters, the residuals, and a number of quality metrics. In Figure 14, we present the images for array `CLL6`. More striking examples can be found in Chapter 3 of [14].

Other quality assessments can be made, and often they lead us back to the sorts of problems mentioned in Section 3.1 of plotting distributions. For example, the relative log expression (RLE) values are computed for each probeset on an array, as the log of the expression value on that array to the median expression value for that probeset across all arrays [14]. Assuming

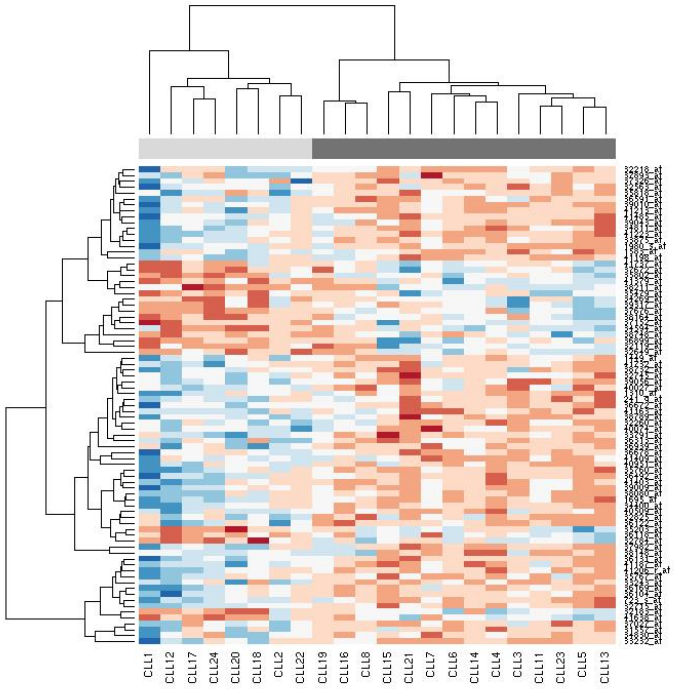

Figure 12: A heatmap representation of the CLL data. Genes were selected using a t-test comparing those with a progressive disease and those with a stable disease. Colors indicate low (blue) and high (red) values of the expression estimates.

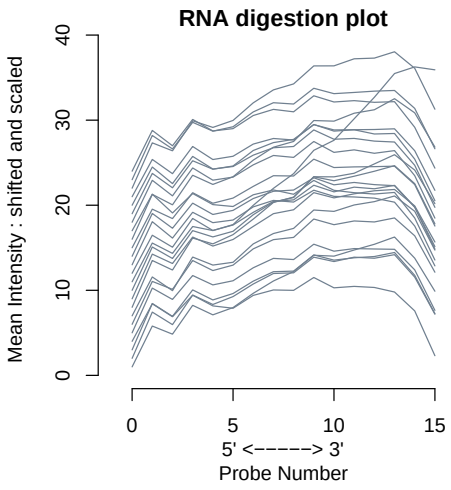

Figure 13: RNA degradation plot for the 24 CLL microarrays.

that most genes are not changing in expression across arrays means most RLE values will be near 0. Comparing per array boxplots indicates arrays where there is potentially a problem. For the CLL data RLE is plotted in the left frame of Figure 15, where only the array labeled 11 seems odd. Another quality measure is the normalized unscaled standard errors (NUSE). To compute the NUSE the standard error estimates obtained for each gene on each array are standardized across arrays so that the median standard error for each gene is one across all arrays [14]. Any array with elevated NUSE relative to the other arrays is often of lower quality. Again, we have a distribution of numbers, per array and boxplots can be used to display these. In Figure 15 we plot the NUSE values for the CLL arrays. We see that there are two such arrays, the one numbered 1 (CLL10) and the one numbered 11 (CLL19).

5 Plotting in genomic coordinates

Many biopolymers have a linear structure, and many (although certainly not all) of their properties can be viewed and understood as features arranged along a linear coordinate. In this section, we will discuss two uses of visualization along the sequence of nucleotides in a chromosome. In Section 5.1, we map transcript abundance to chromosomal location. A reasonable assumption is that the true abundance should be either piecewise continuous

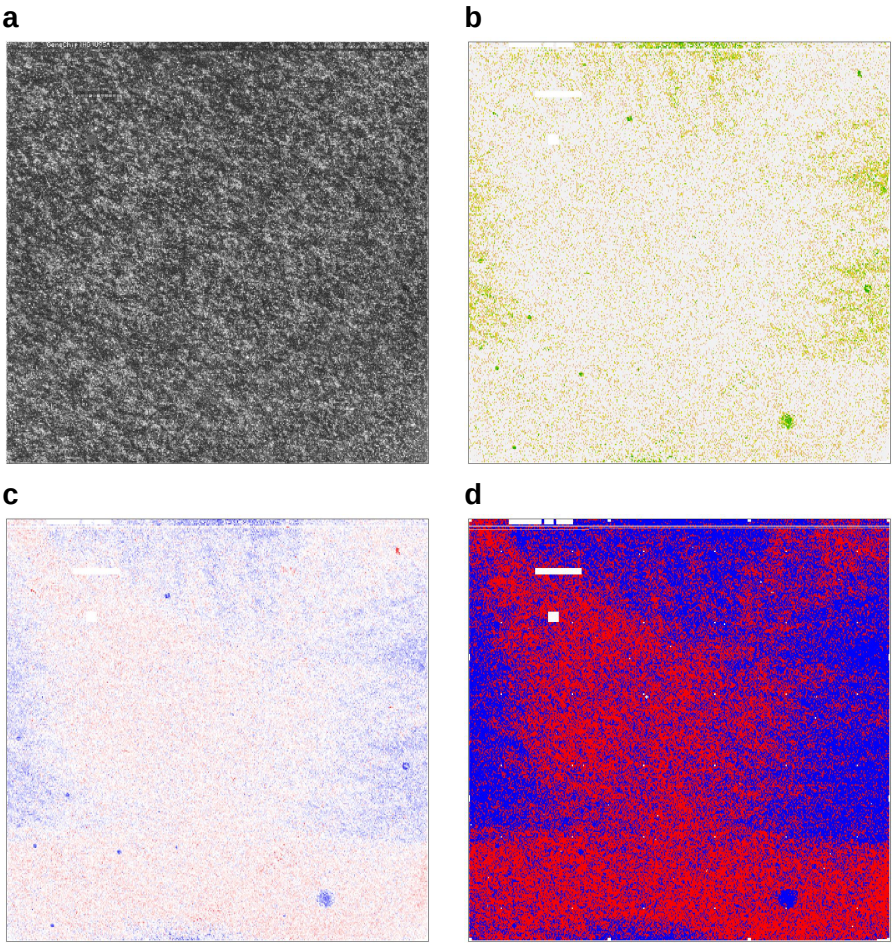

Figure 14: Image plots for Array CLL 6. *a*: raw data. *b*: weights used in fitting the model, green values correspond to low weights, light values to high weights. *c*: residuals *d*: signs of the residuals; red corresponds to positive residuals, blue to negative residuals (see on-line complements for a colored version).

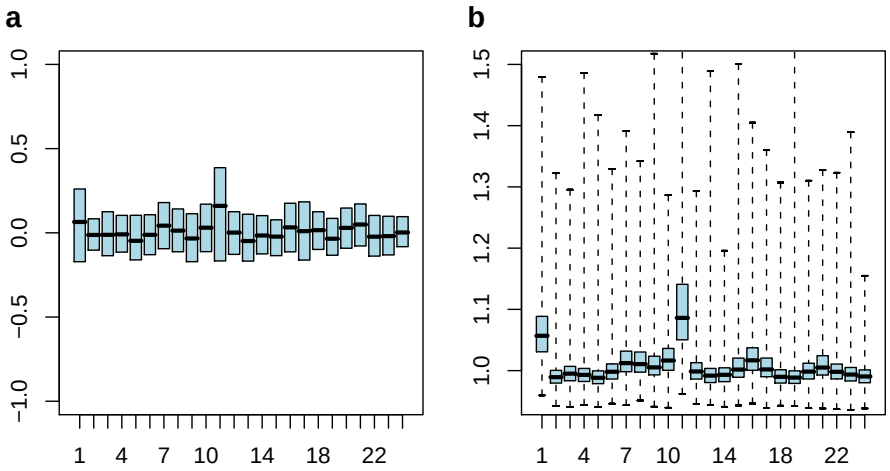

Figure 15: Interquartile ranges of RLE (a) and box-and-whiskers plots of NUSE values (b) for the CLL data.

or piecewise constant over loci and noise can be reduced by applying smoothing procedures that make use of this assumption. Furthermore, ordering by locus can provide a more appropriate context for interpretation of the data. In Section 5.2, we look at the integration and presentation of large amounts of public sequence data and metadata in so-called genome browsers.

5.1 Along-chromosome plots of high-density tiling array data

Along-chromosome plots of microarray data measuring the abundance of RNA transcripts are shown in Figures 16 and 17. Figure 16 shows the data from one condition where the data are essentially two-dimensional: at each genomic coordinate, an intensity measurement quantifies the amount of target molecules transcribed from that site through a real number. Figure 17 is a generalization to situations where we have an additional dimension, in this case, time. The two spatial dimensions of the plot area are now used for genomic coordinate and time, respectively, while the intensity is coded through a false color scheme. The plots were produced with the *plotAlongChrom* function in the *tilingArray* package.

.

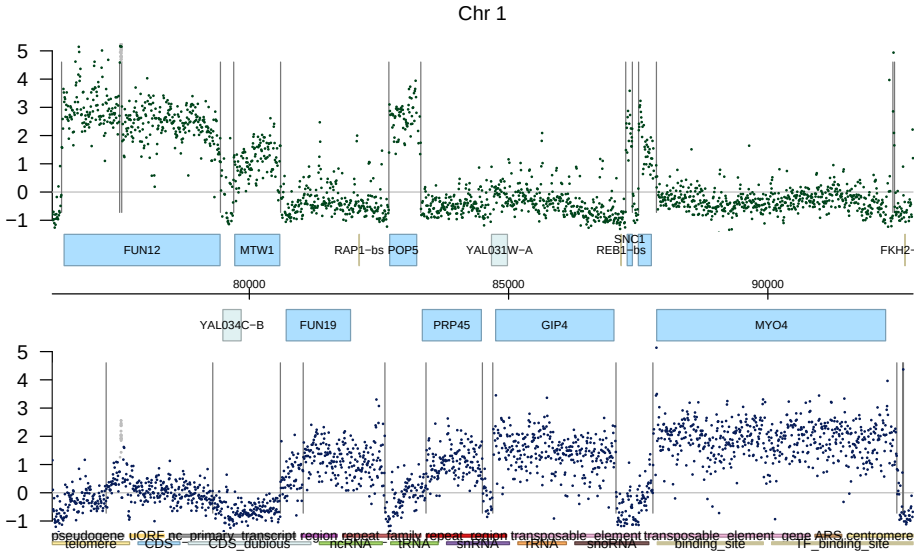

Figure 16: Visualization of hybridization intensities along 16 kB of yeast chromosome 1, measured on a high-density DNA tiling array. The array covers the genome with 25mer probes along both strands of each chromosome in steps of 8 bases. The y-axis shows the generalized logarithm [18] of the background-corrected and scaled hybridization intensities, the x-axis corresponds to the genomic coordinates (in bp) of the probes. Upper and lower scatter plots correspond to Watson and Crick strands, respectively. Annotated features are shown as boxes for each strand. Vertical lines are segment boundaries estimated through a change point detection algorithm. The background threshold is shown as a horizontal line. The abbreviation CDS in the legend refers to coding sequence; uORF, upstream open reading frame; ncRNA, non-coding RNA; TF, transcription factor.

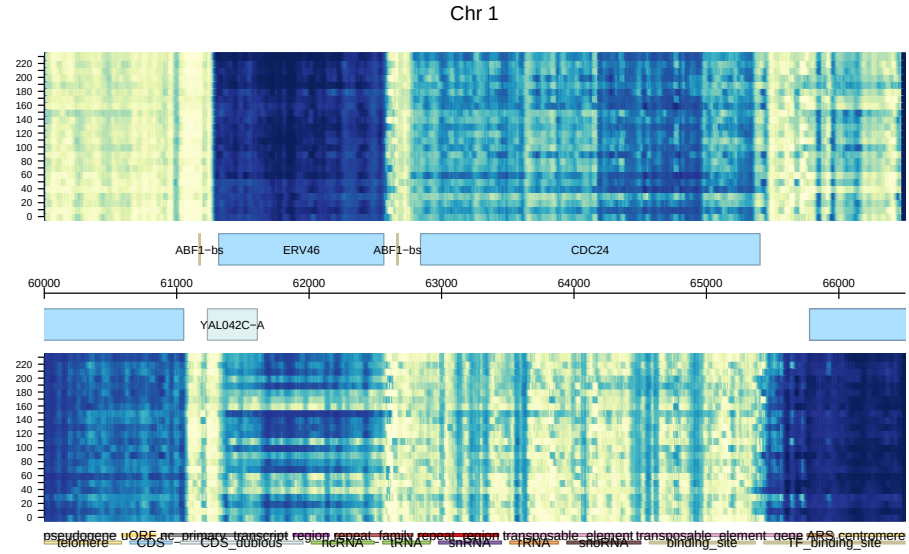

Figure 17: Visualization of hybridization intensities along 6.5 kB of yeast chromosome 1 for 24 time points $t = 0$ min, 10 min,..., 230 min. This time span corresponds to about two full cell cycles. The genomic location is plotted along the x–axis, analogous to Figure 16, time along the y–axis, and hybridization intensity is represented by a false-color scheme in which dark colors represent high values. Due to the limited dynamic range of such color scales, the choice of mapping between data values and colors is crucial. Here, intensities were rank transformed and then mapped to an interpolation between the 9 colors of the *YlGnBu* palette of the *RcolorBrewer* package [4, 5]

5.2 Genome browsers

There are a number of publicly funded genome annotation projects that are trying to integrate, annotate, and interpret the genome and transcriptome sequence data and related information that have been obtained through world-wide efforts over the recent years. So-called genome browsers are provided that can navigate and visualize these data and metadata. They are freely available and can be accessed through the world-wide web using a generic web browser [3, 23, 35].

An example from Ensembl is shown in Figure 18. Again, we recommend inspecting a colored version of the figure from the online-complements since genome-browsers make extended use of color coding and most of the features will not be clear in black and white. The size of features with which the genomic sequence is annotated ranges from Megabases down to individual bases, and the hierarchy of size ranges is used to organize the navigation and information display. Figure 18a shows the cytogenetic bands of human chromosome 3, which has a length of 2×10^8 basepairs. A region near the telomere of the p-arm is marked by a box. An overview over this region of about 10^6 basepairs is shown in Figure 18b, which displays regions of synteny with other animals and the loci of genes. A region in the middle around the VHL gene is marked by a box. Figure 18c spans about 10^4 basepairs and shows the current experimental evidence for the structure of the VHL gene (introns, exons) and its products (transcripts and proteins). A region in the middle is marked. It leads to a fourth panel (not shown), which goes down to the resolution of individual basepairs and shows the encoded protein sequence and restriction enzyme sites.

The distributed annotation system (DAS) [12] allows users to add additional annotation tracks along the genomic coordinates to such displays without going through the effort of installing and configuring their own server. These tracks can be obtained from other public databases, allowing configuration of the display in ways the maintainers of the genome browser did not consider, or they can be derived from a user's own data, facilitating viewing them in the context of genome annotation, as in Figures 16 and 17.

6 Graphs

A graph is a set of *nodes* and a set of *edges*, which we shall denote $G = (V, E)$. The term *vertices* is often used interchangeably with nodes. Our treatment is incomplete, more details on graph theory can be found in [15] while details on capabilities in R and Bioconductor can be found in [14]. Graphs can be

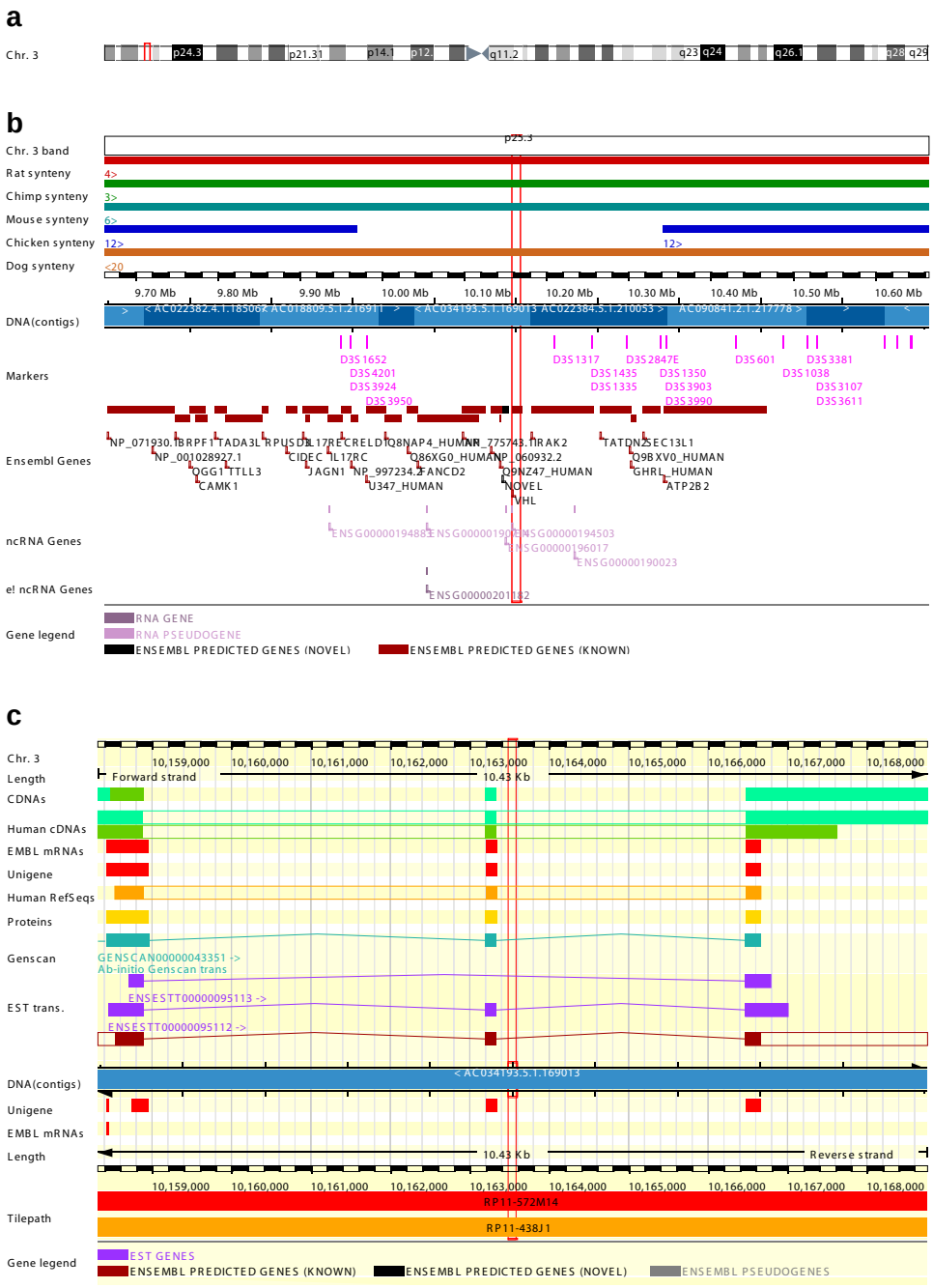

Figure 18: Genome browser visualization of genomic annotation at three different levels of resolution. The plots are taken from the Ensemble genome browser (http://www.ensembl.org). See main text for details.

used to represent binary relationships between the entities represented by the nodes, where the different relationships could be encoded using different types of edges. For genomic data, and especially for recent work in systems biology there is a growing interest in tools for representing, rendering and computing various statistics or quantities on graphs.

The Bioconductor packages, *graph*, *RBGL* and *Rgraphviz*, provide software tools for representing graphs, for graph algorithms and for graph layout, respectively. The tools in *RBGL* largely rely on the Boost Graph Library [27], and those in *Rgraphviz* rely largely on Graphviz [13].

There are many different specializations of graphs, for example a directed graph is a graph where all of the edges have a direction, and directed acyclic graphs (referred to as DAG) arise in different settings. The Gene Ontology (GO), [31] provides a structured vocabulary, or ontology, for genes and gene products in the form of a DAG.

Other specialized graphs of some interest in biology are bipartite graphs. A graph is said to be a bipartite graph if the nodes can be divided into two disjoint sets where all edges are between members of one set and members of the other and there are no within-set edges. Bipartite graphs arise in many contexts such as co-citation, where we would like to study genes that are cited in the same paper (here one set of nodes are the genes, the other set the papers and hence the conditions for a bipartite graph are trivially true). A second, similar example is the bipartite graph that arises when studying the association between genes and pathways. While there is no specific definition of a pathway, it is generally taken to be a set of genes that operate in some coordinated fashion to achieve a major biological objective. Things such as apoptosis, integrin mediated cell adhesion etc. are pathways, and KEGG [21] is one source of pathway data.

6.1 The different graph layout engines in graphviz

In Figure 19 the same graph is laid out using three different graph layout algorithms. The views are remarkably different, and the important message, is that the algorithm chosen matters. In some sense graph layout algorithms are not really visualization methods. Most layout algorithms are based on some optimization problem, minimizing edge length, minimizing edge crossings, a spring model and so on, rather than on enhancing a particular visual impression. But it is also the case that a well laid out graph, such as some of the pathway graphs in KEGG is incredibly informative. These graphics are no different from any other plot and the effort expended in selecting appropriate values for the many parameters will be repaid in terms of effective

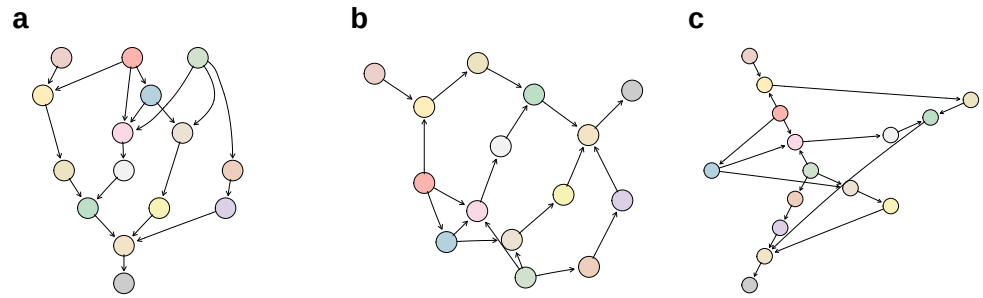

Figure 19: Three different graph layout algorithms applied to the same directed graph. *a*: **dot:** aims at visualizing the hierarchies in a directed graph. *b*: **neato:** tries to arrange the nodes in a way that, as much as possible, the edges do not overlap and have the same length. *c* **twopi:** aims at visualizing the radial structure of the graph.

visualization.

Figure 20 shows a plot of a portion of one of the ontologies from GO. GO defines three ontologies that describe genes and gene products; the molecular function ontology, the biological process ontology and the cellular component ontology. Each ontology is a set of terms, that are related to each other and are represented as a directed acyclic graph (or DAG), with a root node. In Figure 20 each node represents a specific term, and edges go from more specific terms to related less specific terms. Genes are mapped to the different terms in the ontology by a different initiative, named GOA, [6]. In this graphic we show the subgraph of the molecular function ontology that is induced by a set of genes selected to distinguish between the stable and progressive disease in the CLL sample. The filtering of genes was as described in Section 2 except that we reduced the p-value criterion to 0.0001 in order to produce a plot that would be readable.

6.2 Bipartite graphs

Using the same subset of the genes as for Figure 20 we found all papers from PubMed that refer to those genes and created a bipartite graph. The resultant graph is shown in Figure 21. It can be seen that one paper cites all genes, while three are cited together in one other paper. Starting from this, an investigator can follow up by reading these papers and determine if any relationships are revealed in the data.

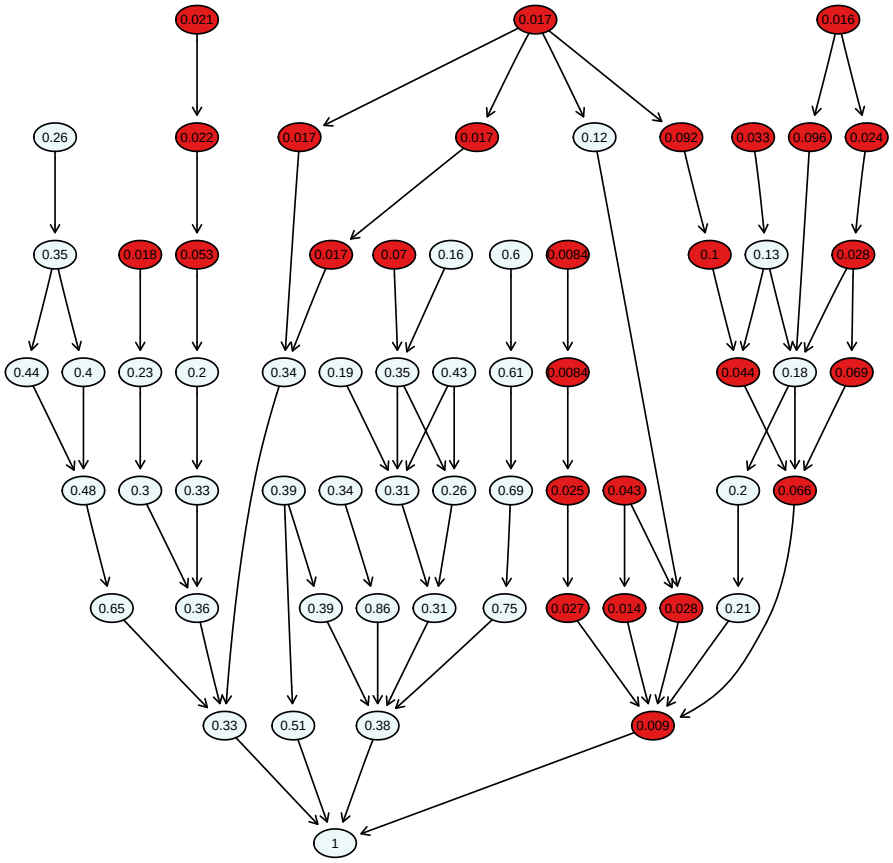

Figure 20: The induced GO graph colored according to unadjusted Hypergeometric p-values, whose values are given in the nodes. Nodes where the p-values are less than 0.1 are colored.

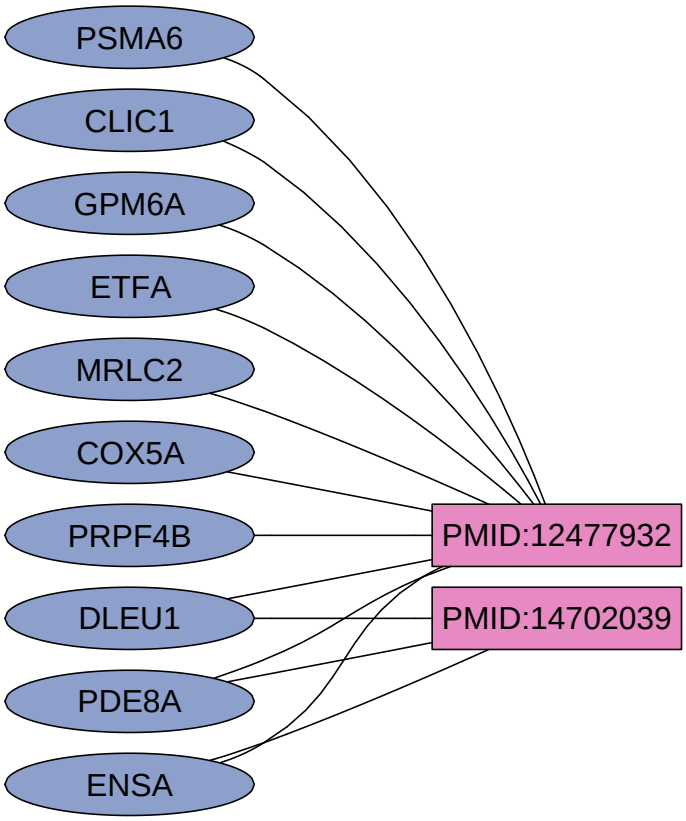

Figure 21: The bipartite graph linking genes to papers, via PubMed.

7 Acknowledgements

We would like to thank Dorit Arlt, Sabina Chiaretti, Sandra Clauder-Münster, Lior David, Jerome Ritz, Annemarie Poustka, Lars Steinmetz, Holger Sültmann, and Stefan Wiemann for allowing us the use of their data. Ross Ihaka provided expert advice on the use of colors. Ting-Yuan Liu provided help in producing some of the figures used in this chapter.

References

[1] B. Alberts, D. Bray, J. Lewis, M. Raff, K. Roberts, and J. D. Watson. *Molecular Biology of the Cell*. Garland Science, 2002. 2

[2] D. Arlt, W. Huber, et al. Functional profiling: from microarrays via cell-based assays to novel tumor relevant modulators of the cell cycle. *Cancer Res*, 65(17):7733–7742, 2005. 3

[3] E. Birney, D. Andrews, M. Caccamo, et al. Ensembl 2006. *Nucleic Acids Res*, 34(Database issue):556–561, Jan 2006. 28

[4] C. A. Brewer. Color use guidelines for mapping and visualization. In A. MacEachren and D. Taylor, editors, *Visualization in Modern Cartography*. Elsevier Science, Tarrytown, NY, 1994. 27

[5] C. A. Brewer. Guidelines for use of the perceptual dimensions of color for mapping and visualization. In J. Bares, editor, *Color Hard Copy and Graphic Arts III*, volume 2171, pages 54–63. Proceedings of the International Society for Optical Engineering (SPIE), Bellingham, 1994. 27

[6] E. Camon, M. Magrane, D. Barrell, et al. The Gene Ontology annotation (GOA) database: sharing knowledge in Uniprot with Gene Ontology. *Nucleic Acids Research*, 32:D262–D266, 2004. 31

[7] D. B. Carr, R. Littlefield, W. Nicholson, et al. Scatterplot matrix techniques for large N. *Journal of the American Statistical Association*, 83:424–436, 1987. 16

[8] Chipping forecast. *The chipping forecast. Special supplement to Nature Genetics*, volume 21, 1999. 2

[9] W. S. Cleveland. *Visualizing Data*. Hobart Press, Summit, New Jersey, 1993. 2, 4

[10] W. S. Cleveland. *The Elements of Graphing Data (Revised)*. Hobart Press, Summit, New Jersey, 1994. 2, 4

[11] T. Cox and M. Cox. *Multidimensional Scaling*. Chapman & Hall CRC, 2001. 18

[12] R. D. Dowell, R. M. Jokerst, A. Day, S. R. Eddy, and L. Stein. The distributed annotation system. *BMC Bioinformatics*, 2:7, 2001. 28

[13] E. R. Gansner and S. C. North. An open graph visualization system and its applications to software engineering. *Software Practice and Experience*, 30:1203–1233, 1999. 30

[14] R. Gentleman, V. Carey, W. Huber, R. Irizarry, and S. Dudoit, editors. *Bioinformatics and Computational Biology Solutions Using R and Bioconductor*. Springer, 2005. 2, 21, 23, 28

[15] J. L. Gross and J. Yellen. *Graph theory and its applications*. CRC Press, Boca Raton, 1998. 28

[16] J. A. Hartigan. Direct clustering of a data matrix. *Journal of the American Statistical Association*, 67:123–129, 1972. 18

[17] K. Hornik and F. Leisch, editors. *Color for Presentation Graphics*, Vienna, Austria, 2003. 6, 8

[18] W. Huber, A. von Heydebreck, H. Sültmann, A. Poustka, and M. Vingron. Variance stabilization applied to microarray data calibration and to the quantification of differential expression. *Bioinformatics*, 18 Suppl. 1:S96–S104, 2002. 26

[19] C. Hurley. Clustering graphics. *Journal of Computational and Graphical Statistics*, To appear. 9

[20] R. A. Irizarry, B. M. Bolstad, F. Collin, L. M. Cope, B. Hobbs, and T. P. Speed. Summaries of affymetrix genechip probe level data. *Nucleic Acids Res*, 31(4):e15, Feb 2003. 13

[21] M. Kanehisa and S. Goto. KEGG: Kyoto encyclopedia of genes and genomes. *Nucleic Acids Research*, 28:27–30, 2000. 30

[22] L. Kaufman and P. J. Rousseeuw. *Finding Groups in Data*. Wiley, 1990. 16

[23] W. J. Kent, C. W. Sugnet, T. S. Furey, et al. The human genome browser at UCSC. *Genome Res*, 12(6):996–1006, Jun 2002. 28

[24] C. Li and W. H. Wong. Model-based analysis of oligonucleotide arrays: expression index computation and outlier detection. *Proc Natl Acad Sci U S A*, 98(1):31–36, Jan 2001. 13

[25] J. Mollon. Seeing colour. In T. Lamb and J. Bourriau, editors, *Colour: Art and Science*. Cambridge Unversity Press, 1995. 8

[26] P. Murrell. *R Graphics*. Chapman & Hall/CRC, New York, 2005. 2

[27] J. G. Siek, L.-Q. Lee, and A. Lumsdaine. *The Boost Graph Library*. Addison Wesley, Boston, 2002. 30

[28] T. P. Speed. *Statistical Analysis of Gene Expression Microarray Data*. Chapman & Hall/CRC, 2003. 2

[29] T. Strachan and A. Read. *Human Molecular Genetics*. Garland Science/Taylor & Francis Group, 3 edition, 2003. 2

[30] H. Sultmann, A. von Heydebreck, W. Huber, et al. Gene expression in kidney cancer is associated with cytogenetic abnormalities, metastasis formation, and patient survival. *Clin Cancer Res*, 11(2 Pt 1):646–655, Jan 2005. 11

[31] The Gene Ontology Consortium. Gene Ontology: tool for the unification of biology. *Nature Genetics*, 25:25–29, 2000. 30

[32] E. Tufte. *Envisioning Information (2e)*. Graphics Press, Cheshire, 1990. 2

[33] E. Tufte. *The Visual Display of Quantitative Information (2e)*. Graphics Press, Cheshire, 2001. 2

[34] H. von Helmholtz. *Handbuch der Physiologischen Optik*. Leopold Voss, 1867. 7

[35] D. L. Wheeler, T. Barrett, D. A. Benson, et al. Database resources of the National Center for Biotechnology Information. *Nucleic Acids Res*, 33(Database issue):39–45, Jan 2005. 28

[36] Z. Wu, R. Irizarry, R. Gentleman, F. Martinez Murillo, and F. Spencer. A model based background adjustement for oligonucleotide expression arrays. *Journal of the American Statistical Association*, in press, 2005. 3