Missing Values Estimation in Microarray Data with Partial Least Squares Regression*

Kun Yang[1], Jianzhong Li[1], and Chaokun Wang[1,2]

[1] Department of Computer Science and Engineering,
Harbin Institute of Technology, Harbin, China
[2] School of Software, Tsinghua University, Beijing, China

Abstract. Microarray data usually contain missing values, thus estimating these missing values is an important preprocessing step. This paper proposes an estimation method of missing values based on Partial Least Squares (PLS) regression. The method is feasible for microarray data, because of the characteristics of PLS regression. We compared our method with three methods, including ROWaverage, KNNimpute and LLSimpute, on different data and various missing probabilities. The experimental results show that the proposed method is accurate and robust for estimating missing values.

1 Introduction

Microarray technology can be used to detect the expression levels of thousands of genes under a variety of conditions. Microarrays have been successfully applied in many studies over a broad range of biological processes [1, 2, 3, 4]. Despite the popular usage of microarray, there are frequently missing values in microarray data. The missing-value phenomenon can occur for various reasons, including insufficient resolution, slide scratches, dust or hybridization error.

However, many multivariate analyses, such as Support vector machines (SVMs) [5], principal component analysis (PCA) [6] and singular value decomposition (SVD) [7], have difficulty to be applied straightforwardly to the data with missing values. One solution to the missing values problem is to repeat the experiments. But this strategy is often unfeasible for economic reasons and the limitations of available biological material. Thus, it is an important preprocessing step to estimate these missing values.

There are some approaches for estimating the missing values. A simple and common used method is to fill missing values by Zero (ZEROimpute) or by the row(or gene)/ column(or sample) average (ROWaverage) [4]. Furthermore, some advanced estimation methods have been introduced. The SVD-based method (SVDimpute) and weighted k-nearest neighbors (KNNimpute) have been proposed by Troyanskaya *et al* [8]. Recently, several methods including Bayesian

* This work was supported by the 863 Research Plan of China under Grant No. 2004AA231071 and the NSF of China under Grant No. 60533110.

V.N. Alexandrov et al. (Eds.): ICCS 2006, Part II, LNCS 3992, pp. 662–669, 2006.

PCA (BPCA) [9], least squares imputation (LSimpute) [10] and local least squares imputation (LLSimpute) [11] have been introduced.

Partial least squares(PLS) regression is a novel multivariate data analysis method and popularly used in the field of chemometrics. PLS regression has many advantages, which ordinary multiple linear regression does not have, such as avoiding the harmful effects in modeling due to the collinearity of explanatory variables and regressing when the number of observation is less than the number of explanatory variables, etc[12, 13, 14].

This paper presents a specialized missing values estimation method based on Partial Least Squares (PLS) regression. This method is referred as PLSimpute, which uses PLS regression to construct the prediction equation between the target gene with missing values and the similar genes, and then estimates the missing values. The estimation accuracy of our method is compared with that of the wildly used ROWaverage, KNNimpute and recent LLSimpute by introducing artificial missing values. In addition, the normalized root mean squared error (NRMSE) is used to quantitatively measure the estimation accuracy.

The remainder of the paper is organized as follows. Section 2 presents a brief description the missing values problem, while the method PLSimpute is detailed in Section 3. Section 4 provids the experimental results and discussion. Section 5 contains the conclusion.

2 Preliminaries

Throughout the paper, microarray gene expression data is represented as an $n \times p$ matrix A, where n is the number of samples (observations) and p is the number of genes. The rows correspond to samples, the columns correspond to the genes, and element $A_{i,j}$ is the expression value of gene j in sample i. The i-th row vector of A, noted by s_i, and the j-th column vector, denoted by g_j, are called the expression profile of the i-th sample and the j-th gene, respectively.

Assume the target gene g_1 has one missing value in sample 1. For $2 \leq j \leq p$, the gene expression vector g_j is noted by $(w_j, x_j^T)^T$, and let $g_1 = (a, y^T)^T$. Then the matrix A can be denoted by

$$\mathbf{A} = \begin{pmatrix} a & w_2 & \dots & w_p \\ y & x_2 & \dots & x_p \end{pmatrix}.$$

Firstly we should find others genes, which have a value in sample 1, highly similar to g_1, based on the expression profile from sample 2 to sample n. Then, the values of those similar genes in sample 1 are used to predict the value of a. That is, in order to estimate the value of a, we should find the genes which have expression vectors x_j similar to y, and use the corresponding w_j to recover a. After finding the first k similar gene expression vectors (denoted as $x_j, 1 \leq j \leq k$), vector y can be represented as a linear combination

$$y = b_1 x_1 + b_2 x_2 + \ldots + b_k x_k + \varepsilon, \tag{1}$$

where b_i are the coefficients of the linear combination. Accordingly, the missing value a can be predicted by

$$\hat{a} = b_1 w_1 + b_2 w_2 + \ldots + b_k w_k. \tag{2}$$

The equation(1) is a multiple linear regression equation between y and $x_1, \ldots, x_k$, and these parameters can be estimated based on least squares principle.

It is essential for good estimation of the parameters in a multiple linear regression equation that the number of variables is a part of the number of observations. When there are much more variables than the observations in presence, the original least squares method can not be applied straightforwardly. But in microarray data, the number of samples is much smaller than the number of genes. Thus, ordinary multiple regression is only feasible when few genes is included in the regression model. This may lost some useful information. Furthermore, the collinearity of the varibles x_i will degrade the prediction ability of equation(1).

3 Method

Partial Least Squares (PLS) regression is considered especially useful for constructing prediction equations when there are many explanatory variables and comparatively little samples. PLS can overcome the collinearity problem of the explanatory variables. Our method based on Partial Least Squares regression consists of two main steps. The first step is to select k highly similar genes. In the second step, univariate partial least squares regression is used to form prediction equation, then the prediction equation is used for estimating the missing values.

3.1 Selecting genes

To predict the missing values in gene g_j in A, the top k similar genes for g_j are selected. The similarity is measured by the Euclidean distance or the absolute Pearson correlation coefficients. In the process of finding similar genes, the components, corresponding to missing values, of the target gene g_j are ignored in computation. After computing the similarity between g_j and these candidate genes, top k highly similar genes are selected.

3.2 Partial Least Squares Regression

To describe the PLS regression, some notations are introduced here. Let $X = (x_1, \ldots, x_p)$ be an $N \times p$ matrix of N samples and p genes, where column vector x_i is N-dimensional, corresponding to the expression profile of the i-th similar gene on N samples. Also, let Y be the expression profile of the target gene on N samples. In the situation of regression, Y is the response variable while $x_i (1 \leq i \leq p)$ are the explanatory variables. Additionally, we should note that p can be much larger than N.

The main idea of PLS regression is to construct new explanatory variables (called *components* or *factors*) that capture most of the information in the X,

where each component is a linear combination of x_i, while reducing the dimensionality of the regression problem by using fewer components than the number of variables x_i to predict Y.

The objective criterion of forming components is to sequentially maximize the covariance between the response variable Y and the component t, which is a linear combination of the columns in X, i.e. $t = Xw$. Thus the objective can be summarized as the following formula

$$w_k = \underset{\|w\|=1}{arg\,\mathrm{Max}}\ \mathrm{cov}^2(\mathrm{Xw}, \mathrm{Y})$$
$$\text{S. T.:} \quad w^T S w_j = 0 \text{ for all } \quad 1 \le j < k \tag{3}$$

where $S = X^T X$. The maximum number of components is the rank of X. The details of PLS algorithm are given in Table1. According the algorithm of PLS, we can use equation $Y \approx X \cdot \alpha = X \cdot \sum_{k=1}^{d} \gamma_k \cdot \prod_{j=1}^{k-1} (I - w_j \cdot p_j^T) \cdot w_k$ to predict the missing values. For some theoretical aspects and properties of PLS, please see [12, 13, 14].

Table 1. The Algorithm of PLS

1 For $k = 1$ To d
2 $w_k = X_{k-1}^T \cdot Y_{k-1} / \|X_{k-1}^T \cdot Y_{k-1}\|$;
3 $t_k = X_{k-1} \cdot w_k$;
4 $p_k = X_{k-1} \cdot t_k / \|t_k\|^2$;
5 $\gamma_k = Y_{k-1} \cdot t_k / \|t_k\|^2$;
6 $Residual: \ X_k = X_{k-1} - t_k \cdot p_k^T$; (with $X_0 = X$);
7 $Residual: \ Y_k = Y_{k-1} - \gamma_k \cdot t_k$; (with $Y_0 = Y$);
8 End For

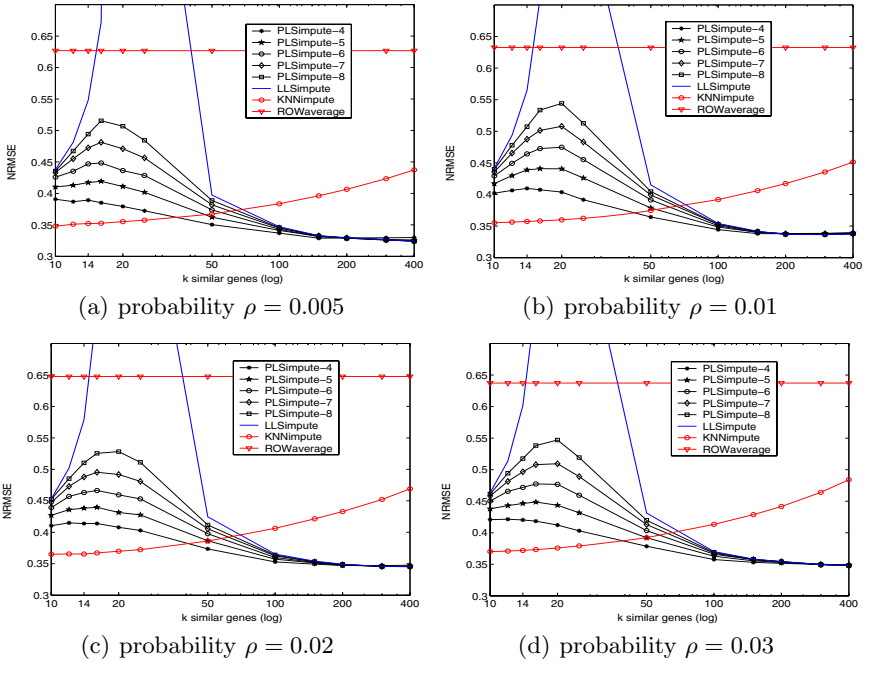
The last problem is to decide the number of components in the regression model. Helland[12] reported that partial least squares regression often needs few components to give its optimal prediction. So we can determine the number of components easily at empirical results. Furthermore, another method for this problem is to use cross validation procedure[15]. In this procedure, the optimal number of components is found at the minimal sum of squared errors of prediction.

4 Results and Discussion

In this section, we compare the performance of our method with three missing values estimation methods, including ROWaverage, KNNimpute and LLSimpute. Two microarray data have been used in our experiments, including both time series and no-time series data. The first data (*Sp*) is attained from 784 cell-cycle-regulated genes in 14 experiments, which was studied by Spellman *et al* [16]. After deleting gene row with missing values, this data has 474 genes

Table 2. Probabilty ρ and the corresponding expected number of genes with and without missing values on two data

	Colon				Sp			
ρ	0.005	0.01	0.02	0.03	0.005	0.01	0.02	0.03
E_0	1792	1603	1283	1023	442	412	358	309
$p - E_0$	208	397	717	977	32	62	116	165



(a) probability $\rho = 0.005$ (b) probability $\rho = 0.01$

(c) probability $\rho = 0.02$ (d) probability $\rho = 0.03$

Fig. 1. NRMSEs of various methods on the Colon data over a wide range of similar genes (k) used in estimation

and 14 experiments (i.e. samples), and is the same data used in LLSimpute [11]. The second data (*Colon*) consists of 22 normal tissues samples and 2000 human genes from the colon data analyzed initially by Alon *et al*[17].

To evaluate the performance of different methods, artificial missing entries are created from a complete expression matrix (i.e. without missing values) according to the method of Ouyang *et al*[18]. That is, each entry in the complete matrix is randomly and independently treated as missing value with a probability ρ. If a probability ρ and a complete matrix A with N samples and p genes are given, then the expected number of genes with k missing values in the artificial data can be calculated by function(4).

$$E_k = p \cdot C_N^k \cdot \rho^k \cdot (1-\rho)^{N-k}. \tag{4}$$

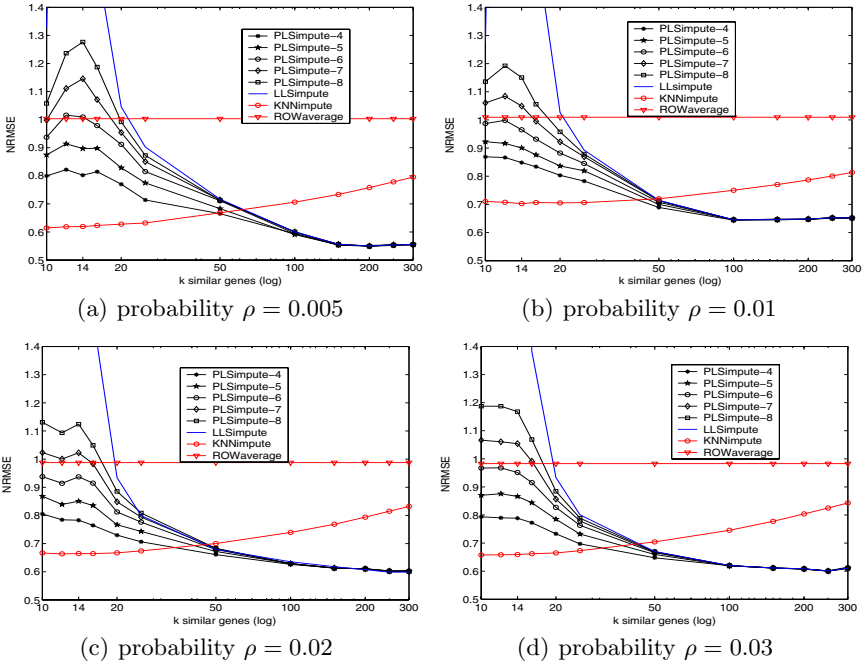
Table 2 shows the different probabilities ρ used in experiments on both data and the corresponding expected numbers of genes with and without missing values.

For the missing entries are artificial, the performance of each method is measured by normalized root mean squared error(NRMSE).

$$NRMSE = \sqrt{\text{mean}[(\text{y}_{\text{guess}} - \text{y}_{\text{answer}})^2]} \,/\text{std}[\text{y}_{\text{answer}}] \tag{5}$$

where the mean and the SD are calculated over all missing entries in the whole matrix.

The experimental method for each complete data is: randomly create a artificial missing matrix with probability ρ, use all methods to estimate these missing values, calculate the NRMSE of each method by the values of estimation and that of the origin. This procedure is repeated 10 times. And the similarity is measured by the Euclidean distance. The results of all methods are the average NRMSEs on 10 randomly created data with an exact probability ρ. Because Troganskaya *et al* [8] reported that KNNimpute was insensitive to the exact value of k in the range 10-20 and the best results were in this range, different values of k (i.e. 10, 12, 14, 16, 20) are always used in every experiment. Furthermore, in order to investigate the effect of the number of components used in PLS regression on the performance of PLSimpute, the results attained from several fixed



(a) probability $\rho = 0.005$ (b) probability $\rho = 0.01$

(c) probability $\rho = 0.02$ (d) probability $\rho = 0.03$

Fig. 2. NRMSEs of various methods on the SP data over a wide range of similar genes (k) used in estimation

numbers of components (i.e. 4,5,6,7,8) are reported in all experiments. Figure 1 and 2 show the experimental results on both data. We observe the following:

For various missing probabilities ρ on both datasets, PLSimpute shows the lowest estimation error, while LLSimpute shows less accurate results than PLSimpute. The performance of ROWaverage is the worst one and the method of KNNimpute is the medium. This result is the same as previous studies [8, 11].

We can see that PLSimpute just needs few components used in PLS regression to give a good estimation. This result is coincident with the result reported by Helland[12].

Moreover, the estimation errors of PLSimpute with different numbers of components are always less than LLSimpute when various numbers of similar genes are used in estimation. Additionally, When the components used in PLS regression increase, the performance of PLSimpute comes near to LLSimpute. It turns out that finding an empirical result of the feasible number of components in PLSimpute is relatively easy, despite the optimal number of components is possibly difficult to be selected. Anyway, we can determine the number of components used in PLSimpute by cross validation procedure [15].

The same as KNNimpute and LLSimpute, the number of similar genes (parameter k) is important to select for a high performance of PLSimpute, while the number of principal axes (eigenvectors) is important for both BPCA and SVDimpute. We can determine the parameter k by a heuristic method [11]: treating some no-missing elements as missing values, and making a estimation, then select the special value of k in respect to the best result of estimation.

5 Conclusion

The missing values usually appear in microarray data. The process of missing values in microarry data is an important preprocessing step in gene expression data analysis, because many analysis methods require complete gene expression data. This paper proposes a method of missing values estimation based on Partial Least Squares regression (PLSimpute) in Microarray data. The method can be used when many similar genes and comparatively little experiments are included in the estimation procedure. Furthermore, the method can reduce the harmful effect of the collinearity of the similar genes used for prediction.

Our experimental results show that PLSimpute gives better performance than ROWaverage, KNNimpute and LLSimpute. As reported in the previous study of LLSimpute [11], LLSimpute has a better ability for estimation than BPCA. These results turn out that PLSimpute is a robust and accurate method for estimating missing values in microarray data.

References

1. Chu, S., DeRisi, J., *et al*: The transcriptioal program of sporulation in budding yeast. Science **278** (1998) 680–686
2. Alon, U., Barkai, N., *et al*: Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotid arrays. Proc. Natl. Acad. Sci. USA **96** (1999) 6745–6750

3. Golub, T.R., Slonim, D.K., *et al*: Molecular classification of cancer: class discovery and class prediction by expression monitoring. Science **286** (1999) 531–537
4. Alizadeh, A.A., Eisen, M.B., *et al*: Distinct types of diffuse large B-cell lymphoma identified by gene expression profiling. Nature **403** (2000) 503–511
5. Vapnik, V. : The Nature of Statistical Learning Theory. Springer-Verlag. New York (1995)
6. Raychaudhuri, S., Stuart, J.M. and Altman, R.: Principal components analysis to summarize microarray experiments: application to sporulation time series. Pac. Symp. Biocomput. (2000) 455–466
7. Alter, O., Brown, P.O. and Botstein, D. : Singular value decomposition for genome-wide expression data processing and modeling. Proc. Natl Acad. Sci. USA **97** (2000) 10101–10106
8. Troyanskaya, O., Cantor, M. *et al* : Missing value estimation methods for DNA microarray. Bioinformatics **17** (2001) 520–525
9. Oba, S., Sato, M., *et al*: A Bayesian missing value estimation method for gene expression profile data. Bioinformatics **19** (2003) 2088–2096
10. Bø, T.H., Dysvik, B. and Jonassen, I. : LSimpute: accurate estimation of missing values in microarray data with least squares methods. Nucleic Acids Res. **32(e34)** (2004)
11. Kim, H., Golub, G.H. and Park, H. : Missing value estimation for DNA microarray gene expression data: local least squares imputation. Bioinformatics **21** (2005) 187–198
12. Helland, I.S. : On the structure of partial least squares regression. Commun. Stat. -Simul. Comput. **17** (1988) 581–607
13. Garthwaite, P.H. : An interpretation of partial least squares. J. Am. Stat. Assoc. **89** (1994) 122-127
14. Wang, H.: Partial Least-squares Regression — Method and Applications. National Defence Industry Press. China (1999)
15. Stone, M. : Cross-validatory choice and assessment of statistical predictions. J. Roy. Statist. Soc. **36** (1974) 111–133
16. Spellman, P.T., Sherlock, G., *et al*: Comprehensive identification of cell cycle-regulated genes of the yeast Saccharomyces cerevisiae by microarray hybridization. Mol. Biol. Cell **9** (1998) 3273–3297
17. Alon, U., *et al*: Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by ologonucleotide arrays. Proc. Natl Acad. Sci. USA, **96** (1999) 6745–6750
18. Ouyang, M., Welsh, W.J. and Georgopoulos P.: Gaussian mixture clustering and imputation of microarray data. Bioinformatics. **20** (2004) 917–923