BIOINFORMATICS

Vol. 17 no. 6 2001
Pages 520–525

Missing value estimation methods for DNA microarrays

Olga Troyanskaya[1], Michael Cantor[1], Gavin Sherlock[2], Pat Brown[3], Trevor Hastie[4], Robert Tibshirani[4], David Botstein[2] and Russ B. Altman[1,*]

[1]Stanford Medical Informatics, [2]Department of Genetics, Stanford University School of Medicine, Stanford, CA, USA, [3]Department of Biochemistry, Stanford University School of Medicine, and Howard Hughes Medical Institute, Stanford, CA, USA and [4]Departments of Statistics and Health Research and Policy, Stanford University, Stanford, CA, USA

Received on November 13, 2000; revised on February 22, 2001; accepted on February 26, 2001

ABSTRACT

Motivation: Gene expression microarray experiments can generate data sets with multiple missing expression values. Unfortunately, many algorithms for gene expression analysis require a complete matrix of gene array values as input. For example, methods such as hierarchical clustering and K-means clustering are not robust to missing data, and may lose effectiveness even with a few missing values. Methods for imputing missing data are needed, therefore, to minimize the effect of incomplete data sets on analyses, and to increase the range of data sets to which these algorithms can be applied. In this report, we investigate automated methods for estimating missing data.

Results: We present a comparative study of several methods for the estimation of missing values in gene microarray data. We implemented and evaluated three methods: a Singular Value Decomposition (SVD) based method (SVDimpute), weighted K-nearest neighbors (KNNimpute), and row average. We evaluated the methods using a variety of parameter settings and over different real data sets, and assessed the robustness of the imputation methods to the amount of missing data over the range of 1–20% missing values. We show that KNNimpute appears to provide a more robust and sensitive method for missing value estimation than SVDimpute, and both SVDimpute and KNNimpute surpass the commonly used row average method (as well as filling missing values with zeros). We report results of the comparative experiments and provide recommendations and tools for accurate estimation of missing microarray data under a variety of conditions.

Availability: The software is available at http://smi-web.stanford.edu/projects/helix/pubs/impute/

Contact: russ.altman@stanford.edu

*To whom correspondence should be addressed.

INTRODUCTION

DNA microarray technology allows for the monitoring of expression levels of thousands of genes under a variety of conditions (DeRisi *et al.*, 1997; Spellman *et al.*, 1998). Microarrays have been used to study a variety of biological processes, from differential gene expression in human tumors (Perou *et al.*, 2000) to yeast sporulation (Chu *et al.*, 1998). Various analysis techniques have been developed, aimed primarily at identifying regulatory patterns or similarities in expression under similar conditions. Commonly used analysis methods include clustering techniques (Eisen *et al.*, 1998; Tamayo *et al.*, 1999), techniques based on partitioning of data (Heyer *et al.*, 1999; Tamayo *et al.*, 1999), as well as various supervised learning algorithms (Alter *et al.*, 2000; Brown *et al.*, 2000; Golub *et al.*, 1999; Raychaudhuri *et al.*, 2000; Hastie *et al.*, 2000).

The data from microarray experiments is usually in the form of large matrices of expression levels of genes (rows) under different experimental conditions (columns) and frequently with some values missing. Missing values occur for diverse reasons, including insufficient resolution, image corruption, or simply due to dust or scratches on the slide. Missing data may also occur systematically as a result of the robotic methods used to create them. Our informal analysis of the distribution of missing data in real samples shows a combination of all of these, but none dominating. Such suspicious data is usually manually flagged and excluded from subsequent analysis (Alizadeh *et al.*, 2000). Many analysis methods, such as principle components analysis or singular value decomposition, require complete matrices (Alter *et al.*, 2000; Raychaudhuri *et al.*, 2000). Of course, one solution to the missing data problem is to repeat the experiment. This strategy can be expensive, but has been used in

validation of microarray analysis algorithms (Butte *et al.*, 2001). Missing $\log_2$ transformed data are often replaced by zeros (Alizadeh *et al.*, 2000) or, less often, by an average expression over the row, or 'row average'. This approach is not optimal, since these methods do not take into consideration the correlation structure of the data. Thus, many analysis techniques, as well as other analysis methods such as hierarchical clustering, k-means clustering, and self-organizing maps, may benefit from using more accurately estimated missing values.

There is not a large published literature concerning missing value estimation for microarray data, but much work has been devoted to similar problems in other fields. The question has been studied in contexts of non-response issues in sample surveys and missing data in experiments (Little and Rubin, 1987). Common methods include filling in least squares estimates, iterative analysis of variance methods (Yates, 1933), randomized inference methods, and likelihood-based approaches (Wilkinson, 1958). An algorithm similar to nearest neighbors was used to handle missing values in CART-like algorithms (Loh and Vanichsetakul, 1988). Most commonly applied statistical techniques for dealing with missing data are model-based approaches. We have tried to minimize the influence of specific modeling assumptions in our methods.

In this work, we describe and evaluate three methods of estimation for missing values in DNA microarrays. We compare our KNN- and SVD-based methods to the row average method, which is likely the most sophisticated estimation technique currently employed for microarray missing data estimation.

SYSTEM AND METHODS

Experimental methods

We implemented and evaluated three data imputation methods: a method based on K Nearest Neighbors (KNN) algorithm, a Singular Value Decomposition based method, and simple row (gene) average.

Three microarray data sets were used: a study in yeast *Saccharomyces cerevisiae* focusing on identification of cell-cycle regulated genes (Spellman *et al.*, 1998), an exploration of temporal gene expression during the metabolic shift from fermentation to respiration in *Saccharomyces cerevisiae* (DeRisi *et al.*, 1997), and a study of response to environmental changes in yeast (Gasch *et al.*, 2000). Two of the datasets were time-series data (DeRisi *et al.*, 1997; Spellman *et al.*, 1998) and one contained a non-time series subset of experiments from Gasch *et al.* (2000). In addition, one of the time-series data sets contained less apparent noise (Botstein, personal communication) than the other. We refer to those data sets by their characteristics: time series, noisy time series, and non-time series.

Each data set was pre-processed for the evaluation by removing rows and columns containing missing expression values, yielding 'complete' matrices. The methods were then evaluated over each dataset as follows. Between 1 and 20% of the data were deleted at random to create test data sets. Each method was then used to recover the introduced missing values for each data set, and the estimated values were compared to those in the original data set. The metric used to assess the accuracy of estimation (henceforth referred to as normalized RMS error) was calculated as the Root Mean Squared (RMS) difference between the imputed matrix and the original matrix, divided by the average data value in the complete data set. This normalization allowed for comparison of estimation accuracy between different data sets.

We examined different parameter sets for the KNN- and SVD-based algorithms. For KNN, the number of neighboring genes optimal for estimation was varied, whereas for SVD, different numbers of principal components, here termed 'eigengenes' in the sense of Alter *et al.* (2000), were used. Thus the experimental design allowed us to assess the accuracy of each method under different conditions (type of data, fraction of data missing) and determine optimal parameters.

KNNimpute algorithm

The KNN-based method selects genes with expression profiles similar to the gene of interest to impute missing values. If we consider gene A that has one missing value in experiment 1, this method would find K other genes, which have a value present in experiment 1, with expression most similar to A in experiments 2–N (where N is the total number of experiments). A weighted average of values in experiment 1 from the K closest genes is then used as an estimate for the missing value in gene A. In the weighted average, the contribution of each gene is weighted by similarity of its expression to that of gene A.

After examining a number of metrics for gene similarity (Pearson correlation, Euclidean distance, variance minimization), we determined that Euclidean distance was a sufficiently accurate norm. This finding is somewhat surprising, given that the Euclidean distance measure is often sensitive to outliers, which could be present in microarray data. However, we found that log-transforming the data seems to sufficiently reduce the effect of outliers on gene similarity determination.

SVDimpute algorithm

In this method, we employ singular value decomposition (1) to obtain a set of mutually orthogonal expression patterns that can be linearly combined to approximate the expression of all genes in the data set. These patterns, which in this case are identical to the principle components of the gene expression matrix, are further referred to

as eigengenes (Alter *et al.*, 2000; Anderson, 1984; Golub and Van Loan, 1996).

$$A_{m\times n} = U_{m\times m}\Sigma_{m\times n}V^T_{n\times n}. \quad (1)$$

Matrix V^T now contains eigengenes, whose contribution to the expression in the eigenspace is quantified by corresponding eigenvalues on the diagonal of matrix Σ. We then identify the most significant eigengenes by sorting the eigengenes based on their corresponding eigenvalue. Although it has been shown by Alter *et al.* (2000) that several significant eigengenes are sufficient to describe most of the expression data, the exact fraction of eigengenes best for estimation needs to be determined empirically.

Once k most significant eigengenes from V^T are selected, we estimate a missing value j in gene i by first regressing this gene against the k eigengenes and then use the coefficients of the regression to reconstruct j from a linear combination of the k eigengenes. The jth value of gene i and the jth values of the k eigengenes are not used in determining these regression coefficients.

It should be noted that SVD can only be performed on complete matrices; therefore we originally substitute row average for all missing values in matrix A, obtaining A'. We then utilize an expectation maximization method to arrive at the final estimate, as follows. Each missing value in A' is estimated using the above algorithm, and then the procedure is repeated on the newly obtained matrix, until the total change in the matrix falls below the empirically determined threshold of 0.01.

RESULTS AND DISCUSSION

KNNimpute

Performance of the KNN-based method was assessed over different data sets (both types of data and percent of data missing) and over different values of K (Figure 1). The method is very accurate, with the estimated values showing only 6–26% average deviation from the true values, depending on the type of data and fraction of values missing. Notably, this method is successful in accurate estimation of missing values for genes that are expressed in small clusters. Other methods, such as row average and SVD, are likely to be more inaccurate on such clusters because the clusters themselves do not contribute significantly to the global parameters upon which these methods rely. When errors for individual values are considered, approximately 88% of the values are estimated with normalized RMS error under 0.25, with KNN-based estimation for a noisy time series data set with 10% entries missing (Figure 2). Under low apparent noise levels in time series data, as many as 94% of values are estimated within 0.25 of the original value.

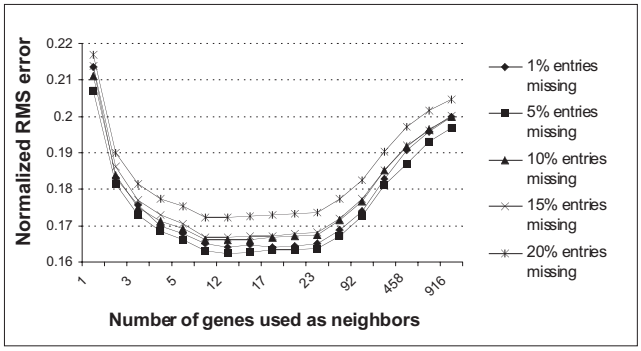

Fig. 1. Effect of number of nearest neighbors used for KNN-based estimation on noisy time series data. Different curves correspond to experiments performed for data sets with different percent of entries missing.

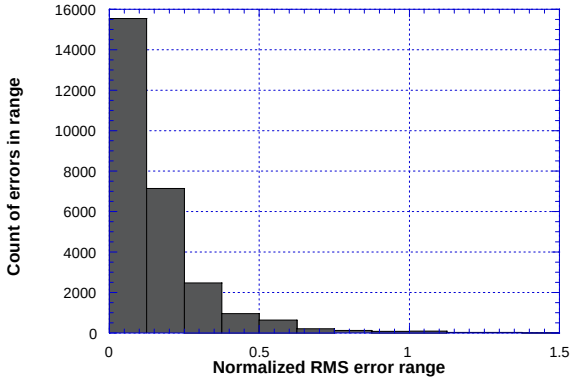

Fig. 2. Distribution of errors for KNN-based estimation on a noisy time-series data set. Individual errors from estimation with K = 15 at 10% of data missing are displayed in a histogram. Most of the normalized RMS errors are under 0.25.

Although a smaller percentage of missing data makes data imputation more precise, the algorithm is robust to increasing the percent of values missing, with a maximum of 10% decrease in accuracy with 20% of the data missing (Figure 1). In addition, the method is relatively insensitive to the exact value of K within the range of 10–20 neighbors (Figure 1). Performance declines when a lower number of neighbors is used for estimation, primarily due to overemphasis of a few dominant expression patterns. However, when the same gene is present twice on the arrays, the method appropriately gives a very strong weight to that gene in the estimation. The deterioration in performance at larger values of K (above 20) may be explained as follows. First, the inclusion of expression patterns that are significantly different from the gene of interest can decrease accuracy because the 'neighborhood' has become too large and not sufficiently relevant to the

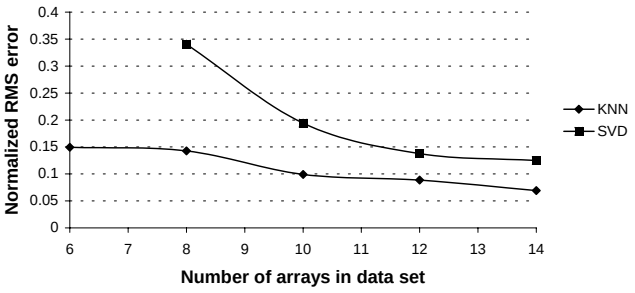

Fig. 3. Effect of reduction of array number on KNN- and SVD-based estimation. On a time series data set, estimation was performed on matrices with successively lower number of columns. The SVD algorithm could not be applied to matrices with less than eight columns.

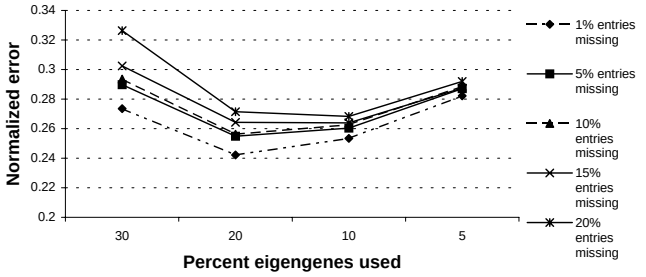

Fig. 4. Performance of SVD-based imputation with different fractions of eigengenes used for estimation. Normalized RMS error was assessed for a non-time course microarray (most challenging estimation) with 5–30% eigengenes used. Different color curves correspond to various percents of data missing from the data set.

estimation problem. In fact, optimal selection of K likely depends on the average cluster size for the given data set. Second, there may be significant noise present in microarray data. As K increases, the contribution of noise to the estimate overwhelms the contribution of the signal, leading to a decrease in accuracy.

To assess the variance in RMS error over repeated estimations for the same file with the same percent of missing values removed, we performed 60 additional runs of missing value removal and subsequent estimation on one of the time series data sets. At 5% values missing and K = 123, the average RMS error was 0.203, with variance of 0.001. Thus, our evaluation method appears to be reliable.

Although microarray experiments typically involve a large number of arrays, sometimes experimenters need to analyze data sets with small numbers of experiments (columns in the matrix). KNNimpute can accurately estimate data for matrices with as low as six columns (Figure 3). We do not recommend using this method on matrices with less than four columns.

SVDimpute

To determine the optimal parameter set for SVDimpute, the method was evaluated using the most significant 5, 10, 20, and 30% of the eigengenes for estimation (Figure 4). The most accurate estimation is achieved when approximately 20% of the eigengenes are used for estimation. In contrast with KNNimpute, where the error curve appears relatively flat between 10 and 20 neighbors, performance of the SVD-based method deteriorates sharply as the number of eigengenes used is changed.

Although SVD-based estimation provides significantly higher accuracy than row average on all data sets, its performance is sensitive to the type of data being analyzed. SVDimpute yields best results on time-series data with low noise level (Figures 5 and 6). Under such conditions the method performs better than KNNimpute if the right number of eigengenes is used for estimation (Figure 6). This likely reflects the signal-processing nature of the SVD-based method. When the expression data is dominated by the combined effect of strong patterns of regulation over time (as in time-series data), SVD is ideally suited to estimating expression of an individual gene in terms of these constituent patterns. In contrast, the KNN-based method exhibits higher performance for both noisy time series data and non-time series data. As SVD-based estimation is essentially a linear regression method in lower-dimensional space, this deterioration in performance is not surprising for non-time series data, where a clear expression pattern is often not present. The slightly lower sensitivity to noise compared to KNNimpute is most likely due to the fact that expression patterns for smaller groups of genes can sometimes not be sufficiently represented in the dominant eigengenes used for estimation.

Row average

Estimation by row (gene) average, although an improvement upon replacing missing values with zeros, yielded drastically lower accuracy than either KNN- or SVD-based estimation (Figure 5). As expected, the method performs most poorly on non-time series data (normalized RMS error of 0.40 and more), but error on other data sets was also significantly higher than both of the other methods. This is not surprising, since this row averaging assumes that the expression of a gene in one of the experiments is similar to its expression in a different experiment, which is often not true. In contrast to SVD and KNN, row average does not take advantage of the rich information provided by the expression patterns of other genes (or even duplicate runs of the same gene) in the data set.

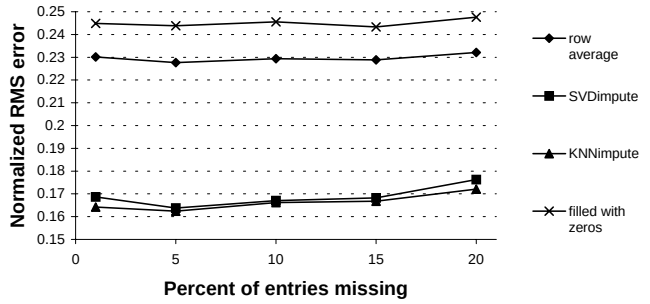

Fig. 5. Comparison of KNN, SVD, and row average based estimations' performance on a noisy time series data set. The same data set (with identical entries missing) was used to assess the accuracy of each method, and normalized RMS error was plotted as a function of fraction of values missing in the data.

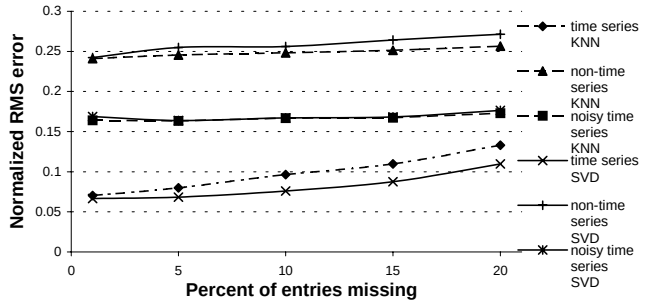

Fig. 6. Performance of KNNimpute and SVDimpute methods on different types of data as a function of entries missing. Best performance of each of the methods was plotted. Three sets of curves represent three data sets (non-time series—top, noisy time series—middle, and time series—bottom).

Although an in-depth study was not performed on column average, some experiments were performed with this method and it does not yield satisfactory performance (results not shown).

Performance

For a matrix of m rows (genes) and n columns (experiments), the computational complexity of the KNNimpute method is approximately $O(m^2n)$, assuming $m \gg k$ and fewer than 20% of the values missing. The computational complexity of a full SVD calculation is $O(n^2m)$. However, SVDimpute utilizes an expectation–maximization algorithm, thus bringing the complexity to $O(n^2mi)$, where i is the number of iterations performed before the threshold value is reached. The row average algorithm is the fastest, with computational complexity of $O(nm)$. The KNNimpute method, implemented in C++, takes 3.23 min on a Pentium III 500 MHz computer to estimate missing values for a data set with 6153 genes and 14 experiments, with 10% of the entries missing.

CONCLUSIONS

KNN- and SVD-based methods provide fast and accurate ways of estimating missing values for microarray data. Both methods far surpass the currently accepted solutions (filling missing values with zeros or row average) by taking advantage of the correlation structure of the data to estimate missing expression values. Based on the results of our study, we recommend KNN-based method for imputation of missing values.

Although both KNN and SVD methods are robust to increasing the fraction of data missing, KNN-based imputation shows less deterioration in performance with increasing percent of missing entries. In addition, the KNNimpute method is more robust than SVD to the type of data for which estimation is performed, performing better on non-time series or noisy data. KNNimpute is also less sensitive to the exact parameters used (number of nearest neighbors), whereas the SVD-based method shows sharp deterioration in performance when a non-optimal fraction of missing values is used. From the biological standpoint, KNNimpute has the advantage of providing accurate estimation for missing values in genes that belong to small tight expression clusters. Missing points for such genes could be estimated poorly by SVD-based estimation if their expression pattern is not similar to any of the eigengenes used for regression.

KNN-based imputation provides for a robust and sensitive approach to estimating missing data for microarrays. However, it is important to exercise caution when drawing critical biological conclusions from data that is partially imputed. The goal of this method is to provide an accurate way of estimating missing values in order to minimally bias the performance of microarray analysis methods. However, estimated data should be flagged where possible and its significance on the discovery of biological results should be assessed in order to avoid drawing unwarranted conclusions.

ACKNOWLEDGEMENTS

We would like to thank Soumya Raychaudhari and Joshua Stuart for thoughtful comments on the manuscript and discussions, and Orly Alter and Mike Liang for helpful suggestions. O.T. is supported by a Howard Hughes Medical Institute predoctoral fellowship and by a Stanford Graduate Fellowship. M.C. is supported by NIH training grant LM-07033. T.H. is partially supported by NSF grant DMS-9803645 and NIH grant ROI-CA-72028-01. R.T. is supported by the NIH grant 2 R01 CA72028, and NSF grant DMS-9971405. D.B. is partially supported by CA 77097 from the NCI. R.B.A. is supported by NIH-GM61374, NIH-LM06244, NSF DBI-9600637, SUN Microsystems and a grant from the Burroughs-Wellcome Foundation.

REFERENCES

Alizadeh,A.A., Eisen,M.B., Davis,R.E., Ma,C., Lossos,I.S., Rosenwald,A., Boldrick,J.C., Sabet,H., Tran,T., Yu,X., Powell,J.I., Yang,L., Marti,G.E., Moore,T., Hudson,Jr,J., Lu,L., Lewis,D.B., Tibshirani,R., Sherlock,G., Chan,W.C., Greiner,T.C., Weisenburger,D.D., Armitage,J.O., Warnke,R. and Staudt,L.M., *et al.* (2000) Distinct types of diffuse large B-cell lymphoma identified by gene expression profiling. *Nature*, **403**, 503–511.

Alter,O., Brown,P.O. and Botstein,D. (2000) Singular value decomposition for genome-wide expression data processing and modeling. *Proc. Natl Acad. Sci. USA*, **97**, 10101–10106.

Anderson,T.W. (1984) *An Introduction to Multivariate Statistical Analysis*. Wiley, New York.

Brown,M.P., Grundy,W.N., Lin,D., Cristianini,N., Sugnet,C.W., Furey,T.S., Ares,Jr.,M. and Haussler,D. (2000) Knowledge-based analysis of microarray gene expression data by using support vector machines. *Proc. Natl Acad. Sci. USA*, **97**, 262–267.

Butte,A.J. and Ye,J., *et al.* (2001) Determining significant fold differences in gene expression analysis. *Pac. Symp. Biocomput.*, **6**, 6–17.

Chu,S., DeRisi,J., Eisen,M., Mulholland,J., Botstein,D., Brown,P.O. and Herskowitz,I. (1998) The transcriptional program of sporulation in budding yeast. *Science*, **282**, 699–705.

DeRisi,J.L., Iyer,V.R. and Brown,P.O. (1997) Exploring the metabolic and genetic control of gene expression on a genomic scale. *Science*, **278**, 680–686.

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.

Gasch,A.P., Spellman,P.T., Kao,C.M., Carmel-Harel,O., Eisen,M.B., Storz,G., Botstein,D. and Brown,P.O. (2000) Genomic expression programs in the response of yeast cells to environmental changes. *Mol. Biol. Cell.*, in press.

Golub,G.H. and Van Loan,C.F. (1996) *Matrix Computations*. Johns Hopkins University Press, Baltimore, MD.

Golub,T.R., Slonim,D.K., Tamayo,P., Huard,C., Gaasenbeek,M., Mesirov,J.P., Coller,H., Loh,M.L., Downing,J.R., Caligiuri,M.A., Bloomfield,C.D. and Lander,E.S. (1999) Molecular classification of cancer: class discovery and class prediction by gene expression monitoring. *Science*, **286**, 531–537.

Hastie,T., Tibshirani,R., Eisen,M., Alizadeh,A., Levy,R., Staudt,L., Chan,W., Botstein,D. and Brown,P.P. (2000) 'Gene shaving' as a method for identifying distinct sets of genes with similar expression patterns. *Genome Biol.*, **1**, research0003.1–research0003.21.

Heyer,L.J., Kruglyak,S. and Yooseph,S. (1999) Exploring expression data: identification and analysis of coexpressed genes. *Genome Res.*, **9**, 1106–1115.

Little,R.J.A. and Rubin,D.B. (1987) *Statistical Analysis with Missing Data*. Wiley, New York.

Loh,W. and Vanichsetakul,N. (1988) Tree-structured classification via generalized discriminant analysis. *J. Am. Stat. Assoc.*, **83**, 715–725.

Perou,C.M., Sorlie,T., Eisen,M.B., van de Rijn,M., Jeffrey,S.S., Rees,C.A., Pollack,J.R., Ross,D.T., Johnsen,H., Akslen,L.A., Fluge,O., Pergamenschikov,A., Williams,C., Zhu,S.X., Lonning,P.E., Borresen-Dale,A.L., Brown,P.O. and Botstein,D. (2000) Molecular portraits of human breast tumours. *Nature*, **406**, 747–752.

Raychaudhuri,S., Stuart,J.M. and Altman,R.B. (2000) Principal components analysis to summarize microarray experiments: application to sporulation time series. *Pac. Symp. Biocomput.*, 455–466.

Spellman,P.T., Sherlock,G., Zhang,M.Q., Iyer,V.R., Anders,K., Eisen,M.B., Brown,P.O., Botstein,D. and Futcher,B. (1998) Comprehensive identification of cell cycle-regulated genes of the yeast *Saccharomyces cerevisiae* by microarray hybridization. *Mol. Biol. Cell*, **9**, 3273–3297.

Tamayo,P., Slonim,D., Mesirov,J., Zhu,Q., Kitareewan,S., Dmitrovsky,E., Lander,E.S. and Golub,T.R. (1999) Interpreting patterns of gene expression with self-organizing maps: methods and application to hematopoietic differentiation. *Proc. Natl Acad. Sci. USA*, **96**, 2907–2912.

Wilkinson,G.N. (1958) Estimation of missing values for the analysis of incomplete data. *Biometrics*, **14**, 257–286.

Yates,Y. (1933) The analysis of replicated experiments when the field results are incomplete. *Emp. J. Exp. Agric.*, **1**, 129–142.