Matrix Visualization

III.15

Han-Ming Wu, ShengLi Tzeng, Chun-Houh Chen

15.1 Introduction

The graphical exploration of quantitative/qualitative data is an initial but essential step in modern statistical data analysis. Matrix visualization (Chen, 2002; Chen et al., 2004) is a graphical technique that can simultaneously explore the associations between thousands of subjects, variables, and their interactions, without needing to first reduce the dimensions of the data. Matrix visualization involves permuting the rows and columns of the raw data matrix using suitable seriation (reordering) algorithms, together with the corresponding proximity matrices. The permuted raw data matrix and two proximity matrices are then displayed as matrix maps via suitable color spectra, and the subject clusters, variable groups, and interactions embedded in the dataset can be extracted visually.

Since the introduction of exploratory data analysis (EDA, Tukey, 1977), boxplots and scatterplots, aided by interactive functionality, have provided the statistical community with important graphical tools. These tools, together with various techniques for reducing dimensions, are useful for exploring the structure of the data when there are a moderate number of variables and when the structure is not too complex. However, with the recent rapid advances in computing, communications technology, and high-throughput biomedical instruments, the number of variables associated with the dataset can easily reach tens of thousands, but the need for practical data analysis remains. Dimension reduction tools often become less effective when applied to the visual exploration of information structures embedded in high-dimensional datasets. On the other hand, matrix visualization, when integrated with computing, memory, and display technologies, has the potential to enable us to visually explore the structures that underlie massive and complex datasets.

This chapter on matrix visualization unfolds as follows. We briefly review studies in this field in the next section. The foundation of matrix visualization, under the framework of generalized association plots (GAP, Chen, 2002), is then discussed in Sect. 15.3, along with some related issues. This is followed, in Sect. 15.4, by some generalization. Section 5 provides a matrix visualization examples involving 400 variables (arrays) and 2000 samples (genes). A comparison of matrix visualization with other popular graphical tools in terms of efficiency versus number of dimensions is then given in Sect. 15.6. Section 15.7 illustrates matrix visualization for binary data, while Sect. 15.8 discusses generalizations and extensions. We conclude this chapter with some perspectives on matrix visualization in Sect. 15.9.

15.2 Related Works

The concept of matrix visualization was introduced by Bertin (1967) as a reorderable matrix for systematically presenting data structures and relationships. Carmichael and Sneath (1969) developed taxometric maps for classifying OUTs (operational taxonomy units) in numerical phenetics analysis. Hartigan (1972) introduced the direct clustering of a data matrix, later known as block clustering (Tibshirani, 1999). Lenstra

(1974) and Slagle et al. (1975) related the traveling salesman and shortest spanning path problems to the clustering of data arrays. The color histogram of Wegman (1990) was the first color matrix visualization to be reported in the statistical literature. Minnotte and West (1998) extended the idea of color histograms to a data image package that was later used for outlier detection (Marchette and Solka, 2003).

Some matrix visualization techniques were developed to explore only proximity matrices: Ling (1973) looked for factors of variables by examining relationships using a shaded correlation matrix; Murdoch and Chow (1996) used elliptical glyphs to represent large correlation matrices; Friendly (2002) proposed corrgrams (similar to the reorderable matrix method) to analyze multivariate structure among the variables in correlation and covariance matrices. Chen (1996, 1999, and 2002) integrated the visualization of a raw data matrix with two proximity matrices (for variables and samples) into the framework of generalized association plots (GAP). The Cluster and TreeView packages of Eisen et al. (1998) are probably the most popular matrix visualization packages due to the proliferation of gene expression profiling for microarray experiments.

The permutation (ordering) of the columns and rows of a data matrix, and proximity matrices for variables and samples, is an essential step in matrix visualization. Several recent statistical works have touched on the issue of reordering variables and samples: Chen (2002) proposed the concept of the relativity of a statistical graph; Friendly and Kwan (2003) discussed the idea of effect-ordering of data displays; Hurley (2004) used scatterplot matrices and parallel coordinates plots as examples to address the issue of placing interesting displays in prominent positions. Different terms (such as the reorderable matrix, the heatmap, the color histogram, the data image and matrix visualization) have been used in the literature to describe these related techniques. We use matrix visualization (MV) to refer to them all.

The Basic Principles of Matrix Visualization 15.3

We use the GAP (Chen, 2002) approach to illustrate the basic principles of matrix visualization for continuous data, using the 6400 genes and 851 microarray experiments collected in the published yeast expression database for visualization and data mining (Marc et al., 2001), which is designated henceforth as Dataset 0. Detailed descriptions of data preprocessing are given for the yeast Microarray Global Viewer (http://transcriptome.ens.fr/ymgv/). For the purposes of illustration, we have selected 15 samples and 30 genes across these samples ("Dataset 1"), where rows correspond to genes and columns to microarray experiments (arrays). In various gene expression profile analyses, the roles played by rows and columns are often interchangeable. This interchangeability is well suited to the GAP approach to matrix visualization, where samples and variables are treated symmetrically and can be interchanged directly.

15.3.1 Presentation of the Raw Data Matrix

The first step in the matrix visualization of continuous data is the production of a raw data matrix $X_{30\times15}$, and two corresponding proximity matrices for the rows, $R_{30\times30}$, and the columns, $C_{15\times15}$, which are calculated with user-specified similarity (or dissimilarity) measures. The three matrices are then projected through suitable color spectra to construct corresponding matrix maps in which each matrix entry (raw data or proximity measurement) is represented by a color dot. The left panel in Fig. 15.1 shows the raw data matrix of $\log_2$-transformed ratios of expressions coded by a bidirectional green–black–red spectrum for Dataset 1, with Pearson correlations for between-array relations coded by a bidirectional blue–white–red spectrum, and Euclidean distances for between-gene relations coded by a unidirectional rainbow spectrum.

In the raw data matrix map, a red (green) dot in the ijth position of the map for $X_{30\times15}$ means that the ith gene at the jth array is relatively up (down)-regulated. A black dot stands for a relatively nondifferentially expressed gene/array combination. A red (blue) point in the ijth position of the $C_{15\times15}$ matrix map represents a positive (negative) correlation between arrays i and j. Darker (lighter) intensities of color stand for stronger absolute correlation coefficients, while white dots represent no correlations. A blue (red) point in the ijth position of the $R_{30\times30}$ matrix map represents a relatively small (large) distance between genes i and j, while a yellow dot represents a median distance.

Data Transformation

It may be necessary to apply transformations such as log, standardization (zero mean, unit variance), or normalization (normal score transformation) to the raw data before the data map is constructed or proximity matrices calculated in order to get a meaningful visual representation of the data structure, or comparable visual effects between displays. The transformation–visualization process may have to be repeated several times before the embedded information can be fully explored.

Selection of Proximity Measures

Proximity matrices have two major functions: (1) to serve as the direct visual representation of the relationships among variables and between samples; (2) to serve as the medium used to reorder the variables and samples for better visualization of the three matrix maps. The selection of proximity measures in matrix visualization plays a more important role than it does in numerical or modeling analyses. Pearson correlation often serves as the measure of proximity between variables, while Euclidean distance is commonly employed for samples (Fig. 15.1). For potential nonlinear relationships, Spearman's rank correlation and Kendall's tau coefficient can be used instead of the Pearson correlation to assess the between-variable relationship, while some nonlinear feature extraction methods such as the Isomap (Tenenbaum et al., 2000) distance can be used to measure nonlinear between-sample distances. More sophisticated kernel methods can also be applied when users see the need for them.

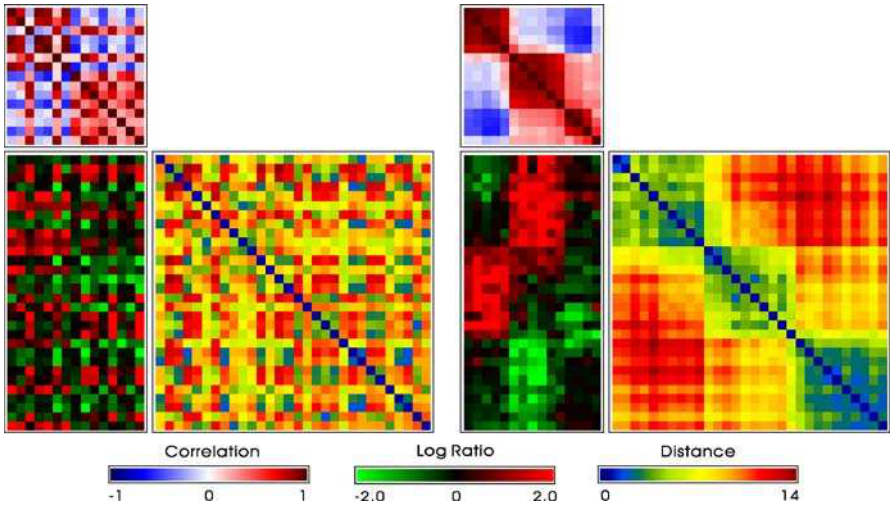

Figure 15.1. [This figure also appears in the color insert.] *Left*: unsorted data matrix (log ratio gene expression) map with two proximity matrix (Pearson correlation for arrays and Euclidean distance for genes) maps for Dataset 1. *Right*: application of elliptical seriations to the three matrix maps on the left panel

Color Spectrum

The selection of an appropriate color spectrum can be critical and is user-dependent in visualization of and information extraction from data and proximity matrices. The selection of a suitable color spectrum should focus on the capacity to express numerical nature individually and globally in the matrices. The choices for gene expression profiles that we mentioned above may well give way to others in different circumstances. Thus, illustrated in Fig. 15.2 is a correlation matrix map of fifty psychosis disorder variables (Chen, 2002) coded with four different bidirectional color spectra. While displays (a) and (b) appear more agreeable to our human perception, displays (c) and (d) actually provide better resolution for distinguishing different levels of correlation intensities. The relative triplet color codes (red, green, blue) in the RGB cube for these four color spectra are shown in Fig. 15.3.

Display Conditions

The display conditions are analogous to data transformations for colors. Usually, the whole color spectrum is used to represent the complete range of values in the data matrix. The matrix conditions can be switched to row or column conditions to emphasize individual variable distributions or subject profiles. For a bidirectional color spectrum (green–black–red for differential gene expressions, blue–white–red for correlation coefficients), the center matrix condition symmetrizes the color spectrum around the baseline numeric value (1:1 for log2 ratio gene expression, zero for the correlation coefficient). On occasion, we might want to downweight the effects of extreme values in the dataset, and it is possible to use ranks as a replacement for numerical values. This is termed the rank matrix condition.

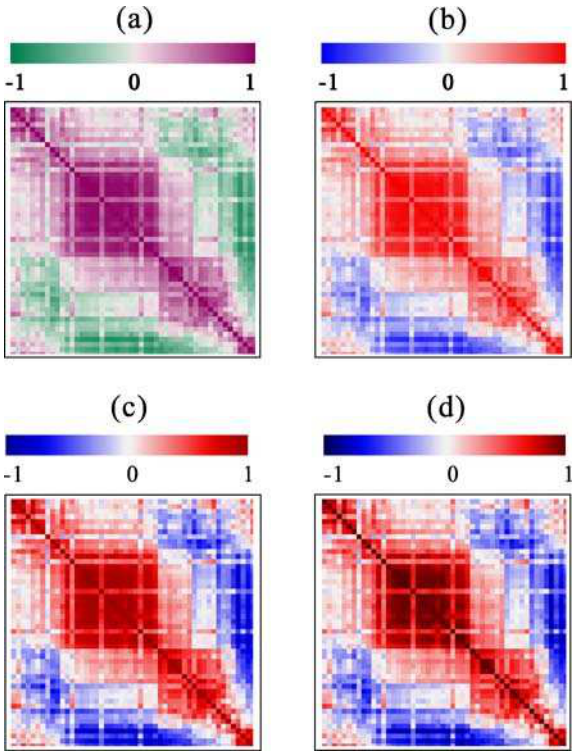

Figure 15.2. [This figure also appears in the color insert.] Four color spectra applied to the same correlation matrix map for fifty psychosis disorder variables (Chen, 2002)

Resolution of a Statistical Graph

If the data matrix or proximity matrices contain potential extreme values, the relative structure of these extreme values compared to the main data cloud will dominate the overall visual perception of the raw data map and the proximity matrix maps. This problem can be handled by using rank conditions or by compressing the color spectrum to a suitable range. We can apply a logarithm or similar transformation to reduce the outlier effect or to simply remove the outlier.

15.3.2 Seriation of Proximity Matrices and the Raw Data Matrix

Without suitable permutations (orderings) of the variables and samples, matrix visualization is of no practical use for visually extracting information (Fig. 15.1, left panel). It is necessary to compute meaningful proximity measures for variables and samples, and to apply suitable permutations to these matrices, before matrix visualization is used to reveal the information structure of the given dataset. We discuss some con-

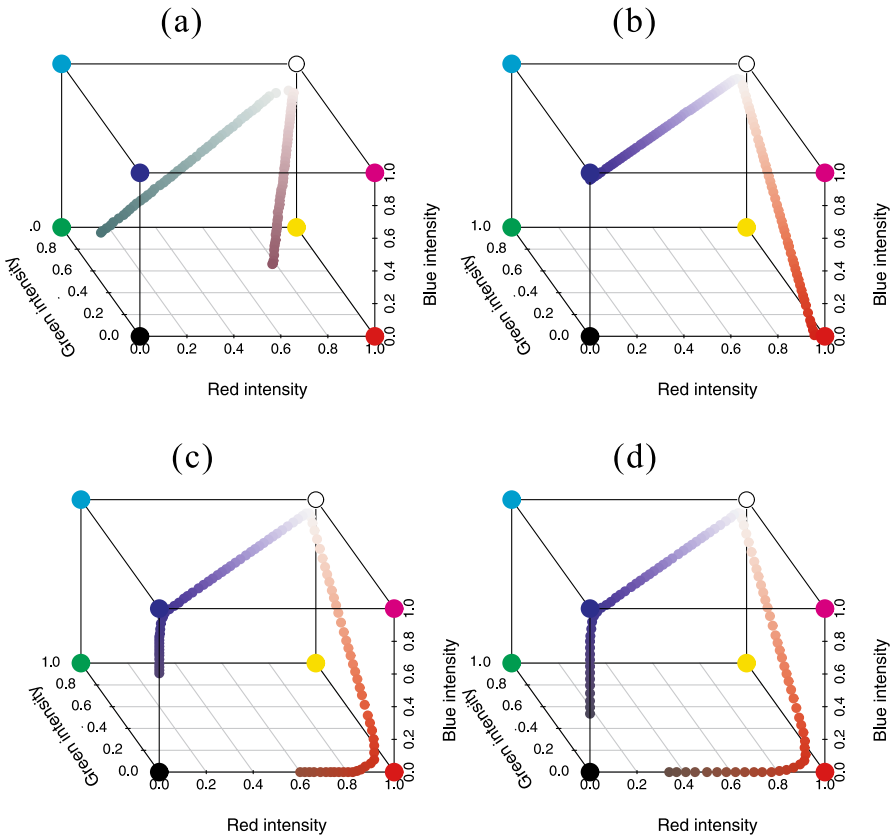

Figure 15.3. Relative (red, green, blue) hues in the RGB cubes for the four color spectra in Fig. 15.2

cepts and criteria for evaluating the performances of different seriation algorithms in reordering related matrices below.

Relativity of a Statistical Graph

Chen (2002) proposed a concept, the relativity of a statistical graph, for evaluating general statistical graphic displays. The idea is to place similar (different) objects at closer (more distant) positions in a statistical graph. In a continuous display, such as the histogram or a scatterplot, relativity always holds automatically. This is illustrated by the histogram of the Petal Width variable and the scatterplot of the Petal Width and Petal Length variables for 150 Iris flowers shown in Fig. 15.4 (Fisher, 1936). Two flowers, denoted with the × and ∘ symbols, are placed next to each other on these two displays automatically, because they share similar petal widths and lengths. Friendly and Kwan (2003) proposed a similar concept to order information in general visual displays, which they called the effect-ordered data display. Hurley (2004) also studied related issues using examples involving scatterplot matrices and parallel coordinates plots.

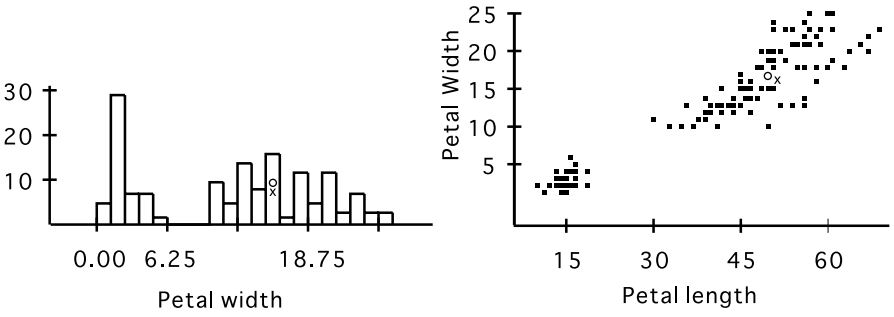

Figure 15.4. Concept of the relativity of a statistical graph for a continuous dataset (the Iris data)

The relativity concept does not usually hold for a matrix visualization or parallel coordinates type of display, since one can easily destroy the property with a random permutation. It is common practice to apply various permutation algorithms to sort the columns and rows of the designated matrix, so that similar (different) samples/variables are permuted to make them closer (more distant) rows/columns.

Global Criterion: Robinson Matrix

It is usually desirable to permute a matrix to make it resemble a Robinson matrix (Robinson, 1951) as closely as possible, because of the smooth and pleasant visual effect of permuted matrix maps. A symmetric matrix is called a Robinson matrix if its elements satisfy $r_{ij} \leq r_{ik}$ if $j < k < i$ and $r_{ij} \geq r_{ik}$ if $i < j < k$. If the rows and columns of a symmetric matrix can be permuted to those of a Robinson matrix, we call it pre-Robinson. For a numerical comparison, three anti-Robinson loss functions (Streng, 1978) are calculated for each permuted matrix, $D = \{d_{ij}\}$, for the amount of deviation from a Robinson form with distance-type proximity:

$$AR(i) = \sum_{i=1}^{p} \left[\sum_{j<k<i} I(d_{ij} < d_{ik}) + \sum_{i<j<k} I(d_{ij} > d_{ik}) \right],$$

$$AR(s) = \sum_{i=1}^{p} \left[\sum_{j<k<i} I(d_{ij} < d_{ik}) \cdot |d_{ij} - d_{ik}| + \sum_{i<j<k} I(d_{ij} > d_{ik}) \cdot |d_{ij} - d_{ik}| \right],$$

$$AR(w) = \sum_{i=1}^{p} \left[\sum_{j<k<i} I(d_{ij} < d_{ik}) |j-k| |d_{ij} - d_{ik}| + \sum_{i<j<k} I(d_{ij} > d_{ik}) |j-k| |d_{ij} - d_{ik}| \right].$$

$AR(i)$ counts only the number of anti-Robinson events in the permuted matrix; $AR(s)$ sums the absolute values of the anti-Robinson deviations; $AR(w)$ is a weighted version of $AR(s)$ penalized by the difference in the column indices of the two entries.

Elliptical Seriation

Chen (2002) introduced a permutation algorithm called rank-two elliptical seriation that extracts the elliptical structure of the converging sequence of iteratively formed

correlation matrices using eigenvalue decomposition. Given a p-dimensional proximity matrix D, a sequence of correlation matrices $R = (R^{(1)}, R^{(1)}, \cdots)$ is iteratively formed from it. Here $R^{(1)}$ is the correlation matrix of the original proximity matrix D, and $R^{(n)}$ is the correlation matrix of $R^{(n-1)}$ for $n > 1$. The iteratively formed sequence of correlation matrices gradually cumulates the variation information to the leading eigenvectors. At the iteration with rank two, there are only two eigenvectors left with nonzero eigenvalues, and all information is reduced to the ellipse spanned by the two eigenvectors. Every object has its relative position on this two-dimensional ellipse, and a unique permutation is obtained. Elliptical seriation usually identifies very good global permutations, and is useful for identifying global clustering patterns and smooth temporal gene expression profiles (Tien et al., 2006) by optimizing the Robinson criterion.

Local Criterion: Minimal Span Loss Function

The minimal span loss function $MS = \sum_{i=1}^{n-1} d_{i,i+1}$ for a permuted matrix $D = \{d_{ij}\}$ focuses on the optimization of local structures. The idea is to find a shortest path through all data elements, as in the traveling salesman problem. The local seriation method produces tighter blocks than the global method does around the main diagonal of the proximity matrix. In addition, we can combine the anti-Robinson measure and minimal span loss into a measure in which a band along the diagonal of a proximity matrix is selected with width w $(0 < w < n)$, and the anti-Robinson measurement is computed within that band.

Tree Seriation

The hierarchical clustering tree with a dendrogram (Eisen et al., 1998) is the most popular method for two-way sorting the gene-by-array matrix map employed in gene expression profiling. The ordering of terminal nodes generated by an agglomerative hierarchical clustering tree automatically keeps good local grouping structure, since the tree dendrogram is constructed through a sequential bottom-up merging of "most similar" subnodes. On the other hand, a divisive hierarchical clustering tree usually retains better global patterns through a top-down splitting of "most heterogeneous" substructures. Divisive hierarchical clustering trees are rarely used due to their computational complexity.

Flipping of Intermediate Nodes

One critical issue when applying the leaves of the dendrogram in order to sort the rows/columns of an expression profile matrix is the flipping of the intermediate nodes. As illustrated in Fig. 15.5 with a schematic dendrogram (Fig. 15.5a), the $n-1$ intermediate nodes for a dendrogram of n objects can be flipped independently (Fig. 15.5b), resulting in $2^{(n-1)}$ different dendrogram layouts (Figure 15.5c, for example) and corresponding permutations for the n objects with identical proximity matrices (Pearson correlation or Euclidean distance) and the same tree linkage method (single, complete, average or centroid). The flipping mechanism of intermediate nodes can be guided by either an external or an internal reference list. For example, the Cluster

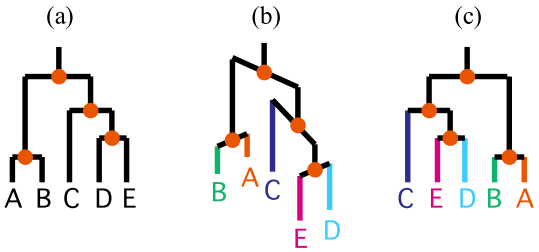

Figure 15.5. Flipping mechanism for intermediate nodes of a dendrogram

software developed by Eisen's lab (1998) guides the tree flips based on the average expression level. He also suggests that one can use the results of a one-dimensional self-organizing map (SOM, Kohonen, 2001) to guide the tree seriation. This makes the tree seriation as close to the external references as possible. In Alon et al. (1999), it is suggested that one should order the leaf nodes according to the similarity between a node and its parent's siblings. Bar-Joseph (2001) proposed a fast optimal leaf ordering method for hierarchical clustering that maximizes the sum of the similarities of adjacent leaves in the ordering. These are two examples of internal references.

15.4 Generalization and Flexibility

15.4.1 Summarizing Matrix Visualization

Sorted matrix maps are capable of displaying the raw expression patterns and the association structures among genes and arrays. One can go one step further and identify clusters in the permuted matrix maps using the dendrogram branching structure or other partitioning methods, such as the converging sequence of Pearson's correlation matrices (Chen, 2002) and block searching (Hartigan, 1972). Once the partitioned matrix maps are obtained (Fig. 15.6, left panel), a summarizing matrix visualization which Chen (2002) coined "sufficient matrix visualization" can be constructed by representing individual data points and proximity measures in each identified subject–subject, variable–variable and subject–variable block by the summary statistic (means, medians or standard deviations) for that particular block.

The three maps in Fig. 15.6, right panel, summarize the sufficient information of the data matrix, and the corresponding proximity matrices for the gene expression profiles are shown in the left panel. In the sufficient MV of Fig. 15.6, right panel, users can easily extract the within and between correlation structure for the three array groups, the relative clustering patterns of the four gene clusters, and the interaction behavior of the four gene clusters on the three array groups. There are three requirements for ensuring the effectiveness of a sufficient MV at extracting the overall information structure embedded in the original data matrix and two proximity matrices: (1) appropriate permuted variables and samples; (2) carefully derived partitions for variables and samples, and; (3) representative summary statistics.

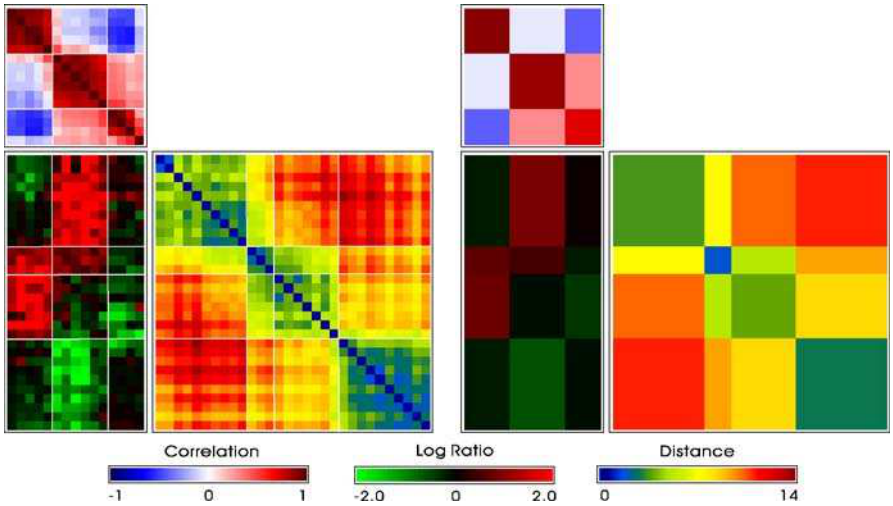

Figure 15.6. *Left*: partitioned data and proximity matrix maps for Dataset 1. *Right*: sufficient data and proximity matrix maps

Sediment Display 15.4.2

The sediment display of a row data matrix for rows (columns) is constructed by sorting the column (row) profiles for each row (column) independently according to their magnitudes. This display expresses the distribution structure for all rows (columns) simultaneously. The middle panel of Fig. 15.7 has the sediment display for all 30 gene expression profiles, while the right panel has the expression distributions for each of the 15 selected arrays. The sediment displays for genes and arrays convey similar information to that given by a boxplot when the color strips at the quartile positions are extracted.

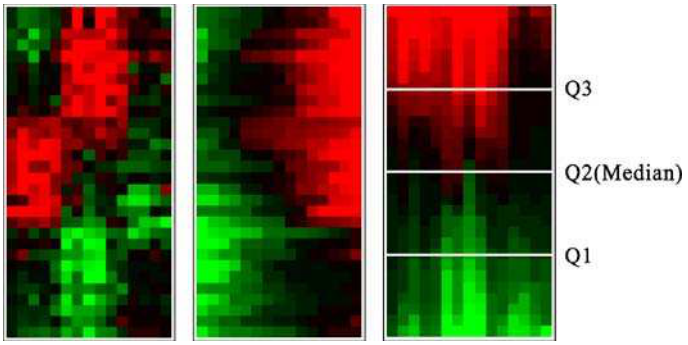

Figure 15.7. Sediment displays for genes (*middle panel*) and arrays (*right panel*) for the permuted data matrix (*left panel*) of Dataset 1

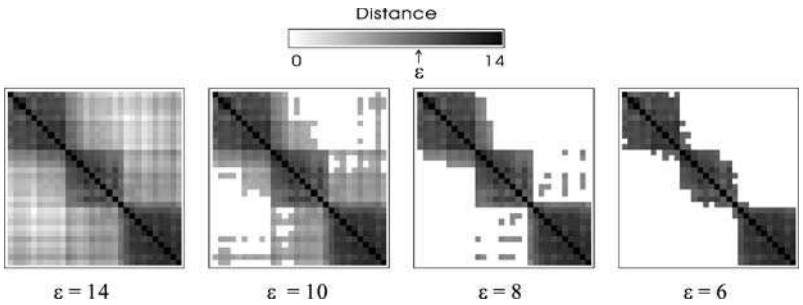

Figure 15.8. Sectional displays for the permuted gene distance map. Only distances smaller than the threshold, ϵ, are displayed

15.4.3 Sectional Display

The purpose of a sectional display is to exhibit only those numerical values that satisfy certain conditions in the data or the associated proximity maps. For example, one can choose to ignore the values below some threshold by not displaying the corresponding color dots. For a permuted distance map, one can emphasize more coherent neighboring structure by displaying only the corresponding neighbors dynamically. Figure 15.8 shows a series of such sectional displays (in grey scale) for the distance matrix for the genes in Fig. 15.1, right panel (and Fig. 15.6, left panel).

15.4.4 Restricted Display

Outlying data points or proximity measures can mask detailed color resolutions. This situation can be improved by displaying only rank conditions instead of original magnitudes, or by compressing the color spectrum to represent only the main body of the data values, i.e., one displays data values that fall within some range of the data using the whole color spectrum. Figure 15.9, left panel, shows a restricted display of Fig. 15.8 with an artificial outlier observation added. The relatively large distance of this outlier from the other observations causes the color spectrum to mask the main

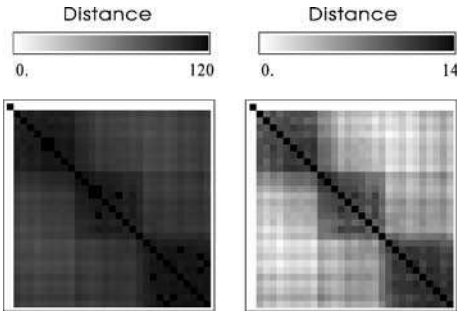

Figure 15.9. Original (*left*) and restricted (*right*) displays for the permuted gene distance map in Fig. 15.8 with an outlying gene added

feature embedded in the distance matrix. The right panel of Fig. 15.9 uses the whole grey spectrum to represent the distance range of 0–14 only, which reveals the main three-group structure. The use of nonlinear color mapping (for the distances), like the one implemented in MANET (Unwin, 1998), can also resolve this problem.

An Example

15.5

Construction of an MV Display

Many of the microarrays in Dataset 0 have lots of missing values due to technical issues and because different experiments studied different sets of genes in the yeast genome. Two thousand genes with four hundred arrays that had relatively few missing values were then selected from the original Dataset 0, resulting in Dataset 2. Illustrated in Fig. 15.10 is the MV display of Dataset 2. Pearson's correlation coefficient is used to measure associations between genes and between arrays, a common practice in gene expression profile analysis. Average linkage clustering trees are then grown on the two correlation matrices for genes and arrays. The relative positions of the terminal nodes of the two dendrograms are then used to sort the corresponding correlation matrix maps and the data matrix map (the gene expression profile). The basic gene clustering structure and array (experiments) grouping patterns can be identified using these tree-sorted matrix maps.

The enlarged, permuted data matrix map used for gene expression profiling is displayed in Fig. 15.11. Red dots represent a relatively high expression of message RNA for the gene-experiment combination, green dots indicate relatively low expression, and black dots designate relatively little differential expression. Missing values are coded in white, it is clear that many of the arrays (experiments) still contain some missing observations. Such an MV display presents each gene expression profile as a horizontal strip of color dots across all arrays (experiments), and the important visual information is carried by the relative variations in color hues.

Without suitable permutations that sort rows of similar genes so that they are close together and place identical arrays next to each other, so that the relativity property holds, an MV display is basically useless. Based on this two-way permuted display, one looks for horizontal strips of genes that share similar expression profiles, and vertical strips of arrays that exhibit close experimental results. The blocks of the two directions illustrate the interaction patterns of gene clusters and experiment groups. All of the numerical information is displayed in this raw expression profile map (with proximity maps for genes and arrays and corresponding dendrograms). Careful and patient examination of these color maps can lead to valuable insights into the embedded information structure.

Examination of an MV Display

As with other visualization tools, both proper training and experience are required to extract as much information out of these complex matrix visualization displays as possible. While examining complex MV displays such as those shown in Figs. 15.10 and 15.11, several general steps should be taken:

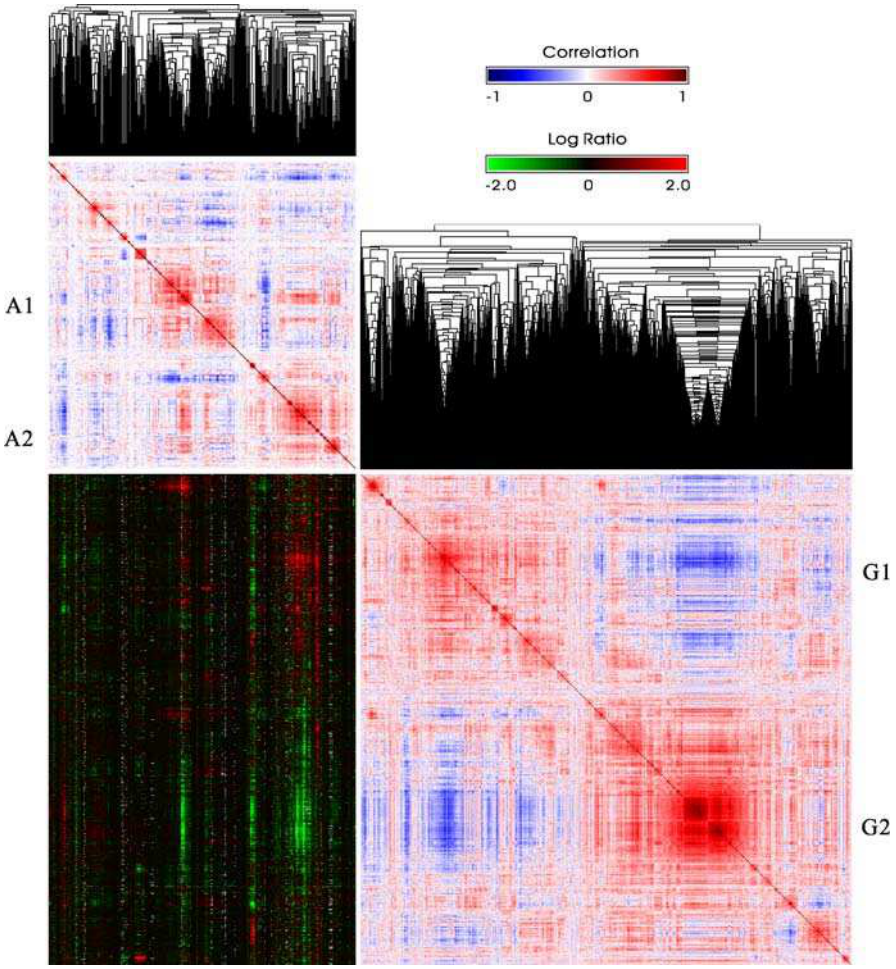

Figure 15.10. [This figure also appears in the color insert.] Data matrix map ($\log_2$ ratio gene expression) with two proximity matrix maps (Pearson correlation for both genes and arrays) for Dataset 2 permuted by two average linkage trees (for genes (*rows*) and arrays (*columns*))

1. For the column (array) proximity matrix:
   a) Search for coherent clusters of arrays along the main diagonal of the correlation (maybe distance in other circumstances) matrix with dark red points. Two dominant groups of arrays can be identified around the middle and the lower-right corner of the correlation matrix, with several small but coherent clusters scattered along the main diagonal. Let's denote these two major groups of arrays as A1 and A2. The arrays grouped into these clusters must have similar expression patterns across all 2000 genes (which will be examined in later steps).

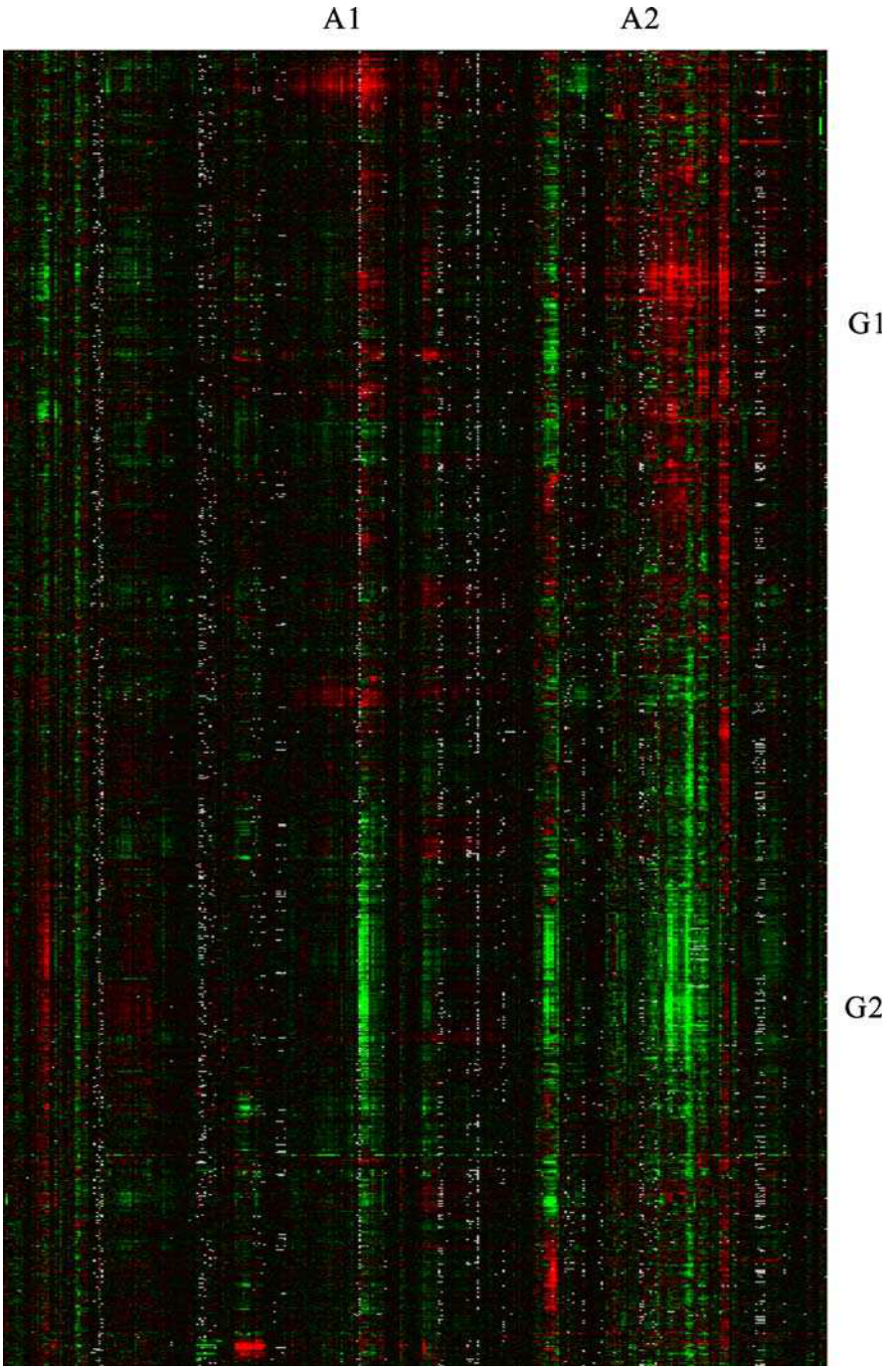

Figure 15.11. Enlarged expression profile matrix map of Fig. 15.10 (missing observations are coded in white)

b) Look for interactions between the array clusters at off-diagonal locations. Various types of between-cluster correlation patterns with substructure are also easily pinned down.
c) The arrays represent many different biological assays for various functions of *Saccharomyces cerevisiae* yeast, such as cell-cycle control, stress (environmental changes, relevant drug-affected), metabolic/genetic control, transcriptional control and DNA binding (http://transcript-ome.ens.fr/ymgv/). Different biomedical assays activate and suppress expression patterns of certain functional groups of genes. We need to integrate this biological/medical knowledge with the numerical/graphical findings in 1a and 1b to validate known information and more importantly to explore and interpret novel interesting patterns.
d) Hierarchical clustering trees for arrays and genes also yield a partial visual exploration of the data and proximity structure, but this is not as comprehensive as direct visualization of the two proximity matrix maps, since the dendrograms only retain some of the information in the proximity matrices from which they are constructed.

2. For the row (gene) proximity matrix:
   a) Similar procedures to those described in 1a and 1b for arrays (columns) must be repeated for the gene (row) proximity matrix. Of particular interest is the dichotomous pattern of these 2000 genes. The up-regulated (red) genes at the upper half and the down-regulated (green) genes at the bottom half of the A2 arrays are responsible for this dichotomous structure. We denote these two clusters of genes as G1 and G2 here. Several small subclusters of genes within G1 and G2 can also be identified along the main diagonal.
   b) It is necessary to go one step further and consult various annotation databases for more detailed interpretations of and explanations for the potential clusters of genes identified this way. Some of the genes have not been annotated yet. Their potential functions can be roughly determined through the positively correlated (up-regulated) gene clusters and the negatively correlated (down-regulated) patterns.

3. For the raw data (gene expression profile) matrix:
   a) Many major and minor array groups and gene clusters were found in steps 1 and 2. In step 3, we use the raw data (gene expression profile) matrix map to search for the interaction patterns of the gene clusters on the array groups. It is also necessary to use vertical strips of expression profiles to contrast between array group structure variations and horizontal strips to distinguish between gene cluster distribution differences. With careful examination, one can associate certain blocks of expression in the raw data matrix with the formation of each array group and gene cluster and the between-group (cluster) differentiation.
   b) There are about 10 000 (~1.25 %) missing observations in this data matrix of 400 arrays with 2000 genes. It is clear that the pattern of missing observa-

tions is not a random one. Different array group and gene cluster combinations have different proportions of missing observations. The visualization of the missing structure is a great aid to users when they attempt to choose a more appropriate missing value estimate or imputation mechanism for further analyses.

c) Visual exploration provides valuable insights into more advanced studies, such as the confirmation of existing metabolite pathways (see Sect. 7 for an example) and the exploration of novel pathways.

This paragraph has only discussed some general issues associated with examining an MV display. If we have access to expert knowledge and biologists familiar with experiments related to *Saccharomyces cerevisiae* yeast, there are actually many more interesting patterns that can be explored. In Figs. 15.10 and 15.11 we demonstrated an MV can easily handle thousands of samples. An MV display can also handle thousands of variables, since samples and variables are treated symmetrically in the MV framework.

Comparison with Other Graphical Techniques

15.6

In this section, we compare the visualization efficiencies of the scatterplot (SP), the parallel coordinates plot (PCP) (Inselberg, 1985; Wegman, 1990), and matrix visualization (MV), based on varying dimensionality of the dataset.

Low-Dimensional Data

For one-dimensional data, scatterplot and the PCP displays amount to dotplots, while a one-dimensional MV yields a colored bar chart. In any event, it is unlikely that any method of displaying one-dimensional data will prove more popular than the histogram. A scatterplot is the most efficient graphical display for two-dimensional data. While a PCP relies on the n connecting line segments between two vertical dotplots to represent the association between the two variables, MV displays each sample as a single row with two colored dots. The efficiency of scatterplots decreases as dimension increases. For three-dimensional data, a rotational scatterplot is commonly used to extract geometric structure by viewing a sequence of two-dimensional scatterplots over a range of angles controlled by the user. The usefulness of PCP and MV displays of three-dimensional data is a subtle point, and the best permutation of variables is definitely needed to enforce relativity for both types of displays.

High-Dimensional Data

A scatterplot matrix (SM) is used to simultaneously visualize the information structure embedded in all $C(p, 2)$ pairs of variables for data with more than three di-

mensions. Grand tours (Asimov, 1985) are sometimes undertaken in the hope of extracting high-dimensional data structure by rotating randomly projected three-dimensional plots. Dimension reduction techniques, such as principal component analysis, are also useful for displaying structural information from high-dimensional data in low-dimensional displays. Figure 15.12 shows a scatterplot matrix display of the first 30 variables (arrays) in Dataset 2, while Fig. 15.13 gives the corresponding PCP for these data. We note that a PCP of high-dimensional data with a large sample size can simultaneously display all of the samples, but it is usually necessary to use some interactive mechanism to select subsets of samples in order to study the relative structure across all variables, as in Fig. 15.13. Moreover, for these plots, more than one pixel width is needed to display each variable.

In general, a scatterplot matrix needs $C(p,2) \times n$ dots to display a dataset with n samples measured on p variables, a PCP needs p vertical lines plus $(p-1) \times n$ line segments, and an MV plot requires $n \times p$ dots. When p becomes large, larger than 15

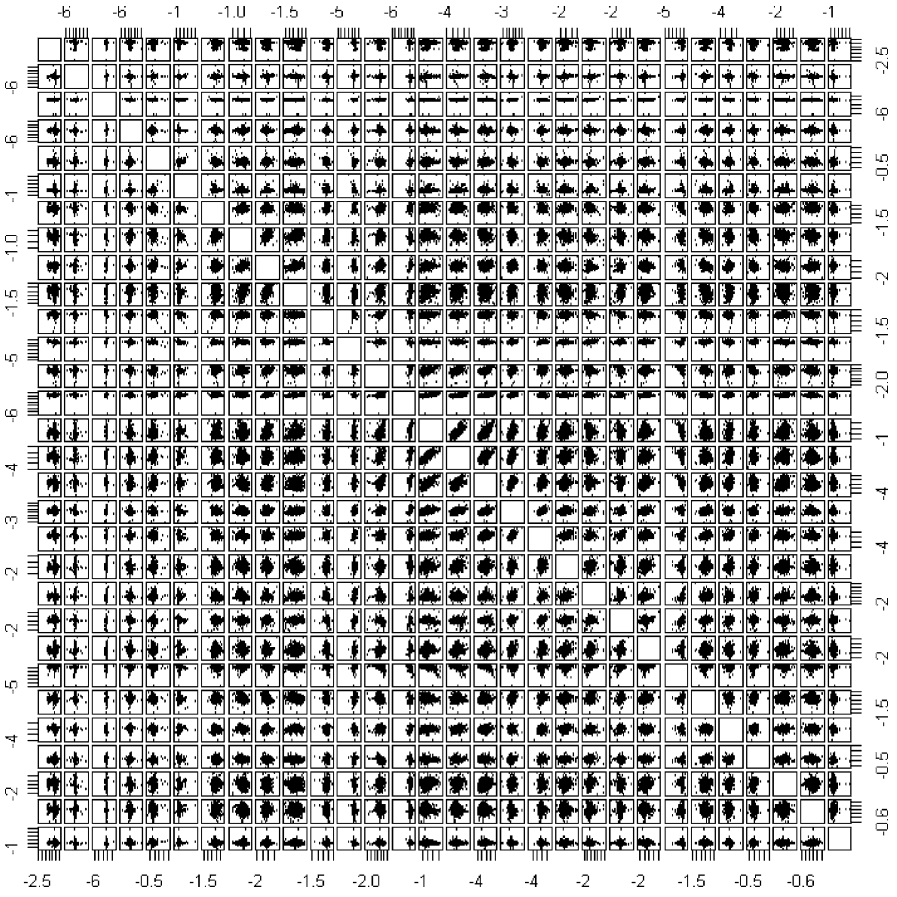

Figure 15.12. Scatterplot matrix for the first thirty arrays of Dataset 2

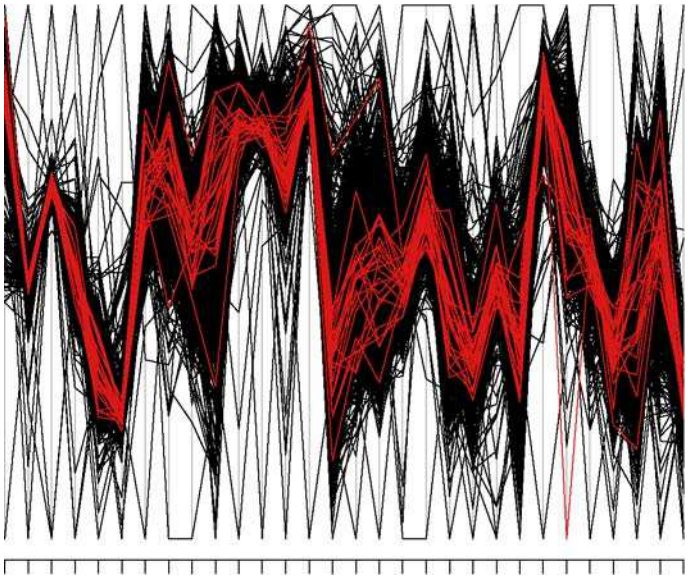

Figure 15.13. Parallel coordinates plot for the first thirty arrays of Dataset 2

for instance, a scatterplot matrix is basically useless. A PCP display does well for up to a few hundred variables, but founders for more due to the space required to display the line segments that connect sample points. A scatterplot matrix also wastes a high proportion of the display space. An MV display, on the other hand, utilizes every column pixel to display a variable on a computer screen. PCP has an advantage over MV on the sample side, but MV plots provide better resolution.

Overall Efficiency

Figure 15.14 is a diagram of efficiency against dimensionality for a conventional scatterplot (matrix) and dimension-free visualization tools such as the parallel coordinates plot (PCP) and matrix visualization (MV). While direct visual perception of the geometric pattern makes scatterplots the most efficient type of display for visualizing low-dimensional data, MV and PCP are definitely better for visualizing datasets with fifteen or more variables.

Missing Values

It is very difficult to display missing values in a scatterplot, while it is always possible to display missing values above or below the regular data range of each variable in a PCP display. The MANET system by Unwin et al. (1996) can be used to display missing information interactively. In an MV plot, a missing value can be simply displayed at the corresponding position (row and column) with a color that can be easily distinguished from the color spectrum of the numerical values. The missing values of the gene expression profiles in Figs. 15.10 and 15.11 are coded in white. With appropriate permutations for rows and columns, the corresponding variable–sample

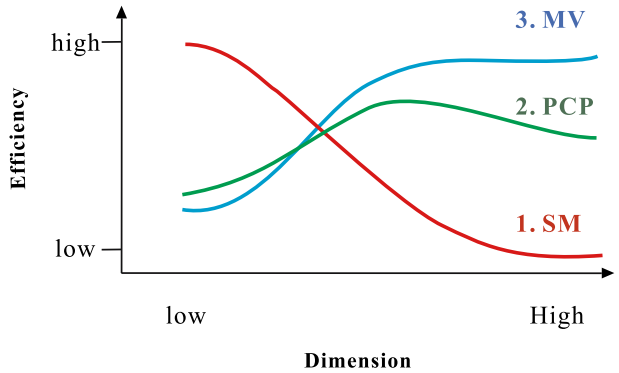

Figure 15.14. Schematic illustration of the relative efficiencies of the scatterplot matrix, the parallel coordinates plot, and matrix visualization, for varying numbers of dimensions

combinations for the missing structure can be accessed visually. MV users can benefit from a simple visual perception of the mechanism associated with the missing observations (random or not, ignorable or unignorable) before formal statistical modeling of the missing values is implemented.

15.7 Matrix Visualization of Binary Data

While scatterplots, PCP, and MV displays have their own advantages and disadvantages for continuous data structures of various dimensions, an MV display is the only statistical graph that can meaningfully display binary data sets over all dimensions. We use the KEGG (Kyoto Encyclopedia of Genes and Genomes) metabolism pathways (http://www.genome.jp/kegg/pathway.html) for *Saccharomyces cerevisiae* yeast to illustrate how an MV display can be generalized to visually extract all of the important information embedded in multivariate binary data. The KEGG website provides detailed information on the 1177 related genes involved in 100 metabolism pathways in *Saccharomyces cerevisiae* yeast. We simplified the complex information structure down to a two-way binary data matrix of 1177 genes by 100 pathways. This binary data matrix is called Dataset 3 in our study. A one (zero) encoded at the ith row and jth column of the matrix means that the ith gene is (not) involved in jth pathway activities.

15.7.1 Similarity Measure for Binary Data

The usual measures used to evaluate associations between samples and variables for continuous data – Euclidean distance and correlation coefficients – cannot be applied directly to binary data sets. Two issues are noted here in relation to the selection of similarity measures for binary data in an MV display.

Symmetric and Asymmetric Binary Variables

A binary variable is considered symmetric if both of its states are equally valuable; that is, there is no preference over which outcome should be coded as 0 or 1. A binary variable is asymmetric if the outcomes of the states are not equally important, such as the positive and negative outcomes of a disease diagnosis. Conventionally, the most important outcome (the rarer one) is coded 1, the other 0. Thus, asymmetric binary variables are often considered "monary" (as if there is only one state).

Sparseness and Dimensionality

Asymmetric binary variables are usually sparse in nature, and it is difficult to identify an appropriate association measure that could be used to assess the relationships among samples and between variables. Dimension reduction techniques also fail in attempts to summarize high-dimensional data structures in low-dimensional fashions. Listed in Fig. 15.15 are some similarity measures commonly applied to binary data. For sparse data, it is common practice to use the Jaccard coefficient instead of the simple matching coefficient.

Binary Data		Object B		
		1	0	
Object A	1	a	b	(a+b)
	0	c	d	(c+d)
		(a+c)	(b+d)	(a+b+c+d)

Similarity for Binary Data	Formula
Kulczynskl	$\frac{a}{b+c}$
Rao	$\frac{a}{a+b+c+d}$
Jaccard	$\frac{a}{a+b+c}$
simple match	$\frac{a+d}{a+b+c+d}$
Sneath	$\frac{a}{a+2b+2c}$
Rogers	$\frac{a+d}{a+2b+2c+d}$
Hamman	$\frac{a+d-(b+c)}{a+b+c+d}$
Phi	$\frac{ad-bc}{\sqrt{(a+b)(a+c)(d+b)(d+c)}}$
Yule	$\frac{ad-bc}{ad+bc}$

Figure 15.15. Some similarity measures for binary data

15.7.2 Matrix Visualization of the KEGG Metabolism Pathway Data

The 1−Jaccard distance coefficient is used to compute the proximity matrices for both genes and pathways in Fig. 15.16. Elliptical seriations (Chen, 2002) are employed to permute the two 1−Jaccard distance matrices and the binary pathway data matrix. One can easily see, from the binary data matrix map and the proximity matrix map for genes, that there are many genes that are only involved in the activities of a single pathway. We then exclude those genes from further analysis, since they provide no association information. This reduces the original 1177 genes by 100 pathways binary matrix to a 432 genes by 88 pathways matrix (some pathways are also excluded after excluding genes). When there are too few horizontal or vertical pixels for an MV

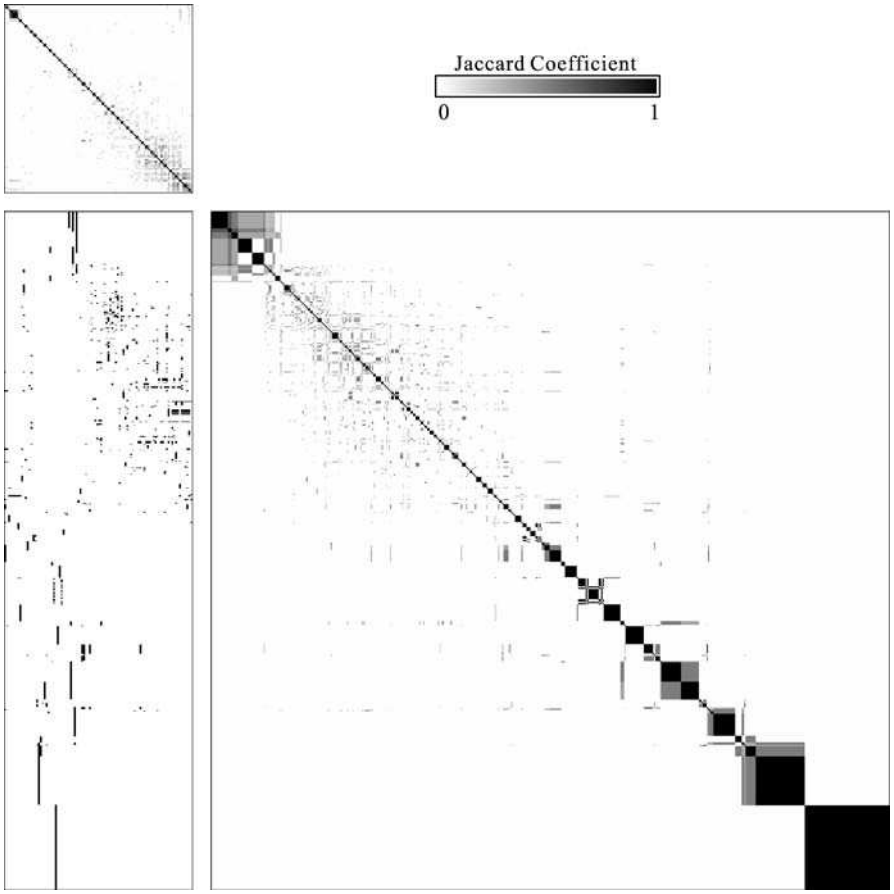

Figure 15.16. Binary data matrix map for the Dataset 3 (KEGG metabolite pathway database with 1177 genes (*rows*) for 100 pathways (*columns*)) with two Jaccard proximity matrices for genes and for pathways sorted by elliptical seriations in both directions

display, users can either use the scroll bars to visualize a certain region of the display or to zoom out in order to visualize the overall structure with an averaging effect, as used in a typical computer graph.

Average linkage clustering trees are then employed to sort the resulting 1–Jaccard distance matrices for genes and pathways and the corresponding binary data matrix, see Fig. 15.17. The association structure between genes and among pathways can now be comprehended using the three corresponding permuted matrix maps. In the upper left corner of the data matrix and the upper-left corners of the proximity maps for genes and pathways we can identify several groups of genes that are involved in the activities of only a few pathways, and several small groups of pathways that share

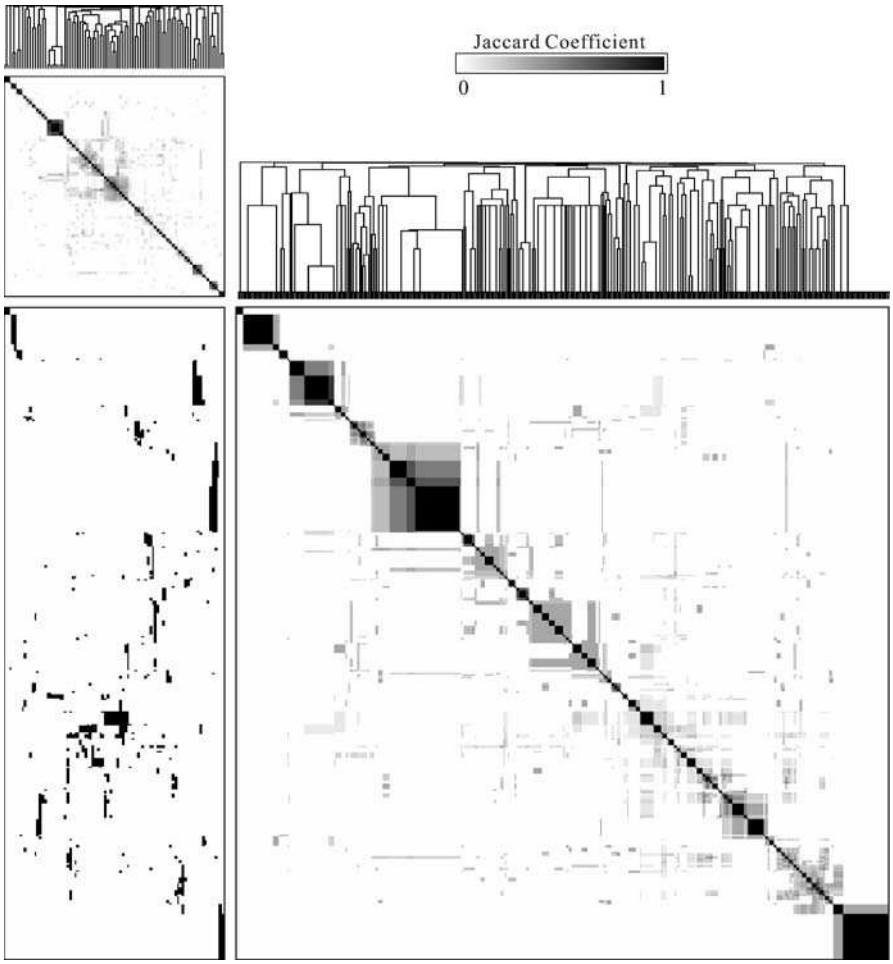

Figure 15.17. Binary data matrix for the reduced Dataset 4 (432 genes (*rows*) for 88 pathways (*columns*)) with two Jaccard proximity matrices for genes and for pathways sorted by average linkage trees in both directions

very similar groups of genes. The other genes and pathways have more complicated interactions between activities. It is of course possible to further exclude pathways and genes with simpler behavior, and to focus on the details of interactions of the more active genes and pathways.

15.8 Other Modules and Extensions of MV

So far we have introduced the fundamental framework for matrix visualization in the GAP (Chen, 2002) approach to the visualization of continuous and binary data matrices, with corresponding derived proximity matrices. We have also presented some generalizations, such as the sufficient MV and the sediment, sectional, and restricted displays. In practice, observed data can be highly complex, to the degree that basic matrix visualization procedures are not rich enough to comprehend the data structure. In some situations, one may not be able to apply MV directly to the given data or proximity matrices. This section discusses ongoing projects and future directions that will make matrix visualization a more promising statistical graphical environment. One important feature of the GAP (Chen, 2002) approach to matrix visualization is that it usually contains four basic procedures: (1) color projection of the raw data matrix; (2) computation of two proximity matrices for variables and sample; (3) color projection of the two proximity matrices, and; (4) permutations of variables and sample. Most extensions of MV are related to the first two procedures. It is simple to adapt the aforementioned algorithms for the other two steps once the first two procedures are fixed.

15.8.1 MV for Nominal Data

It is much more difficult to perform MV for nominal data than it is for binary data, since one can use black/white to code 1/0 if the binary data is asymmetric, and the Jaccard and other coefficients for binary data in order to derive the relationships between variables and between samples. There is no natural way to guide the color-coding for multivariate nominal data in such a way that the color version of the relativity of a statistical graph still holds (Chang et al., 2002). The derivation of meaningful between-variable and between-sample proximity measures for nominal data is another challenging issue. Chen (1999) and Chang et al. (2002) utilized the Homals (de Leeuw, 1998) algorithm and developed a categorical version of matrix visualization that naturally resolved the two critical problems.

15.8.2 MV for Covariate Adjustment

Quite often, covariate data (such as gender and age) are collected in a study in addition to the variables of primary interest. When the effects of covariates are an issue, covariate adjustment must be taken into consideration, much as in a formal statistical modeling process. Wu and Chen (2006) introduced a unified regression model

approach which partitions the raw data matrix into model and residual matrices, and ordinary MV can be applied to these two derived matrices. The covariate adjustment process is accomplished through by estimating conditional correlations. For a discrete covariate, a correlation matrix for variables is decomposed into within- and between-component matrices. When the covariate is continuous, the conditional correlation is equivalent to the partial correlation under the assumption of joint normality.

Data with Missing Values 15.8.3

The relativity of a statistical graph (Chen, 2002) is the main concept used in seriation algorithms to construct meaningful clustered matrices. This property can be used to develop a weighted pattern method to impute the missing values. The initial proximity measurements for rows and columns with missing values can be computed with pair-wise complete observations, and then imputed values are estimated and updated iteratively for the subsequent proximity calculations and imputation.

Modeling Proximity Matrices 15.8.4

Many statistical modeling procedures try to visually explore the high-dimensional pattern embedded in a proximity matrix that records pair-wise similarity or dissimilarity for a set of objects through a low-dimensional projection. Multidimensional scaling, hierarchical clustering analysis, and factor analysis are three such statistical techniques. Four types of matrices are usually involved in the modeling processes of these statistical procedures. The input proximity matrix is transformed into a disparity matrix prior to fixing the statistical model that summarizes the information in the output distance matrix. A stress (residual) matrix is calculated to assess the goodness of fit for the modeling. Such a study aims to create a comprehensive diagnosis environment for statistical methods through various types of matrix visualization for the numerical matrices involved in the modeling process.

Conclusion 15.9

Matrix visualization is the color order-based representation of data matrices. It is beneficial to employ human vision to explore the structure in a matrix in the pursuit of further appropriate mathematical operations and statistical modeling. A good matrix visualization environment helps us to gain comprehensive insights into the underlying process. Rather than rely solely on numerical characteristics, it is suggested that matrix visualization should be performed as a preliminary step in modern exploratory data analysis, and research into and applications of matrix visualization continue to be of much interest.

Matrix visualization displays provide five levels of information: (1) raw scores for every sample/variable combination; (2) an individual sample score vector across all

variables and an individual variable vector across all samples; (3) an association score for every sample–sample and variable–variable relationship; (4) a grouping structure for variables and a clustering effect for samples, and; (5) an interaction pattern for sample clusters on variable groups.

With the capacity to display thousands of variables in a single picture, the flexibility to work with all types of data, and the ability to handle various manifestations of extraordinary data patterns (missing value, covariate adjustment), we believe that matrix visualization has the potential to become one of the most important graphical tools applied to exploratory data analysis (EDA) in the future.

References

Alon, U., Barkai, N., Notterman, D.A., Gish, K. Ybarra, S., Mack, D. and Levine, A.J. (1999). Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays, *Proc Natl Acad Sci USA.*, 96(12):6745–6750.

Asimov, D. (1985). The grand tour: a tool for viewing multidimensional data, *SIAM Journal of Scientific and Statistical Computing*, 6(1):128–143.

Bar-Joseph, Z., Gifford, D.K. and Jaakkola, T.S. (2001). Fast optimal leaf ordering for hierarchical clustering, *Bioinformatics*, 17:S22–S29.

Bertin, J. (1967). *Semiologie Graphique*, Paris: Editions gauthier-Villars. English translation by William J. Berg. as Semiology of Graphics: Diagrams, Networks, Maps. University of Wisconsin Press, Madison, WI, 1983.

Carmichael, J.W. and Sneath, P.H.A. (1969). Taxometric maps, *Systematic Zoology*, 18:402–415.

Chang, S.C., Chen, C.H., Chi, Y.Y. and Ouyoung, C.W. (2002). Relativity and resolution for high dimensional information visualization with generalized association plots (GAP), *Section for Invited Papers, Proceedings in Computational Statistics 2002 (Compstat 2002)*, Berlin, Germany, 55–66.

Chen, C.H. (1996). The properties and applications of the convergence of correlation matrices. In *1996 Proceedings of the Section on Statistical Graphics of the American Statistical Association*, 49–54.

Chen, C.H. (1999). Extensions of generalized association plots, *1999 Proceedings of the Section on Statistical Graphics of the American Statistical Association*, 111–116.

Chen, C.H. (2002). Generalized association plots: information visualization via iteratively generated correlation matrices, *Statistica Sinica,* 12:7–29.

Chen, C.H., Hwu, H.G., Jang, W.J., Kao, C.H., Tien, Y.J., Tzeng, S. and Wu, H.M. (2004). Matrix visualization and information mining, *Proceedings in Computational Statistics 2004 (Compstat 2004)*, pp. 85–100, Physika Verlag, Heidelberg.

Eisen, M.B., Spellman, P.T., Brown, P.O. and Botstein, D. (1998). Cluster analysis and display of genome-wide expression patterns, *Proc Natl Acad Sci USA,* 95:14863–14868.

Fisher, R.A. (1936). The use of multiple measurements in taxonomic problems, *Annals of Eugenics*, 7:179–188.

Friendly, M. (2002). Corrgrams: exploratory displays for correlation matrices. *The American Statistician*, 56(4):316-324.

Friendly, M. and Kwan, E. (2003). Effect ordering for data displays. *Computational Statistics and Data Analysis*, 43(4):509–539.

Hartigan, J.A. (1972). Direct clustering of a data matrix. *Journal of the American Statistical Association*, 78:123–129.

Hurley, C.B. (2004). Clustering visualization of multidimensional data, *Journal of Computational and Graphics Statistics*, 13:788–806.

Inselberg, A. (1985). The plane with parallel coordinates, *The Visual Computer*, 1:69–91.

Lenstra, J. (1974). Clustering a data array and the traveling salesman problem, *Operations Research*, 22:413–414.

Ling, R.L. (1973). A computer generated aid for cluster analysis, *Communications of the ACM*, 16(6):355–361.

Marc, P., Devaux, F. and Jacq, C. (2001). yMGV: a database for visualization and data mining of published genome-wide yeast expression data, *Nucleic Acids Research*, 29(13):e63.

Marchette, D.J. and Solka, J.L. (2003). Using data images for outlier detection, *Computational Statistics and Data Analysis,* 43:541–552.

Michailidis, G. and de Leeuw, J. (1998). The Gifi system for descriptive multivariate analysis, *Statistical Science*, 13:307–336.

Minnotte, M. and West, W. (1998). The data image: a tool for exploring high dimensional data sets, in *Proceedings of the ASA Section on Statistical Graphics,* Dallas, TX, 25–33.

Murdoch, D.J. and Chow, E.D. (1996). A graphical display of large correlation matrices, *The American Statistician*, 50:178–180.

Robinson, W.S. (1951). A method for chronologically ordering archaeological deposits, *American Antiquity* 16:293–301.

Slagle, J.R., Chang, C.L. and Heller, S.R. (1975). A clustering and data-reorganizing algorithm, *IEEE Trans. Syst. Man Cybern*, 5:125-128.

Streng, R. (1991). Classification and seriation by iterative reordering of a data matrix. In Classification, Data Analysis, and Knowledge Organization: Models and Methods with Applications (Edited by H.H. Bock and P. Ihm), 121-130. Springer, New York.

Tenenbaum, J.B., de Silva, V. and Langford, J.C. (2000). A global geometric framework for nonlinear dimensionality reduction, *Science*, 290(5500):2319–2323.

Tibshirani, R., Hastie, T., Eisen, M., Ross, D., Botstein, D. and Brown, P. (1999). Clustering methods for the analysis of DNA microarray data. Technical Report, Stanford University, Oct. 1999.

Tien, Y.J., Lee, Y.S, Wu, H.M. and Chen, C.H. (2006). Integration of clustering and visualization methods for simultaneously identifying coherent local clusters with smooth global patterns in gene expression profiles. Technical Report 2006-11, Institute of Statistical Science, Academia, Taiwan.

Tukey, J.W. (1977). *Exploratory Data Analysis*. Addison-Wesley, Reading, MA.

Unwin, A.R., Hawkins, G., Hofmann, H. and Siegl, B. (1996). Interactive graphics for data sets with missing values – MANET, *Journal of Computational and Graphical Statistics* 5:113–122.

Unwin, A.R and Hofmann, H. (1998). New interactive graphics tools for exploratory analysis of spatial data. *In Innovations in GIS 5, ed. S Carver,* pp. 46–55. Taylor & Francis, London.

Wegman, E.J. (1990). Hyperdimensional data analysis using parallel coordinates. *Journal of the American Statistical Association*. 85:664–675.

Wu, H.M. and Chen, C.H. (2005). *Covariate adjusted matrix visualization*. Technical Report. Institute of Statistical Science, Academia, Taiwan.